Paris Mobile Configuration and Navigation

Last updated: September 14, 2020
Some examples and graphics depicted herein are provided for illustration only. No real association or connection to ServiceNow products or services is intended or should be inferred.

If you have comments about this documentation, submit your feedback to:

docfeedback@servicenow.com
# Table of Contents

**ServiceNow Mobile**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Now Mobile app</td>
<td>5</td>
</tr>
<tr>
<td>Now Mobile app demo</td>
<td>10</td>
</tr>
<tr>
<td>Agent mobile app</td>
<td>11</td>
</tr>
<tr>
<td>Agent app demo</td>
<td>14</td>
</tr>
<tr>
<td>Onboarding Mobile app</td>
<td>15</td>
</tr>
<tr>
<td>Onboarding app demo</td>
<td>20</td>
</tr>
<tr>
<td>Considerations before implementation</td>
<td>21</td>
</tr>
<tr>
<td>Base system functionality for ServiceNow mobile</td>
<td>25</td>
</tr>
<tr>
<td>Mobile publishing</td>
<td>29</td>
</tr>
<tr>
<td>Mobile themes</td>
<td>36</td>
</tr>
<tr>
<td>Mobile security</td>
<td>62</td>
</tr>
<tr>
<td>Edge Encryption for ServiceNow mobile</td>
<td>80</td>
</tr>
<tr>
<td>Mobile GovCommunityCloud (GCC) Compliance</td>
<td>81</td>
</tr>
<tr>
<td>Enterprise mobility management (EMM)</td>
<td>83</td>
</tr>
<tr>
<td>Tested devices and supported versions for ServiceNow mobile apps</td>
<td>121</td>
</tr>
<tr>
<td>Using the mobile apps</td>
<td>123</td>
</tr>
<tr>
<td>Log in to an instance with a mobile app</td>
<td>123</td>
</tr>
<tr>
<td>Mobile app structure</td>
<td>125</td>
</tr>
<tr>
<td>Mobile search</td>
<td>132</td>
</tr>
<tr>
<td>Mobile activity stream</td>
<td>136</td>
</tr>
<tr>
<td>Mobile filters</td>
<td>138</td>
</tr>
<tr>
<td>Mobile platform dashboards</td>
<td>143</td>
</tr>
<tr>
<td>Mobile app settings</td>
<td>163</td>
</tr>
<tr>
<td>Accessing an instance on a mobile device web browser</td>
<td>183</td>
</tr>
<tr>
<td>Mobile plugins and upgrades</td>
<td>184</td>
</tr>
<tr>
<td>Mobile plugins</td>
<td>185</td>
</tr>
<tr>
<td>Migrate from Classic mobile to ServiceNow mobile</td>
<td>188</td>
</tr>
<tr>
<td>Migration from Madrid to New York mobile</td>
<td>190</td>
</tr>
<tr>
<td>Migration from New York and later releases</td>
<td>202</td>
</tr>
<tr>
<td>Building and configuring mobile apps</td>
<td>202</td>
</tr>
<tr>
<td>Mobile tutorials</td>
<td>203</td>
</tr>
<tr>
<td>Design considerations for mobile apps</td>
<td>232</td>
</tr>
<tr>
<td>Mobile hierarchy</td>
<td>232</td>
</tr>
<tr>
<td>Mobile Studio</td>
<td>241</td>
</tr>
<tr>
<td>Create a mobile application using Guided Application Creator</td>
<td>247</td>
</tr>
<tr>
<td>Navigation bar</td>
<td>248</td>
</tr>
<tr>
<td>Applet launchers</td>
<td>256</td>
</tr>
<tr>
<td>Data items</td>
<td>297</td>
</tr>
<tr>
<td>Mobile dashboard preview</td>
<td>311</td>
</tr>
<tr>
<td>Applets</td>
<td>317</td>
</tr>
<tr>
<td>Mobile styles</td>
<td>449</td>
</tr>
<tr>
<td>Mobile functions</td>
<td>473</td>
</tr>
<tr>
<td>Mobile push notifications</td>
<td>513</td>
</tr>
<tr>
<td>Siri shortcuts</td>
<td>529</td>
</tr>
<tr>
<td>Displaying campaigns on ServiceNow mobile</td>
<td>530</td>
</tr>
<tr>
<td>Additional mobile app configuration options</td>
<td>539</td>
</tr>
<tr>
<td>Mobile authentication</td>
<td>551</td>
</tr>
<tr>
<td>Configurations for Now Mobile</td>
<td>552</td>
</tr>
<tr>
<td>Configure a personalized greeting</td>
<td>561</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.

ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Configure My Requests to track open records ................................................................. 561
Disable people search .............................................................................................................. 562
Configure catalogs ................................................................................................................... 562
Configure knowledge bases .................................................................................................... 562
Configure Siri shortcuts for Now Mobile ................................................................................. 563
Add a quick action in your mobile applications ........................................................................ 567
Enable Live Agent in the Now Mobile app .............................................................................. 569
Enable voice search ................................................................................................................ 570
Populate search suggestions in the mobile apps ...................................................................... 570

User Experience Analytics ........................................................................................................... 571
Mobile location tracking configuration .................................................................................... 573
Offline mode ............................................................................................................................. 574
  Request Offline Mode .............................................................................................................. 574
  Configure Offline Mode behavior ........................................................................................... 576
  Configure Offline Mode for applications or applets ................................................................. 577
  Configure Offline properties for action functions ..................................................................... 579
  Scheduled offline caching ....................................................................................................... 581
Mobile bar-code scanning ........................................................................................................... 582
  Multi-scan for mobile applications ........................................................................................... 585
  Configure a grouped input for multiple scans .......................................................................... 588
Localization on mobile devices .................................................................................................. 590
Domain separation in mobile .................................................................................................... 593
ServiceNow Classic mobile app ................................................................................................... 593
  Device security for ServiceNow Classic ................................................................................... 595
  Mobile configuration for ServiceNow Classic .......................................................................... 600
  Get started with the ServiceNow Classic app .......................................................................... 637
  Support for ServiceNow Classic ............................................................................................. 679

Index ........................................................................................................................................... 681
ServiceNow Mobile

Access your instance from anywhere using the ServiceNow® Now Mobile, Mobile Agent, and Mobile Onboarding apps on your mobile device.

Take your work wherever you go

Your instance is available whether you're away from your desk or at the office. Manage incidents, collaborate with your teams, respond to approval requests, access the knowledge base, or receive push notifications with your mobile device.

Note: The Now Mobile platform is only available for Madrid and later releases. For London and earlier releases, see Mobile configuration for ServiceNow Classic.

View and download the full infocard for a highlight of ServiceNow mobile features.

<table>
<thead>
<tr>
<th>Download the mobile apps</th>
</tr>
</thead>
<tbody>
<tr>
<td>the Now Platform provides three mobile apps. Each app is designed for a specific set of users. Download these apps for your Android or iOS devices using the Google Play store or Apple app stores.</td>
</tr>
<tr>
<td>Note: The Apple App Store and the Google Play Store are the only two official distribution channels that are supported for these mobile applications.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Get started quickly using available base system applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Take advantage of pre-configured mobile applications to get your users started with common functionality like approvals for Field Service Management.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Use the features on your mobile device with your ServiceNow instance.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceNow apps can use your mobile device's features to make calls, send SMS messages, and display navigation information. For example, you can use the camera on your mobile device to scan bar codes, or create image attachments.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Use Studio to create custom applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Build applications that you can customize for your organization's specific needs. With Studio, you can quickly create applications in a simple, low-code interface.</td>
</tr>
</tbody>
</table>
Download the mobile apps

Your users access ServiceNow mobile from a client such as Mobile Agent, Now Mobile, or Mobile Onboarding. These applications are available in the Google Play or Apple App Store. Each app works with a specific area of the ServiceNow platform. To learn how each app works with your instance, see ServiceNow Mobile.

Tailored experiences

Take advantage of pre-configured applications. These applications can get you started right away with the ServiceNow mobile platform.

Now Mobile

Now Mobile is tailored for your employees. Employees can find answers, create requests, and get help across all departments.

Mobile Agent

Mobile Agent is tailored for your service agents. Agents can get work done anywhere through an easy to navigate native mobile interface.

Mobile Onboarding

Mobile Onboarding is tailored for your new hires. New employees can get informed and become productive in a matter of minutes.
Now Support

Use Now Support to conduct key customer support tasks from a modern mobile app that is powered by the ServiceNow mobile platform.

Note: The Now Support is designed to interact with customer support, and is not a customizable mobile app.

Use the features on your mobile device with your ServiceNow instance.
Use built-in features on your mobile device to perform certain actions in your mobile apps. Use your device's camera to upload photos, perform image-based searches, or scan bar-codes to catalog your assets. The Now Mobile app uses your mobile device's photo, phone, keyboard, messaging, and email applications.

**Use Studio to create custom applications**

Mobile Studio is part of the existing Studio environment. With mobile Studio, you can create, modify, and manage applets for use in ServiceNow mobile apps using an intuitive, low code interface.
Get started

- To learn more about the elements of ServiceNow mobile, see *Mobile app structure*.
- For more information on how to begin setup with ServiceNow mobile, see *Building and configuring mobile apps*.
- To learn about the migration process from classic mobile to ServiceNow mobile, see *Migrate from Classic mobile to ServiceNow mobile*.
- Learn more about how to create your own applications on the *ServiceNow® Developer Site*. The Developer Site contains courses and examples to get you started on creating custom mobile applications suited to your organization's specific needs. To access the mobile content on the Developer Site, see *Mobile application training*.

Applications and features

- *Agent mobile app*
- *Considerations before implementation*
- *Now Mobile app*
- *Using the mobile apps*
- *Mobile Onboarding app*
- *Building and configuring mobile apps*
### Now Mobile app

Enable your users to submit incidents and requests, manage tasks, and access company resources from anywhere using the Now Mobile app.

#### Features of the Now Mobile app

The Now Mobile app is a downloadable app designed to help your employees with the following tasks:

- Get help by submitting requests and issues.
- Find answers using global search and knowledge base.
- Submit and manage approvals.
- View and complete tasks.
- Upload images and attachments to ServiceNow records.
- Request help from Agent Chat and Virtual Agent.
- Use pre-configured Siri shortcuts unique to the Now Mobile app to complete common actions.

**Note:** For more information, see Base system functionality for ServiceNow mobile.

#### Download the app

Download the Now Mobile app for Apple iOS or Google Android from the Apple App Store or the Google Play store.

#### Access the demo

Take a look at the Now Mobile app demo. The demo is a curated experience designed to show how your employees can accomplish their work from anywhere. For more information on the app, see Now Mobile app demo.
Now Mobile app demo

Access the demo in your Now Mobile app to learn how the app can help you work from anywhere using your mobile device.

The Now Mobile demo includes curated demonstrations to show how employees can get work done on their mobile device. Employees can perform tasks such as submit incidents and service catalog requests, manage tasks, and access company resources.

Access the demo

In the Now Mobile app, tap the Try with a demo account button at the bottom of your login screen. Then, tap the Launch Demo button to start using the demo. The demo automatically logs in to a ServiceNow instance with an example account.
Enter the instance address or scan a QR code.

- example.service-now.com

Enter nickname

Nickname

Save and Login

TRY WITH A DEMO ACCOUNT
Demo highlights

The Now Mobile demo displays an applet launcher page divided into six UI sections. Tap an item in any section to open the item, or tap See All to view all items in that section. For details on how applet launchers, applets, actions work together in the mobile environment, see Mobile hierarchy.

Tech Lounge

The Tech Lounge campaign display shows how you can easily present targeted information to employees based on site, organization, and other key metrics. Selecting Schedule Now takes you through the experience of booking Tech Lounge visit, as well as leaving the queue if necessary.

News

The News UI section displays information from a knowledge base. You can view knowledge article information, as well as subscribe, rate, and comment on the article.

My Approvals

The My Approvals section shows a list of approvals in the requested state, where you are the assigned approver. You can tap any approval record to view it. You can approve or reject approvals using the buttons on the form. You can also quickly approve or reject approvals directly from the list by swiping to the left on an approval record.

My Requests

The My Requests section shows a list of all active requests and their status, including both catalog items and incidents. You can tap any record here to see working details and all order activities, as well as add additional comments or pictures.

Help with an Issue

The help with an issue section allows you to easily submit an incident for any issue. Once submitted that incident is available immediately to view or edit in My Requests.

Scan for Help

The Scan for Help option demonstrates how you can use QR codes to save time when creating requests. By scanning the QR code associated with a conference room your mobile device’s camera, your facilities team has all the information they need about the room. All you have to do is scan the code and select an issue.

For more information on incident management using mobile applications, see Mobile experience with ITSM

Mobile agent

Book a Room

The applets in the Book Conference Rooms section demonstrate how to use applets to automate common tasks without the need to use another app. You can book a conference room by tapping an available room, then tapping the Book button. You can also quickly book a room by swiping the room on the list to the left.

The Book Now applet gives you a list of all available rooms. The Book for later applet displays a list of rooms based on information you provide through inputs in the applet. These inputs are an example of mobile parameters. For more information on how to use mobile parameters in your applets, see Mobile parameter tutorials.

White Glove Support

The White Glove Support option quickly creates an incident record based on the text you enter into a field. You can enter text with a keyboard, or use your phone’s voice-to-text capability.

People

Use the People section to view and communicate with your contacts using your mobile device’s built in capabilities. You can call, email, or navigate to one of your contacts using field functions configured on the contact form. You can also call or email or using swipe actions on the list that appears when you tap See All.
For more information on employee directory applets, see *Configure an employee directory applet*.

For information on configuring functions like those used to make calls and send emails, see *Mobile functions*.

**Answers**

The Answers section contains knowledge articles you have subscribed to or viewed recently to give you quick access to the information need. Tap articles within the applets to view them. You can subscribe or unsubscribe to articles using a button on the article, or through swipe actions on the list.

**Chat with a Virtual Agent**

Use the chat button to connect to a virtual agent. Virtual agent is an automated conversational interface you can use to address common issues, such as password resets, or email account setup. The Now Mobile uses a quick action to access chat. For more information on Virtual Agent, see *Virtual Agent*.

The chat button on the demo launch page is a quick action. For details on how to add a virtual agent to your mobile apps, see *Enable Live Agent in the Now Mobile app*.

**Learn more about the Now Mobile app**

Use the following links to learn more about on how to use and configure the Now Mobile app.

- *Now Mobile app*
- *Configurations for Now Mobile*
- *Now Mobile app*
- *Building and configuring mobile apps*

**Agent mobile app**

Use the ServiceNow Agent mobile app to update records, coordinate with coworkers, track your location, and work without an internet connection — all from your mobile device. Download ServiceNow Agent for Apple iOS or Google Android from the Apple App Store or the Google Play store.
Features of the Mobile Agent app

- Give your agents a mobile interface to triage, address, and resolve requests.
- Use the ServiceNow Agent mobile app's offline mode to work in areas without an internet connection. For details see [Offline mode](#).
- Use the ServiceNow Agent mobile app's location tracking feature to keep a record of where an agent last worked on an instance. For more detail, see [Mobile location tracking configuration](#).

**Note:** For more information, see [Base system functionality for ServiceNow mobile](#).

### Download the app

Download the Mobile Agent app for Apple iOS or Google Android from the Apple App Store or the Google Play store.

### Access the demo

Take a look at the Mobile Agent app demo. The demo is a curated experience designed to show how your agents can complete tasks wherever they are. For more information on the app, see [Now Mobile app demo](#).

---

**Agent app demo**

Access the demo in your Mobile Agent app to learn how the app can help your agents work from anywhere using their mobile device.
The Mobile Agent demo includes curated demonstrations to show how your agents can perform tasks on their mobile device. Agents can track incoming work, access important knowledge, and collaborate with teams on the go.

Access the demo

In the Mobile Agent app, tap the TRY WITH A DEMO ACCOUNT button at the bottom of the login screen. Then, select the role that matches the demo you’d like to see:

- Field Service Agent – manage their schedules, access knowledge and parts inventories, and track their work for the day
- Service Desk Agent – prioritize their tasks for the day, manage their shifts, and swiftly respond to major incidents
- IT Operator – triage and assign alerts, research root cause, and collaborate to act and resolve issues quickly

Finally, tap the Launch Demo button to start using the demo. The demo automatically logs you in to a ServiceNow instance with an example account.
Instances

Enter the instance address or scan a QR code.

- example.service-now.com

Enter nickname

Nickname

Save and Login

Choose a persona

- Customer Service Agent
- Field Service Agent
- IT Operator
- Project Manager
- Security Analyst
- Service Desk Agent

Launch Demo

By tapping on the Launch Demo button to use the demo, you agree to be bound by the terms of the End User License Agreement (EULA), and in particular the terms of the demo section. If you do not agree to the terms of the EULA, you must not use the demo.

The ServiceNow Agent mobile demo shows how ServiceNow helps business professionals simplify their work and increase productivity. The screens have been modified to simplify the demo experience across a variety of ServiceNow solutions and are representative of what is possible using ServiceNow Agent.
Demo highlights

The ServiceNow Agent mobile demo displays an applet launcher page divided into sections. Tap an item in any section to open the item or tap See All to view all items in that section. For details on how applet launchers, applets, actions work together in the mobile environment, see Mobile hierarchy.

Customer Service Agent

Manage priority cases

Agents use Cases that need attention to quickly identify priority cases needing their response. These are cases where customers have responded back, or where internal blocking tasks have been completed. Agents can open cases to view the latest updates, add comments or work notes, edit case details, and propose solutions. Related lists show SLAs, tasks, child cases, and other case-related lists.

Initiate case workflows

From the New cases list, agents can swipe to accept or reject a case. They have quick access to active critical cases. From this case list, agents view case details and can initiate standard workflows, such as proposing a major case or escalating a case.

Approve escalations and change requests

If users such as managers and supervisors have approval roles, they see their escalation approvals and change request approvals as well. They can open the approvals, view details, and either approve or reject the requests. Managers can also easily reassign cases to other teams or individual users.

Field service agent

Optimize my daily schedule

My Task Map displays locations for each of an agent's assigned tasks. Use the Optimize Today’s Route option to reorder tasks and show the quickest way to get to work locations.

Access knowledge articles and parts inventories

Agents view the parts required for the tasks they're currently assigned, and also view a full inventory of on-hand and defective stock. Agents can locate needed parts and see a map of warehouses and technicians that carry the parts they need. The Mobile Agent app includes knowledge management capabilities to provide immediate access to the knowledge that’s relevant to the task at hand. These capabilities include FAQs, error codes, schematics, and even videos.

Manage my work order tasks

Once tasks are prioritized, your agents select their first work order and accept it. Thus acceptance provides visibility to the dispatcher and updating the agent’s location. With a swipe, they can track the parts needed to fix a problem or scan a barcode to consume an asset from inventory. As part of completing certain tasks, your agents can document conformance to a safety checklist, document equipment readings, or fill out an inspection questionnaire. These actions are done using forms specific to the type of work or equipment or even customer. Then agents can close their tasks, capturing a customer signature and adding any important work notes.

For more information on Field Service Management using mobile applications, see Mobile experience for Field Service Management.
IT operator

**Triage and assign alerts in a services context**

Use the Services section to view prioritize which services need attention to address operational alerts. View service details, then view the primary alerts grouped by machine learning. View individual alerts to see if any other services are impacted by the alert, providing insight into the scope of the issue.

**Investigate root cause of alerts**

Drill into the alert details, review the secondary alerts that are automatically grouped with the alert. These alerts provide key insights into the root cause of the problem. View related incidents, repeated alerts, and log files attached to a related incident. Access knowledge articles that provide more information about the problem, and suggestions actions to address it.

**Collaborate and take action**

With the root cause and recommended solutions, agents can add work notes to the alert and even initiate actions. Actions can include like kicking off a workflow or proposing a major incident.

For more information on Event Management using mobile applications, see *Mobile experience for Event Management.*

Project Manager

**Monitor all your projects on the go**

Quickly view key project status and details, with a click or a swipe, all from this single, native, mobile app. Filter to view color-coded, at-a-glance status of projects by project manager, portfolio, or business unit. See Critical Projects and Projects in Red on the opening page of the app - and simply click in for more actionable insights.

**Raise awareness and take action**

Gain mobile visibility into the number of open risks, issues decisions, actions, and changes (RIDAC) that may affect the overall project health. Enter new RIDAC items right from your mobile device throughout the project life cycle.

**Collaborate with project stakeholders**

Provide timely project updates anytime, anywhere. The native mobile app enables you to easily send emails, add work notes, or upload project documents. Team members and project stakeholders always know exactly where things stand with their projects.

Security Analyst

**Manage security incidents anywhere**

Quickly view security incidents and review details, including the incident priority, affected user, configuration item, attacker information, and more. Analysts can search by keyword, and filter by security incidents, users, and vulnerability groups, to find exactly what they are looking for.

**Take action with the tap of a button**

Open IT requests, initiate orchestration tasks like firewall blocks and host isolation, and communicate with affected users, all from within the security incident on your mobile device.

**Review active vulnerabilities on the go**
Analysts can review vulnerability groups assigned to them to determine the next steps; view remediation status, assignment group, individual vulnerable items, and more. Then, create a Change Request and resolve the vulnerability or opt to defer until a later date.

Service desk agent

Manage my work

View new incidents coming in, incidents currently at risk, and even breached incidents on the My Work tab. Agents can easily add comments, reassign, resolve, and approve incidents and tasks. Agents can also view and manage shifts to ensure adequate support coverage. The app includes a visual calendar view that shows your schedule as well as the shifts they cover for other agents. Request shift coverage or even request time off right through the app.

Manage the team’s work

Agents use the My Team dashboard to direct resources no matter where they are. Review unassigned work and take ownership of a critical incident. You can also assign it to a member of their team based on availability for the fastest incident response. Manage the team’s shift coverage using the app to ensure that all-time slots are adequately covered. Agents can use their phone’s native voice-to-text capabilities to provide the input to the Virtual Agent to see all schedules where they may have an upcoming coverage gap.

Engage in major incidents as if you’re in the office

Agents use the Mobile Agent app to engage and respond to major incidents. They can see proposed major incident candidates and quickly review and promote them. Collaborate by kicking off a conference call right from the app to serve as an in-person war room for the team.

For more information on incident management using mobile applications, see Mobile experience with ITSM Mobile agent

Learn more about the Mobile Agent app

Use the following links to learn more about on how to use and configure the Mobile Agent app.

Onboarding Mobile app

Enable your new hires to complete onboarding to-dos, view relevant media sections, view relevant banners, chat with an agent, and more using the Mobile Onboarding app.
Features of the Mobile Onboarding app

The Mobile Onboarding app works with the HR Service Delivery application to prepare your new hires for work. Use the Mobile Onboarding app to:

- Engage new employees before they start.
- Guide to-dos across all departments, including IT and HR.
- Provide your new hires with relevant media, chat options, and more.

Note: For more information, see Base system functionality for ServiceNow mobile.

Download the app

Download the Mobile Onboarding app for Apple iOS or Google Android from the Apple App Store or the Google Play store.

Access the demo

Take a look at the Mobile Onboarding app demo. The demo is a curated experience designed to show how your new hires can prepare for work. For more information on the app, see Onboarding app demo.

Note: To use the Mobile Onboarding app, you must have the HR Service Delivery application installed on your instance. For more detail on this product, see HR Service Delivery

Onboarding app demo

Access the demo in your Mobile Onboarding app to learn how the app can give you a simple, effective new hire experience.
The Mobile Onboarding demo includes curated demonstrations to show how your new hires can perform tasks such as complete onboarding to-dos and view relevant content for new employees.

Access the demo

In the Mobile Onboarding app, tap the **Try with a demo account** button at the bottom of your login screen. Then, tap the **Launch Demo** button to start using the demo. The demo automatically logs in to a ServiceNow instance with an example account.

Demo highlights

The Mobile Onboarding demo displays an applet launcher page divided into four UI sections. Tap an item in any section to open the item, or tap **See All** to view all items in a section. For details on how applet launchers, applets, actions work together in the mobile environment, see *Mobile hierarchy*.

**Welcome Section**

The first section of the Mobile Onboarding demo shows a short introductory video. This video is embedded in a media section within the applet launcher. For information on configuring a media section in your onboarding app, see *Configure a media section for the Mobile Onboarding app*.

**Get Ready**

The **Get Ready** section shows a list of common tasks from across the organization that new employees may complete as part of onboarding. These tasks are lifecycle activities, which are part of the Enterprise Onboarding and Transitions application. For more information on lifecycle activities, see *Configure a lifecycle event activity*.

**Get Connected**

Use the contacts in the **Get Connected** section to view and connect with colleagues using your mobile device’s built in capabilities. You can call, email, or navigate to your contacts using field functions configured on the contact form. You can also call or email or using swipe actions on the contact list when you tap **See All**.

For more information on employee directory applets, see *Configure an employee directory applet*.

For information on configuring functions like those used to make calls and send emails, see *Mobile functions*.

**Get Informed**
The **Get Informed** section illustrates the concept of using targeted mobile campaigns to push relevant information to employees at the right moment. This example uses a vertical item section rather than the mobile content feature available in the Orlando release. The section is configured to use the **Hide header** option, so the **See All** link is unavailable. The section uses a list applet as a destination screen. New hires are directed to a list of records when they tap the section. The data item for the list is defined to return only one item. The pattern (item view) is updated with a custom view for the item.
Tapping the Get Informed section opens the screen shown in the image. This action is achieved using defined a List Item function instance that opens this screen. This screen is a form screen which contains three embedded details screens, Location, Checklist, and FAQs.

The Location displays the Name, Phone, and Website screen fields. These fields use smart functions to open the location in a map, call the phone number and opening the website in an external browser.

The Checklist screen displays two screen fields. The first displays a static value, and the second is a Checklist screen field.

The FAQs screen displays a single HTML screen field to display the FAQ.
Learn more about the Mobile Onboarding app

Use the following links to learn more about on how to use and configure the Mobile Onboarding app.

- Mobile Onboarding app

Considerations before implementation

Use these guidelines and considerations when implementing ServiceNow® mobile on your instance.

Base system functionality for ServiceNow mobile

Learn about the base system functionality included in with ServiceNow® mobile apps.

Base system functionality for ServiceNow Agent

Take advantage of the ServiceNow Agent app by activating the base system applications available from ServiceNow. For a list of plugins available for ServiceNow Agent, see Mobile plugins for ServiceNow Agent.

To use the ServiceNow Agent mobile app, you must install additional plugins, which provide capability such as customer service, asset receiving, or IT service management.
Base system functionality for Now Mobile

Employees can use the Now Mobile app to do these tasks:

Search
Find answers using global search and knowledge base.

Popular services
View and request items from the service catalog.

Popular articles
Search and view company resources and useful articles.

For Me tab
Log in to the app to view your home page. Tap different items to see more information.

Use My To-Dos to see if you have anything on your to-do list. For example, approve an item or finish an onboarding task.

Use My Requests to track the status of your requests and update request details. If one of your requests is taking longer than you expect, you can ask for an update here.

Services tab
View things you've ordered and things you own.

Use My Locker to view the company items that you own, such as your laptop or phone. Go here to report an issue if one of your company items isn't working properly.

Information tab
Search for or browse company articles. Go here to look for a solution to a problem or to learn company information.

Notifications tab
Enable or disable notifications here. Go here to view all of your notifications in one place.

Siri shortcuts
If you're an iOS user, you can use Siri to open some pages in the app. These Siri shortcuts are available:

- Open a chat window.
- Browser items and services.
- Open your tasks.
- Open your requests.
Add functionality to the Now Mobile app by installing plugins. For a list of the available plugins, see Mobile plugins for Now Mobile.
Base system functionality for Mobile Onboarding

New hires can use the Mobile Onboarding app to:

**View HR Requests**

Employees can view their HR requests by navigating to the **For Me** tab and tapping **See All** from the My Requests section. Opening an HR request takes them to the HR ticket page, where they can learn more about the details of their case.

**Request Help**

Employees can submit HR requests by navigating to the **Services** tab and searching or browsing for HR services. They can also tap the plus icon, and then tap **Create HR case** to create a general inquiry HR case.

**Complete HR tasks**

Employees can view and complete their HR tasks, such as signing electronic documents, uploading attachments, and watching videos.

**View relevant banners**

Employees can view relevant banners. See *Configure mobile content* for further information.

**Chat with a virtual agent**

If enabled, employees can chat with a virtual agent for answers to common questions.
Mobile publishing

Publish secure branded mobile applications using your unique company identity and management method.

Use mobile publishing to create a visually distinct version of ServiceNow mobile apps. Tailor your apps to your unique company identity. Configure each of your mobile apps with a unique name, splash screen, icon, security vendor, and theme colors.

The ServiceNow publishing program complies with Google and Apple’s suggested practices for branded app releases and gives you a guided path to achieve your branding goals. Google allows you more flexibility to distribute the app to your employees via your MAM configuration or through any other internal method you choose. The overall process is outlined below.

You can order a branded version of any ServiceNow mobile app through your instance. With branding, you can make the following customizations to your mobile apps:

- App icon
- App name
- App description
- App splash screen
- In app theme colors
- Add MAM (Mobile App Management) support

For more information on this process, see Request and publish a branded mobile app.

Mobile themes

Use mobile themes change the color scheme of your mobile apps. Themes can further enhance the appearance of your apps by matching your color scheme with the splash screen and icons. For more detail on mobile themes see Mobile themes.

Request branded apps

The mobile publishing plugin is a paid plugin that ServiceNow personnel must activate. For more information on requesting mobile branded apps, see Request and publish a branded mobile app.

Distribution

For information on your distribution options, see Mobile distribution.
Request and publish a branded mobile app

Publish secure custom mobile apps with your unique company identity by requesting branded versions of ServiceNow apps.

Role required: admin

To use branded applications, you must have a configured Google Firebase account or Apple Device Enrollment Program (DEP) ID.

To use mobile publishing on any instance, you must have the branding [com.glide.sn-mobile-whitelabel] plugin installed on your instance. The mobile publishing plugin is a paid plugin that can be installed from the ServiceNow Store once your license is executed. For more detail on the ServiceNow Store, see Install a ServiceNow Store application.

Note:

Apple is deprecating their legacy binary protocol, which will affect customers who use custom push notifications. Custom push notifications for customers with mobile branding will stop working as of Nov 2020. This deprecation only affects customers using New York and Orlando instances who have apps customized with Mobile publishing.

- New York customers using mobile publishing customized apps can prevent the push issue by upgrading to New York Patch 10.
- Orlando customers using mobile publishing customized apps can prevent the push issue by upgrading to Orlando Patch 5.

Paris customers are not affected.

For more information, see KB0830082.

1. Navigate to System Mobile > Request Branded App.
2. Click Request Custom Branding.

3. On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Branding Setup form</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Field</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>Platforms</td>
<td></td>
</tr>
<tr>
<td>Deploy app on</td>
<td>Mobile operating systems that you can select for your mobile applications. You may select more than one option to create applications for more than one operating system. Selecting an option reveals the branding setup fields for that operating system.</td>
</tr>
<tr>
<td>Requester’s Contact Information</td>
<td></td>
</tr>
<tr>
<td>First Name</td>
<td>First name for the requester of the branded application</td>
</tr>
<tr>
<td>Last Name</td>
<td>Last name for the requester of the branded application</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Email</td>
<td>Email address that you use to receive notifications about the build status of your request.</td>
</tr>
<tr>
<td>Testers</td>
<td>For iOS app branding, users are required to receive notifications from Apple TestFlight to test the app. Click the Add icon to add a tester for your application. Your testers receive notifications from TestFlight to test the app. For more information on TestFlight, see <a href="https://developer.apple.com/testflight/">https://developer.apple.com/testflight/</a>.</td>
</tr>
<tr>
<td>Note:</td>
<td>To access completed iOS apps, you must use the email address associated to your Apple ID.</td>
</tr>
<tr>
<td>Note:</td>
<td>This field is only applicable when creating an app for Apple iOS.</td>
</tr>
<tr>
<td>Brand Your App</td>
<td></td>
</tr>
<tr>
<td>Long Name</td>
<td>Unique name for your app. The Long Name cannot exceed 30 characters and must be unique. This name displays for your app in the Apple B2B app store.</td>
</tr>
<tr>
<td>Short Name</td>
<td>App icon name. This name appears under the icon on your mobile device. The Short Name cannot exceed 13 characters. This name displays below the app icon on your user's mobile device.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for your app. This description appears in the Apple B2B app store.</td>
</tr>
<tr>
<td>MAM Support</td>
<td>Type of MAM (Mobile App Management) vendor that you want to use with your apps:</td>
</tr>
<tr>
<td>Note:</td>
<td>None: App does not use MAM.</td>
</tr>
<tr>
<td>Note:</td>
<td>BlackBerry Dynamics: Use BlackBerry MAM.</td>
</tr>
<tr>
<td>Note:</td>
<td>Microsoft Intune: Use Intune MAM.</td>
</tr>
<tr>
<td>In-app Color Branding</td>
<td>In-app Color Branding uses the mobile color theme on your instance. For information on configuring a color theme for your app, see <a href="#">Mobile themes</a>.</td>
</tr>
<tr>
<td>iOS Branding Setup</td>
<td></td>
</tr>
<tr>
<td>Apple DEP ID / Organization ID</td>
<td>Volume purchase program ID. For more information, see <a href="#">Branding requirements for mobile applications</a>. For more information on the Apple volume purchase program see <a href="https://volume.itunes.apple.com/">https://volume.itunes.apple.com/</a>.</td>
</tr>
<tr>
<td>Organization Name</td>
<td>Organization name you used for your Apple volume purchase program.</td>
</tr>
<tr>
<td>Note:</td>
<td>This name is linked to your Apple DEP ID. Entering this value incorrectly will result in an unsuccessful build.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>App Icon</strong></td>
<td>App icon for your app. Click <strong>Choose file</strong> to select a local image file to upload as the app icon for your app.</td>
</tr>
<tr>
<td><strong>iOS Splash Screen</strong></td>
<td>Splash screen for your app. Click <strong>Choose file</strong> to select a local image file to upload as the splash screen for your app.</td>
</tr>
<tr>
<td><strong>iOS Splash Screen Background Color</strong></td>
<td>Colored box where you select the splash screen background color for your app.</td>
</tr>
<tr>
<td><strong>Android Branding Setup</strong></td>
<td><strong>Android/Firebase Token</strong> - App ID of the Android app in your Google Firebase project. For more information, see <a href="https://firebase.google.com/">Branding requirements for mobile applications</a>. Your app id should use the following format: <code>com.servicenow.b2b.{app_name_long_lower_case_no_spaces}.{client_type}</code>. For example, if you are creating an Agent app with a short name “Agent” and long name “Acme Agent”, the package name would be <code>com.servicenow.b2b.acmeagent.fulfiller</code>. Note: Each Firebase app ID is unique to the banded app request. If you request multiple apps, each app must have its own unique firebase app id. For more information on Firebase, see <a href="https://firebase.google.com/">https://firebase.google.com/</a>.</td>
</tr>
<tr>
<td><strong>Adaptive Foreground Launcher Icon</strong></td>
<td>Foreground launcher icon for your app. Click <strong>Choose file</strong> to select a local image file to upload as the adaptive foreground launcher icon for your app.</td>
</tr>
<tr>
<td><strong>Adaptive Background Launcher Icon</strong></td>
<td>Background launcher icon for your app. Click <strong>Choose file</strong> to select a local image file to upload as the adaptive background launcher icon for your app.</td>
</tr>
<tr>
<td><strong>Legacy Launcher Icon</strong></td>
<td>Legacy launcher icon for your app. Click <strong>Choose file</strong> to select a local image file to upload as the legacy launcher icon for your app.</td>
</tr>
<tr>
<td><strong>Android Notification Icon</strong></td>
<td>Notification icon for your app. Click <strong>Choose file</strong> to select a local image file to upload as the notification icon for your app.</td>
</tr>
<tr>
<td><strong>Android Splash Screen</strong></td>
<td>Splash screen for your app. Click <strong>Choose file</strong> to select a local image file to upload as the splash screen for your app.</td>
</tr>
<tr>
<td><strong>Android Splash Screen Background Color</strong></td>
<td>Colored box where you select a color for the splash screen background color for your app.</td>
</tr>
</tbody>
</table>

4. **Click Submit.**

You have successfully submitted a request for a branded app. You receive notifications regarding the status of the build at the email address entered in the **Email** field.
Note: You can only submit one build at a time. You cannot submit another build until the current build is canceled, failed, or completed.

ServiceNow begins creating your app. You can see the status of the build by navigating to System Mobile > Request Branded App and viewing the Latest Builds section below each app. Changes to your build status are sent to the email address in the Email Address for Notifications field.

Branding requirements for mobile applications

Android and iOS have specific requirements for custom applications. Understand the branding requirements for iOS and Android mobile apps to ensure that your branded application is ready for distribution these platforms.

iOS distribution requirements

For an iOS deployment, you must have a valid Apple Customer DEP ID. This ID enables ServiceNow to publish an app to your Apple B2B store on your behalf. You can then distribute the branded app to your employees via MDM or through B2B store redemption codes. You receive Apple Customer DEP ID when you sign up for the Apple VPP program. Details on this program can be found at https://www.apple.com/business/it/

iOS image requirements

Images used in iOS branded applications must meet the image requirements defined by Apple. Select images for your icon and splash screens that meet the requirements defined here.

<table>
<thead>
<tr>
<th>Image type</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>App Icon</td>
<td>• File must be in the PNG format.</td>
</tr>
<tr>
<td></td>
<td>• Color space must be sRGB.</td>
</tr>
<tr>
<td></td>
<td>• Layers must be flattened.</td>
</tr>
<tr>
<td></td>
<td>• Image must not have any transparency.</td>
</tr>
<tr>
<td></td>
<td>• Image must be a 1024 px by 1024 px square, with no rounded corners.</td>
</tr>
<tr>
<td>Splash Screen</td>
<td>• File must be in the PNG format.</td>
</tr>
<tr>
<td></td>
<td>• Image size must be 2048 px by 2048 px.</td>
</tr>
<tr>
<td></td>
<td>• Image must contain the company name and logo on a transparent background.</td>
</tr>
<tr>
<td>Splash Screen Background Color</td>
<td>• Must be in hex color format</td>
</tr>
</tbody>
</table>

Android notification requirements

Branded Android apps require a Firebase account to provide push notification support. You must provide the Firebase application ID.
## Android image requirements

Images used in Android branded applications must meet the image requirements defined by Google. Select images for your icons that meet the requirements defined here.

<table>
<thead>
<tr>
<th>Image type</th>
<th>Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adaptive foreground launcher Icon</td>
<td>• File must be in the PNG format.</td>
</tr>
<tr>
<td></td>
<td>• Image size must be 2048 px by 2048 px.</td>
</tr>
<tr>
<td></td>
<td>• Image must be an icon of a company logo with no text. The icon must take up most of the image.</td>
</tr>
<tr>
<td></td>
<td>• Image must be on a transparent background.</td>
</tr>
<tr>
<td>Adaptive background launcher Icon</td>
<td>• File must be in the PNG format.</td>
</tr>
<tr>
<td></td>
<td>• Image size must be 2048 px by 2048 px.</td>
</tr>
<tr>
<td></td>
<td>• Image must have no logo or text.</td>
</tr>
<tr>
<td>Legacy launcher Icon</td>
<td>• File must be in the PNG format.</td>
</tr>
<tr>
<td></td>
<td>• Image size must be 2048 px by 2048 px.</td>
</tr>
<tr>
<td></td>
<td>• Image must have rounded corners.</td>
</tr>
<tr>
<td></td>
<td>• Image must have a solid background.</td>
</tr>
<tr>
<td></td>
<td>• Image must not have any transparency.</td>
</tr>
<tr>
<td></td>
<td>• The icon must take up most of the image.</td>
</tr>
<tr>
<td>Android notification icon</td>
<td>• File must be in the PNG format.</td>
</tr>
<tr>
<td></td>
<td>• Image size must be 2048 px by 2048 px.</td>
</tr>
<tr>
<td></td>
<td>• Image must contain the company logo with no text on a transparent background.</td>
</tr>
<tr>
<td>Splash Screen</td>
<td>• File must be in the PNG format.</td>
</tr>
<tr>
<td></td>
<td>• Image size must be 2048 px by 2048 px.</td>
</tr>
<tr>
<td></td>
<td>• Image must contain the company logo and name on a transparent background.</td>
</tr>
<tr>
<td>Android Splash Screen Background Color</td>
<td>• Must be in hex color format</td>
</tr>
</tbody>
</table>
Ensure that an image or text in your icon are at least 350 pixels from the edge to account for the rounded corners in Android icons.

Images that do not follow this guideline may appear cut off, as shown in this example.

Mobile distribution

Branded apps for Android

The ServiceNow branding program complies with Google’s suggested practice for branded app releases, and gives you a guided path to achieve your branding goals. Google allows you more flexibility to distribute the app to your
employees via your MAM configuration or through any other internal method you choose. The overall process is outlined here.

**Branded apps for iOS**

The ServiceNow branding program complies with Apple's suggested practice for branded app releases and gives you a guided path to achieve your branding goals.

Depending on your distribution model, for an iOS deployment you must have signed up for Apple’s Business Manager to use Apple’s VPP program and/or have a valid Apple Customer DEP ID. This enables ServiceNow to publish an app to your Apple Business Manager on your behalf. Apple then allows you to distribute the app to your employees via MDM or through Apple Business Manager store redemption codes. The overall process is outlined here.

**Building and configuring in branded mobile apps**

Learn about configurations unique to branded mobile apps.

Most app configuration processes for branded apps are the same as non-branded apps. Use the following sections to learn about configuration processes unique to branded mobile apps.

**Push Notifications**

To use push notifications for your branded apps, you must change your push application settings. For the necessary changes for iOS apps, see *Configure push applications for iOS branded apps*. For the necessary changes for Android apps, see *Configure push applications for Android branded apps*.

**Mobile themes**

Use themes on your mobile applications so that your customers and employees can recognize your unique company identity.
Use mobile themes to change the color scheme of your mobile apps. The colors for elements such as headers, links, buttons, and icons can be controlled using themes. You can configure each of your mobile apps with a unique theme.

For details on creating a theme, see "Create a theme for your mobile applications".

To see how your selected theme colors affect the appearance of your mobile apps, see the "Mobile theme color guidelines".

Note: You can configure themes on your apps without custom branding.
Virtual Agent theming

Theming for virtual agent is handled separately from mobile theming. For details on configuring the appearance of Virtual Agent, see *Configure chat branding and the chat menu*.

Create a theme for your mobile applications

Use mobile themes to give your applications a unique appearance that works with your company identity.

Role required: admin

1. Navigate to **System Mobile > Mobile Branding > Client Themes**.
2. Open an existing theme record, or click **New** to create a new theme.
3. On the form, fill in the fields.

### Client theme form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name for your theme.</td>
</tr>
<tr>
<td>Palette</td>
<td>Palette field that consists of four properties. Each property has a name value on the left and a color value in hexadecimal format on the right. Enter or modify the color values to change the color for your theme. For details on how to use these properties in your mobile applications, see the <em>Mobile theme color guidelines</em>.</td>
</tr>
</tbody>
</table>

**Note:** Do not modify the property names. Your instance uses these property names to assign colors to the correct locations in your mobile applications.

4. Click **Submit** or **Update**. Your theme is ready to use on your instance. The next steps cover the process of assigning a theme to the application of your choice.

5. Navigate to **System Mobile > Mobile Publishing > Native Clients**.
6. Open the record for the application where you want to apply your theme.
7. In the **Client Theme** field, select the theme that you created in the previous steps.
8. Click **Update**.

**Mobile theme color guidelines**

Learn how the colors that you select in your mobile themes are applied on your mobile applications.

**Color choice considerations**

Consider the following information when you select colors for your mobile applications:

- How your color choices look with colors in other parts of the app, such as your applet icons and UI styles. Avoid using too many colors. Simple designs of two to three colors result in a cleaner, more accessible layout.
- How your color choices look with colors in other elements that may display within your app. Elements like knowledge articles and service portal pages may appear in your apps. Avoid color schemes that don't work well with your existing elements.
- How higher contrast, complementary colors can guide your user’s focus. Use color selection to highlight areas of your apps that require your user’s attention.
- How color can be used to provide contrast for greater readability. In areas where text appears, ensure that your text and background colors have a high level of contrast to accommodate both low-vision and color-blind users.

**Theme color locations in mobile apps**

This example shows the colors that are used in the default mobile app theme. Use the numbers in this image to see where these colors appear in the different areas of your mobile apps.

<table>
<thead>
<tr>
<th>Color</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Brand color – Gable Green - #293E40</td>
</tr>
<tr>
<td>2</td>
<td>Primary – Tropical Rain - #1F3476</td>
</tr>
<tr>
<td>3</td>
<td>Secondary – Evening Sea - #165C53</td>
</tr>
<tr>
<td>D</td>
<td>Destructive – Mahogany - #C83C36</td>
</tr>
</tbody>
</table>

**Mobile Features**

Use the following figures to see how your color choices affect the elements within your mobile apps.

**Application Launch Page**
Navigation Bar
Hello User

To Do

Complete the following as part of your new hire checklist

Due in 2 days

Requested for Mary Smith

My Stuff

Schedule

Employee Finder

Get Help

List
Virtual Agent
Done

their leave after that period this benefit will be assessed b...

now

Alright, how else can I help? Here's what I can do:

3 weeks ago

It seems you have left the conversation.

Just now

now

How can I help you?

You can type your request below, or use the button to see everything that I can help with.

See all topics

Stuff Send
Form screen and components

Form screen
Checklist
Activity stream
No Recents
See recently opened documents here, or use the Browse tab to see all documents and folders.

Android

Links
E-signature
Applet templates

Employee Directory
Note: The **Pin color** field in the item stream record that is associated to your map applet determines the color of the pins on your maps.

**URL template**
Settings and notifications

Notifications
Settings
Offline mode settings
Geolocation settings
PIN settings
Siri shortcuts settings (iOS Only)
Modals

Android modals
iOS modals
Mobile security

Learn about the security features of the ServiceNow mobile platform.

ServiceNow mobile architecture

ServiceNow mobile apps consist of ServiceNow server instance and native apps for iOS and Android. The apps use full native code and are not a hybrid approach. The mobile apps transmit and receive data with the server across the wireless network.
Overview of key features of ServiceNow mobile platform security

- The mobile apps rely on the secure ServiceNow platform and its APIs to provide a seamless mobile experience to its users.
- App/server interactions are protected through OAuth authorization framework.
- Most of the user interface on the ServiceNow app is driven through meta data delivered by the ServiceNow platform.
- The ServiceNow mobile apps fetch all their data from the ServiceNow platform and store it in a local cache on the app client layer.
- For government community cloud (GCC) ServiceNow instances, locally data stored is encrypted.
  - For iOS apps, ServiceNow uses the OS level FIPS 140-2 validated disk encryption on Core Data, by forcing a device level PIN or Biometrics security.
  - For Android apps, ServiceNow uses the SQLCipher SDK. This SDK provides encryption using FIPS 140-2 validated crypto module for all the app data stored in Room DB.

App flow overview

ServiceNow mobile apps start fetching the initial user experience after a successful sign-in. The mobile app fetches the metadata to render the landing home screen from the instance. The app then uses this metadata to render the home screen.
Data retrieval

Read data

When a user requests to view information on the mobile app, the following steps occur.

1. The mobile app sends a request to access data from the instance. The request includes the token and any relevant data field needed for the request.
2. The instance receives the request and checks if the token is valid.
3. If the token is valid, the instance directs the request to the relevant API to fetch the information.
4. The instance returns the information to the mobile app.

Downloading documents

When a user requests to download documents from the app, the following steps occur.

1. The mobile app sends a request to access the document. The request includes the token.
2. The instance receives the request and checks if the token is valid.
3. The instance checks the access control list (ACL) rules.
4. If valid, the document is available to view.

Write-backs for updating fields

When a user updates a field in the mobile app, the following steps occur.

1. The mobile app sends the token and the action metadata to the instance. For example, the ID, or the field to update.
2. The instance directs the action based on the relevant API.
3. The instance completes the action and sends a response to the mobile app.
4. Based on the response, the mobile app reflects the field changes and action availability in the UI.

Write-backs for attaching files

When attaching files, the following steps occur.

1. The mobile app prompts the user to attach a file, for example, an image.
2. The mobile app sends the file and token to the instance.
3. The instance places the file based on the relevant API.
4. The instance sends a response back to the mobile app.

Mobile authentication

ServiceNow mobile apps support platform authentication using OAuth 2.0. Authentication mechanisms include multi provider SSO, MFA, LDAP, Local DB, and Digest. ServiceNow mobile apps use an authentication methodology called AppAuth. AppAuth uses an external mobile browser to log the user in.

Authentication flow
When a user signs in to an app on their mobile device, the app uses the user's credentials to negotiate an OAuth Token with the instance. The iOS Keychain stores the token for iOS devices. Android devices use KeyStore. The keychain encryption is AES 256 in Galois/Counter Mode (GCM).

At first login, the instance gives the user an access token and a refresh token. These tokens are valid for an amount of time that can be configured on your instance. When a user opens the mobile app, the client checks to see if the access token is valid. If valid, the user can continue with the session. If not valid, the client checks if the refresh token is valid. If valid, the refresh token is used to fetch a new valid access token for the user, and the session can continue. If the refresh token is not valid, the user must reauthenticate.

**Access and refresh tokens**

- The mobile apps never store the user password.
- The mobile apps do store the client ID, which is necessary for getting the OAuth token as part of the authentication flow.

**User termination**

- When an administrator deletes or removes a user from the instance, the access token is no longer valid, and any operation logs the user out.
Multi-provider SSO

The multi-providers SSO plugin [com.snc.integration.sso.multi.installer] provides SAML authentication support. The login process (AppAuth) uses this plugin to redirect the user to the IDP (SAML provider) login page when using SAML.

Multifactor authentication

Users can access the instance via Multifactor Authentication using the MFA plugin [com.snc.integration.multifactor.authentication]. The mobile app directs users to their login page after selecting their instance in the mobile app.

LDAP

Use LDAP authentication to access using LDAP credentials. The user sees the same login page as the local login (DB based), but the back end to the LDAP server deletes the authentication.

Data security

ServiceNow mobile apps use SSL/TLS over-the-air (OTA) communication encryption for data security. The OAuth authorization endpoints are HTTPS.

Data at rest

Application preference data such as favorites, home screen, and the mobile navigator items are stored and cached locally on the device. The mobile apps do not store record data such as incidents and problems on the device unless your organization has specifically enabled offline syncing for field service. Record data stored during offline mode is encrypted with FIPS 140-2 validated modules. (iOS cryptographic modules and SQL Cipher for Android which uses this cryptographic module for encryption).
Data at rest

Data in motion
Data in motion is over a secure SSL/TLS channel and encrypted with FIPS 140-2 validated modules.

Data loss prevention
ServiceNow provides data loss prevention features without the need for the device and application be managed by an enterprise mobility management (EMM) suite. These features includes restrict copy/paste, enforce PIN, block attachment, and/or blur functionality.

Restrict copy/paste
Copy/paste restrictions are defined by a property in the system properties table.

<table>
<thead>
<tr>
<th>System property field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>glide.sg.clear_pasteboard_when_backgrounded</td>
</tr>
<tr>
<td>Type</td>
<td>true</td>
</tr>
<tr>
<td>Value</td>
<td>true</td>
</tr>
</tbody>
</table>

**Require an app PIN**

Require users to enter a six-digit PIN each time they sign in from their mobile device, or when the application has been inactive for five minutes. Requiring an app PIN is controlled a property in the system properties table.

<table>
<thead>
<tr>
<th>System property field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>glide.sg.require_mobile_application_pin</td>
</tr>
<tr>
<td>Type</td>
<td>true</td>
</tr>
<tr>
<td>Value</td>
<td>true</td>
</tr>
</tbody>
</table>

**Disabling attachments on a mobile device**

You can configure ACL rule to block attachments specifically on mobile devices. Use the `isMobile` method to check if a request comes from a mobile device. For example, you could add an ACL rule for the attachment [sys_attachment] table where the read and write scripted ACLs includes the following check.

```java
if( gs.isMobile() ){
    answer = false;
}
```

You can also add this code to any existing ACL rules you have for the attachment table. If you have multiple attachment ACL rules, all will need to have **Admin override** option unchecked.

**Enable the blur app option**

Blur the mobile app when not in focus on a mobile device using the following system property in the system properties table.

<table>
<thead>
<tr>
<th>System property field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>glide.sg.blur_ui_when_backgrounded</td>
</tr>
<tr>
<td>Type</td>
<td>true</td>
</tr>
<tr>
<td>Value</td>
<td>true</td>
</tr>
</tbody>
</table>

**Penetration testing**

ServiceNow engages a third party to perform penetration testing of the mobile app. This typically happens annually but sometimes occurs more frequently. The results of these tests are available to customers. For more detail on security testing, see KB0538598: Customer Instance Security Testing | Policy and Procedure.

**Security patching**

In the event a security patch is needed, the mobile development team aligns with standard SDLC properties in order to patch.

**User data collection**
The mobile app does not specifically collect any user data.

User transactions or usage within the app is tracked on the ServiceNow instance just as it is on the web. For user credentials, after a user logs in, the mobile app negotiates an OAuth Token that is stored in the Apple Keychain or the Android Keystore. User credentials are never saved. If the user opts in, the following information is collected:

- Location
- Access to camera
- Notifications

Device security for ServiceNow Mobile apps

This document applies to current ServiceNow apps for iOS and Android for Paris. This document may be subject to change for future mobile releases.

Components and architecture

The ServiceNow mobile apps consist of the ServiceNow server instance and native apps for iOS and Android. The apps use fully native code and are not a hybrid approach. The mobile client applications communicate over a wireless connection with the server and pull live data for the end user.

Component explanations

Apps for iOS

The ServiceNow apps for iOS are fully native iPhone and iPad applications. The applications can be pulled, dynamically configured, and distributed using MDM (more information available in EMM section). ServiceNow does not currently distribute the ipa file to customers.

Apps for Android

The ServiceNow apps for Android is a fully native applications for Android phone. They can be downloaded from the Google Play store directly by a user, or can be pulled, dynamically configured, and distributed using MDM (more information available in EMM section). ServiceNow only supports apk file distribution to customers for branded versions of Android mobile apps. For details on branded applications, see Mobile publishing.

Identity and access management

Learn about user authentication, third party authentication, and user session termination for mobile applications.

User authentication for ServiceNow mobile apps

ServiceNow mobile apps support platform authentication using OAuth 2.0. Authentication mechanisms include multi provider SSO, MFA, LDAP, Local DB, and Digest.

AppAuth authentication

The ServiceNow mobile apps use a new authentication methodology called AppAuth. AppAuth uses an external mobile browser to log the user in. The following steps detail login using AppAuth.

1. The client creates and records a secret named the code_verifier, and derives a transformed version \( t(code\_verifier) \) (referred to as the code_challenge). This code_challenge is sent in the OAuth 2.0 Authorization Request along with the transformation method \( t\_m \).

2. The Authorization Endpoint responds as usual but records \( t(code\_verifier) \) and the transformation method.
3. The client then sends the authorization code in the Access Token Request as usual but includes the code_verifier secret generated in the previous steps.

4. The authorization server transforms code_verifier and compares it to t(code_verifier) from the previous steps. Access is denied if they are not equal.

Abstract protocol flow

**Single sign-on**

ServiceNow mobile apps require multi-provider single sign-on in order to use external authentication. The multi providers SSO plugin [com.snc.integration.sso.multi.installer] provides SAML authentication support. The login process (AppAuth) uses this plugin to redirect the user to the IDP (SAML provider) login page when using SAML.

For more information on this plugin, see *External single sign-on (SSO).*

For more information on configuring multi provider SSO, see *Single sign on for the ServiceNow Classic mobile app.*

**Multifactor authentication**

Users can access the instance via Multifactor Authentication using the MFA plugin [com.snc.integration.multifactor.authentication]. Users are directed to their login page after selecting their instance in a mobile app.
You will be redirected automatically.

For details on configuring Multi-factor Authentication, see Configure multifactor authentication (MFA)
LDAP

Use LDAP authentication to access using LDAP credentials. The user sees the same login page as the local login (DB based) but the back end to the LDAP server deletes the authentication. For more information on LDAP configuration, see *LDAP integration and authentication*.

Local DB

The user name and password in the user record in the instance database.

Not officially supported

- Kerberos
- Certificate-based authentication (AppAuth’s external browser may solve for some certificate based mechanisms)

Storage/Keychain

When you sign in to an app on your mobile device, the app uses your credentials to negotiate an OAuth Token with the instance. The iOS Keychain stores the token and Android uses KeyStore. The keychain encryption is AES 256 in Galois/Counter Mode (GCM).

The ServiceNow mobile apps never store the user password.

The mobile app does store the Client ID which is necessary for getting the OAuth token as part of the authentication flow.

Access and Refresh Tokens

ServiceNow mobile apps use access and refresh tokens to determine valid user sessions.

At first login, a user is given an access token and a refresh token. These tokens are valid for a configurable amount of time. When the user opens a mobile app, the client checks to see if the access token is valid. If valid, the user is able to continue with the session. If not valid, the client then checks if the refresh token is valid. If valid, the refresh token is used to fetch a new valid access token for the user, and the session can continue. If the refresh token is not valid, the user must re-authenticate.

User termination

When an administrator deletes or removes a user from the system, the Access Token is no longer valid and any operation will log the user out.

Mobile data flow for ServiceNow mobile apps

Data can be retrieved, downloaded from, and written back to a mobile device.

*Retrieval*

The following describes how data is retrieved from ServiceNow mobile apps.

Read data

When a user requests to view information on the mobile app, the following steps occur.

1. The mobile app sends a request to access data from the instance. The request includes the token and any relevant data field needed for the request.
2. The instance receives the request and checks if the Token is valid.
3. If the token is valid, the request is directed to the relevant API to fetch the information.
4. The information is returned to the mobile app.
Downloading documents

When a user requests to download documents from the app, the following steps occur.

1. The mobile app sends a request to access the document. The request includes the Token.
2. The instance receives the request and checks if the Token is valid.
3. If valid, the document becomes available to view or take further actions on the device.

Write-backs
The following describes how data is written back from ServiceNow mobile apps.

Updating fields

When a user updates a field in a mobile app, the following steps occur.

1. The mobile app sends the Token and the action metadata, for example the ID, or the field to be updated, to the instance.
2. The instance directs the action based on the relevant API.
3. The instance completes the action and sends a response to the mobile app.
4. Based on the response, the mobile app reflects the field changes and action availability in the UI.

Attaching documents

When attaching documents, the following steps occur.

1. The mobile app asks the user to attach a document, for example, an image.
2. The mobile app sends the document and Token to the instance.
3. The instance places the document based on the relevant API.
4. The instance sends a response back to the mobile app.

Internal mobile app distribution

Internal distribution of ServiceNow mobile apps is supported through all major EMM vendors. Customers are able to pull the app for iOS or Android from the Apple App store and Google Play respectively, dynamically configure the apps to point to the correct ServiceNow instance, and distribute using the EMM hub. This way, the MDM can fully manage the app as part of a customer portfolio.

Note: ServiceNow does not currently distribute the ipa files, or any other unpublished app to customers as it breaches the Apple Enterprise Developer License Agreement.

Mobile app distribution providers:
- AirWatch Mobile Device Management (MDM)
- BlackBerry Unified Endpoint Management (UEM)
- Intune mobile device management (MDM)
- IBM MaaS360 mobile device management (MDM)
- MobileIron Mobile Device Management (MDM)
Data security for ServiceNow mobile apps

ServiceNow mobile apps use SSL/TLS for Over-the-Air (OTA) communication encryption for data security. The OAuth authorization endpoints are HTTPS.

Data stored in your mobile apps

Application preference data such as favorites, home screen, and the mobile navigator items are stored and cached locally on the mobile device. ServiceNow mobile apps do not store record data such as incidents and problems on the device unless your organization has specifically enabled offline syncing for Field Service. The record data is encrypted with AES 128.

Information stored in mobile apps

- Databases
  - User defined instances
  - Favorite application IDs
  - Push Notifications
  - Geolocation updates
  - Offline data
- Preferences stored in mobile apps
  - sys_id, display name, username, and initials of the current user
  - URL and name of the current instance
  - Last activity timestamp
  - Encrypted PIN code
  - Offline cache warning period
  - Server Properties
    - LOCATION_PROXIMITY
    - IS_PIN_CODE_REQUIRED
    - IS_BLURRED_IN_BACKGROUND
    - IS_BLOCK_ATTACHMENT_SHARING
    - LOCATION_TRACKED
    - IS_CLEARING_CLIPBOARD_IN_BACKGROUND
    - IS_HIDE_APPLICATIONS_SCREEN_IMAGE
    - IS_OFFLINE_ENABLED
    - LOCATION_FREQUENCY
  - key_analytics_initial_app_launch flag
- Information stored in the system Account Manager
  - Login date
  - Instance URL
  - Access Token
  - Refresh Token

Data in motion

Data in motion is over a secure SSL/TLS channel and encrypted with HTTPS.
Offline access and data cache configuration

Choose specific applets and actions to be enabled offline from with Studio. On the mobile device, your users can select offline and choose to “cache data” from Settings. The offline flows that you designate are downloaded and cached to the device.

You can encrypt offline cached data by using native encryption. This encryption expires at a specified period of time. The default is 48 hours or when a user signs out of the mobile app.

Offline data is protected by local-auth and the app PIN that can be optionally enabled by administrators. When enabled, users are required to enter a PIN on login, or when the application is inactive for five minutes.

Disabling mobile attachments

You can disable attachments for mobile apps by using access control rules. For more details on this process see Disable attachments in mobile apps.

Push notifications

Administrators create push notifications and users are able to receive them.

Cloud

For more information on the push notification system including process, configuration, and architecture, see Push notification system. Administrators can configure push notification delays using scheduled jobs. To view an example included with the base system, navigate to System Scheduler > Scheduled jobs, then search for a job with the name Push. 5 seconds is the minimum time allowed for the push delay.

Mobile security practices

Mobile security practices include mobile-specific system properties, attachment control, password reinforcement, security patching, and controlling shared data.

Security controls
Configure security controls to restrict copy/paste, enforce PIN, or block attachment functionality.

Restrict copy/paste

Copy/paste restrictions are defined in the system properties [sys_properties] table. There are two applicable security properties.

- glide.sg.clear_pasteboard_when_background: Clears the copy/paste clipboard when a ServiceNow app enters the background. For more information on clearing the clipboard, see Configure clear clipboard

Require an app PIN

Require users to enter a six digit PIN each time they sign in from their mobile device, or when an app has been inactive for five minutes. Requiring an app PIN is controlled by the glide.sg.require_mobile_application_pin system property. For more information on requiring an app PIN, see Require an app PIN for the mobile app.
Disabling attachments on a mobile device

Use an ACL to block specific access on mobile. Use the `isMobile` method to check if a request comes from a mobile device. For example, you could add an ACL for the attachment [sys_attachment] table where the read and write scripted ACLs includes the following check. You can also add this code to any existing ACLs you have for the attachment table. If have multiple attachment ACLs, all of the need to have `Admin override` option unchecked.

```java
if( gs.isMobile() ) {
    answer = false;
}
```

Note: You need elevated privileges to create ACLs.

Enable the blur app option

Blur a mobile app when not in focus on a mobile device using the following system property in the system properties [sys_properties] table.

- `glide.sg.blur_ui_when_backgrounded`

For more information, see Configure the blur app option

Penetration testing

ServiceNow engages a third party to perform penetration testing of a mobile app. This typically happens annually but sometimes occurs more frequently. The results of these tests are available to customers on CORE. Customers can test the mobile application in conjunction with a pen test of their instance per the process outlined in KB0538598.

Security patching

In the event a security patch is needed, the mobile development team aligns with standard SDLC properties in order to patch.

User data collection

ServiceNow mobile apps do not specifically collect any user data.

Any user transactions or usage within an app is tracked on the ServiceNow instance just as it is on the web. For user credentials, after a user logs in, the mobile app negotiates an OAuth Token that is stored in the Apple Keychain or the Android Keystore. User credentials are never saved. If the user opts in, the following information is collected:

- Location
- Access to camera
- Notifications

Shared data

ServiceNow mobile apps communicate with a third party software for app crash reporting. No customer information is shared.

- iOS:
  - Uses Crashlytics for crash reporting.
- Android:
  - Uses Crashlytics for crash reporting.
Incident reporting
ServiceNow mobile app issues should be reported through the standard support channels. You can report incidents by contacting ServiceNow Technical Support.

Edge Encryption for ServiceNow mobile

Users can view and edit data protected with Edge Encryption within their mobile device. The data appears in readable form on the mobile device but is encrypted in the database.

ServiceNow® Edge Encryption encrypts sensitive data on your company premises before sending it over the Internet to your instance (encrypted in flight). This encrypted data is then stored and protected within your database (encrypted at rest).

The diagram shows an example of how Edge Encryption operates within the mobile platform. A field is configured and protected with Edge Encryption. When the user enters a value in an encrypted field on the mobile device, it remains in a readable format. However, in the database, the value is displayed as an encrypted value.
For more information, see *Edge Encryption*.

**Mobile GovCommunityCloud (GCC) Compliance**

Learn about how ServiceNow mobile apps comply with security standards for the GovCommunityCloud environment.
ServiceNow GovCommunityCloud (GCC) compliance is designed for U.S. Federal, State, and local government customers. This environment is FedRAMP High and DoD Impact Level 4 authorized and compliant. Each ServiceNow mobile app (Now Mobile, Mobile Agent, and Mobile Onboarding) use FIPS 140-2 certified encryption modules and are GCC-compliant.

As part of this accreditation, ServiceNow mobile apps include the following:

**Encryption**
ServiceNow uses FIPS 140-2 validated encryption when connecting to GovCommunityCloud instances.

**Enforced security feature enablement**
Enforced device PIN or biometric enablement when connecting to GovCommunityCloud instances.

**Encryption for locally stored data**
Locally stored app data such as user preferences and offline data are encrypted.

**Blur feature**
The blur feature is automatically enabled when the app is in the background.

### iOS FIPS 140-2 Compliance

- On iOS devices, ServiceNow mobile apps use Apple's validated cryptographic modules. These modules are available on all devices using iOS 11 and up.
- To enforce iOS FIPS 140-2 encryption, the ServiceNow mobile apps require that a user’s device has a pass code enabled when connecting to a GCC instance.
- All locally stored mobile app data such as user preferences and offline data use FIPS 140-2 validated encryption when pass code enablement is confirmed.

For more information on Apple's validated cryptographic modules, see [https://support.apple.com/en-us/HT202739](https://support.apple.com/en-us/HT202739)

### Android FIPS 140-2 Compliance

- On Android devices, ServiceNow mobile apps are integrated with a third party SDK that uses a FIPS 140-2 validated certificate.
- With this SDK, Android versions of ServiceNow mobile apps are FIPS 140-2 compliant for data at rest. All locally stored app data such as user preferences and offline data use the same level of encryption.
- ServiceNow mobile apps also require that a device pass code is enabled when a user connects to a GCC instance.
Note: This feature requires Android version 7.0 Nougat and up.

For more information on the certificate used with the third party SDK, see https://csrc.nist.gov/projects/cryptographic-module-validation-program/Certificate/1747

Mobile system properties related to GCC compliance

Enforcing FIPS 140-2 Encryption

Use the glide_sg.device_encryption_enabled system property to enforce encryption and require that a device pass code is configured. This system property is automatically added and defaults to true for GCC instances. For non-GCC instances, this property defaults to false. Enable this property on these instances to take advantage of encryption and device pass code enablement.

Disabling offline mode

On GCC instances, offline mode is disabled by default when the offline mode plugin is installed. To enable offline mode on a GCC instance, an administrator must create the glide_sg.offline.enabled system property on the [sys_properties] table, and set the value of this property to true.

For non-GCC commercial instances, offline mode is enabled by default when the offline mode plugin is installed. To disable offline mode on a non-GCC instance, an administrator must create the glide_sg.offline.enabled system property on the [sys_properties] table, and set the value of this property to false.

For more information on offline mode, see Offline mode.

Screen blur on background

Use the glide_sg.blur_ui_when_backgrounded system property to blur the app screen when in background. This property was introduced in the Madrid release. Beginning in the Paris release, this property is automatically enabled with a value of true.

Note: This property is not overridden for existing customers who upgrade to the Paris release.

FedRAMP

The Federal Risk and Authorization Management Program (FedRAMP) creates a set of processes to ensure cloud security for the government. For more detail on this program, see https://www.fedramp.gov/.

Enterprise mobility management (EMM)

Use an EMM suite to distribute ServiceNow mobile apps or the ServiceNow Classic mobile app.

Enterprise mobility management (EMM)

Apply your corporate app protection policies to ServiceNow mobile apps by either using your EMM suite, or, with an embedded mobile application management (MAM) SDK for personal devices. ServiceNow only supports Intune and BlackBerry SDKs.
AppConfig

AppConfig is a standard approach for configuring mobile apps using key-value pairs. AppConfig was created by leading EMM providers like MobileIron, SAP, IBM, and VMWare. For more information on application configuration, please read your MDM product documentation. ServiceNow supports two app configurations:

- Pre-configure the default instance
- Change the default browser

For details on these configurations, see AppConfig for Mobile Applications.

Mobile application management (MAM) integration

Use Microsoft Intune or BlackBerry Dynamics to secure and protect sensitive information in mobile applications, even in cases where customers use their own mobile devices. For more detail on MAM integration, see Mobile application management (MAM) integration.

Unsupported MDM/MAM features

The following MDM and MAM features are not supported by ServiceNow.

- Application distribution
- App protection policies
- GA (general availability) binary files (ipa and apk) will not be distributed.
- App wrapping and/or resigning are not allowed.
- Troubleshooting mobile gateway or proxy rules.
- App Config is supported for on-prem MDM but creating plist file is not supported.

Mobile application management (MAM) integration

Use Microsoft Intune or BlackBerry Dynamics to secure and protect sensitive information in mobile applications, even in cases where customers use their own mobile devices.

Use Microsoft Intune or BlackBerry Dynamics mobile management tools to control, secure, and enforce policies for ServiceNow mobile apps. These tools provide a central point of control for securing your data on mobile apps, even in scenarios where you are not the owner of the mobile device.

These apps are available in the Apple App store, Google Play, and BlackBerry marketplace stores. Get started right away using these publicly available apps, or request custom branded versions of these apps with your unique company identity.

Available apps by version

<table>
<thead>
<tr>
<th>MAM Platform</th>
<th>Mobile Agent Availability</th>
<th>Now Mobile Availability</th>
<th>Mobile Onboarding Availability</th>
</tr>
</thead>
</table>
Microsoft Intune

Take advantage of the policy enforcement and management features of Microsoft Intune for your ServiceNow mobile apps.

ServiceNow provides versions of the Mobile Agent, Now Mobile, and Mobile Onboarding apps designed to integrate with your Microsoft Intune mobility management tools.

Using these apps, you can set and manage policies for each app using your organizations Intune administration portal. The following mobile application management features are supported:

- App protection without device enrollment
- Restrict copy/paste
- Attachment Control
- Remote wipe
- MSFT conditional access

BlackBerry Dynamics

Manage your ServiceNow mobile apps using your organization's BlackBerry Dynamics mobility management tools.

ServiceNow provides versions of the Mobile Agent, Now Mobile, and Mobile Onboarding apps designed to integrate with your BlackBerry Dynamics mobility management tools. BlackBerry versions of ServiceNow are available in the BlackBerry marketplace store.
Using these apps, you can set and manage policies for each app using the BlackBerry Dynamics SDK. The following mobile application management features are supported:

Your branded apps are integrated with the BlackBerry Dynamics SDK to help manage policy enforcement and employee access to your data. The following mobile application management features are supported:

- App protection without device enrollment
- Restrict copy/paste
- Attachment Control
- Remote wipe
- Dynamic VPN

**Package and Bundle IDs**

To enforce app policies your mobile application management software may require the bundle or package ID associated with the mobile apps. For information on these IDs see *Bundle and Package IDs for ServiceNow mobile apps*.

**Bundle and Package IDs for ServiceNow mobile apps**

To enforce app policies your mobile application management software may require the bundle or package ID associated with the mobile app. See the following reference for these IDs.
Mobile app bundle and package IDs for Microsoft Intune

iOS bundle IDs

<table>
<thead>
<tr>
<th>App</th>
<th>Bundle ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceNow® Agent - Intune</td>
<td>com.servicenow.intune.fulfiller</td>
</tr>
<tr>
<td>Now® Mobile - Intune</td>
<td>com.servicenow.intune.requestor</td>
</tr>
<tr>
<td>ServiceNow® Onboarding - Intune</td>
<td>com.servicenow.intune.onboarding</td>
</tr>
</tbody>
</table>

Android package IDs

<table>
<thead>
<tr>
<th>App</th>
<th>Package ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceNow® Agent - Intune</td>
<td>com.servicenow.fulfiller.mam.intune</td>
</tr>
<tr>
<td>Now® Mobile - Intune</td>
<td>com.servicenow.requestor.mam.intune</td>
</tr>
<tr>
<td>ServiceNow® Onboarding - Intune</td>
<td>com.servicenow.onboarding.mam.intune</td>
</tr>
</tbody>
</table>

Mobile app bundle and package IDs for BlackBerry Dynamics

iOS bundle IDs for BlackBerry Dynamics

<table>
<thead>
<tr>
<th>App</th>
<th>Bundle ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceNow® Agent - BlackBerry</td>
<td>com.servicenow.blackberry.fulfiller</td>
</tr>
<tr>
<td>Now® Mobile - BlackBerry</td>
<td>com.servicenow.blackberry.requestor</td>
</tr>
</tbody>
</table>

Android package IDs for BlackBerry Dynamics

<table>
<thead>
<tr>
<th>App</th>
<th>Package ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceNow® Agent - BlackBerry</td>
<td>com.servicenow.fulfiller.mam.blackberry</td>
</tr>
<tr>
<td>Now® Mobile - BlackBerry</td>
<td>com.servicenow.requestor.mam.blackberry</td>
</tr>
</tbody>
</table>

AppConfig for Mobile Applications

Pre-configure the default instance

When you distribute ServiceNow mobile apps through an EMM suite or an embedded MAM SDK app, you can use AppConfig to pre-configure the default instance URL.
### Change the default browser

Because ServiceNow uses AppAuth for authentication, the mobile apps use the default browser on the mobile device. iOS uses Safari, while Android uses Chrome. You may have browser security requirement where their app protection policy only allows their MDM managed browser or a specific browser. A common use case is the support for per-app VPN.

<table>
<thead>
<tr>
<th>Platform</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>iOS</td>
<td>iOS Authentication Browser</td>
<td>Safari Apple Safari</td>
</tr>
<tr>
<td></td>
<td>Chrome</td>
<td>Google Chrome</td>
</tr>
<tr>
<td></td>
<td>Firefox</td>
<td>Mozilla Firefox</td>
</tr>
<tr>
<td></td>
<td>Edge</td>
<td>Microsoft</td>
</tr>
<tr>
<td></td>
<td>WorkspaceONE</td>
<td>AirWatch VMWare</td>
</tr>
<tr>
<td></td>
<td>WebAtWork</td>
<td>MobileIron Web@Work</td>
</tr>
<tr>
<td></td>
<td>BlackBerry Access</td>
<td>BlackBerry Access</td>
</tr>
<tr>
<td></td>
<td>MaaS360</td>
<td>IBM MaaS360</td>
</tr>
<tr>
<td>Android</td>
<td>Android Authentication Browser</td>
<td>Chrome Google Chrome</td>
</tr>
<tr>
<td></td>
<td>Firefox</td>
<td>Mozilla Firefox</td>
</tr>
<tr>
<td></td>
<td>Samsung</td>
<td>Samsung Internet Browser</td>
</tr>
<tr>
<td></td>
<td>WorkspaceONE</td>
<td>AirWatch VMWare</td>
</tr>
<tr>
<td></td>
<td>WebAtWork</td>
<td>MobileIron Web@Work</td>
</tr>
<tr>
<td></td>
<td>BlackBerry Access</td>
<td>BlackBerry Access</td>
</tr>
<tr>
<td></td>
<td>MaaS360</td>
<td>IBM MaaS360</td>
</tr>
</tbody>
</table>

Note:
- Keys are case sensitive.
- AppConfig key-values are still supported for non-managed devices if the user installs a ServiceNow app with a MAM SDK.

### Supported mobile device management vendors

Learn how to configure mobile device management software from supported vendors to distribute ServiceNow mobile apps.
AirWatch Mobile Device Management (MDM)

Use VMware AirWatch to distribute and manage the ServiceNow mobile app on user devices.

Use the following steps to configure the ServiceNow mobile app for iOS and Android.

You should have access to AirWatch before attempting any configuration.

Add the ServiceNow app for iOS to AirWatch

Configure the ServiceNow app for iOS for AirWatch distribution.

Role required: admin

1. Sign in to the AirWatch portal.
3. Click Add Application.
4. In the Platform field, select Apple iOS.
5. In the Source field, select Search App Store then search for ServiceNow.
6. Next to the ServiceNow App search result, click Select.
7. Click Save & Assign.
8. Click Add Assignment.
9. Click Assignment Group.
10. From the Assignment Group list, select Application Configuration Policy.
11. From the Application Configuration field, select Enabled.
12. In the Application Configuration section, enter the following information.

Application Configuration fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration key</td>
<td>SNDefaultInstanceURL</td>
</tr>
<tr>
<td>Value type</td>
<td>String</td>
</tr>
<tr>
<td>Configuration value</td>
<td>https://&lt;instance name&gt;.service-now.com/</td>
</tr>
</tbody>
</table>
Would you like to enable Data Loss Prevention (DLP)?

DLP policies provide controlled exchange of data between managed and unmanaged applications on the device.

To prevent data loss on this application, make it "Managed Access" and create "Restriction" profile policies for desired device types.

- **Managed Access**: Enabled
- **Remove On Unenroll**: Enabled
- **Prevent Application Backup**: Enabled
- **Make App MDM Managed if User Installed**: Enabled
- **App Tunneling**: Enabled
- **Application Configuration**: Enabled

Enter Key-Value pairs to configure applications for users:

<table>
<thead>
<tr>
<th>Configuration Key</th>
<th>Value Type</th>
<th>Configuration Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNDefaultInstanceURL</td>
<td>String</td>
<td>https://&lt;instance name&gt;.aervil</td>
</tr>
</tbody>
</table>

Add

Add

Cancel
13. Click Add.
14. Click Save & Publish.
15. Click Publish.

After adding the app to AirWatch, it appears in the Apps & Books > Applications > Native > Public section of the AirWatch portal. Mobile users can download the app from the AirWatch App Catalog app.

Add the ServiceNow app for Android to AirWatch

Configure the ServiceNow app for Android for AirWatch distribution.

Role required: admin

1. Sign in to the AirWatch portal.
3. Click Add Application.
4. In the Platform field, select Android.
5. In the Source field, select Search App Store then search for ServiceNow.
   If the search cannot find the ServiceNow application, the app might not be approved for your organization. You can approve the app in the Google Play store.
6. Next to the ServiceNow App search result, click Select.
7. Click Save & Assign.
8. Click Add Assignment.
9. Click Assignment Group.
10. From the Assignment Group list, select Application Configuration Policy.
11. From the Application Configuration field, select Enabled.
12. In the Application Configuration section, enter the following information.

Application Configuration fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Server URL</td>
</tr>
<tr>
<td>Description</td>
<td>The full ServiceNow instance URL</td>
</tr>
<tr>
<td>Value</td>
<td>https://&lt;instance name&gt;.service-now.com/</td>
</tr>
</tbody>
</table>
### ServiceNow - Add Assignment

**Select Assignment Groups**
- All Devices (ServiceNow, Inc.)

**App Delivery Method**
- Auto
- On Demand

**Policies**

- **Adaptive Management Level:** Managed Access

  - Apply policies that give users access to apps based on administrative management of devices.

---

**Would you like to enable Data Loss Prevention (DLP)?**

DLP policies provide controlled exchange of data between managed and unmanaged applications on the device.

To prevent data loss on this application, make it "Managed Access" and create "Restriction" profile policies for desired device types.

- **Managed Access:**
  - Enabled
  - Disabled

- **App Tunneling:**
  - Enabled
  - Disabled

- **Application Configuration:**
  - Enabled
  - Disabled

**Enter Key-Value pairs to configure applications for users:**

<table>
<thead>
<tr>
<th>Title</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server URL</td>
<td>The full ServiceNOW Instance URL</td>
<td>https://&lt;instance name&gt;.service-now</td>
</tr>
</tbody>
</table>
13. Click Add.
14. Click Save & Publish.
15. Click Publish.

After adding the app to AirWatch, it appears in the Apps & Books > Applications > Native > Public section of the AirWatch portal. Mobile users can download the app from the AirWatch App Catalog app.

If your instance is on an internal network, you might need to configure the VPN. For more information on configuring the VPN, see the AirWatch documentation on configuring per-app tunnel profiles.

For more information on app distribution with AirWatch, see the official AirWatch documentation.

**BlackBerry Unified Endpoint Management (UEM)**

Use BlackBerry UEM (Unified Endpoint Management) as a mobile device management system to distribute and manage the ServiceNow mobile app on user devices.

**Get the BlackBerry UEM registered versions of ServiceNow mobile apps**

ServiceNow has BlackBerry integrated versions of the Now Mobile and Mobile Agent apps. These apps integrate with the BlackBerry UEM portal.

These apps are available in the BlackBerry Marketplace at [https://marketplace.blackberry.com/apps](https://marketplace.blackberry.com/apps)

**Managing ServiceNow mobile apps with BlackBerry UEM**

Learn how to manage ServiceNow mobile applications in the BlackBerry UEM (Unified Endpoint Management).

Role required: admin

1. Sign into the BlackBerry UEM.
2. In the menu bar to the left, click **Apps**.
You see a list of available apps tied to your BlackBerry UEM.
3. In the search bar above the list of apps, type ServiceNow.

4. Click the app you want to register.

5. When the app opens, click on the blue text labeled App Config With Default Values to create the app configuration for this app.
6. In the app configuration form, fill in the fields as needed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Instance URL</td>
<td>Your instance URL. For example, <code>https://INSTANCENAME.service-now.com</code>.</td>
</tr>
<tr>
<td>iOS Authentication Browser</td>
<td>The browser your app will use for authentication on iOS devices.</td>
</tr>
<tr>
<td>Android Authentication Browser</td>
<td>The browser your app will use for authentication on Android devices.</td>
</tr>
</tbody>
</table>
7. Click Save to save the app configuration.
8. Click Save to save the app.

**Note:** If using the instructions above to setup the BlackBerry Access browser as your authentication browser, you must also have an app configuration setup for the BlackBerry Access app. This is necessary to successfully authenticate to ServiceNow integrated apps.

*Configure BlackBerry Access for mobile apps*

Configure the BlackBerry Access browser to authenticate ServiceNow mobile apps.

Role required: admin

The following steps are only necessary if you are using the BlackBerry Access browser as the authentication browser for your ServiceNow mobile apps. The BlackBerry Access browser must be configured in order to authenticate with your ServiceNow mobile apps.

1. Sign into the BlackBerry UEM.
2. In the menu bar to the left, click Apps.
You see a list of available apps tied to your BlackBerry UEM.
3. In the search bar above the list of apps, type **BlackBerry Access**.

4. Click **BlackBerry Access** to open the app.

5. When the app opens, click on the blue text labeled **App Config With Default Values** to create the app configuration for this app.
6. Enable **Allow external apps** to open HTTP/HTTPS URLs through BlackBerry Access.

7. Enable **Enable 3rd Part Applications**.

8. Add these values: `snappauth`, `snempappauth`

   **Note:** Do not use any spaces in the values above.

   **Note:** The `snappauth` is required for the Mobile Agent app. The `snempappauth` value is required for the Now Mobile app.

9. Click **Save**.
Citrix XenMobile

Configure Citrix XenMobile to distribute and manage the ServiceNow mobile app on user devices.

Use the following steps to distribute the ServiceNow mobile app using the Citrix XenMobile service. You can also configure the mobile app for your default ServiceNow instance for the app.

You should have access to Citrix before attempting any configuration.

*Add the ServiceNow app to Citrix XenMobile*

Configure the ServiceNow for Citrix XenMobile distribution.

Role required: admin

1. Sign in to the XenMobile console.
2. Click **Configure > Apps**.
3. Click **Add**, then choose **Public App Store**.
4. In the App Information pane, complete the following information.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Type a descriptive name for the app. The name appears under App Name on the apps table.</td>
</tr>
<tr>
<td>Description</td>
<td>Type an optional description of the app.</td>
</tr>
<tr>
<td>App category</td>
<td>In the list, click the category you want to add the app to.</td>
</tr>
</tbody>
</table>

5. Click **Next**.
6. Under Platform, select the following platforms, or the platforms that are applicable to your distribution plan.
   - iPhone
   - iPad
   - Android for Work

7. For each of the platforms, complete the following steps.
   a) In the search bar, search for ServiceNow.
   b) Click the ServiceNow tile.
      The App Details fields are pre-populated with information related to the app.
   c) Configure the deployment rule depending on your organizations deployment plan.
   d) Click **Next**.

If Android for Work cannot find the ServiceNow application, the app might not be approved for your organization. You can approve the app in the Google Play store.

If the search results still do not display in Citrix, click **Didn't find the app you were looking for?** then enter the URL for the app: https://play.google.com/work/apps/details?id=com.servicenow.servicenow.

8. Click **Next**.
9. Optional: From the Approvals page, configure the workflow for your company.
   
   You only need to use workflows when you need approval when creating user accounts. If you do not need to set up approval workflows, skip to the next step. For more information on configure workflows for Citrix, see the **Citrix product documentation**.

10. From the Delivery Group Assignments page, select the delivery groups for users that you want to have the ServiceNow application. You can also configure a deployment schedule.

After successfully adding the ServiceNow app from the iOS store platforms, install the ServiceNow application from the Citrix Hub app from your devices. For Android, the ServiceNow app appears in the Google Play Store managed by your Android Work Profile.
Configure the ServiceNow app for the default instance
Configure apps that support managed configuration by deploying an XML configuration file to users' iOS devices.

Configuring the app for a default instance is only available for iOS devices.

1. In the XenMobile console, click Configure > Device Policies.
2. Click Add.
3. In the Add a New Policy page, under Apps, click App Configuration.
4. On the Policy Information page, enter the following information.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Name</td>
<td>Type a descriptive name for the policy.</td>
</tr>
<tr>
<td>Description</td>
<td>Optionally, type a description of the policy</td>
</tr>
</tbody>
</table>

5. Click Next.
6. In the Policy Platforms panel, select iOS.
7. In the Identifier list, select Add new.
8. Enter the following in the add new identifier field: com.servicenow.servicenow.
9. In the Dictionary content field, enter the following text.

```xml
<dict>
  <key>SNDefaultInstanceURL</key>
  <string>https://<instance_name>.service-now.com</string>
</dict>
```

Use the name of your own instance where it says <instance_name>.

10. Click Check Dictionary to verify the XML.
12. Click Next.
13. From the Delivery Group Assignments page, select the delivery groups for users that you want to have the ServiceNow application. You can also configure a deployment schedule.
14. Click Save.

IBM MaaS360 mobile device management (MDM)

Use IBM MaaS360 to monitor and manage smartphones, tablets, and other mobile devices from a web-based portal. Add the ServiceNow mobile app to the IBM MaaS360 portal to monitor and manage distribution.

Before adding the ServiceNow app to the IBM MaaS360 portal, complete the IBM MaaS360 setup procedures.

Add the ServiceNow app for iOS to the IBM MaaS360 portal
Configure the ServiceNow app for iOS for IBM MaaS360 distribution.

Role required: admin

1. Sign in to the IBM MaaS360 portal.
2. From the portal homepage, navigate to Apps > Catalog.
3. Click Add to display the list of apps.
5. In the App Details tab, type ServiceNow into the app field.
6. Select the ServiceNow app from the results.
7. In the Policies and Distribution tab, select the applicable policies.
8. In the Configuration tab, in the App Config Source list, select **Key/Value**.
9. In the Key/Value section add the following information.

   **Key/Value configuration**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute name</td>
<td>SNDefaultInstanceURL</td>
</tr>
<tr>
<td>Attribute value</td>
<td>The full URL for the instance. For example,</td>
</tr>
<tr>
<td></td>
<td>https://&lt;instance name&gt;.service-now.com/</td>
</tr>
</tbody>
</table>

10. Click **Add**.

After adding the ServiceNow app from the Apple store, it appears in the App Catalog application. Users can download the app to their iOS devices.

*Add the ServiceNow app for Android to the IBM MaaS360 portal*

Configure the ServiceNow app for Android for IBM MaaS360 distribution.

Role required: admin

1. Sign in to the IBM MaaS360 portal.
2. From the portal homepage, navigate to **Apps > Catalog**.
3. Click **Add** to display the list of apps.
5. In the App Details tab, type ServiceNow into the app field.
6. Select the ServiceNow app from the results.
7. In the Policies and Distribution tab, select the applicable policies.
8. In the Configuration tab, select the **Configure App Settings** checkbox.
9. In the Server URL field, enter the full URL for the instance. For example, https://<instance name>.service-now.com/.

10. Click Add.

After adding the ServiceNow app from the Google Play store, it appears in the App Catalog application. Users can download the app to their Android devices.

If your instance is on an internal network, you might need to configure the VPN. For more information on configuring the VPN, see the IBM MaaS360 documentation on the Maas360 VPN module.

For more information on app distribution, see the following IBM documentation for each operating system.

- Add an iTunes App Store app to the App Catalog
- Add a Google Play app to the App Catalog

**Intune mobile device management (MDM)**

Microsoft Intune uses Azure to manage mobile devices and apps. With some additional configuration, you can manage the ServiceNow mobile app in Intune.

You need to have access to an Azure account in order to add the ServiceNow mobile app to the store. Complete the Intune configuration steps before adding any apps to the Intune portal.

*Add an Intune integrated app to Microsoft Azure*

Configure your ServiceNow app for distribution with Microsoft Intune.

Role required: admin

1. Sign in to the Microsoft Azure portal.
2. In the search bar, enter Client Apps.

![Welcome to Azure!](image)

3. In the navigator menu on the left side of the screen, navigate to Manage > Apps.
4. Click Add.

5. Select the App type.

- **Note:** Both iOS and Android store apps will go through the same process. When adding an iOS app, Azure offers a shortcut for you to find the app on the Apple App Store seen here:
If you are adding an Android store app, there is no shortcut to add in the apps. Android apps are found in the following locations.

- Mobile Agent Intune can be found here.
- Now Mobile Intune can be found here.
- Mobile Onboarding Intune can be found here.

6. Click Next.
7. Add the specific group or all users who will be using this app.

8. Click Next.
9. Review the information you entered. When you are finished, click Create.

Add an Intune integrated apps to a protection policy in Microsoft Azure
Learn how to add your ServiceNow mobile apps to your existing Microsoft Azure protection policies.

Role required: admin
The following steps describe how to add your Intune integrated apps to an existing protection policy. For information on creating protection policies, refer to your Microsoft Azure documentation.

1. Sign in to the Microsoft Azure portal.
2. In the search bar, enter Client Apps.
3. In the navigator menu on the left side of the screen, navigate to Manage > App protection policies.
Home  >  Client apps

Client apps
Microsoft Intune

Search (Cmd+/)

Overview

Manage

Apps

App protection policies

App configuration policies

App selective wipe

iOS app provisioning profiles

S mode supplemental policies
4. Open the app protection policy where you want to add your apps.
5. In the left navigator, open Manage > Properties.
Intune App Protection
iOS Policy

Manage

Properties

Help and support

Diagnose and solve problems
6. To the right of the **Apps** header click **Edit** to add an app.

7. In the Edit policy form, click **Select public apps**.
8. Add the selected apps, then click Select.

Configure the default authentication browser for in Microsoft Azure
Learn how to configure the default authentication browser for your ServiceNow mobile apps managed by Microsoft Azure.

Role required: admin

The following steps describe how to add your Intune integrated apps to an existing configuration policy. For information on creating configuration policies, refer to your Microsoft Azure documentation.

1. Sign in to the Microsoft Azure portal.
2. In the search bar, enter Client Apps.

Welcome to Azure!
Don’t have a subscription? Check out the following options:

3. In the navigator menu on the left side of the screen, navigate to Manage > App configuration policies.
4. Open the configuration policy where you want to add your ServiceNow mobile apps.
5. To the right of the Settings header, click Edit.
6. In the **General configuration settings** section, add the following **Name** and **Value** entries:

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNDefaultInstanceURL</td>
<td><a href="https://INSTANCENAME.service-now.com/">https://INSTANCENAME.service-now.com/</a></td>
</tr>
<tr>
<td>SNAuthenticationBrowseriOS</td>
<td>Enter the value for the browser to use for apps on iOS mobile devices. For a list of values, see <em>AppConfig for Mobile Applications</em></td>
</tr>
</tbody>
</table>

**Note:** Replace INSTANCENAME with the name of your instance.
<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SNAuthenticationBrowserAndroid</td>
<td>Enter the value for the browser to use for apps on Android mobile devices. For a list of values, see AppConfig for Mobile Applications</td>
</tr>
</tbody>
</table>
7. Click **Review + save**
Jamf Pro mobile device management (MDM)

Use Jamf Pro to distribute and manage the ServiceNow mobile app on user devices.

Use the following steps to configure the ServiceNow mobile app for iOS and Android.

You should have access to Jamf Pro before attempting any configuration.

Configure the ServiceNow mobile app for Jamf Pro

Use app configuration to pre-configure the instance URL in Jamf Pro.

Role required: admin

1. Log in to Jamf Pro.
2. Click Devices.
3. In the Devices panel, click Mobile Device Apps.
4. On the mobile device apps page, click New.
5. Under Choose an App type, select App Store app or VPP store app.
6. Click Next.
7. In the search field, type ServiceNow.
8. Next to the ServiceNow app, click Add.
10. In the Preferences field, paste the following string.

```xml
<dict>
  <key>SNDefaultInstanceURL</key>
  <string>https://<instance_name>.service-now.com</string>
</dict>
```

Use the name of your own instance where it says <instance_name>.

11. Click Save.
After the instance is configured for the mobile app, configure the app settings to distribute the app to mobile devices in the scope.

MobileIron Mobile Device Management (MDM)

MobileIron is a cloud-based service you can use to manage applications for your users. Users can access apps from MobileIron while you manage and secure content on the network.

Before adding the ServiceNow app to MobileIron, complete the MobileIron setup procedures.

Add the ServiceNow app for iOS to MobileIron

Configure the ServiceNow app for iOS for MobileIron distribution.

Role required: admin

1. Sign in to the MobileIron portal.
2. Navigate to Apps > App Catalog, then click Add.
3. From the search menu list, select the Apple App Store.
4. In the search bar, type ServiceNow.
5. Select the ServiceNow app from the results.
6. Click Next.
7. Optional: Describe and add screenshots of the app.
8. Select a distribution level for this configuration of the app.
• Everyone: The app is added to all user-compatible devices.
• No one: The app is staged for distribution at a later date.
• Custom: The app is distributed only to users or groups you choose.

9. Click Next.
10. In the App Configuration section, click the add icon next to iOS Managed App Configuration.
11. In the Name field, enter a name for the configuration, for example, Default Instance.
12. Enter the key and value pairing.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key</td>
<td>SNDefaultInstanceURL</td>
</tr>
<tr>
<td>Value</td>
<td>The URL for the instance, for example, https://&lt;instance name&gt;.service-now.com/</td>
</tr>
</tbody>
</table>

13. Set a distribution level for this configuration of the app.
• Everyone: The app is added to all user-compatible devices.
• No one: The app is staged for distribution at a later date.
• Custom: The app is distributed only to users or groups you choose.

14. Click Next.
15. Click Done.

The ServiceNow app for iOS appears in the App Catalogs section on the MobileIron Portal. Mobile users can download it from the MobileIron App Catalog.

Configure the ServiceNow app for Android for MobileIron distribution.

Role required: admin

1. Sign in to the MobileIron portal.
2. Navigate to Apps > App Catalog, then click Add.
3. From the search menu list, select the Google Play store.
4. In the search bar, type ServiceNow.
5. Select the ServiceNow app from the results.
6. Click Next.
7. Optional: Describe and add screenshots of the app.
8. Click Next.
9. Set a distribution level for this configuration of the app.
• Everyone: The app is added to all user-compatible devices.
• No one: The app is staged for distribution at a later date.
• Custom: The app is distributed only to users or groups you choose.

10. In the App Configuration Summary section, click Managed Configurations for Android.
11. Add the name and configuration value for the default instance configuration.
### Configuration Setup fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for the configuration, for example, Default instance.</td>
</tr>
<tr>
<td>Configuration value</td>
<td>The full instance URL, for example, https://&lt;instance name&gt;.service-now.com/</td>
</tr>
</tbody>
</table>

12. Set a distribution level for this configuration of the app.
   - Everyone: The app is added to all user-compatible devices.
   - No one: The app is staged for distribution at a later date.
   - Custom: The app is distributed only to users or groups you choose.
ServiceNow Agent
ServiceNow | Version 7.1 | Delegation Status: App is delegated

App Configurations Summary - Managed Configurations for Android

Managed Configurations
Apply these configurations to the app when silently distributed to the users.
- Blocks app from sharing widget across profiles.
- Blocks the user from uninstalling the app.

Managed Configurations

<table>
<thead>
<tr>
<th>Configuration Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server URL</td>
<td>https://&lt;instance name&gt;</td>
</tr>
<tr>
<td>Server Name</td>
<td>The ServiceNow Instance Name</td>
</tr>
</tbody>
</table>

Runtime Permissions
For Apps created with API 23+ and Android 6.0+

Manage Permissions

Runtime Permissions

Distribute this App Config
Choose one of these options

Everyone with App
All Users who have the app

No One
Stage this App Config for later distribution

Custom
This config goes to a custom defined set of users and/or user groups
13. Click Next.
14. Click Done.

The ServiceNow app for Android from the Google Play store appears in the App Catalogs section on the MobileIron Portal. Mobile users can download it from the MobileIron App Catalog.

For more information on the MobileIron App Catalog, see the MobileIron documentation.

If your instance is on an internal network, you might need to configure the VPN. For more information, see the MobileIron documentation on VPN configuration.

Approve the ServiceNow app for Google Play

Android for Work may not be able to find the ServiceNow application. If that is the case, the ServiceNow App might not be approved for your organization. You can approve the app by finding it in the Google Play store.

Role required: none, however, you must either go to Google Play through the MDM, or use the same email address with the MDM that you use to sign in to the Google Play store.

1. Navigate to the Google Play store.
2. Search for the ServiceNow mobile app.
3. Click Approve.

Tested devices and supported versions for ServiceNow mobile apps

Use the following information to ensure that your mobile devices and operating systems are ready to use ServiceNow mobile apps.
Operating system version support

ServiceNow supports latest two versions of Apple iOS, and the latest four versions of Google Android. ServiceNow reserves the right to further reevaluate support based on OS capabilities and the product development roadmap.

Supported operating system and instance versions

Use this table to find the minimum iOS and Android versions necessary to use ServiceNow mobile apps.

Supported operating system versions by app version

<table>
<thead>
<tr>
<th>ServiceNow mobile app version</th>
<th>Minimum iOS version</th>
<th>Minimum Android version</th>
<th>Maximum Instance version</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.x</td>
<td>iOS 11</td>
<td>5.0 Lollipop</td>
<td>Madrid</td>
</tr>
<tr>
<td>7.x</td>
<td>iOS 11</td>
<td>5.0 Lollipop</td>
<td>Madrid</td>
</tr>
<tr>
<td>8.x</td>
<td>iOS 11</td>
<td>5.0 Lollipop</td>
<td>New York</td>
</tr>
<tr>
<td>9.x</td>
<td>iOS 12</td>
<td>5.0 Lollipop</td>
<td>Orlando</td>
</tr>
<tr>
<td>10.x</td>
<td>iOS 12</td>
<td>7.0 Nougat</td>
<td>Paris</td>
</tr>
<tr>
<td>11.x</td>
<td>iOS 13</td>
<td>7.0 Nougat</td>
<td>Quebec</td>
</tr>
</tbody>
</table>

Note:

The latest mobile apps are backward compatible and can be connected to any instance version. However, the mobile apps are not forward compatible. For example, the 8.x client should only be used in New York and earlier instances. Avoid experience issues or crashes by following these maximum instance version guidelines.

Mobile tested devices

ServiceNow® mobile apps have been officially tested on the following devices, although additional devices might work as well. ServiceNow® mobile apps require the Madrid release of ServiceNow or later.

<table>
<thead>
<tr>
<th>Operating system</th>
<th>Tested devices</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apple® iPhone®</td>
<td>• iPhone 6</td>
</tr>
<tr>
<td></td>
<td>• iPhone 7 and 7s</td>
</tr>
<tr>
<td></td>
<td>• iPhone 8 and 8s</td>
</tr>
<tr>
<td></td>
<td>• iPhone X</td>
</tr>
<tr>
<td></td>
<td>• iPhone XS</td>
</tr>
<tr>
<td></td>
<td>• iPhone XS Max</td>
</tr>
<tr>
<td>Apple® iPad®</td>
<td>• iPad</td>
</tr>
<tr>
<td></td>
<td>• iPad pro</td>
</tr>
</tbody>
</table>
### Operating system

**Android™**

<table>
<thead>
<tr>
<th>Tested devices</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Samsung S8 and S8 Edge</td>
</tr>
<tr>
<td>• Samsung S9</td>
</tr>
<tr>
<td>• Samsung S10+</td>
</tr>
<tr>
<td>• Samsung Galaxy Tab A</td>
</tr>
<tr>
<td>• Samsung Galaxy Tab 3</td>
</tr>
<tr>
<td>• Samsung Galaxy Tab S4</td>
</tr>
<tr>
<td>• Samsung Galaxy Tab S5e</td>
</tr>
<tr>
<td>• Xiaomi Mi8</td>
</tr>
<tr>
<td>• Huawei P20</td>
</tr>
<tr>
<td>• Google Pixel 2 and 2 XL</td>
</tr>
<tr>
<td>• Google Pixel 3 and 3a</td>
</tr>
<tr>
<td>• Google Pixel 3 XL and 3a XL</td>
</tr>
<tr>
<td>Android tablet is not officially tested.</td>
</tr>
</tbody>
</table>

### Using the mobile apps

Learn how use ServiceNow mobile apps to get your work done from anywhere using your mobile device.

**Log in to an instance with a mobile app**

Learn how to access an instance using a ServiceNow mobile app.

Role required: none

If you downloaded the mobile app from an enterprise mobility management (EMM) suite, enter your user credentials as prompted. For more detail on EMM, see *Enterprise mobility management (EMM)*

If you did not download the mobile app from an EMM suite, follow the subsequent steps.

1. Tap the icon for one the ServiceNow mobile apps
   ![icon](image)
   on your device

2. In the instance address field, enter the instance address in one of two ways:
Enter the instance address or scan a QR code.

- example.service-now.com

Nickname (Optional)

Save and log in
Note: If you have already logged into another instance using the app, you can the plus button to open a different instance.

3. Tap Login.
4. In iOS versions of the apps, a window appears to remind you that you are connecting your mobile device to the server. Tap Continue.

![iOS window prompt](image)

5. In iOS versions of the apps, you see the prompt to connect to the server. Tap Allow.

Note: The preceding two steps are a requirement of the iOS operating system, are not configurable.

Mobile app structure

Learn how applets, screens, and settings are organized within ServiceNow® mobile apps to understand how to find the tools and data you're looking for.
Components of a ServiceNow app
App Icons

Each ServiceNow app has its own icon on your mobile device. You can configure your apps to show a badge indicating the number of unread notifications you have in the app. For information on enabling or disabling this feature, see Manage push notifications for mobile.

Note: If your company has provided you with branded ServiceNow mobile apps, your icons may not match the image shown here.

Navigation bar

Each ServiceNow app displays a navigation bar along the bottom of the screen. This bar displays icons called tabs, which you can use to access settings, notifications and commonly used applets or application launch pages.

Your administrator can choose which tabs appear in the navigation bar, and the order in which they appear. If a navigation bar has six or more icons, a More tab appears. Tap the More tab to see the additional tabs in a list format.

For more detail on navigation bars, see Navigation bar.
Applet Launchers

Applet launchers serve as landing pages or home pages. Using an applet launcher, you can access applets in various formats, as well as search, do quick actions, and find user information.

Header

The header of the applet launcher defines how the title of the screen appears and what information is shown in the header. For detail on configuring headers, see Applet launcher headers.

Search

Use global search to quickly search through defined search sources configured on your instance. For more details, see Global search for mobile.

Sections

Sections appear as containers within applet launchers. Within sections, you can see applets, charts, record information, and more. For more detail on these sections, see Applet launcher UI sections.

Quick Actions

Quick actions are a way to access commonly used functions, such as transferring records or using a template in the mobile app. For more detail on quick actions, see Quick actions.

For more information on applet launchers configuration, see Applet launchers.
Use the settings screen to configure security options and review information about the app. The following information and options are found in the security screen:

**Header**

The header displays the name of the logged in user and last login time and date.

**Instance**

The ServiceNow instance your app is logged into.

**Version**

The version of the mobile app on your mobile device.

**Security**

Tap to display a configuration page for your security options.

**Privacy Policy**

Tap to view the apps privacy policy.

**Legal**

Tap to view the apps legal disclosures

**Log out**

Tap to log out of the connected ServiceNow instance.

**Analytics**

For more detail about analytics settings, see Analytics settings for mobile

**Notification preferences**

Manage the notifications preferences on your app to ensure you get the information most relevant to you. For more detail, see Manage push notifications for mobile.
Notifications

The notifications screen is accessed from the Notifications tab on the navigation bar. This screen displays the push notifications sent to you. Tap any notification to view its contents.

You can enable or disable all notifications on your mobile device from this tab. For more information on notifications, see Mobile push notifications.
Applets are collections of screens within your mobile apps, designed for a specific task. Each applet provides one or more screens, which you see when you select an applet. The template may include other optional screens that you can access for additional information. All screens display information that is based on records in a data item.

**Segments**

Applets can contain one or more segments, which display information from a specific data item. If an applet has more than one segment, you will be able to switch between segments using a tabbed interface.

**Embedded screens**

Within a form applet, you might see one or many embedded screens. An embedded screen could be a details screen, an activity stream screen, a related list screen, or an embedded list screen.

**Functions**

Applets can be associated with one or more functions. Access these functions through the top menu or swipe actions. Use functions to perform tasks relating to the current record, or navigate to another applets, applet launchers, or external resources.

For more information on applets and how to configure them, see *Applets*.

**Mobile search**

Learn how to use global search to find information within search sources configured for your mobile apps.
ServiceNow apps have several configurable ways to search to help you find what you need to get work done on your mobile device.

### Global search

Use search in the applet launcher header to quickly find information within the defined search sources while on your mobile device.

Your administrator can configure global search within an applet launcher header and define a set of tables that can be searched through. For details on how an administrator can configure global search, see *Enable global search in your applet launcher*.

### Voice search

If enabled by your administrator, you can tap to talk on the microphone icon (●) to the right of the search bar. You can then speak to your device and your voice is captured as search text. For details on how an administrator can configure voice search, see *Enable voice search*. 

© 2020 ServiceNow, Inc. All rights reserved.

ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Mobile voice search

Find items, articles, and people using native speech recognition from an app on their mobile device.

Using voice search

Take advantage of the speech recognition feature of your mobile device. Use voice search by tapping the microphone icon that appears on the right side of a search bar. Use your device's voice recognition interface to record your search query, which is then converted to text and entered as the search query in your app.

The first time you use the voice search feature, your mobile device may prompt you to allow your ServiceNow app to record audio.

Note: Your administrator must enable this feature. For details on enabling voice search on your instance, see Enable voice search.

The first time you use the voice search feature, your mobile device may prompt you to allow your ServiceNow app to record audio.
If you deny this permission, you cannot use the voice search feature. The microphone icon changes to a disabled microphone icon.

Third-party data usage

Depending on your mobile device's operating system, the voice recordings you create with the voice search feature are sent to Google or Apple to be processed into a text query. ServiceNow does not have control of the recording once it has been sent.

Mobile photo search

Configure photo search to give your users the ability to perform image-based searches using the objects around them.
Photo searches in your mobile apps

When photo search is configured on your instance, a photo icon appears in your applet launcher search bar. Your users can tap this icon to use the camera on their mobile device to take a picture. The picture is identified using the Google Vision API, which returns one or more results. Users can select a result, which is used as their search query.

Third-party data usage

Images you take using the photo search feature are sent to Google for identification using Google Vision API. ServiceNow does not have control of the image once it has been sent. For details on how Google Vision API handles your image, see https://cloud.google.com/vision/docs/data-usage.

Mobile activity stream

Use the activity stream to submit work notes and comments to a record with a mobile app.
Access the activity stream for a record by opening a record and tapping the activity stream tab ( ).

To post to the activity stream, tap the plus icon ( ) in the record title bar.

You can post any of the following items to the activity stream:

- Work notes
- Additional comments
- Photos
- Videos
- Files

**Note:** A work note appears to employees only, but an additional comment appears to employees and customers.
Mobile filters

Learn how to filter screens to display only the records that are most relevant to you.

When you tap the filter icon, you can filter your records. Filtering records in the mobile app works like filtering with a condition builder on the Now Platform.

Filtering your list

Begin by selecting the category or categories to filter by. Your administrator defines which categories are available. These categories appear horizontally across the top of the filter screen.

Select or de-select a category by tapping it.
Tap on a category to see the fields available in that category. When you select multiple categories, you can see fields common to both categories.
For date fields, the filter displays a calendar interface. Swipe to the left or right to view previous or future months. To select or de-select a date, tap the day. To search a date range, tap a day, then tap a second day. The filter searches for dates matching that range.
For fields where a value must be selected rather than entered, the available options display as check boxes. Tap one or more options to select or de-select, and then tap **Apply**.
After you have defined your filter, tap **Apply**.
Mobile platform dashboards

Use a mobile dashboard to display data in graphical format. Adding reports and Performance Analytics widgets helps users more easily identify trends and turning points through indicator scores and visual representation.

A mobile dashboard is an applet launcher that enables you to see preconfigured reporting and analytical views via trend lines and score visualizations. Tap chart UI sections in your applet launcher to open reports and analytical information displaying the status of instance data. For example, you can review the number of incidents opened per week or the average resolution time of projects.

For more information about adding chart UI sections to your applet launcher, see Configure chart UI sections.
Mobile dashboard applet launcher showing Performance Analytics and reporting data

Click the names of the Performance Analytics and Reporting charts listed in the sections below for general, non-mobile specific information.
Performance Analytics charts

Performance Analytics charts display views of key performance indicators over time, for example, the percentage mean time to resolve incidents.

The Performance Analytics chart available on the mobile platform is latest score.

Reporting charts

Reporting charts display views of current state instance data.

Four types of Reporting charts are available on the mobile platform: time series trend-by charts, score reporting, bar and donut. These charts display information such as the number of high priority work orders or the number of pending approvals over a defined period.

The available score report is single score.

The available bar report is vertical bar.

The available pie and donut report is donut.

The available time series trend-by charts are:

- Area
- Line
- Spline
- Step line

For more information

For general information on the benefit and use of Performance Analytics and Reporting concepts (not specifically for mobile), see the following documentation topics:

Understanding Performance Analytics

Getting started with reports

Performance Analytics compared to Reporting

Displaying Performance Analytics charts for mobile dashboard

Use Performance Analytics charts to help identify trends and turning points through indicator scores and visual representation. These charts are used when reporting on KPIs and metrics.

The mobile platform supports the Performance Analytics latest score widget, which shows a single indicator score and the change in that score compared to a previous period. For example, you can configure the widget to show the number and percentage difference of high priority incidents over a time period.
Performance Analytics chart

Score - latest score widget
Latest score widgets show a single indicator score and the change in that score compared to a previous period. The latest score widget in the applet launcher displays the single score value, the Performance Analytics calculation, the time of the change, and the compare-to date.

Tap the widget to display a detailed chart of the Performance Analytics data. Alternatively, you can navigate to a chart screen via a navigation button. To magnify or contract the chart view, spread or pinch your fingers on the screen. A small sparkline chart at the bottom of the screen highlights the magnified section of the chart. Tap the zoom-out button to return the screen to its original magnification.

Note: Unlike on the web-based UI, you cannot tap on an x axis for details or records on a specific area of a report.
Mobile dashboards support the following Performance Analytics features from the web-based UI:

- Data configuration: Direction property; target and gap calculations in chart screen
- Style configuration: Change color

**Unavailable Performance Analytics features**

The following Performance Analytics features from the web-based UI are not available in mobile dashboards:

- Date configuration: Value when nil field; "compare to" date score
- Style configuration: Templates, score color

**For more information**

For general information about Performance Analytics widgets and when to use them, see *Understanding Performance Analytics*.

For instructions about enabling charts to be displayed on mobile apps, see *Configure chart UI sections*.

**Displaying reporting charts for mobile dashboard**

Use Reporting charts to help identify trends and turning points through indicator scores and visual representation. These charts display the current state of instance data.

The mobile platform supports the following reporting charts.
Time series reports

Time series reports show you data points over a period of time, for example, the number of cases resolved within the first response. For more information, see *Time series reports for mobile*. 

![Mobile Device Screen](image-url)
Single score reports

Single score reports display a single key value, for example, the number of critical incidents opened by the QA team. For more information, see [Single score reports for mobile](#).
Donut reports

Donut reports show how one grouping relates to the total amount, for example, the number of tasks completed by a team compared with the status of other task definitions. For more information, see Donut reports for mobile.
Bar reports

Bar reports compare two or more values. Use a stacked bar report to show information in segments that are proportional to the values they represent. A bar report, for example, can display the number of incidents assigned to employees while a stacked bar report can segment the incidents into categories. For more information, see Bar reports for mobile.
For more information

For general information about reports and when to use them, see *Getting started with reports.*

For instructions about enabling charts to be displayed on mobile apps, see *Configure chart UI sections.*

Time series reports for mobile

Time series reports display a series of data points over successive intervals of time, for example, the number of incidents opened over a certain period. The mobile platform supports the following time series reports, area, line, spline, and step line.
Tap a time series dashboard preview in an applet launcher to display a detailed screen of the report. Alternatively, you can navigate to a chart screen via a navigation button. Depending on the configuration settings, you can then tap on the report screen to see a complete list of records relating to the report. To magnify or contract the chart view, spread or pinch your fingers on the screen. A sparkline at the bottom of the screen highlights the magnified section of the chart. Tap the zoom-out button ( ) to return the screen to its original magnification. For time series charts that support multiple lines, select lines from the legend that you want to include or omit from the chart.

To list records corresponding the point highlighted in the chart line, tap the navigation arrow ( ) in chart header. Tap the list icon ( ) in the screen header to display a list of all the records contained within the time series chart.

Available time series report types

Mobile dashboards support the following time series report types:

- Area
- Line
• *Spline*
• *Step line*

**Available time series features**
Mobile dashboards support the following time series report features from the web-based UI:
• Data configuration: Multiple line reports; accessibility line patterns
• Style configuration: Chart colors; chart titles; y-axis settings

**Unavailable time series features**
The following time series features from the web-based UI are not available in mobile dashboards:
• Drill-down views; data labels on charts
• Legend configuration by admin
• Grouping by data set for more than one level of grouping
• Styling (other than chart colors)

Note the following display behavior within chart applets and charts:
• For some time series charts, the title for the y axis is not visible in the applet launcher preview of a report.

**Single score reports for mobile**
A single score report is a single value which you can use to share metrics or scores that are key to your business needs, for example the number of high priority incidents opened by your team.

Tap on the single score to see a complete list of records relating to the report. You can then tap on individual records for additional information.
Available single score features

Mobile dashboards support the following single score report features from the web-based UI:

- Data configuration: Aggregation type
- Style configuration: Default colors; color rules; chart titles

Unavailable single score features

The following single score features from the web-based UI are not available in mobile dashboards:

- Drill-down views
Donut reports for mobile

Donut reports enable you to see the portions and percentages between different priorities at a single glance. For example, a donut report showing open incidents by priority, enables you to quickly see whether incident counts of different priorities are within acceptable ranges.

To display a donut report, tap on the dashboard preview in an applet launcher. Alternatively, you can navigate to a chart screen via a navigation button. The percentage figure displayed in the center of the chart corresponds to the percentage of records for that selected segment compared to the remaining segment records within the chart.
To display the number of records contained in that group, tap on a donut segment. To list records related to the selected segment, tap the segment, then tap the navigation arrow (↑) in the chart header. Tap the list icon (≡) in the screen header, to display a list of all the records contained within the donut chart.

Use the legend at the bottom of the chart to include or omit segments from the donut. As you add or remove filters, the percentage in the center of the donut report changes to correspond to the records currently displayed in the donut report.

**Available donut report features**

Mobile dashboards support the following donut report features from the web-based UI:

- Data configuration: aggregation type
- Style configuration: Chart colors; set palette; chart titles

**Unavailable donut report features**

The following donut report features from the web-based UI are not available in mobile dashboards:

- Grouping by data set for more than one level of grouping
- Legend configuration by admin
- Title (other than chart title)
- Styling (other than chart colors and palette)

**Note:** Admins can configure both donut and pie reports in the web-based UI. Both these reports have a donut visualization in the mobile platform.

**Bar reports for mobile**

Use bar reports to compare individual or aggregate scores across data dimensions. Each bar represents a specific category of data. A stacked bar report helps you differentiate between each segment within a category. Mobile dashboards support vertical bar graphs.
To display a bar report, tap a dashboard preview of a bar report in an applet launcher. Alternatively, you can navigate to a chart screen via a navigation button.

Tap on a bar or a segment of a bar to display the number of records contained in that selection. To view the records contained within a bar or a segment of a bar, tap the navigation arrow (>) in the chart header. To display a list of all the records contained within the bar report, tap the list icon (<) in the screen header.

To magnify or contract the chart view, spread or pinch your fingers on the screen. A sparkline at the bottom of the screen highlights the magnified section of the chart. Tap the zoom-out button ( ) to return the screen to its original magnification.

A maximum of four bars are displayed when viewing the chart in a vertical position. Use the horizontal position to view additional bars.

For stacked bars, use the legend at the bottom of the chart to either include or omit segments from the bar report.
Mobile dashboards support the following bar report features from the web-based UI:

- Data configuration: Aggregation type, stacked bar
- For stacked bars visualizations, support of one group by and stack by option
- Style configuration: Chart colors; set palette; chart title

**Unavailable bar report features**

The following bar report features from the web-based UI are not available in mobile dashboards:

- Only vertical bar graphs are supported. Horizontal, pareto, and histogram bar reports are not supported.
- Legend configuration by admin
- Title (other than chart title)
- Styling (other than chart colors and palette)

**Change chart view for accessibility**

Gives you the option to change chart views from color segments to black and white patterns. This option can be used for accessibility purposes.

Role required: user

The **Enable data visualization patterns for charts** button gives you the option to view charts in either black and white patterns or colored segments, for all chart types. Using black and white patterns assist people with accessibility requirements.

1. Tap More

    ( )

    in the banner frame.

2. Tap Settings

    ( )

3. Enable the **Enable data visualization patterns for charts** to view charts in black and white patterns. Do not enable the button to view charts as colored segments.

   **Note:** If you do not see the **Enable data visualization patterns for charts** button, refer to your admin to enable this feature.

Select the display option best suited for your requirements.
Comparison of graph with and without accessibility option enabled
Mobile app settings

Use the settings in your mobile applications to manage accessibility, notifications, location tracking, and more.

App PIN settings for mobile

Improve the security of your

Using a PIN with your mobile app

A security PIN is a 6-digit code you must enter for added security. When this feature is enabled, you are prompted to enter your pin:

• When logging into your app
• When the app has been inactive for five minutes
• When accessing the security settings for the app

If your mobile device uses faceID, touchID, or similar biometric security, you can use biometric authentication in place of the PIN.

For information for administrators on PIN settings for your instance, see Require an app PIN for the mobile app.

Create a PIN for a mobile app

Create a PIN for your mobile app.

Role required: admin

1. Log into a ServiceNow mobile app.
2. On the navigation bar, tap the Settings.
4. Tap Require an app PIN.
5. When prompted Enter a 6 digit PIN for your app.
6. When prompted, re-enter the same 6 digit PIN. Your app is now configured with a security PIN.

7. Optional: Enable Unlock with Fingerprint Sensor to use your device's fingerprint biometric security instead of your PIN.

Change your PIN for a mobile app

Role required: admin

1. Log into a ServiceNow mobile app.
2. On the navigation bar, tap the Settings.
4. Tap Change app PIN.
5. At the prompt, enter your current app PIN.
6. When prompted, enter a new 6 digit PIN.
7. When prompted, re-enter the same 6 digit PIN.
Your app is now configured with a new security PIN.

**Remove your PIN from a mobile app**

Role required: admin

1. Log into a ServiceNow mobile app.
2. On the navigation bar, tap the **Settings**.
3. On the settings page, tap **Security**.
4. Tap **Disable app PIN**.
   Your app no longer configured with a security PIN.

**Analytics settings for mobile**

Enable analytics tracking of your user journey in a mobile application.

When launching a mobile application for the first time, European users are prompted to select whether to consent to analytics tracking. Non-EU/non-US users see a notice of enabled analytics tracking.

All users can enable/disable tracking in the mobile application at **Settings > Analytics > Enable Analytics**.

For detail on how an administrator can configure and track analytics for mobile, see *User Experience Analytics*.

**Enable accessibility for charts**

Enable the chart accessibility feature to use data visualization patterns rather than colors in your ServiceNow mobile app charts.

Role required: admin

Enable chart accessibility to change the appearance of your charts to accommodate both low vision and color-blind users. The chart accessibility feature uses high contrast colors and patterns in place of the standard colors.
2. On the navigation bar, tap the **Settings**.
3. On the settings page, tap to enable the **Chart accessibility** toggle.

### Location tracking for mobile

Turn on location tracking from your mobile device and in the mobile app so that you can keep a record of where you were when you last worked on an instance. Location tracking continues even when there is no internet connection.

The location setting on your mobile device takes precedence over the location setting in the mobile app. Grant permission on your mobile device to enable the app to track your location.

#### Note:
To access geolocation settings on your mobile device, your administrator must install the geolocation plugin and enable location tracking on your user record. For more detail on geolocation configuration, see [Mobile location tracking configuration](#).

---

#### Enable location tracking

Access by tapping the **Settings** tab in the navigation bar, and selecting **Location Tracking**.

On the **Geolocation** page, enable **Geolocation tracking**.

#### Note:
You may see a prompt from your mobile device to grant location tracking permission to the mobile app. Select **OK** to enable geolocation tracking.

Use the **Track For** field to select the duration for which the app tracks your location. Once you have set a duration, you can see a time stamp indicating when location tracking ends on your **Settings** page.

#### Using location tracking

When enabled, geolocation keeps a record of where you are when you last work on an instance. Location tracking continues even when there is no internet connection.

For more information on location tracking, see [Location tracking](#).
Offline mode for mobile

Access and submit actions to records even if you do not have an internet connection.

Watch this three-minute video on how offline mode works. Learn how to download data, enable and disable offline mode, synchronize your outbox, and resolve synchronization errors.

Plan ahead when you use offline mode. If you will be working in an area with no internet access, download what you want to work on ahead of time while you are still connected to the internet.

When you are in offline mode, the changes that you make to your records are logged in your outbox. Your outbox tracks all the actions that you made on your cached records. After your device has internet access, you can synchronize your device with the instance. The cached changes in your outbox update to the instance.

Enable offline mode

Enable offline mode in your Settings tab. Tap Offline and then toggle on Offline Mode.

If you have not already downloaded the offline cache, you see a dialog box that asks you to download it. Tap Download and Go Offline.
Offline Cache
Download cache to enable Offline Mode.

Cancel Download

Offline Mode

Outbox
Your Outbox is empty

BACKGROUND DOWNLOADING

Enable

Wi-Fi only
Navigate the mobile app in offline mode

When you are in offline mode, a banner that reads "Offline Mode" appears across the top of all screens.

Depending on how your administrator configures the mobile app, you are unable to submit certain actions while you are in offline mode. These actions are grayed out on the user interface.
When you submit an action while you are in offline mode, the change gets marked with a yellow border. Changes remain marked until your device synchronize to the server.
Disable offline mode and synchronize outbox

To return online, in the mobile app, navigate to Settings > Offline Mode. On the offline mode screen, toggle off Offline Mode.
A dialog box asks you to synchronize the changes in your outbox. To push all the changes that you made in offline mode to the instance, tap **Go Online & synchronize**.

**Note:** After the synchronization completes, you are back online and your offline cache is deleted.

**Cache expiration**

Your administrator configures a default length of time after which your offline cache expires.

When a cache expires, you lose all the data that you saved to the cache. If you do not synchronize the cache to an instance before the cache expires, none of your changes show on the instance.

Warning messages appear periodically to remind you to synchronize your cache before it expires. To avoid losing your data due to a cache expiration, always synchronize your cache before and after going offline.
Resolve synchronization errors

Problematic changes that you made in offline mode do not synchronize to the instance. They remain in the outbox until they are resolved.

You cannot synchronize changes that contradict changes that were made by other users while you were offline. For example, you may receive an error message if you try to synchronize changes to a record that another user closed while you were working in offline mode.

To view the errors in your cached changes, navigate to Settings > Offline Mode > Outbox. Error messages indicate where errors occurred in your cached records while you were offline. From the application home page, navigate to the indicated records and resolve these errors.
After you resolve an error, return to the outbox to delete the error messages. In the outbox, swipe left on the error message. Then, tap **Delete**.
Scheduled offline caching

Enable scheduled offline caching to automatically download your cache according to your work schedule. Scheduled caching works in the background, so you are able to continue to use the app while the download completes. You
can enable or disable this feature in your app settings. Enable **Wi-Fi only** to allow downloads only when you are connected to Wi-Fi.

**Manage push notifications for mobile**

Receive push notifications in ServiceNow mobile apps for the records that are most relevant to you by changing the notification settings on your mobile device and in the mobile app.

Role required: none

Push notifications appear on your device lock screen when they first arrive. You can tap a push notification to open the corresponding record. A badge also appears on the mobile app icon when you receive notifications.

**Note:** The notifications setting on your mobile device takes precedence over the notifications setting on the mobile app. If you enable notifications in the mobile app but do not allow push notifications on your mobile device, you can't receive push notifications.

1. Access your notification settings by navigating to **Settings > Notification preferences**
Settings

System Administrator
Last login: May 29, 2020 at 12:40 PM

Instance
adamgolab

Version
10.0.4 (202005281530.20)

Security

Offline

Notification preferences

Location tracking
Off

Legal
2. Tap the **Enable Notifications** switch to enable or disable push notifications.

**Manage push notifications for mobile by category**

If your administrator enabled the option, you can enable and disable notifications by category in your mobile notification settings.

Role required: admin

This feature must be enabled on your instance by an administrator. For details on enabling push notifications for mobile, see *Enable push notification categories in mobile settings*.

1. Access your notification settings by navigating to **Settings > Notification Preferences**
2. Tap the **Enable Notifications** switch to enable or disable push notifications.

3. Under the **Enable Notifications** section are your notification categories. Tap a category, then tap to enable or disable a type of notification.
**Accessing an instance on a mobile device web browser**

Access an instance from anywhere using your mobile device. Connect using the mobile app or a web browser on a mobile device.

The updated mobile UI includes the new ServiceNow branding as well as an all new app.

The mobile browser experience is consistent with the ServiceNow Classic mobile app.

**Mobile web UI**

Access an instance using the browser on your mobile device.

The mobile web UI is similar to the ServiceNow Classic mobile app. Some features are optimized more for the ServiceNow Classic app and may not perform as well on a mobile browser.

Administrators can configure home pages by role so different users may see different mobile experiences depending on their role.

**Mobile web unsupported features**

The following capabilities are not currently supported in mobile browsers, though they may work to varying degrees.

- Switching to the standard browser interface from the mobile interface
- HTML fields
- CODE tags to render HTML in Journal Fields
- UI Scripts
- Legacy Chat
- Field styles
- Formatters
- Form Templates
- Timeline visualizations
- Embedded lists
- Assessments, surveys, and legacy surveys
- Mobile service catalog features
  - Order guides
  - Wizards
  - Content items
- These variable types:
  - Macro With Label
  - UI Page
  - List Collector
  - HTML
  - Macro
  - Label
  - Break
- Data lookup rules
- Custom auto-complete scripts
Tablet interface

Use a tablet to access your instance either app or from a browser.

Use the native tablet app for an experience similar to the native mobile app. Applications or modules that you have favored in a desktop instance appear as tiles on your mobile app homescreen.

The tablet web UI mimics the desktop experience in UI16. $tablet.do has been disabled in UI16 because the tablet interface is the same as the desktop.

Connect to an instance using the browser on your tablet for an experience similar to the standard desktop user interface.

Tablet features with limited support

- Editing lists: You cannot edit field values in a record from the list view. Access the record form to modify any field values.
- Dependency Views map, schema map, graphical Workflow Editor, Gantt chart, and visual dispatch tool: Graphics-based tools can be viewed but not modified from the tablet interface. Data presented by these tools is read-only when accessed through the tablet interface.
- Calendars: You can access calendar reports but scrolling around the calendar as you would on the desktop interface is not supported.
- Video and image attachment upload: Attach videos and images with both iOS 9 and Android. Other file type attachments are not allowed.

Unsupported tablet features

- Field watcher: Administrators must use the desktop version to access the Field Watcher.
- JavaScript debugger: Administrators must use the desktop version to access the JavaScript debug window.
- Language picker: Even if the internationalization plugin is enabled, the language picker does not appear in the tablet UI toolbar. Language selected through the desktop interface applies to the tablet UI.
- Domain picker: Tablet users cannot select any other domains that administrators configure for domain-specific personalizations. To select a new domain, use the desktop interface.
- Slushbucket feature: Any lists, fields, or filters that use the slushbucket feature are unsupported on a tablet device. Slushbuckets are only supported in the desktop interface.
- Suffix in the navigation filter: You can use the .list, .do, or .form shortcuts to access a list of records in a table or a new form from the desktop version only.
- Support chat: End users cannot request a chat session nor can support technicians respond to chat requests when using the tablet interface. Help desk chat is only supported in the desktop interface.
- Printer friendly view: This view, which shows the current screen in a pop-up window without frames and the application navigator, is not available from the tablet.
- Timeline sliders and the Timeline Metrics UI actions: Features that use timelines, such as the workflow timeline and the Gantt chart are not supported from the tablet.

Mobile plugins and upgrades

Learn about upgrading to ServiceNow mobile, and how to install plugins to get your users started with the ServiceNow mobile platform.
Mobile plugins

Learn about the plugins used to enable ServiceNow mobile on your instance.

Core plugins

The following mobile plugins include the core functionality of ServiceNow mobile and the ability to interact with ServiceNow mobile apps. Plugins marked as base system plugins are automatically installed on your instance.

<table>
<thead>
<tr>
<th>Name</th>
<th>ID</th>
<th>Description</th>
<th>Base System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mobile Studio</td>
<td>com.glide.sg-studio</td>
<td>Plugin for Mobile Studio and Mobile Agent</td>
<td>Yes</td>
</tr>
<tr>
<td>ServiceNow NowMobile App Screens and Applet Launcher</td>
<td>com.glide.mobile-employee</td>
<td>Application and configurations required to setup the Now Mobile app.</td>
<td>No</td>
</tr>
<tr>
<td>Human Resources Scoped App: Mobile Employee Onboarding</td>
<td>com.sn_hr_onboarding</td>
<td>Activates the Mobile Onboarding app for HR.</td>
<td>No</td>
</tr>
<tr>
<td>Human Resources Scoped App: Mobile</td>
<td>com.sn_hr_mobile</td>
<td>Activates the HR mobile pieces for Now Mobile</td>
<td>No</td>
</tr>
</tbody>
</table>

Supporting plugins

These plugins are not required for ServiceNow mobile, but include extended functionality, such as offline mode, geolocation, and access to Virtual Agent.

<table>
<thead>
<tr>
<th>Name</th>
<th>ID</th>
<th>Description</th>
<th>Base System</th>
</tr>
</thead>
<tbody>
<tr>
<td>SG Offline support</td>
<td>com.glide.sg.offline</td>
<td>Provides offline support for ServiceNow mobile.</td>
<td>No (This plugin is automatically installed with the Field Service Mobile plugin)</td>
</tr>
<tr>
<td>Service Management Geolocation Mobile</td>
<td>com.snc.service_management</td>
<td>Menu in the mobile UI for Service Management Geolocation.</td>
<td>No</td>
</tr>
<tr>
<td>Service Management Geolocation</td>
<td>com.snc.service_management</td>
<td>Enables geolocation capabilities for Service Management</td>
<td>No (This plugin is automatically installed with the Field Service Mobile plugin)</td>
</tr>
<tr>
<td>Geolocation</td>
<td>com.snc.geolocation</td>
<td>Core geolocation capabilities</td>
<td>Yes</td>
</tr>
<tr>
<td>Glide Virtual Agent</td>
<td>com.glide.cs.chatbot</td>
<td>Virtual Agent platform and other necessary plugins</td>
<td>No</td>
</tr>
</tbody>
</table>
Activating plugins on your instance

Install a plugin by searching for the plugin name in you plugins list. For information on the plugin activation process, see Activate a plugin.

Request apps on the Store

Visit the ServiceNow Store website to view all the available apps and for information about submitting requests to the store. For cumulative release notes information for all released apps, see the ServiceNow Store version history release notes.

Mobile plugins for ServiceNow Agent

Use these plugins to extend functionality for the ServiceNow Agent app.

Base system applications for ServiceNow Agent

<table>
<thead>
<tr>
<th>Application</th>
<th>Description</th>
<th>Availability</th>
<th>Paid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Walk-up Experience</td>
<td>Walk-up Experience for pre-built tech lounges. This plugin enables your IT organization to set up a walk-up contact channel to support online and on-site check in.</td>
<td>Instance</td>
<td>Yes</td>
</tr>
<tr>
<td>ITSM Mobile Agent</td>
<td>Delivers mobile-first experiences designed for agents to triage, act on, and resolve incidents. Agents can also view schedules, check who is on-call, respond to major incidents, and more.</td>
<td>Store</td>
<td>No</td>
</tr>
<tr>
<td>Asset Receiving Mobile</td>
<td>Provides the capabilities for receiving personnel to receive purchased assets in ServiceNow Agent.</td>
<td>Instance</td>
<td>Yes</td>
</tr>
<tr>
<td>PPM Mobile</td>
<td>Project Portfolio Suite Mobile user experience. This provides access to project status and project status report in ServiceNow Agent.</td>
<td>Instance</td>
<td>Yes</td>
</tr>
<tr>
<td>Customer Service Mobile</td>
<td>Customer service mobile user experience.</td>
<td>Instance</td>
<td>Yes</td>
</tr>
<tr>
<td>Application</td>
<td>Description</td>
<td>Availability</td>
<td>Paid</td>
</tr>
<tr>
<td>---------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>--------------</td>
<td>------</td>
</tr>
<tr>
<td>GRC: Mobile</td>
<td>Task assignments, requests, approvals, and other follow-up actions for GRC applications directly from the ServiceNow Agent app. Receive timely notifications for current alerts, as well as risk and compliance status for your critical assets, vendors, and impacted essential business services.</td>
<td>Store</td>
<td>Yes</td>
</tr>
<tr>
<td>Security Incident Response Mobile</td>
<td>Receive notifications when security incidents or security response tasks are assigned to you or to your assignment groups and begin remediation using the ServiceNow Agent app. Monitor the security incidents the Security Operations Center (SOC) or your assignment groups are responding to with search criteria that you enter.</td>
<td>Store</td>
<td>Yes</td>
</tr>
<tr>
<td>Vulnerability Response Mobile</td>
<td>Receive notifications when vulnerability groups or vulnerability items are assigned to you or to your assignment group. Begin remediation on your most critical vulnerabilities using the ServiceNow Agent app.</td>
<td>Store</td>
<td>Yes</td>
</tr>
</tbody>
</table>

**Mobile plugins for Now Mobile**

Use these plugins to extend functionality for the Now Mobile app.

**Base system applications for Now Mobile**

<table>
<thead>
<tr>
<th>Application</th>
<th>Description</th>
<th>Availability</th>
<th>Paid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Human Resources Scoped App: Mobile Employee Onboarding</td>
<td>Mobile Onboarding module for HR. Combined with Mobile Employee Experience Native Application plugin, items related to HR Onboarding will be shown in the Now Mobile app.</td>
<td>Instance</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Mobile plugins for Mobile Onboarding

Use these plugins to extend functionality for the Mobile Onboarding app.

Base system applications for Mobile Onboarding

<table>
<thead>
<tr>
<th>Application</th>
<th>Description</th>
<th>Availability</th>
<th>Paid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Human Resources Scoped App: Mobile Employee Onboarding</td>
<td>Mobile Onboarding module for HR. Combined with Mobile Employee Experience Native Application plugin, items related to HR Onboarding will be shown in the Now Mobile app.</td>
<td>Instance</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Migrate from Classic mobile to ServiceNow mobile

Migrate from ServiceNow Classic to ServiceNow® mobile to take advantage of features such as rapid development, offline capability, and integration with native mobile device features.

Understand the benefits of migration to ServiceNow® mobile

ServiceNow® mobile is a native, mobile-first design that introduces new capabilities that were not available in ServiceNow Classic:

• Rapid, low code application development using Studio.
• Base system applications for Approval Management, ITSM, and Field Service Management.
• Integration with native mobile device features, such as camera, phone, SMS, and map apps.
• Offline capabilities, including offline read and write.
• Enhanced security features including data loss prevention.

For more detail on ServiceNow® mobile, see Agent mobile app.

Evaluate your organization's current mobile use

Start off with a list of business goals and determine which features your users need most. Consider which screens, fields, and actions your users use frequently, as well as what functionality your users do not use.

There is no direct method of transferring ServiceNow Classic components into the ServiceNow® mobile product. You can create the same functionality available to your users with base system applications and manually building mobile applications in the ServiceNow® mobile framework.

Use base system applications when possible

ServiceNow provides several base system applications to get you started quickly with commonly used functionality. Base system applications are tested by ServiceNow, and receive updates automatically as you upgrade your instance. The New York release includes applications for:

• Approval Management
• Customer Service Management
• Field Service Management
• Incident Management
• Information Technology Operations Management
• ITSM Software Asset Management
• ITSM
• Security Operations
• Risk
• HR
• Finance

Use these applications right away, or customize them to suit your business needs using Studio.

Mobile experience for Approvals
Approve catalog requests, requested items, or change requests from anywhere using the ServiceNow Agent application. For more detail, see *Fulfiller experience in ServiceNow Agent*

Mobile experience for Customer Service Management
Manage customer service cases from a mobile device with the Customer Service Management mobile application. Stay connected and access information in real time to complete tasks quickly. For more detail, see *Mobile experience for Customer Service Management*

Mobile experience for Field Service Management
Manage your field service tasks anywhere using the Field Service mobile application. With this application, you stay connected with your mobile device, can access important information, and complete your tasks quickly. For more detail, see *Mobile experience for Field Service Management*

Mobile experience for Incident Management
Manage incident tasks anywhere using the incidents mobile application, which enables you to stay connected and access important information to complete tasks quickly. For more detail, see *Mobile experience with ITSM Mobile agent*

Create your own applications
Use Studio to make new applications, and give your users the ability to work in areas not covered by the current base system applications. For details on the components of ServiceNow® mobile and examples of application creation, see *Building and configuring mobile apps.*

Understand the current limitations for ServiceNow® mobile

General
• Client scripts are not supported.
• Geolocation and Offline mode is not supported in the Now Mobile app.
• Siri shortcuts are not available in the ServiceNow Agent app.

Related list
M2M or scripted relationships are not supported.

Screen UI policies
• Screen UI Policies can’t control whether field is read-only.
• Screen UI Policy conditions cannot be scripted.
• Reference & Date/Time fields cannot be used in screen UI Policy conditions.

Offline mode

• Mobile apps do not automatically go into offline mode when internet connection is lost.

Migration from Madrid to New York mobile

Migrate your mobile applications in the New York release to take advantage of the improved features and continue editing within Studio.

Changes made during your upgrade

During the upgrade to New York, the instance updates to the new mobile hierarchy by activating the Mobile Agent Native Client [com.glide.sg.agent_native_client] plugin. This installation creates the following changes:

Native clients

Adds the Native Clients [sys_sg_native_client] table. Records on this table represent the available native clients: ServiceNow Agent, Now Mobile, and Mobile Onboarding.

Navigation bar

Adds the navigations [sys_sg_navigation] table. Records on this table represent a navigation bar for each of the native clients. Records on this table during the migration have their Legacy application [legacy_application] field enabled.

Notifications tab

Adds the notifications tabs [sys_sg_notifications_tab] table. Records on this table represent a tab for notifications on each navigation bar.

Settings tab

Adds the settings tabs [sys_sg_settings_tab] table. Records on this table represent a tab for settings on each navigation bar.
This upgrade includes new features such as application launchers and a configurable navigation bar. Any unmodified base system mobile applications installed on your instance are automatically updated to work with the new design, and can be used with Studio right away. For more detail on the mobile hierarchy used in New York and later, see Mobile hierarchy.

Modified base system applications, and applications that you have created in Madrid will continue to work after the upgrade. These applications will not be configurable in Studio until after you have run the mobile migration script.

**Post-upgrade considerations**

After an upgrade, consider the following information to confirm that your mobile implementation is working as expected, and ensure that mobile migration script runs.

**Modified base system applications**
Document any changes you have made to mobile applications provided by ServiceNow, as well as any applications you have created. Test each of these applications to ensure that they continue to function as you expect.

**Use the Debug Upgrade feature**

The debug upgrade feature can help you to quickly diagnose upgrade issues. For information on this feature, see *Debug upgrade*.

A video training course on this tool is available. To view this course, see *Using Debug Upgrade*.

**Review skipped records**

To prevent overriding your customizations, the upgrade process does not update records that you have modified. Instead, the upgrade process notes this skipped record in the upgrade logs. For more detail on handling skipped records, see *Process the skipped records list*.

A video training course on resolving skipped records is available. To view this course, see *Upgrade Skipped Records*.

**Review functionality after upgrade**

Once you have upgraded your instance and run the migration script, regression testing can help ensure that your users can continue to work as expected after an upgrade. A regression test is a review of your applets, screen ui policies, and functions to make sure that they are working as intended.

**Running Mobile migration script**

This script converts your custom applications and any modified base system application to the new mobile schema available in the New York release. The script only changes the current scope when it runs. If you have more than one scoped mobile application, you must run the script for each scope.

After an upgrade, the option to run the migration script appears when you first access a custom application, or a base system application that you have modified. For example, when opening a modified or custom applet record. You can also see the migration prompt when accessing the applet picker in Studio by browsing to *Mobile Studio > Applets* and clicking the pop-out icon.

The migration prompt displays if any of the applets shown the picker require migration.

![Migration Prompt](image)

After the script completes, you may be prompted to resolve collisions detected by the migration process. Collisions are records created by ServiceNow that you have modified, and are not automatically upgraded. Collisions can only occur when you have modified a base system application before your upgrade to New York or later releases.
Click the View Collisions to resolve these collisions. For detail on this process, see Troubleshooting mobile migration script results.

Changes made by the mobile migration script

Click Migrate to start the migration script for the current scope. The migration script migrates all records within the scope, not just the applet you have opened.

Applications and folders transition to applet launchers

The legacy Madrid schema used mobile applications and folders to organize your applets. The Now Mobile schema, uses applet launcher screens, which are divided into UI sections. Applet launcher is accessed by tapping on tabs in the navigation bar which appears at the bottom of your app screens.
The migration script creates an applet launcher for each mobile application record. The script converts each folder in the original mobile application to a new horizontal icon section within that applet launcher. The script then creates an icon in the icon section for each applet with the folder. Hidden screens do not appear in the icon section. The script then adds a tab to the navigation bar for each of the new applet launchers.

The example image shows how the incidents application appears after the migration process. The original folders (My Incidents and Group Incidents) display as UI sections in the Incidents applet launcher. These UI sections can scroll horizontally to show as many applets as needed. The Incidents application is accessible by tapping the Incidents tab in the navigation bar.

After migration, the script removes the legacy Folder [sys_sg_folder] and Mobile Application [sys_sg_application] records.
For more detail on the navigation bar, applet launchers and their UI sections, see Navigation bar, and Applet launchers.

**Form migration**

The Form applet replaces the master detail screens used to view record forms in the Madrid release. The migration creates a form screen [sys_sg_form_screen] record. The script creates segments for each embedded screen in the original master detail screen. Any button [sys_sg_button] records associated to the original master detail screen change to associate with the new form applet.

**Map migration**

Map applets did not use an item view to display fields in map cards in the Madrid release. The migration script creates an item view [sys_sg_item_view] record for each map applet using the Title, Tag, Sub-title, and Info fields from the original map applet.

**Calendar migration**

The migration script creates time span item stream [sys_sg_time_span_item_stream] records for each calendar, and associates the calendars original data item to the new item stream. The migration script also creates a form applet
[sys_sg_form_screen] record, and migrates the buttons from the calendars original embedded screen to the new form.

**Item streams and master items**

The migration script creates an item stream [sys_sg_item_stream] record for each screen in the scoped application. The original data item record associated with the legacy application changes to associate with the new item stream record. The script creates time span item stream [sys_sg_time_span_item_stream] records for each calendar screen, and location item stream [sys_sg_location_item_stream] records for map screens. These two tables extend from the item stream table, but are used specifically for these screen types.
Screen Cleanup

The following fields are no longer used in Screen records. The script removes these fields from call records on the Screen [sys_sg_screen] table.

- User Roles [application_roles]
- Order [order]
- Parent [parent]
- Parent table [parent_table]
- Data Item [sys_sg_data_item]
- Hidden [hidden]

In addition, the script also removes values from the following fields on Map screen [sys_sg_map_screen] records:

- Data item table [data_item_table]
- Title [title]
- Sub-title [subtitle]
- Info [info]
- Location [location]
- Tag [tag]
- Tag font color [tag_font_color]
- Tag background color [tag_background_color]
- Tag Style [tag_style]
- Phone [phone]
- Pin color type [pin_color_type]
- Pin color [pin_color]

The script removes values from the following fields on Master item [sys_sg_master_item] records:

- Table [table]
- Screen [screen]
- Condition [condition]
- Condition Order [condition_order]

The script removes the value in the Item View [item_view] field of Details screen [sys_sg_details_screen] records.

The script removes the value in the Item View [item_view] field of List screen [sys_sg_list_screen] records.

The script removes the value in the Data Item [data_item] field of Item View [item_view] records.

More Resources

For more information on the migration process, see the Mobile Migration Guide for New York on the ServiceNow community site. [https://community.servicenow.com/community?id=community_article&sys_id=f5121a33dba7f788ff8a345ca961957](https://community.servicenow.com/community?id=community_article&sys_id=f5121a33dba7f788ff8a345ca961957)

Run the mobile migration script

Run the mobile migration script to convert Madrid mobile applications you have created or modified to use the new mobile hierarchy.

Role required: admin
Mobile applications created in the Madrid release still work in the Paris release, but cannot be edited in Studio. To continue editing in Studio, and to take advantage of new features available in the Paris release, run the mobile migration script.

1. Navigate to **System Applications > Studio**.
2. Open a scoped application that you have created, or a base system application that you have modified.
3. Click **Upgrade** when prompted to start the migration script.

![Migrate to New York Version](image)

Troubleshooting mobile migration script results

Find solutions to common issues after running the mobile migration script.

Log error messages

The mobile migration script adds entries to the Log [syslog] table when it encounters an error. You can review these logs by navigating to **System Logs > System Log > All**. Listed here are errors the mobile migration script may add to the logs.

<table>
<thead>
<tr>
<th>Error Message</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invalid instance scope provided</td>
<td>If you see this message, the migration script was aborted. Run the migration script again to complete the migration. You can do rerun the migration script by reopening Studio and selecting the scope.</td>
</tr>
<tr>
<td>Please activate com.glide.mobile-employee first before migration.</td>
<td>The ServiceNow NowMobile App Screens and Applet Launcher [com.glide.mobile-employee] plugin must be active to run the migration script. Ensure that this plugin is active. If you see this message, the migration script was aborted. Run the migration script again to complete the migration.</td>
</tr>
<tr>
<td>Cannot perform migration task on customized record.</td>
<td>The record causing this error appears immediately after this message. A customization on this record prevented the migration script from changing this file. The migration script skips this file, and continues to run. The named file is inaccessible in Studio.</td>
</tr>
</tbody>
</table>
Collisions

Collisions can occur in base system applications that you have customized before the upgrade to New York or later versions. If the migration script detects any collisions, it prompts you to review them after the script execution completes.

Click the View Collisions button to view a filtered list of upgrade details [sys_upgrade_history_log] records. This list shows the records within the current scope that the upgrade process has skipped. To resolve a conflict, click a record on this list to open the record, then click the Resolve Conflicts button.

The Resolve Conflicts form shows the base system version of the record alongside the customized version Fields that are different between versions are highlighted with a darker background.
Use the arrow buttons to move values from one version to the other. After making your changes, click the **Save Merge** button to save your changes. You can also click the **Revert to Base System** button to discard your customizations and use the unmodified version of the record.

**Note:** Use an update set to capture the changes you make while resolving conflicts. You can use this update set to apply your changes in other instances. For details on using update sets see *System update sets*.
A common collision issue is Master detail [sys_sg_master_detail_screen] records. Master details records are no longer a part of the mobile schema as of the New York release. These records are replaced with new list [sys_sg_list_screen] and form [sys_sg_form_screen] screen records. They are normally deleted as part of the upgrade process, but if they have been customized, the script does not automatically delete them. If you have, for example, renamed a base system application, this kind of collision can occur.

To resolve the issue, check studio to make sure that your applet is still available and working as expected. Once you have confirmed that the applet is available, you can delete the master detail record.

Common migration issues

An applet is missing

After migration, your applets should be visible in the Applications tab in the navigation bar. If the applets do not appear, you can manually migrate these applets.

1. In Studio, open Mobile Studio > Application Menu in the application explorer, and select the app where you are missing an applet.
2. In the Navigation Tabs related list, click the Applications navigation tab.
3. Note the Applet Launcher associated with the Applications tab.
4. In Studio, open Mobile Studio > Applet Launchers, and open the applet launcher noted in the last step.
5. In the applet launcher form, select a UI section with the same name as the folder the missing applet was located in before the migration.
6. Find the missing applet in the All Applets list, and move it to the Selected Applets list.
7. Click Save.

A related list is missing from an applet

This issue may be the result of an outdated reference on the related lists mapping [sys_sg_related_list_map] table. You can re-associate the Destination Screen for your related list to resolve the issue.

1. In Studio, navigate to Mobile Studio > Applets, and select the applet with the missing segment.
2. Click the Form Screen tab.
3. Click Body.
4. Click the Related Lists button.
5. Check the list for items that have an empty Destination Screen value.
6. Click the list item, and select a value in the Destination Screen field.

Users are not prompted to enter input parameters in Field Service Management or ITSM applets

Normally parametrized applets prompt your users for a value. If you are no longer seeing this prompt after a migration, use these steps to correct the issue.

1. Open the UI Parameter [sys_sg_ui_parameter] list by entering sys_sg_ui_parameter.list in the filter navigator for your instance.
2. Find the parameter which is not generating a prompt for your users.
3. Check the value of the **Screen** field. If this field appears empty, it may be pointing to a Master-detail screen [sys_sg_master_detail_screen] record.

4. Update the field by selecting the applet [sys_sg_screen] record used by this parameter.

**Incorrect results for a customized Field Service Management or ITSM applet.**

This issue can occur if you have added a customer parameter to a base system.

1. Open the Screen Parameters mappings [sys_sg_screen_param_map] list by entering `sys_sg_screen_param_map.list` in the filter navigator for your instance.

2. Find the record with the **Item Parameter** field matching the item parameter you have added to your data item.

3. Check the value of the **Screen** field. If this field appears empty, it may be pointing to an unused Master-detail screen [sys_sg_master_detail_screen] record.

4. Update the field by selecting the applet [sys_sg_screen] record used by this parameter.

**Migration from New York and later releases**

Learn about migration from the New York to later releases.

Upgrading from the New York release to later releases of ServiceNow mobile does not require any special configuration steps. The upgrade process will have no impact on your existing mobile applications.

**Building and configuring mobile apps**

Build and configure a mobile experience for any of the three ServiceNow mobile apps.

**Requirements**

ServiceNow mobile is available for instances upgraded to Madrid or later releases.

The ServiceNow mobile platform, like any other application, is built using records on tables within your instance. While configuration can be done through the platform user interface, the ServiceNow mobile platform has its own experience built out in Studio. With Mobile Studio, you can use to quickly and easily create mobile experiences.

**Mobile tutorials**

Use tutorials to see how you can build or customize mobile applications suited to your business needs. These tutorials include step-by-step examples for creating mobile applications for common business cases.

- For a list of available tutorials on ServiceNow mobile, see [Mobile tutorials](#).

**Design considerations for ServiceNow mobile**

Before you begin building and customizing your applications for ServiceNow mobile, take a look at the design considerations documentation. This content contains information from mobile designers and product managers to help you design an application optimized for the mobile environment. View the design considerations documentation at [Design considerations for mobile apps](#).
Mobile Hierarchy

The ServiceNow mobile platform is designed using a hierarchy structure. At the top, are the three different mobile apps, Now Mobile, Mobile Agent, and Mobile Onboarding. Familiarize yourself with is hierarchy to deliver an exceptional mobile experience. Before getting into configuration, find more details about the mobile fundamentals and hierarchy at Mobile hierarchy.

Mobile Customization

Use Studio to modify base system mobile app components, or create your own components. Studio provides an environment where you can perform most customization and creation tasks that relate to mobile apps. For more details on Studio, see Mobile Studio.

To test your configuration, you can use the app that is installed on your mobile device. To get the mobile apps, go to the Apple App Store or the Google Play store and search for ServiceNow to download the available mobile apps.

Mobile tutorials

Use this reference to find all tutorials available in the ServiceNow mobile documentation.

Mobile parameter tutorials

Parameters are a way of creating a variable or a placeholder that is waiting for input from either the user or the database. The variable then queries the database or the user for more information. You can add parameters to a data item or an action item.

When you add parameters to a data item, the parameter looks for additional information before opening a screen. For example, you could create a data item that allowed users to filter incidents by priority. The data item parameter would hold a place in the Priority field, so when the user opened the screen they would may select the priority. For more information on configuring a data item with parameters, see Configure a parametrized data item.

When you add parameters to an action function, that parameter looks for information from the user or the database before updating a record. For example, you could create an action function that allowed a user to update the assignee field from a swipe action. The action function parameter would hold the space of the assigned to field. When a user uses the swipe action, they are prompted to select an assignee. For more information on creating an action function with parameters, see Create an action function with parameters.

Use the following tutorials to create action and data items with parameters and associate them with their respective functions.

Tutorial: Configure a data item with parameters

Use a parameter to query a user for a priority value, then display a list of incident records matching that priority. This example demonstrates how you can use a parameter to get a value from user input and apply that value to an applet.

Tutorial: Configure an action with parameters

Configure a swipe action on a list to assign a record from the list to a user. This example demonstrates how you can use a parameter to get user input and apply that input to a record using an action.

Tutorial: Configure a navigation function with parameters

Navigate from an incident record to the user record for the incident's caller. In this example demonstrates how you can use a parameter to store information from a record (the caller), and use that to navigate to a specific record on another table.
Tutorial: Configure a data item with parameters

Follow this tutorial to create an applet that allows the user to filter by priority before viewing a list. Use this tutorial as a guideline to help you understand how data items with parameters work in the mobile application.

In this example, you use a parameter to prompt a user for a priority, then display a list of incidents matching that priority. The parameter accepts the selection from the user and passes that information to a data item which is used to filter the list of incidents. Setting up an applet that uses a parametrized data item includes several steps. In this tutorial you will:

- Create a data item with parameters.
- Configure the applet to use the parametrized data item.

Before you begin, navigate to Studio (System Applications > Studio) and create an application for your mobile app. For more information on setting up Studio for your mobile app, see Create a mobile application using Guided Application Creator.

Create a data item with parameters
Create a parametrized data item for a list of incidents.

Role required: admin

In these steps you will create a parametrized data item. This means creating a parameter within the data item form, and adding the parameter to your data item's conditions, so the value of the parameter is used to filter your data item.

1. In Studio, navigate to Mobile Studio > Data Items.
2. Click the pop out icon to open the data items list in a tab.
3. Click Create a new data item.
4. In the New Data Item form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name for your data item. For example, Open Incidents</td>
</tr>
<tr>
<td>Table</td>
<td>Table from which the data items gets it's records. In this example, choose Incident [incident].</td>
</tr>
<tr>
<td>Description</td>
<td>Enter a description for your data item</td>
</tr>
<tr>
<td>Condition Type</td>
<td>Method used to create conditions for the data item. For this example, select Declarative.</td>
</tr>
</tbody>
</table>

5. In the Query condition, add the following conditions.
   - Active is true
   - State is one of New, In progress, or On hold

6. Click Save.
Data item completed using the preceding steps.

7. In the Parameter Definition section, click the add icon (]+

8. In the parameter definition form, fill in the fields.

Parameter definition form

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the parameter. For this example, use <strong>Priority</strong>. <strong>Note:</strong> You can have multiple parameters with the same name. Choose a name that you can distinguish easily when creating parameters.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of parameter. For this example, select <strong>String</strong>.</td>
</tr>
</tbody>
</table>
10. In the Data Item tab, from the condition builder, click the **AND** button that appears to the left of the **State** condition.

11. From the Choose field list, select **Priority**.

12. In the same row as the **Priority** field in the condition builder, click the contextual reference value icon.

13. Add the item parameter you created in previous steps.

14. Click **Save**.

Data item conditions using the parameter created in the preceding steps.

Assign the data item to an applet

After you create a data item with parameters, assign it to an applet.

In these steps you will create a list applet. The list applet uses the parametrized data item you created in the preceding steps.

1. In Studio, navigate to **Mobile Studio > Applets**.

2. Click the pop out icon to open the applets tab.

3. Click **Create an applet**.

4. On the Create an applet form, fill in the fields. Any fields not mentioned in the table can be left at their default values.

Create an applet form

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the applet. For this example enter <strong>Open incidents</strong>.</td>
</tr>
<tr>
<td>Icon</td>
<td>Icon displayed for this applet. Select any icon.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for your applet. Enter a description.</td>
</tr>
<tr>
<td>Choose a template</td>
<td>The template used for the applet. For this example, select <strong>List</strong>.</td>
</tr>
</tbody>
</table>
5. Click **Create New**.
6. In the Open incidents applet tab, from the Data Item list, select **Open incidents**.
7. In the Field Configurations section, use the list to add the following fields.
   - Number
   - Priority
   - Short description
   - Assigned to
   - Assigned to > Avatar
   - State
8. Click the **Form Screen** tab.
9. Above the **Selected Fields** list, click the **Replicate from primary** button.
10. Click **Save**.
11. Click the **List Screen** tab to return to the list configuration.
    The parameter settings section is now visible under the data item.
12. In the **User input parameter definition** section, click the add button.
13. In the **User Input** pop-up form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for your user input. For this example enter <strong>Priority</strong>.</td>
</tr>
<tr>
<td>Input type</td>
<td>The type of input. Select <strong>Choice List</strong>.</td>
</tr>
<tr>
<td>Table</td>
<td>Table from which your choice list will be created. Select <strong>Incident [incident]</strong>.</td>
</tr>
<tr>
<td>Field name</td>
<td>Field from which your choice list will be created. The available fields are determined by the table selected in the <strong>Table</strong> field. Select <strong>Priority</strong>.</td>
</tr>
<tr>
<td>Input style</td>
<td>Style for the user input. Select <strong>Inline</strong>.</td>
</tr>
<tr>
<td>Default value type</td>
<td>Default value type for the input. Select <strong>None</strong>.</td>
</tr>
<tr>
<td>Mandatory</td>
<td>Whether the user input is mandatory. Leave this option unselected.</td>
</tr>
<tr>
<td>Placeholder text</td>
<td>Text that appears in the input before the user makes a selection. Enter <strong>Priority</strong>.</td>
</tr>
<tr>
<td>Multi-select</td>
<td>Whether the user can select more than one value from the list. Leave this option unselected.</td>
</tr>
</tbody>
</table>
14. Click **Save**.
15. In the **Screen UI Parameter Mapping** section, click the add button.
16. In the **Screen UI Parameter mapping** pop-up form, fill in the fields as needed.
Screen UI Parameter mapping form

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item parameter</td>
<td>Select the parameter from the data item you created in previous steps.</td>
</tr>
<tr>
<td>UI parameter</td>
<td>Select the UI parameter you created in this applet in the previous steps.</td>
</tr>
</tbody>
</table>

17. Click **Save**.
18. In the Applet form, click **Save**.

When the user opens the applet, they should be prompted to select a priority level before opening the list.

**Tutorial: Configure an action with parameters**

Follow this tutorial to create a swipe action that allows a user to assign an incident from a mobile list. Use this tutorial as a guideline to help you understand how actions with parameters work in a mobile app.

In this tutorial, you create a parametrized action item to enable your users to assign an incident from a list using a swipe action. Creating a parametrized action item requires several steps. In this tutorial you will do the following:

- Create an **Assign to** action item with parameters.
- Create an action function to change the incident assignee.
- Associate the action function with an incident list applet.

Create an action item with parameters

Create an Assign to action item with parameters.

Role required: admin
In these steps you will create an action item with parameters.

1. In Studio, navigate to **Mobile Studio > Action items.**
2. Click the pop out icon to open the Action items list in a tab.
3. Click the **Create New** button.
4. On the Action Item form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the action item. For this example, enter <strong>Assign to</strong></td>
</tr>
<tr>
<td>Description</td>
<td>Description of the action item</td>
</tr>
<tr>
<td>Type</td>
<td>Type of action item. Select <strong>Update.</strong></td>
</tr>
<tr>
<td>Application</td>
<td>Scoped application for the action item. This field is automatically populated with the current application.</td>
</tr>
<tr>
<td>Table</td>
<td>Table used for the action item. Select <strong>Incident</strong> [incident].</td>
</tr>
<tr>
<td>Use current record as condition</td>
<td>Whether the action item will use the current record as its condition. Leave this field selected.</td>
</tr>
<tr>
<td>Set field values</td>
<td>Field values that this action item will set. Leave this field blank for now. You will create field values in later steps.</td>
</tr>
</tbody>
</table>

5. Click **Submit.**
6. In the **Item Parameters -> Writeback** related list, click **New.**
7. On the Item Parameters -> Writeback form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the item parameter. For this example, enter <strong>Assignee</strong></td>
</tr>
<tr>
<td>Type</td>
<td>Item parameter type. Select <strong>String.</strong></td>
</tr>
</tbody>
</table>

8. In the Item Parameter tab, click **Submit.**
9. In the Action Item tab, from the Set field values condition builder, search for **Assigned to.**
10. In the same row as the Assign to field in the condition builder, click the contextual reference value icon. 
11. Add the **Assignee** item parameter you created in previous steps.
12. Click **Update**.

*Create an action function*

After creating the action item with parameters, configure an action function to associate the action item with.

In these steps, you use the action item you created in previous steps in a function. You will use this function in a list applet to allow your users to assign incidents.

1. In Studio, navigate to **Mobile Studio > Functions > Actions**.
2. Click the pop out icon
   ![Pop out icon](image)
   to open the Action items list in a tab.
3. Click **Create New**.
4. In the **Function** form, fill in the fields.
Fields not mentioned in the table can be left at their default values.

**Function form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for the function. For this example, enter Assign.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the function</td>
</tr>
<tr>
<td>Type</td>
<td>Type of function. Select Action Item</td>
</tr>
<tr>
<td>Application</td>
<td>Scoped application for the function. This field is automatically populated with the current application.</td>
</tr>
<tr>
<td>Context</td>
<td>Context for the function. Select Record</td>
</tr>
<tr>
<td>Action Item</td>
<td>Action item used in the function. Select the Assign to action item created in the previous steps.</td>
</tr>
</tbody>
</table>

5. At the bottom of the form, open the **Condition** tab.
6. In the **Table** field, select Incident [incident].
7. At the bottom of the form, open the **Messages** tab.
8. In the **Success message** field, enter {{number}} has successfully updated.
9. Click **Submit**.
10. In the UI parameter related list, click **New**.
11. In the **UI Parameter** form, fill in the fields.
Fields not mentioned in the table can be left at their default values.

### UI Parameter form

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the UI Parameter. For this example, enter Assign.</td>
</tr>
<tr>
<td>Parameter type</td>
<td>The parameter type. Select <strong>Button</strong>.</td>
</tr>
<tr>
<td>Input Style</td>
<td>Determines where the input is handled inline or in a pop-up. Select <strong>Pop-up</strong>.</td>
</tr>
<tr>
<td>Button</td>
<td>The function used when the user taps or swipes to activate the function. This field is automatically filled with the function created in the previous steps.</td>
</tr>
<tr>
<td>Input source</td>
<td>Input source for the action. Select <strong>User Input</strong>.</td>
</tr>
<tr>
<td>Input type</td>
<td>The method the user uses to enter input. Select <strong>Choice List</strong>.</td>
</tr>
<tr>
<td>Table Name</td>
<td>The table related used in this action. Select <strong>Incident [incident]</strong>.</td>
</tr>
<tr>
<td>Field Name</td>
<td>The field used for this action. Select <strong>Assigned to</strong>.</td>
</tr>
</tbody>
</table>

12. **Click Submit.**
13. In the **Assign** function record, in the Action parameter mappings related list, click **New**.
   a) In the **Button** field, if the field is not completed already, search for **Assign**.
b) In the Item Parameter tab, search for Assignee.

c) In the UI parameter field, search for Assign.

d) Click Submit

![Completed item parameter form]

14. From the Assign action function, click Update.

Assign the action function to an applet

After you create an action function with a parametrized action item, assign the action item to an applet.

Now that you have an action function, you a list applet to apply your action to. In these steps you will create a data item, list applet, and swipe function definition.

1. Create a data item for open incidents.
   a) In Studio, navigate to Mobile Studio > Data Items.
   b) Click the pop out icon

![Open Incidents]

to open the data items list in a tab.

c) In the Data Items tab, click Create New.

d) In the New Data Item form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Title for the data item. For this example, enter Open Incidents</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Table</td>
<td>The name of the table you want the data item to pull information from. Select <strong>Incident [incident]</strong></td>
</tr>
<tr>
<td>Description</td>
<td>Optional additional details about the data item.</td>
</tr>
<tr>
<td>Condition Type</td>
<td>Whether the data item condition is declarative or scripted. Select <strong>Declarative</strong>.</td>
</tr>
</tbody>
</table>

e) In the query condition, add the following condition.
   - `<State> <is one of> <New, In Progress, On Hold>`

f) Click **Save**.

---

2. Create an applet for open incidents.
   a) In Studio, navigate to **Mobile Studio > Applets**.
   b) Click the pop out icon
      ![Pop out icon](image)
      to open the applets tab.
   c) Click **Create an applet**.

---
d) Create a new applet using the list template.

![New applet with the list template](image)

e) On the applet form, in the **Data Item** list, select **Open incidents**.

f) In the field configurations section, add the following fields to the list header.

- Number
- Priority
- Short description
- Assigned to
- Assigned to > Avatar
3. Add the action function to a swipe action for the Open incidents list.
   a) From the primary screen tab of your Open incidents applet, click **Functions**.
   b) In the Swipe Functions section, click the Add icon.
   c) In the Label field, type **Assign**.
   d) In the Function field, select **Action > Assign**.
   e) Click **Done**.
Action function example

**Tutorial: Configure a navigation function with parameters**

Follow this tutorial to create a navigation from a field on a mobile form. Use this tutorial as a guideline to help you understand how navigation functions with parameters work in the mobile application.
Creating a navigation function with parameters requires several steps. In this tutorial, you will do the following:

- Create a data item with parameters.
- Create an applet to navigate to.
- Create a navigation function.
- Associate the navigation function with an applet.

Before you begin, navigate to Studio (System Applications > Studio) and create an application for your mobile app. For more information on setting up Studio for your mobile app, see Create a mobile application using Guided Application Creator.

Create a data item with parameters

Create a data item containing your caller records.

Role required: admin

In these steps you create a data item that contains your caller records. The data item uses a parameter to filter the User [sys_user] list, so only the callers from your incident record appear.

1. In Studio, navigate to Mobile Studio > Data Items.
2. Click the pop out icon
   ( )
   to open the data items list in a tab.
3. In the Data Items tab, click Create New.
4. In the New Data Item form, fill in the fields.
5. In the query condition, add the following condition.
   • `<Active> <is> <true>`

6. Click **Save**.

7. In the Parameter Definition section, click the add icon.

8. On the Parameter Definition form, fill in the fields.

   **Parameter definition form**

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the UI parameter. For this example, enter <code>caller_sys_id</code>.</td>
</tr>
<tr>
<td>Note:</td>
<td>You can have multiple parameters with the same name, so choose a name that you can distinguish easily.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of parameter. Select <strong>String</strong>.</td>
</tr>
</tbody>
</table>

9. In the Item Parameter tab, click **Save**.

10. In the Data Item tab, click **And** to add a new row to your condition builder.

11. From the Choose field list, select **Sys ID**.

12. In the same row as the Sys ID field in the condition builder, click the contextual reference value icon.

13. Add the item parameter you created.

14. Click **Save**.
Create an applet to navigate to
Before you create a navigation function, you must have an applet to navigate to.

In these steps you create the applet that your users will see when they use the navigation function.

1. In Studio, navigate to Mobile Studio > Applets.
2. Click the pop out icon
   ![pop out icon]
   to open the ITSM application in a tab.
3. Click Create an applet.
4. Create a new applet using the list template.
5. In the Callers applet tab, from the Data Item list, select Caller.
6. Above the Item preview example, click Change Pattern.
7. In the List Item Patterns window, select pattern 9 in Pattern Set 1.
8. Click Done.
9. In the Field Configurations section, use the list to add the following fields to the list header.
   • Department
   • Name
   • Avatar
   • Title
10. Switch to the Form screen tab of the Callers applet.
11. Above the Selected Fields list, click the Replicate from primary button.
12. Click Save.
    A Parameter Setting section and sub sections appear under the Data item field.
13. In the User input parameters definition section, click the add icon.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the user input parameter. For this example, enter caller_sys_id.</td>
</tr>
<tr>
<td>Input Type</td>
<td>Type of input for the user input parameter. Select Text.</td>
</tr>
</tbody>
</table>
15. In the Screen UI Parameter Mapping section, click the add icon.
17. In the Screen UI Parameter mapping pop-up, click **Save**.
18. In the Callers applet, click **Save**.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item Parameter</td>
<td>Select the <code>caller_sys_id</code> item parameter created in the previous steps.</td>
</tr>
<tr>
<td>UI Parameter</td>
<td>Select the <code>caller_sys_id</code> UI parameter created in previous steps.</td>
</tr>
</tbody>
</table>
Data and Field Functions

Data Item

Caller

Parameter Setting

Screen UI Parameter Mapping

These parameters were defined in the data item you selected above and need to be fulfilled (using UI parameters) in order for the action item to work.

<table>
<thead>
<tr>
<th>Data Item Parameter</th>
<th>UI Parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>caller_sys_id</td>
<td>caller_sys_id</td>
</tr>
</tbody>
</table>

UI Parameter

User Input Parameter Definition

If you define user input UI parameters, and map them to the data item parameters, your end user will be asked to fulfill these inputs with values before this screen is launched.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>caller_sys_id</td>
<td>text</td>
</tr>
</tbody>
</table>

AutoFill Parameter Definition

If you define autoFill UI parameters, and map them to the data item parameters, then those data item parameters will be automatically fulfilled by the system when this screen is called.

Field Configurations

Assign a field for each header element (e.g. E1, E2, ...)

List Item

<table>
<thead>
<tr>
<th>List Item</th>
<th>Change Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Department</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>Avatar</td>
<td></td>
</tr>
<tr>
<td>Title</td>
<td></td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Configure a navigation function with parameters

After you create an applet to navigate to, create a navigation function.

1. In Studio, in a mobile application, navigate to Mobile Studio > Functions > Navigations.

2. Click the pop out icon to open the navigation function list in a tab.

3. In the Navigations tab, click Create New.


<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for the navigation function. For this example, enter Navigate to caller</td>
</tr>
<tr>
<td>Destination Type</td>
<td>Whether the destination is an applet or applet launcher. Select Applet.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional description for the navigation function.</td>
</tr>
<tr>
<td>Destination</td>
<td>Destination for the application. Select the Callers applet you created in previous steps.</td>
</tr>
<tr>
<td>Context</td>
<td>Context for the navigation function. Select Record.</td>
</tr>
<tr>
<td>Table</td>
<td>Table for the navigation function. Select Incident [incident]</td>
</tr>
</tbody>
</table>

5. Click Save.

After saving, the Parameter Setting section appears on the form.

6. In the Parameter Setting, click the first row in the table.

<table>
<thead>
<tr>
<th>Parameter name</th>
<th>Type</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>caller_sys_id</td>
<td>Field</td>
<td></td>
</tr>
</tbody>
</table>

Parameter setting section

7. From the Redirection Parameter table, fill in the fields as needed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Destination UI parameter</td>
<td>Automatically filled with the name of the parameter created for the data item.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Type</td>
<td>Where the information for the parameter comes from. Since your users access this navigation from the Caller field on the incident form, select Field.</td>
</tr>
<tr>
<td>Source field</td>
<td>The field you want the user to navigate from. Select Caller.</td>
</tr>
</tbody>
</table>

8. Click Save.
9. In the main navigation tab, click Save.

**Associate the navigation function with an applet**

Now that you have a navigation function and destination applet, you need an incident applet to act as a starting point.

You can use any applet for these steps, including one from a previous tutorial. The first steps include instructions on creating an applet. Make sure that the Caller field is included as part of the Body on the details screen of the app. Otherwise, you do not have a field to assign the navigation to.

1. Create a data item for open incidents.
   a) In Studio, navigate to Mobile Studio > Data Items.
   b) Click the pop out icon to open the data items list in a tab.
   c) In the Data Items tab, click Create New.
   d) In the New Data Item form, fill in the fields.

   **Data item form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Title for the data item. For this example, enter Open Incidents</td>
</tr>
<tr>
<td>Table</td>
<td>The name of the table you want the data item to pull information from. Select Incident [incident]</td>
</tr>
<tr>
<td>Description</td>
<td>Optional additional details about the data item.</td>
</tr>
<tr>
<td>Condition Type</td>
<td>Whether the data item condition is declarative or scripted. Select Declarative.</td>
</tr>
</tbody>
</table>

   e) In the query condition, add the following condition.

   *  \(<\text{State}\> \text{<is one of> <New, In Progress, On Hold>}\)

   f) Click Save.
2. Create an applet for open incidents.
   a) In Studio, navigate to **Mobile Studio > Applets**.
   b) Click the pop out icon
      
      to open the applets tab.
   c) Click **Create an applet**.
   d) Create a new applet using the list template.
e) On the applet form, in the **Data Item** list, select **Open incidents**.

f) In the field configurations section, add the following fields to the list header:

- Number
- Priority
- Short description
- Assigned to
- Assigned to > Avatar
- State
Field configuration for the list applet

g) Click **Save**.

3. From the **Form Screen** tab of the applet, select **Body**.
4. Use the list to add the Caller field to the applet.
5. Click **Save**.
6. In the Details tab, open the Functions section.
7. In the **Field functions** section, click the Add icon.
8. From the Field Functions window, in the **Field** list, select **Caller**.
9. In the Function list, select **Navigation > Navigate to caller**.
10. Click **Save**.
Navigate to a caller example

Design considerations for mobile apps

Use mobile design and product principles to give your users the best mobile experience.

Before designing your mobile experience, take a look at the Building Mobile Apps with ServiceNow guide. The purpose of this document is to provide ServiceNow mobile app creators product principles and design considerations to create consumer-grade mobile experiences.

Download the Building Mobile Apps guide at https://community.servicenow.com/community?id=community_article&sys_id=6dac508dbaf8905ed4a851ca961978

Mobile hierarchy

Learn the components of ServiceNow® mobile and how they work together to assist you in configuring, modifying, and creating applications.

Components of the ServiceNow mobile framework

This image represents the structure of the ServiceNow mobile framework. The next sections detail specific areas of the overall hierarchy, and descriptions of individual components.
Native clients

Native client

Your users access ServiceNow® mobile from a native client such as ServiceNow Agent, Now Mobile, or Mobile Onboarding. After logging in, your users will see an applet launcher screen, and a navigation bar.

Navigation bar
Each app has a navigation bar that appears at the bottom of the screen. A navigation bar can have up to five icons, called navigation bar tabs. By default, navigation bars have a notification and settings navigation bar tab. For more information on navigation bars, see Navigation bar.

Navigation bar tabs

Each tab in the navigation bar represents an applet or applet launcher. When you add more than five tabs to the navigation bar, a More tab appears. Tapping the More tab opens a list view showing additional tabs.

For information on how to create a navigation bar, see Configure the navigation bar.

For information on how to create navigation bar tabs, see Configure an applet launcher tab.

Applet launchers

Applet launchers serve as a landing pages or home pages for your users. Applet launchers contain a configurable header, and UI sections to provide access to applets in several formats. You can also configure applet launchers with a global search bar, and quick actions, which give your users access to commonly used functions.

For more detail on applet launchers, see Applet launchers.

Applets

Applets provide your users a method to view and modify data on your instance. Applets can display information as lists, maps, record forms, and other formats. You can find more detail on applet components in the next section.
Applets

Segments

Applets contain one of more segments, which display information from your instances to your users. Segments represent the lists, calendars, maps, and forms your users see within the app. If an applet has more than one segment, your users can switch between segments using a tabbed interface.
Segments records for lists, calendars, and maps are located on the item stream segment [sys_sg_item_stream_segment] table. Segments records for forms are on the form screen segment [sys_sg_form_segment] table.

Icons

Each applet has an icon. This icon represents the applet when it is displayed in an applet launcher or the navigation bar.

Icon records are located on the Icons [sys_sg_icon] table.

For more information on icons, see *Mobile icons*.

Item streams

An item stream is the source for the data shown in your applet. An item stream gets its data from a single source, called a data item. You can associate more than one item stream to a segment to include data from multiple tables. For example, you could create two item streams to display items from both the incident [incident] and request item [sc_req_item] tables in a single list.

Item streams are also associated with one or more master items, which provides a pattern controlling how the data appears in your segment.

Item stream records are located in the Item Stream [sys_sg_item_stream] table.

Data items

Data items provide the data presented in an applet. A data item is a dataset correlated with a table in an instance. A data item can include a filter condition to restrict what data the item returns. Associate data items with applets so that the applets can transform the dataset into human-readable information.

Data items are located on the Data Items [sys_sg_data_item] table.

For more detail on data items, see *Data items*.

Master items

Master items provide a pattern for data in your applet, and control how your data appears within a segment. For more detail on how a master item controls the appearance of your data, see the master item section.

Master items are located on the Master Item [sys_sg_master_item] table.

More information

For more information, See:
Segments

Applets contain one of more segments, which display information from your instances to your users. Segments represent the lists, calendars, maps, and forms your users see within the app. If an applet has more than one segment, your users can switch between segments using a tabbed interface.
Segments records for lists, calendars, and maps are located on the Item Stream Segment [sys_sg_item_stream_segment] table. Segments records for forms are located on the Form Screen Segment [sys_sg_form_segment] table.

Screens

Screen types determine what an applet looks like and how your users are able to interact with it. You can create applets with these screen types:

- Calendar
- Chart
- Employee directory
- Form
- Grouped list
- List
- Map
- URL

In addition to these types, you can add the following screens to segments in your form applet:

- Details screen
- Activity stream screen
- Related list screen
- Embedded list screen

Functions

Your users can use functions to perform tasks in the mobile app such as assigning a task, or navigating from a record to related record. Actions can also interact with your mobile device to send emails, navigate using map software, or make a phone call.

Functions are located on the Function [sys_sg_button] table.

For more information on functions, see Mobile functions.
Master items

Item view

Item views determine the formatting and appearance for fields in your applets. When you select a screen template when creating an applet, you are actually selecting the item view. Item views use JSON to determine the configuration for fields in your applet.

Item views are located on the Item view [sys_sg_item_view] table.
Conditions

Master items include condition a **Condition** field. This field sets filter conditions which restrict what records appear when a master item is used. The conditions used by the master item are used in addition to any conditions in place in the data item.

Table

Each item view record contains a **Table** field. This field determines the content the item view appears.

Pattern

Each item view record contains an **Item view JSON** field that contains a JSON. This JSON defines a pattern to control how data appears on the screen. The instance automatically creates this JSON when you select an applet template while designing applets in Studio. It is possible to manually create a pattern within an item view record, but Studio may not recognize these custom patterns.

UI styles

UI styles change the color of fields in an applet. You can associate an item view record with one or more UI styles. Each UI style applies to a single table, and only applies under conditions you set within the UI style record.

UI styles are located on the UI Styles [sys_sg_ui_style] table.

For details on using UI styles, see **Mobile UI styles**.

Screen UI Policies

Screen UI policies are similar to the UI policies that are used on forms in the instance, but screen UI policies are designed for the mobile app. Policies contain a set of conditions that you can use to determine when the policy applies. When a condition you have defined triggers a screen UI policy, all UI policy rules that are associated with that policy apply.

For information on creating screen UI policies, see **Screen UI policies**.

Mobile Studio

Use Studio to create and modify mobile components for use in any of the ServiceNow mobile apps.
Use the application explorer to browse mobile components

The Application Explorer appears on the left edge of the screen and displays your mobile components, such as applets, application launchers, and functions. Open your items by clicking them, or select a section, and use the pop-out icon to display a card view of the components in that section.
Use the card view to filter and search for specific components

<table>
<thead>
<tr>
<th>Applets</th>
<th>Sort by</th>
<th>Updated Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>All</td>
<td>Updated Time</td>
<td></td>
</tr>
</tbody>
</table>

- **Create an Applet**
- **Work Order**
- **Part Requirements**
- **Task SLA**
- **Caller**
- **Work Order Tasks**
- **Time Worked**
- **Upcoming Work Orders**
- **Asset Usage**
- **Work Order Task**
The card view displays each record in the selected section as a card. Each card displays the title of the component, as well as its creator, last update time, and whether the component is active.

Use the **Sort by** field to sort your components by name or update time. You can also use the search field to filter your results by name.
Create and modify mobile components
Use studio to design layouts for your applets and launchers, set permissions, determine offline-mode availability. See documentation on specific components for instructions on how you can use Studio to make changes and create modify your mobile experience.

Create a mobile application using Guided Application Creator

Use the Guided Application Creator in Studio to create a base scoped application and mobile app components. These components can be used in any of the ServiceNow mobile apps.

Role required: admin

Studio is a development environment where your application developers can work on custom applications in one centralized location. Much of the configuration for mobile apps in this section take place in Studio (System Applications > Studio). For more information about Studio, see ServiceNow Studio.

Role required: admin

1. Navigate to System Applications > Studio.
2. In Studio, click Create Application or select an existing application from the list.
   The application that you create here is a new scoped application which will contain your app. Scoped applications help restrict data and application files to just this one application. For more information on scoped applications, see Application scope.
   You can also access the Guided Application Creator outside Studio by navigating to System Applications > My Company Applications, and then click Create new.
   Studio opens a Guided Application Creator window where you create your application.
3. Optional: If you are launching Guided Application Creator for the first time, click Let's get started on the welcome screen.
4. Follow the steps on the screen to create a name, description, and logo for your application, and then click Create.
5. Optional: In the Roles field, select the roles to associate with your app.
   Users with the selected roles can access your application. If you have selected no roles, users with any role will have access to the application.
6. Optional: Click Create new role to create a new role to associate to the application.
7. Click Continue when you are finished defining roles for your application.
8. Select Mobile as the format for this application, and then click Continue.
9. Select or create tables that you want to use in your mobile app.
   The Guided Application Creator can create list applets for these tables. If you create new applets later, you can will not be limited to the tables you select here.
10. Click Done with Tables when you are finished adding tables to your application.
11. Click the Start button to create applets for your selected tables using the Guided Application Creator.
    If you do not want to create applets at this point, you can click X in the upper right corner of the window to close the Guided Application Creator and return to Studio.
12. Fill in the fields in the Guided Application Creator form as needed:

Guided App Creator Form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for the app. The applet launcher created in this process will use this name.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for the app.</td>
</tr>
<tr>
<td>Tables</td>
<td>Tables for which Guided Application Creator will create applets.</td>
</tr>
</tbody>
</table>

© 2020 ServiceNow, Inc. All rights reserved. ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
### Table

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roles</td>
<td>Roles for your applets. Users with the selected roles can access the created applets.</td>
</tr>
</tbody>
</table>

13. **Click Create.**
   The Guided Application Creator will create a list and form for each of the selected table, including **New**, **Edit**, and **Delete** functions for each table.

14. **Click the Done with apps button, and then the Done button to return to Studio**

You now have a new application that you can access in Studio.

Create apps to add functionality to your mobile app using Guided Application Creator, or close Guided Application Creator and create apps using Studio.

### Navigation bar

Users can quickly access applet launchers, applets, settings, and notifications by using the navigation bar in the mobile app.

The navigation bar consists of these components:
- Navigation bar
- Navigation bar tabs
- Applet launcher tab
- Applet tab
The navigation bar appears at the bottom of each mobile app. Within the navigation bar, you can create navigation bar tabs to give users access to applet launchers or applets.

**Note:** The navigation bar in each mobile app is pre-configured with **Notifications** and **Settings** navigation bar tabs.
When you add more than five tabs to the navigation bar, a More tab appears. Tapping the More tab opens a list view showing additional tabs.
Applet launcher tabs

Use an applet launcher tab to allow a user to access content within an applet launcher.

For details on creating applet launcher navigation tabs, see Configure an applet launcher tab.
Applet tabs

Use an applet tab to allow a user access to a single applet, such as a map, calendar, or list of contacts.

For details on creating applet navigation tabs, see Configure an applet launcher tab.
Configure the navigation bar

Configure the navigation bar that appears at the bottom of the mobile app. Your users use the navigation bar to quickly navigate to applets, applet launchers, settings, and notifications.

Role required: admin

1. In Studio, use the navigation explorer on the left edge of the screen to navigate to Mobile Studio > Application menu. Select an app to open its navigation bar record.
   
   Each mobile app has a navigation record with a matching name. For example, the Mobile Agent app has a navigation record called Mobile Agent.

2. On the form, modify the fields as needed.

   Navigation form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the app that uses this navigation bar. For example, Mobile Agent is the navigation bar for the ServiceNow Agent app.</td>
</tr>
<tr>
<td>Color</td>
<td>The hexadecimal value for the color of the navigation bar.</td>
</tr>
<tr>
<td>Quick Actions Menu Color</td>
<td>The hexadecimal value for the color of the quick actions icon that appears on the screen.</td>
</tr>
<tr>
<td>Application</td>
<td>Scoped application for this record. This field automatically populates with the current application's scope.</td>
</tr>
<tr>
<td>Legacy application</td>
<td>Enabled for applications that are not migrated to the new mobile framework introduced in the New York release.</td>
</tr>
</tbody>
</table>

To add a new tab to your navigation bar, see Configure an applet launcher tab or Configure an applet tab.

Configure an applet launcher tab

Configure a tab on your navigation bar to direct users to an applet launcher.

Role required: admin

Configure an applet launcher tab to direct your users to a commonly used applet launcher.

1. Navigate to System Applications > Studio.

2. In Studio, use the navigation explorer on the left edge of the screen to navigate to Mobile Studio > Application menu. Select an app to open its navigation bar record.
   
   Each mobile app has a navigation record with a matching name. For example, the Mobile Agent app has a navigation record called Mobile Agent.

3. In the Related Links section of the form, click Create New Tab to add a new tab to your navigation bar.
   
   The navigation tab map form displays in a pop-up window. The navigation tab map record is what ties a navigation tab to the navigation bar for a specific mobile app. You will create a record for the navigation tab itself in steps 5 through 10.

4. On the navigation map tab form, fill in the fields as needed.
5. Click the reference icon next to the **Navigation Tab** field.

6. Click **New** to create a new navigation tab record.

7. In the **Select Navigation Type** pop-up, select **Applet Launcher Tab**.

8. Click **OK**.

    The navigation tab form displays after you select a type.


### Tab form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Label</strong></td>
<td>Name that displays for your navigation tab</td>
</tr>
<tr>
<td><strong>Application</strong></td>
<td>Scoped application for this record. This field automatically populates with the current application's scope.</td>
</tr>
<tr>
<td><strong>Applet Launcher</strong></td>
<td>Applet launcher that displays when a user taps on the tab. This field only appears on applet launcher tab records. For detail on creating an applet launcher, see <strong>Applet launchers</strong>.</td>
</tr>
<tr>
<td><strong>Icon</strong></td>
<td>The icon that displays for your navigation tab</td>
</tr>
<tr>
<td><strong>Active</strong></td>
<td>Whether the navigation tab is active. Inactive navigation tabs do not appear on your navigation bar.</td>
</tr>
</tbody>
</table>

10. Click **Submit**.

    After you click submit on the Navigation Tab form, you are returned to the Navigation Tab Map form. The navigation tab you created in steps 5 through 10 should appear in the **Navigation Tab** field.

11. In the Navigation tab map pop-up, click **Submit**.

    **Note:** Modifications to navigation bars are not visible to your users until they log out and back into the app.
Configure an applet tab

Configure a tab on your navigation bar to direct your users to an applet.

Role required: admin

Create an applet navigation tab to direct your users to a commonly used applet, such as a map, calendar, or list of contacts.

1. Navigate to System Applications > Studio.
2. In Studio, use the navigation explorer on the left edge of the screen to navigate to Mobile Studio > Application menu. Select an app to open its navigation bar record.

   Each mobile app has a navigation record with a matching name. For example, the Mobile Agent app has a navigation record called Mobile Agent.

3. In the Related Links section of the form, click Create New Tab to add a new tab to your navigation bar.

   The navigation tab map form displays in a pop-up window. The navigation tab map record is what ties a navigation tab to the navigation bar for a specific mobile app. You will create a record for the navigation tab itself in steps 5 through 10.

4. On the navigation map tab form, fill in the fields as needed.

   **Navigation tab map form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Navigation</td>
<td>The Navigation bar where this tab appears. This field automatically populates with your selected Navigation.</td>
</tr>
<tr>
<td>Application</td>
<td>Scoped application for this record. This field automatically populates with the current application's scope.</td>
</tr>
<tr>
<td>Navigation Tab</td>
<td>The navigation tab record.</td>
</tr>
<tr>
<td>Order</td>
<td>The order in which the tab appears on the navigation bar. Tabs display from left to right, from the lowest order number to the highest.</td>
</tr>
</tbody>
</table>

5. Click the reference icon next to the Navigation Tab field.

6. Click New to create a new navigation tab record.

7. In the Select Navigation Type pop-up, select Applet.

8. Click OK.

   The navigation tab form displays after you select a type.


   **Tab form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>Name that displays for your navigation tab</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Application</td>
<td>Scoped application for this record. This field automatically populates with the current application's scope.</td>
</tr>
<tr>
<td>Screen</td>
<td>Select the applet that displays when a user taps on the tab. For detail on creating an applet, see <em>Create an applet</em>.</td>
</tr>
<tr>
<td>Icon</td>
<td>The icon that displays for your navigation tab</td>
</tr>
<tr>
<td>Active</td>
<td>Whether the navigation tab is active. Inactive navigation tabs do not appear on your navigation bar.</td>
</tr>
</tbody>
</table>

10. **Click Submit.**

    After you click submit on the Navigation Tab form, you are returned to the Navigation Tab Map form. The navigation tab you created in steps 5 through 10 should appear in the **Navigation Tab** field.

11. **In the Navigation tab map pop-up, click **Submit**.**

    **Note:** Modifications to navigation bars are not visible to your users until they log out and back into the app.

**Applet launchers**

Applet launchers serve as landing pages or home pages. Using an applet launcher, you can access applets in various formats, as well as search, do quick actions, and find user information.

Applet launchers serve as landing pages. An applet launcher servers as a container to give your users access to functions and information.
Applet launcher components

When creating a new applet launcher, you configure the following few components within the applet launcher itself:

**Headers**

The header of the applet launcher defines how the title of the screen appears and what information is shown in the header.

**Global search**

Global search give your users the ability to search for information within the defined search sources from a mobile app.

**UI sections**

UI sections contain the applets, charts, and record information on your applet launcher pages.

**Quick actions**

Quick actions give users the ability access commonly used functions from an applet launcher page.

Applet launcher headers

The header of the applet launcher defines how the title of the screen appears and what information is shown in the header. The available header types are **Home** and **Generic**.
Header

**Home Header Type**

Use the **Home** header type for the applet launcher that a user sees when they log in to a mobile app. The title of a **Home** applet launcher is customizable and can automatically includes the name of the user that is logging in to the mobile app.

The avatar or initials for the logged-in user appear in an icon in the upper right corner of the screen. The user can tap this icon to open a form applet that contains the current user profile. You can define the applet for this purpose when you create the applet launcher.

For steps detailing home header configuration, see [Create a home applet launcher header](#).

**Generic Header Type**

Use the **Generic** header type for applet launchers that are not designed to be used as home pages. The title of a **Generic** applet launcher is a static string that you define.

You can optionally configure the **Generic** applet launcher to display an icon in the upper right corner of the screen. If you choose this option, you can select the icon and define which applet opens when your users tap this icon.

For steps detailing generic header configuration, see [Create a generic applet launcher header](#).

---

**Create a generic applet launcher header**

Create a generic applet launcher header for launchers that do not serve as a user’s homepage. Generic applet launchers have a static title. The launcher displays a configurable icon in the upper right corner.

Role required: admin
1. Navigate to **System Applications > Studio**.
2. Select your application.
3. In the Application Explorer, navigate to **Mobile Studio > Applet Launchers** and select **Applet Launchers**.
4. Click the pop-out icon
   ![Pop-out icon](image)
   that appears to the right of **Applet Launchers**.
5. In the applet launchers list, click **Create an Applet Launcher**.
6. In the Create an Applet Launcher window, enter a name for your launcher.
7. Select **Available offline** to make this launcher available to users in offline mode.
8. Click **Create New**.
9. In the **Header type** field, select the **Generic** header type.

   **Note**: For more information on applet launcher header types, see **Applet launcher headers**.

10. On the form, fill in the fields.
Note that some of the listed fields appear only for a specific header type.

### Applet launcher form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Title that appears at the top of the launcher. This field appears only when the <strong>Header type</strong> is <strong>Generic</strong>.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Include Icon</td>
<td>Option for including an icon in the upper right corner of the launcher page to enable users to quickly go to another screen within your application. This field appears only when the <strong>Header type</strong> is <strong>Generic</strong>.</td>
</tr>
<tr>
<td>Icon</td>
<td>Icon displayed in the applet launcher. The color of this icon is not configurable. This field appears when you select the <strong>Include Icon</strong> option.</td>
</tr>
<tr>
<td>Destination</td>
<td>Applet to launch when a user taps the icon. This field appears only when you select the <strong>Include Icon</strong> option.</td>
</tr>
<tr>
<td>Include Search</td>
<td>Option to enable global search on your applet launcher to give users the ability to quickly find people, catalog items, and knowledge base articles in your instance.</td>
</tr>
<tr>
<td>Search Criteria</td>
<td>A previously created global search configuration record used for search. For more information, see <a href="#">Configure search criteria</a>. This field appears only when you select the <strong>Include Search</strong> option.</td>
</tr>
<tr>
<td>Group Search Results</td>
<td>Group search results by categories as defined in the <strong>Search Criteria</strong> field. This field appears only when you select the <strong>Include Search</strong> option.</td>
</tr>
<tr>
<td>Photo Search</td>
<td>Option to perform a search by photo recognition. For more information, see <a href="#">Configure photo search</a>. This field appears only when you select the <strong>Include Search</strong> option.</td>
</tr>
</tbody>
</table>

11. **Click Save.** By default the applet launcher is activated.

   Click the **Active** button at any time, to either activate or deactivate the applet launcher.

   Click the roles icon

   ![roles_icon](#)

   to select roles required to view this applet launcher. If the selected role column is empty, the applet launcher has no role restrictions.

**Create a home applet launcher header**

Create a generic applet launcher header for launchers that serve as a home page for the user. You can configure the title to include the user's name. The launcher icon in the upper right corner displays the user's image or initials. You can configure the icon to link to the user's profile.

Role required: admin
1. Navigate to System Applications > Studio.
2. Select your application.
3. In the Application Explorer, navigate to Mobile Studio > Applet Launchers and select Applet Launchers.
4. Click the pop-out icon that appears to the right of Applet Launchers.
5. In the applet launchers list, click Create an Applet Launcher.
6. In the Create an Applet Launcher window, enter a name for your launcher.
7. Select Available offline to make this launcher available to users in offline mode.
8. Click Create New.
9. In the Header type field, select Home.

   **Note:** For more information on applet launcher header types, see Applet launcher headers.

10. On the form, fill in the fields.
Note that some of the listed fields appear only for a specific header type.
Applet launcher form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Text</td>
<td>Text to appear before the user’s name that is used to create a title for the launcher. This field appears only when the <strong>Header type</strong> is <strong>Home</strong>.</td>
</tr>
<tr>
<td>Variable</td>
<td>Value used for the user name format in the <strong>Home</strong> header type. The available options are:</td>
</tr>
<tr>
<td></td>
<td><strong>First Name</strong> Displays the current user's first name as defined in the <strong>First Name</strong> field in the user's User [sys_user] record.</td>
</tr>
<tr>
<td></td>
<td><strong>Last Name</strong> Displays the current user's last name as defined in the <strong>Last Name</strong> field in the user's User [sys_user] record.</td>
</tr>
<tr>
<td></td>
<td><strong>Full Name</strong> Displays the current user's full name as defined in the <strong>Full Name</strong> field in the user's User [sys_user] record.</td>
</tr>
<tr>
<td></td>
<td><strong>None</strong> Displays no name. The header title will show only the content of the <strong>Text</strong> field. This field appears only when the <strong>Header type</strong> is <strong>Home</strong>.</td>
</tr>
<tr>
<td>Include Search</td>
<td>Option to enable global search on your applet launcher to give users the ability to quickly find people, catalog items, and knowledge base articles in your instance.</td>
</tr>
<tr>
<td>Search Criteria</td>
<td>A previously created global search configuration record used for search. For more information, see <strong>Configure search criteria</strong>. This field appears only when you select the <strong>Include Search</strong> option.</td>
</tr>
<tr>
<td>Group Search Results</td>
<td>Group search results by categories as defined in the <strong>Search Criteria</strong> field. This field appears only when you select the <strong>Include Search</strong> option.</td>
</tr>
<tr>
<td>Photo Search</td>
<td>Option to perform a search by photo recognition. For more information, see <strong>Configure photo search</strong>. This field appears only when you select the <strong>Include Search</strong> option.</td>
</tr>
<tr>
<td>Employee Profile</td>
<td>Applet to launch when a user taps the Employee icon in the upper right corner of the home page. This icon appears as the user's avatar image or initials. This field appears only when the <strong>Header type</strong> is <strong>Home</strong>.</td>
</tr>
</tbody>
</table>

11. Click **Save**. By default the applet launcher is activated.

   Click the **Active** button at any time, to either activate or deactivate the applet launcher.

   Click the roles icon

   ![Roles icon]
to select roles required to view this applet launcher. If the selected role column is empty, the applet launcher has no role restrictions.

**Global search for mobile**

Give your users the ability to search for information within the defined search sources from a mobile app.
Global search bar

You can include a global search bar on your applet launcher. Use global search to give your users the ability to quickly find information within the defined search sources. Examples include people, catalog items, and knowledge base articles in your instance.

For more detail on search in the mobile platform, see Mobile search.
Search criteria

Create search criteria for your applet launcher to help your users find related information quickly. You can configure which search sources the search uses, as well as options like search result limits. For detail on configuring custom search criteria, see *Configure search criteria*.

Voice search

Voice search enables your users to use global search using the speech recognition feature of their mobile device. Users access voice search by tapping the microphone icon ( ), that appears on the right side of a search bar. For more information on configuring voice search, see *Enable voice search*.

Photo search

Photo search enables users to use global search by taking a picture. The Google Vision API identifies the picture and uses that identification as search query. Users access photo search by tapping photo icon ( ) that appears in your applet launcher search bar. For more information on photo search, see *Mobile photo search*.

Enable global search in your applet launcher

Enable global search on your applet launchers to give your users the ability to quickly find information within the defined search sources.

Role required: admin

1. Navigate to **System Mobile > Applet Launcher**
2. Open the record for the applet launcher where you want to add search functionality.
3. In the **Header** tab, select **Homepage Search - Catalog, Knowledge, People** in the **Search Configuration** field.
4. Click **Save**.

Configure global search options for your applet launcher

Configure global search options to control how your app presents search results to your users.

Role required: admin

1. Navigate to **System Mobile > Applet Launchers**.
2. Open the record for the applet launcher where you want to add search functionality.
3. In the **Search Configuration** field, click the reference icon to open the record preview, and then click **Open Record**.
4. In the **Placeholder** field, enter the text that you want to appear in the search bar before a user enters a value into the field.

The placeholder text appears in the search bar before your users enter a search query.
5. Select **Interleave Result** to display interleaved search results.
   If the option is not selected, search results are separated by the search source.
6. Click **Save**.
7. In the **Search Context Configuration** field, click the reference icon
to open the record preview, and then click **Open Record**.
8. In the **Search Results Limit** field, enter the maximum allowed search results.
9. In the **Suggestions Limit** field, enter the maximum allowed search suggestions.
10. Click **Save**.

Enable voice search so that your users can search using native speech recognition. For more details, see .

*Populate search suggestions in the mobile apps*
If you are upgrading from a previous release, run a script to populate search suggestions with data from a platform
search table to provide search suggestions to your users. Alternatively, you can wait until users search for keywords
instead of running this script.

Role required: admin

In new instances, the search suggestions are enabled by default. In upgraded instances, you must enable the search
suggestions. For more information, see **Enable search suggestions**.

The Now Platform collects search data and analytics that generate search suggestions. If you are upgrading from a
previous release, the search analytics do not contain any data yet. To immediately provide suggestions to your users,
you can populate the search suggestions using knowledge, catalog, and user search records from the Text Searches
[text_search] table.

Search suggestions improve over time as more people use the app. Search analytics and suggestions is a Now
Platform feature. For more information, see **Search analytics and suggestions**.

**Caution:** Populating search suggestions can be a resource-intensive task that may take a while to complete. Do not
run this script during peak hours. Populating search suggestions is not supported on domain-separated instances.

1. Navigate to **System Definition > Scheduled Jobs**.
2. Open the **Populate Suggestions to avoid Cold Start - NowMobile App** scheduled job.
3. Activate the record and select **Execute Now**.
   Running this scheduled job populates the Search Events [sys_search_event], Search Source Events
   [sys_search_source_event], and Search Suggestions [sys_search_suggestion] tables with records from the Text
   Searches [text_search] table.

Users see suggestions when they start typing in a search field in the Now Mobile app.
Configure search criteria

Create a custom search for your applet launcher to help your users find related information quickly. Administrators can configure which search sources the search uses, as well as options like search result limits.

Configuration for search consist of linking the following records:

**Applet launcher**

The applet launcher is where the search bar appears within your mobile application.

**Global search configuration**

The global search configuration [sys_sg_global_search] record is what associates your applet launcher with a search context configuration and a master item. This record is also where you can configure placeholder text, and enable photo search.

**Master item**

Master items provide a pattern for data in your applet, and control how your data appears within a segment. In this case, the appearance of your search results.

**Search context configuration**

The search context configuration [sys_search_context_config] record connects your search source with your search configuration. This record is also where you can set limits on search results and search suggestions.

**Search source**

Search source [sys_search_source] records determine what tables on your instance are included in your searches.

---

Create a search source

Create a search source to define what records can be found in your search bar.
Role required: admin

In these steps, you create the search source to define what search results your users can see. Each search source consists of a table, and optional conditions. User will see results from the selected table, limited by the conditions you define in the search source. You could, for example, select the Incident [incident] table, and then create a condition so that only incidents in the open state appear in search results. You can define more than one search source for a search, so your users can find records from multiple tables that match their search input.

1. In the filter navigator, enter sys_search_source.list.
2. Click New.
3. On the Search Source field, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of your search source. Enter a name that includes the name of the table you intend to use to make the search source easy to identify.</td>
</tr>
<tr>
<td>Application</td>
<td>The scoped application for the search source. This field is ready only and automatically contains the name of the current application.</td>
</tr>
<tr>
<td>Table</td>
<td>The table for this search source.</td>
</tr>
<tr>
<td>Conditions</td>
<td>Conditions under which a record from the selected table appears in search results. For example, if you select &lt;Active&gt;&lt;is&gt;&lt;true&gt;, only active records appear in your search results.</td>
</tr>
</tbody>
</table>
A search source configured for the User [sys_user]

1. Enter `sys_sg_global_search.list` in the filter navigator.
2. Click New.
3. In the global search configuration form, fill in the fields:
   - **Name**: Name for your search configuration.
   - **Application**: The scoped application for the search configuration. This field is ready only and automatically contains the name of the current application.
   - **Placeholder**: Text that appears in the search bar before a user enters text.

4. Click Save.
5. Optional: Repeat steps 1 through 4 to define additional sources for your search. Note that additional search sources can impact the performance of your searches.

*Create a global search configuration*

Create a global search configuration. This record connects your applet launcher with the other records used to configure search.

Role required: admin

1. In the filter navigator, enter `sys_sg_global_search.list`.
2. Click New.
3. In the global search configuration form, fill in the fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for your search configuration.</td>
</tr>
<tr>
<td>Application</td>
<td>The scoped application for the search configuration. This field is ready</td>
</tr>
<tr>
<td></td>
<td>only and automatically contains the name of the current application.</td>
</tr>
<tr>
<td>Placeholder</td>
<td>Text that appears in the search bar before a user enters text.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Search Context Configuration | Search context configuration record used for this search configuration.  
Note: Leave this field empty. You create a search context configuration record in the following steps.                                             |
| Interleave Result           | Whether search results are interleaved. Deselect this option to display search results separated by search source.                                                                                       |
| Enable Photo Search         | Whether photo search is available for your search bar. If the glide.sg.image_recognition.search.enable system property is set to false, you cannot select this option. For details on this property see Create the image search system property. |

4. In the **Search Context Configuration** field, click the reference icon.

5. In the search context configuration list, click **New**.

6. In the Search Context Configuration form, fill in the fields:

   **Search context configuration form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for your search context configuration.</td>
</tr>
<tr>
<td>Search Results Limit</td>
<td>Limit for the initial results returned in the search. Scrolling down with load additional results in increments defined by this value. This field has a default value of 10.</td>
</tr>
<tr>
<td>Application ID</td>
<td>The scoped application for the record. Select the same scoped application as the applet launcher where you want to use with this search configuration.</td>
</tr>
<tr>
<td>Suggestions limit</td>
<td>Limit for search suggestions listed for the search. This field has a default value of 10.</td>
</tr>
</tbody>
</table>

7. Right-click the header of the Search Context Configuration form, and click **Save**.
   The Search Context Configuration record is saved. After the save, the Application Search Sources related list appears on the form.

8. In the Application Search Sources related list, click **New**.

9. In the **Source** field of the Application Search Sources form, select a search source you created in the previous steps.

10. In the **Order** field add a value to determine the order in which the search source facets appear in the search results.

    **Note:** Search sources display from the lowest order to the highest.
A search context configuration with an associated search source.

11. Optional: Repeat steps 8 through 10 to add additional search sources if you created them in previous steps.
12. Click Submit.
13. Right-click the header of the global search configuration form, and click Save.
15. In the Global Search M2M Master Item form, click the reference icon

( )

next to the Master Item field.
16. Select the master item you want to use for a table that contains your desired search results. For example, if you want your search to return problem records, select master item that has problem in its Table field.
17. On the Global Search M2M Master Item form, click **Submit**.
18. Optional: Repeat steps 9-12 to add master items for additional record types if you plan to include multiple tables in your search.
19. Click **Save** when you have finished adding master items.

*Add your search configuration to an applet launcher*

Update your applet launcher to use your custom search configuration.

Role required: admin

1. In Studio, navigate to **Mobile Studio > Applet Launchers**, and open the applet launcher where you want to add your search.
2. Click the **Include Search** option to enable it.
3. In the **Search Criteria** field, select the global search configuration record you created in the previous steps.
4. Click **Save**.

*Enable voice search*

Enable your users to search for items, articles, and people using native speech recognition from an app on their mobile device.

Role required: admin

*Caution:* Voice search uses native speech recognition and relies on your operating system's cloud server to transcribe voice into text search. If you have data privacy concerns about search queries moving to the operating system cloud server, do not turn on voice search.

1. In the application navigator, enter **sys_properties.list**.
2. Click New to add a new system property.
3. On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter glide.sg.voice_search.enabled.</td>
</tr>
<tr>
<td>Type</td>
<td>Select true</td>
</tr>
<tr>
<td>Value</td>
<td>Enter True.</td>
</tr>
</tbody>
</table>

4. Click Submit.

The global search bar in the Now Mobile app includes a microphone icon. Users must select this icon and allow the app to access speech recognition on their mobile device to use voice search.

**Mobile photo search**

Configure photo search to give your users the ability to perform image-based searches using the objects around them.

**Photo searches in your mobile apps**
When photo search is configured on your instance, a photo icon appears in your applet launcher search bar. Your users can tap this icon to use the camera on their mobile device to take a picture. The picture is identified using the Google Vision API, which returns one or more results. Users can select a result, which is used as their search query.

**Google Firebase account**

To take advantage of image recognition in your mobile applications, you need a Firebase account with the Google Vision API enabled. Google Firebase and the Google Vision API are third-party services that must be subscribed to separately. Once you have a Firebase account, you can connect that account to your ServiceNow instance by entering your Firebase App ID and Web API key. For details on this process see Configure photo search.

**Third-party data usage**

Images you take using the photo search feature are sent to Google for identification using Google Vision API. ServiceNow does not have control of the image once it has been sent. For details on how Google Vision API handles your image, see [https://cloud.google.com/vision/docs/data-usage](https://cloud.google.com/vision/docs/data-usage).

**Configure photo search**

Configure photo search for your mobile applications by enabling photo search in system properties, creating a Firebase account, and connecting your account to your ServiceNow instance.

Create a Firebase account

Create a Google Firebase project and enable the Google Vision API.

Role required: admin

Photo search requires a Google Cloud / Firebase account to analyze images and return search results. If you already have a Google account, you can use your existing account. If not you can create an account at [https://firebase.google.com/](https://firebase.google.com/).

**Note:** Configuration for your Firebase account includes upgrading your account to the Blaze plan. This plan is a billable service that you must configure with Google.

1. Navigate to Firebase website, and sign in with a new or existing Google account.
2. Click Create Project.
3. Give your project a name and continue through the guided steps. The Firebase project page appears once you have completed the project setup.
4. From the project setup page, add an iOS or Android app to your project by selecting the iOS or Android button.
5. In the App setup page, fill in the form to add an app to your Firebase project.
   a) Enter a bundle ID and App nickname for your app. There are no specific requirements for the bundle ID and app nickname fields.
   b) You do not need to download the configuration file, add an Firebase SDK, add an initialization code, or run your app to verify installation. You can click Next or Skip this step to bypass these steps.

After completing setup you can see the configuration page for your project, with your new application listed.

6. Click the Spark Plan button.
7. Select the Blaze plan.

   Note: The Blaze plan is a billable plan. For details on pricing see Firebase pricing.

8. When prompted, create a billing account and enter in your payment details.
10. Log in to the website using the same Google account used in the previous steps.
11. In the header, select the project you created in previous steps.
12. Use the APIs & Services menu option to locate and enable the Google Vision API.
13. Return to the Firebase website, and open your project settings page.
14. In another browser tab or window, log in to your ServiceNow instance.
15. Navigate to System Mobile > Mobile Publishing > Native Clients, and select the record for the mobile app where you want to configure photo search.
16. In the Vision iOS App ID field, enter the App ID for your iOS app listed on your Firebase project settings page.
17. In the Vision Android App ID field, enter the App ID for your Android app listed on your Firebase project settings page.
18. In the Vision API Key field, enter the Web API Key listed on your Firebase project settings page.
19. Click Update.

Connect your instance to your Firebase project
Enter your project ID and Google Vision API key into your instance to begin using photo search.

Role required: admin
1. Navigate to System Mobile > Mobile Publishing > Native Clients.
2. Select the record for the app for which you want to enable photo search.
3. In the Vision iOS App ID field, enter the iOS App ID from your Firebase project.
4. In the Vision Android App ID field, enter the Android App ID from your Firebase project.
5. In the Vision API Key field, enter the Google Vision API key from your Firebase project.
6. Click Update.
   Your app is configured to use the Google Vision API for photo search. Repeat the steps if for your other mobile apps if you want to include the photo search option in those apps. Before using the photo search, you must still enable the system property as detailed in the next steps.

Create the image search system property
Create and enable a system property to control access to image search on your instance.

Role required: admin
1. In the filter navigator, type sys_properties.list to display the list of system properties for your instance.
2. Click New.
3. Fill in the system property form as needed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>glide.sg.image_recognition.search.enable</td>
</tr>
<tr>
<td>Type</td>
<td>true</td>
</tr>
<tr>
<td>Value</td>
<td>true</td>
</tr>
</tbody>
</table>

4. Click Submit.
   Your instance is configured to use photo search. You can disable photo search on your instance at any time by deleting the property record or by setting the Value field to false.

**Note:** Enabling this system property grants access to image search for all users.

Enable global search in your applet launcher to begin using photo search on your mobile applications. For details on enabling search on your applet launchers see *Enable global search in your applet launcher.*

Enable photo search on your applet launcher
Enable photo search on your applet launcher record in Studio to present the photo search option for your users.
Role required: admin

1. In the application explorer in Studio, navigate to Mobile Studio > Applet Launchers.
2. Open the applet launcher where you want to enable photo search.
3. Enable the Include Search option if it is not already enabled.
4. Enable the Photo Search option.
5. Click Save.

Applet launcher UI sections

UI sections contain the applets, charts, and record information on your applet launcher pages.

UI sections

Use UI sections to display applets and record information on your applet launcher screen. UI sections serve as containers for different types of data displayed within your applet launchers.
Use **Chart** and **Single Score** UI sections to display a preview of a single score report, time series report, or **Performance Analytics** widget.

**Chart** and **Single Score** previews are displayed as cards. When a user taps a chart, they open the chart applet which displays more information on the report or widget.

For information on configuring chart sections see *Configure chart UI sections.*
Use a Content section to display a combination of video, image, and text cards in a carousel format, at the top of your screen. Users can interact with the promotional displays by either viewing videos, being redirected to web pages, or navigated to defined areas on their mobile device.

For more information on mobile campaign configuration, see Configure content UI sections.
Use **Icon** UI sections to display applets using the icon that is defined in each applet. Users can tap on an icon to list, maps, calendars, or any other applet type. You can select a **Horizontal** or **Vertical** layout for your icons.

**Horizontal layout**

Use this section type to display many applets in one horizontal view. Limit the applet name character count to 20 or fewer characters so it does not get cut off. Use this section type to group related applets that do not require counts.

**Vertical layout**

When displayed vertically, you can use the **Display Count** option to display a count of records within each applet. Use this type for actionable items such as approvals or requests.

For detail on creating icon UI sections, see *Configure icon UI sections*.
Use **Item** UI sections to display records which require the user's attention, such as high priority or overdue items, and unassigned tasks. Users tap on an item to see additional record information or take actions. Choose from a **Horizontal** or **Vertical** layout when creating your item section.

Records from the selected list applet are displayed as cards. When a user taps a record, they open the record by using the form screen that is defined in the selected list applet.

**Horizontal layout**

Use horizontal item sections to display a discoverable number of items you want the user to see.

**Vertical layout**

Use vertical sections when you want to display a limited number of records within the applet launcher. Vertical item sections are good for records like critical tasks where you want the user to see details as soon as they land on the page.

For detail on creating item UI sections, see *Configure item UI sections*.

Use **Media** UI sections to display images or videos on your applet launchers. Users can tap a media section to navigate to another screen.

Media sections can display a single line header, and up to two lines of additional to text. You can define a navigation function and the text to appear on the navigation button.

Media sections are also a way to add a static company logo to the top of your applet launcher.

For information on configuring media sections see *Configure media UI sections*. 
Use **Navigation** UI sections to display applets and applet launchers using the icon defined for each applet and applet launcher. Users can tap on an icon to list, maps, reports, calendars, or any other applet or applet launcher type. You can select a **Horizontal** or **Vertical** layout for your icons.

**Horizontal layout**

Use this section type to display many applets or applet launchers in one horizontal view. Limit the name character count to 20 or fewer characters so it does not get cut off. Use this section type to group related applets or applet launchers that do not require counts.

**Vertical layout**

When displayed vertically, you can use the **Display feed count** option to display a count of records within each applet. Use this type for actionable items such as approvals or requests. For details on creating navigation UI sections, see

*Configure navigation UI sections.*

---

**Configure icon UI sections**

Use the icon section type to enable users to easily access your applets. Select how you want to visually represent this option.

Role required: admin

1. Navigate to **System Applications > Studio**.
2. Select your application.
3. In Application Explorer, navigate to **Mobile Studio > Applet Launchers** and select **Applet Launchers**.
4. Open your applet launcher.
5. In the **UI Section** of your applet launcher, click the add button (➕) to create a new item or icon UI section.
6. On the form, fill in the fields.
UI section form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Section Name</td>
<td>Name for the UI section.</td>
</tr>
<tr>
<td>Section Type</td>
<td>Type of UI section. For an icon section, select <strong>Item</strong>.</td>
</tr>
<tr>
<td>Orientation</td>
<td>Orientation of the UI section. You can select either <strong>Horizontal</strong> or <strong>Vertical</strong>.</td>
</tr>
</tbody>
</table>

**Note:**
- Use a vertical view when you want all scores to be visible on the screen without scrolling, or when your titles or values are long.
- Use a horizontal view when you want to display several scores in a scrollable section.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hide when empty</td>
<td>Option for hiding empty UI sections when there is no content to display.</td>
</tr>
<tr>
<td>Select the applets to display</td>
<td>Applets that you select from the Selected Applets list to appear in the UI section. This option is not available for the <strong>Item</strong> section type.</td>
</tr>
</tbody>
</table>

7. Click **Save**.

**Configure item UI sections**

Use the icon UI section types to enable users to easily access your applets. Select how you want to visually represent this option.

Role required: admin

1. Navigate to **System Applications > Studio**.
2. Select your application.
3. In Application Explorer, navigate to **Mobile Studio > Applet Launchers** and select **Applet Launchers**.
4. Open your applet launcher.
5. In the **UI Section** of your applet launcher, click the add button (+) to create a new item or icon UI section.
6. On the form, fill in the fields.

UI section form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Section Name</td>
<td>Name for the UI section.</td>
</tr>
<tr>
<td>Section Type</td>
<td>Type of UI section. For an icon section, select <strong>Item</strong>.</td>
</tr>
<tr>
<td>List applet</td>
<td>Select a list applet to display in your section. The records that are displayed in the section are the same records that appear when you use this applet. This field appears only when <strong>Section type</strong> is set to <strong>Item</strong>.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Section label</td>
<td>Label displayed for the section. This field is automatically populated from the selection made in the List applet field. This field appears only when Section type is set to Item.</td>
</tr>
<tr>
<td>Orientation</td>
<td>Orientation of the UI section. You can select either Horizontal or Vertical.</td>
</tr>
<tr>
<td>Note:</td>
<td>- Use a vertical view when you want all scores to be visible on the screen without scrolling, or when your titles or values are long.</td>
</tr>
<tr>
<td></td>
<td>- Use a horizontal view when you want to display several scores in a scrollable section.</td>
</tr>
<tr>
<td>Hide when empty</td>
<td>Option for hiding empty UI sections when there is no content to display.</td>
</tr>
</tbody>
</table>

**Hide the title of an item UI section**

You can choose not to display the title of your item section type, for example, to reduce the amount of text on your screen or because the title is similar to header of the UI section.

Role required: admin

1. In the web-based UI, enter `sys_sg_item_section.list` in the filter navigator to open a list of UI sections.
2. Open the record for your item section.
3. Clear the Display title option to hide the title for your UI section.
4. Click Update.
The title of your UI section is not visible in your mobile app. Because the header is still visible, you can see the See All button in the upper right corner of the UI section. This example image shows two UI sections. The upper UI section has a hidden title. In this case, the header is displayed, so the spacing has not changed and the See All option is still visible.

Define the number of items to display in a UI section
Define the number of items that are visible in the item section type providing the user with sufficient information for the required task and not requiring too much scrolling.

Role required: admin

1. In the web-based UI, enter `sys_sg_item_section.list` in the filter navigator to open a list of UI sections.
2. Open the record for your item section.
3. In Max Items Display Count enter the maximum number of items to display in a UI section.
   The default value is 15.
4. Click Submit.

Configure chart UI sections
Use the chart UI section type to display dashboard previews that enable users to navigate to reports and Performance Analytics chart widgets.

Role required: admin

1. Navigate to System Applications > Studio.
2. Select your application.
3. In Application Explorer, navigate to **Mobile Studio > Applet Launchers**.
4. Open your applet launcher.
5. In the UI section of your applet launcher, click the add button
   ![add button](image)
to create a new chart UI section.
6. On the form fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Section Name</td>
<td>Name for the UI section.</td>
</tr>
<tr>
<td>Section Type</td>
<td>Type of UI section.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Single Score</strong> sections display single score reports or the latest score Performance Analytics widgets.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Chart</strong> sections display a preview of a reporting charts.</td>
</tr>
<tr>
<td>Orientation</td>
<td>Orientation of the UI section. You can select either <strong>Horizontal</strong> or <strong>Vertical</strong>. Orientation options are not available for the <strong>Chart</strong> section type.</td>
</tr>
<tr>
<td>Hide when empty</td>
<td>Option for hiding empty UI sections when there is no content to display.</td>
</tr>
<tr>
<td>Select the dashboard preview to display</td>
<td>Applets that you select from the All Dashboards Previews list to appear in the UI section. You can move applets to the Selected Dashboard Previews list.</td>
</tr>
</tbody>
</table>

7. Click **Save**.

A dashboard preview is displayed in your applet launcher. Tap on the preview to navigate to a preconfigured chart. See **Mobile platform dashboards** for examples of the charts you can use.

**Configure navigation UI sections**

Use the navigation UI section type to navigate to applets and applet launchers.

Role required: admin

1. **Add and define a navigation UI section to be displayed in the applet launcher.**
   a) In the web-based UI, enter `sys_sg_navigation_section.list` in the filter navigator.
   b) On the Navigation Sections form, click **New**.
   c) On the form, fill in the fields.
Navigation section form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Title of the navigation UI section.</td>
</tr>
<tr>
<td>Active</td>
<td>Option for activating the navigation UI section.</td>
</tr>
<tr>
<td>Hide header</td>
<td>Option for hiding the header of the UI section.</td>
</tr>
<tr>
<td>Orientation</td>
<td>Orientation of the displayed applet icons, either Horizontal or Vertical.</td>
</tr>
<tr>
<td>Display feed count</td>
<td>The number of available items within an applet. This option is available only when Orientation is set to Vertical.</td>
</tr>
<tr>
<td>Required roles</td>
<td>The roles required to view this applet launcher. If the selected role column is empty, the applet launcher has no role restrictions.</td>
</tr>
</tbody>
</table>

4) Click Submit.

2. Define a link between the navigation UI section and either an applet or an applet launcher destination.

<table>
<thead>
<tr>
<th>Destination</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Applet launcher</td>
<td>1. In the web-based UI, enter <code>sys_sg_navigation_section_destination_applet_launcher.list</code> in the filter navigator.</td>
</tr>
<tr>
<td></td>
<td>3. Set the following values.</td>
</tr>
<tr>
<td></td>
<td>• Name: Name of the object displayed in the navigation UI section.</td>
</tr>
<tr>
<td></td>
<td>• Applet launcher: Applet launcher destination of the navigation.</td>
</tr>
<tr>
<td></td>
<td>• Icon: Icon displayed in the navigation UI section.</td>
</tr>
<tr>
<td></td>
<td>4. Click Submit.</td>
</tr>
<tr>
<td>Applet</td>
<td>1. In the web-based UI, enter <code>sys_sg_navigation_section_destination_screen.list</code> in the filter navigator.</td>
</tr>
<tr>
<td></td>
<td>2. On the Navigation Section Destination Screens form, click New.</td>
</tr>
<tr>
<td></td>
<td>3. Set the following values.</td>
</tr>
<tr>
<td></td>
<td>• Name: Name of the object displayed in the navigation UI section.</td>
</tr>
<tr>
<td></td>
<td>• Applet: Applet destination of the navigation.</td>
</tr>
<tr>
<td></td>
<td>• Icon: Icon displayed in the navigation UI section.</td>
</tr>
<tr>
<td></td>
<td>4. Click Submit.</td>
</tr>
</tbody>
</table>

3. Add and map the destination within a selected navigation UI section.

a) In the web-based UI, enter `sys_sg_navigation_section.list` in the filter navigator.
b) On the Navigation Sections form, select a navigation UI section from the list.

c) In the **Navigation Section Destinations** section, click **New**.

d) On the form, fill in the fields.

![Navigation Section Destinations form](image)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Navigation Section</td>
<td>The navigation UI section.</td>
</tr>
<tr>
<td>Navigation Section Destination</td>
<td>The applets or applet launchers to be used as the mapped destination.</td>
</tr>
</tbody>
</table>

e) **Submit**.

4. Add your UI section to an applet launcher:

   - **Note:** Content, media, and navigation UI sections are added to applet launchers in the applet launcher record within the web-based UI. Icon, item, and chart UI sections are added to applet launchers within mobile Studio.

   a) Navigate to **System Mobile > Applet Launchers**.

   b) Open the applet launcher record for the launcher where you want to add your UI section.

   c) Click the **Body** tab.

   d) In the **Applet Launcher Sections** list, click the **Insert a new row** link.

   - **Note:** If the *Insert a new row* link does not appear, you might not be in the same application as the applet launcher record you are editing. Switch to the appropriate application and reload the form.

   e) In the **Order** field, enter a number representing the order in which the UI section is displayed on the applet launcher.

   - UI sections appear on the launcher from the lowest order to the highest.

   f) Press the Tab key to move to the **Section** field.

   g) In the **Section** field, select the UI section and click the green check.

   h) After you have added your UI section, click **Update** to save the applet launcher record.

**Configure content UI sections**

Use the content UI section type to display text, images, and videos as campaign cards at the top of your screen. Use these campaigns to deliver messages and important information to your users.

Role required: admin

1. In the web-based UI, enter `sys_sg_content_section.list` in the filter navigator to open a list of content sections.

2. Click **New** to create a content section.

3. On the content section form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>A title for the content UI section. This title is not displayed in the mobile UI.</td>
</tr>
</tbody>
</table>
4. Click Submit.
5. Add your UI section to an applet launcher:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>Select whether to display the campaign in your mobile instance.</td>
</tr>
</tbody>
</table>

Note: Content, media, and navigation UI sections are added to applet launchers in the applet launcher record within the web-based UI. Icon, item, and chart UI sections are added to applet launchers within mobile Studio.

a) Navigate to System Mobile > Applet Launchers.
b) Open the applet launcher record for the launcher where you want to add your UI section.
c) Click the Body tab.
d) In the Applet Launcher Sections list, click the Insert a new row link.

Note: If the Insert a new row link does not appear, you might not be in the same application as the applet launcher record you are editing. Switch to the appropriate application and reload the form.

e) In the Order field, enter a number representing the order in which the UI section is displayed on the applet launcher.
   UI sections appear on the launcher from the lowest order to the highest.
f) Press the Tab key to move to the Section field.
g) In the Section field, select the UI section and click the green check.
h) After you have added your UI section, click Update to save the applet launcher record.

The content UI section you created can now contain a mobile campaign. For full instructions, see Configure mobile campaign components.

Configure media UI sections

Configure a media UI section type to display images or videos on your applet launcher.

Role required: admin

1. In your instance, enter sys_sg_section.list in the filter navigator.
2. In the mobile sections list click New.
3. In the Select section Type pop-up, select Media Section.
4. Click OK.
   The instance displays a media section form.
5. In the media section form, fill in the fields as needed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Title for your media section</td>
</tr>
<tr>
<td>Active</td>
<td>Whether the record is active. Inactive records do not display in your app.</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Application</td>
<td>The scoped application for this media section. This field is automatically populated with the current scoped application.</td>
</tr>
<tr>
<td>Required Roles</td>
<td>The roles required to see the media section. If this field is left empty, there are no role requirements.</td>
</tr>
<tr>
<td>Headline</td>
<td>Headline for the media section. This headline appears below your image or video.</td>
</tr>
<tr>
<td>Text</td>
<td>Text for the media section. This text appears below the text of your headline.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of media section select one of the following:</td>
</tr>
<tr>
<td></td>
<td>• Image</td>
</tr>
<tr>
<td></td>
<td>• Video</td>
</tr>
<tr>
<td>Image</td>
<td>Image that appears in the media section. This field appears only if the Type field is set to Image.</td>
</tr>
<tr>
<td>Note:</td>
<td>Images used in your media section must be attached to your media section record. Drag your image onto the media section form to attach, then select the image using the reference lookup icon</td>
</tr>
<tr>
<td></td>
<td>Create images optimized for use in media sections by using a height of 160px and a width of 1366px.</td>
</tr>
<tr>
<td></td>
<td>If your users primarily use iPhones, use a height of 160px and a width of 375px to 600px.</td>
</tr>
<tr>
<td>Video</td>
<td>Video that appears in the media section. This field appears only if the Type field is set to Video. This video must be a URL for an externally hosted YouTube or Vimeo video.</td>
</tr>
<tr>
<td>Function instance</td>
<td>Select the function instance that is used when a user taps on the media section. The label of your function appears below the text in the Text field.</td>
</tr>
<tr>
<td>Note:</td>
<td>The selected function instance must meet the following requirements:</td>
</tr>
<tr>
<td></td>
<td>• The Parent field of the selected function instance must have be the launcher where you intend to add your media section.</td>
</tr>
<tr>
<td></td>
<td>• The Location field of the function instance must be Media Section.</td>
</tr>
</tbody>
</table>
6. Click Submit.
   You have a configured media section that you can add to an applet launcher.

7. Add your UI section to an applet launcher:

   ![Image](https://via.placeholder.com/150)

   **Note:** Content, media, and navigation UI sections are added to applet launchers in the applet launcher record within the web-based UI. Icon, item, and chart UI sections are added to applet launchers within mobile Studio.

   a) Navigate to System Mobile > Applet Launchers.
   b) Open the applet launcher record for the launcher where you want to add your UI section.
   c) Click the Body tab.
   d) In the Applet Launcher Sections list, click the Insert a new row link.

   **Note:** If the Insert a new row link does not appear, you might not be in the same application as the applet launcher record you are editing. Switch to the appropriate application and reload the form.

   e) In the Order field, enter a number representing the order in which the UI section is displayed on the applet launcher.
      
      UI sections appear on the launcher from the lowest order to the highest.
   f) Press the Tab key to move to the Section field.
   g) In the Section field, select the UI section and click the green check.
   h) After you have added your UI section, click Update to save the applet launcher record.

### Hide the header of a UI section

You can choose to hide the header of a UI section that contains the title and the See All button, for example, to reduce the amount of noise on your screen or because the header is similar to the title of a UI section.

**Role required:** admin

1. In the web-based UI, enter sys_sg_item_section.list in the filter navigator to open a list of UI sections.
2. Open the record for your UI section.
3. Clear the **Hide header** option to hide the header for your UI section.
4. Click **Submit**.

The header of your UI section, which contains the title and the **See All** button are no longer visible. The example image shows two UI sections. The UI section on the right has a hidden header.

**Quick actions**

Define quick actions for your applet launcher so that your users can access commonly used functions, such as transferring records or using a template in the mobile app.

Role required: admin
Define quick actions for an applet launcher so that your users can access commonly used functions.

Users access quick actions tapping the quick action icon. For Android users, this icon appears as a plus symbol (+) at the bottom right corner of the applet launcher. For iOS users this icon appears as an ellipsis symbol (…).

1. In Studio, navigate to
2. Navigate to Mobile Studio > Applet Launchers, and select the applet launcher where you want to add a quick action.
3. In the Quick Actions list, click the add button (+).
4. Fill in the fields in the Create a Quick Action window as needed.
Create a Quick Action form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>Enter a descriptive label for the quick action</td>
</tr>
<tr>
<td>Icon</td>
<td>Select an icon for the quick action. For more information on mobile icons, see Mobile icons.</td>
</tr>
<tr>
<td>Function</td>
<td>Select a function to use in the quick action, or click the add button to create a new function. For details on creating functions, see Mobile functions.</td>
</tr>
</tbody>
</table>

**Note:** After you have created your quick action, you can change these values by clicking on the quick action in the Quick Actions list.

5. Optional: Click and drag the selection icon to change the order of the quick actions.
6. Optional: Click the delete icon to remove a quick action from your application launcher.

Add your applet launcher to the navigation bar to make it accessible to your users. For detail on navigation bar configuration, see Configure the navigation bar.

**Data items**

Data items provide the data presented in an applet.

Data items are queries on a selected table that bring back data. Applets, which display data from your instance, get their information from data items. For example, if your application contains a list of incidents, the list would use a data item representing records from the incident table.

**Data item types**

There are three different type of data items.

**Standard data items**

A standard data item is used returns data from a single table. A standard data item does not include parameters, but can include a filter to return a specific set of results from the queried table. For example, a standard data item could query all high priority changes from the change request table.

For information on standard data items, see Configure a standard data item.

**Parametrized data items**
A parametrized data item is used to perform a query containing one or more parameters to return data. Parameter are used to pass in variables to your queries. For example you could use a parameter to include the value of the logged in user, to show a list of all incidents assigned to that user.

For detail on parametrized data items, see Configure a parametrized data item. For a detailed walk-through of a parametrized data item configuration, see Tutorial: Configure a data item with parameters.

Relationship data items

A relationship data item is automatically created when an embedded list is added to a form screen through Studio. Though these data items are automatically created, you may modify them after they have been created.

Special use data item configurations

The following data items do not fit into the above categories, and are used only in specific cases detailed below.

Append encoded query

To navigate from a single score dashboard preview or from a chart header function to a list of records, you need to configure a special data item using an instance relative URL. This is a standard data item that uses a different type of condition. Find more info about how to configure this in Chart screen.

Grouped list data item

A grouped data item is used exclusively for a grouped list applet. Any data item, standard and relationship, parametrized or not, can be grouped by a field. Find more info about how to configure these at Grouped list screen.

Configure a standard data item

Configure a standard data item to query data for your applets that do not require parameter input.

Role required: admin

1. In Studio, navigate to Mobile Studio > Data Items, and click the popout icon.
2. In the Data Items tab, click Create New.
3. Complete the following fields as needed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A title for the data item. You can have multiple data items with the same name. Make sure that this name is unique so that you can find it easily.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional additional details about the data item</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Table</td>
<td>The name of the table you want the data item to pull information from.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Custom tables are not available by default. You can change this behavior by modifying the <code>subscription.custom_table.enforce_entitlement</code> system property. For details on making this change, see <em>Allow or restrict access to custom tables in mobile data items</em>.</td>
</tr>
<tr>
<td>Condition type</td>
<td>Determines what type of condition your data item uses. Select from:</td>
</tr>
<tr>
<td></td>
<td><strong>Declarative</strong> Use a declarative condition to create conditions for the data item using the condition builder.</td>
</tr>
<tr>
<td></td>
<td><strong>Scripted</strong> Use a script to determine the conditions of your data item. A text window to enter a script appears below this field when you select this option.</td>
</tr>
<tr>
<td></td>
<td><strong>Append Encoded Query</strong> Use this option only when creating data items for your chart applets. For details on that data item type, see <em>Configure an encoded query data item</em>.</td>
</tr>
<tr>
<td>Add sort</td>
<td>Adds fields to the form that allow you to configure how to sort the list. In the Add Sort window, select the field you want the list to be sorted by. For example, Caller. Then select a to z or z to a to determine which order the list goes in.</td>
</tr>
<tr>
<td>Query Condition</td>
<td>Set of conditions for the data item to conform to. You can create conditions using the <em>condition builder</em>.</td>
</tr>
<tr>
<td>Sorted by</td>
<td>This field only appears if you click <strong>Add sort</strong>. It is automatically populated with the information you added in the Add Sort window.</td>
</tr>
<tr>
<td>Direction</td>
<td>This field only appears if you click <strong>Add sort</strong>. It is automatically populated with the information you added in the Add Sort window.</td>
</tr>
</tbody>
</table>

4. Optional: If you have selected **Scripted** in the **Condition type** field, you must create a script in the **Query Condition Script** field.

Your scripted condition must return a query string, which the instance uses to filter the data item. Use the `input` variable to access information from the current record.

```javascript
(function getQueryString(input) {
    var queryString = '';
    if(input.company) {
        queryString += "company=\"+input.company; 
    }
    if(input.location) {
        queryString += "^location=\"+input.location;
    }
    queryString += "^EQ^ORDERBYDESCsys_updated_on";
});
```
This example uses the input variable filters the data item for records matching the current records company and location. It then appends the text ^EQ^ORDERBYDESCsys_updated_on to the query, which sorts the data item records by the Updated on field.

5. Click Save.

Associate a data item with an applet.

**Configure a parametrized data item**

Configure a parametrized data item to filter and view just the relevant data according to the selected parameters.

Role required: admin

Use the included examples to create a data item that allows users to open an incident list filtered by priority. For more detailed tutorial on how to create a data item with parameters, see Tutorial: Configure a data item with parameters.

1. In Studio, in your mobile app, navigate to **Mobile Studio > Data Items**.
2. Click **Create New** or open an existing data item you want to add parameters to.
3. Complete the Name, table, and condition fields as needed.
   
   For more information on creating a data item, see Configure a standard data item.

4. Click **Save**.
5. In the Parameter Definition section, click the add icon.
6. In the Item Parameter window, in the Name field, type a name for the parameter. Parameter names correlate most often with fields on a form.
   
   For example, type **Priority** as the parameter name.
7. From the Type list, select the type of parameter. The type determines how the user interacts with the mobile UI. For example, a type of Decimal or Integer tells the mobile device to open a numbers-only keypad. Select from the following types:
   
   • String: Uses a full keyboard for input. Use the String type for list parameters, such as priority or state, or for reference fields, such as assigned to or caller.
   • Integer: Opens a numbers-only keypad
   • Decimal: Opens a numbers-only keypad
   • Boolean: Opens a true or false selection option
   • DateTime: Opens a calendar with an exact time selector
   • Date: Opens a calendar
8. Click **Save**.
9. In the data item form, in the Query condition section, add a query condition for you parameter. The condition field should match the parameter you are querying the database for. For example, if you are creating a data item to query the Priority field, create a condition for **Priority is <priority>**.
Open Incidents

Data Item

Properties

- Name: Open incidents
- Table: Incident

Description:

Query condition:

- Active: is true
- State: is one of New, In Progress
- Priority: is Priority>

Parameter Definition

Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Priority</td>
<td>string</td>
</tr>
</tbody>
</table>
10. Click Save.

After you create a data item, you can assign it to an applet. Data items with parameters require additional configuration in the applet. For more information on assigning a data item with parameters to an applet, see Assign a data item with parameters to an applet.

Assign a data item with parameters to an applet

When you associate a parametrized data item with a screen, additional fields appear in the screen configuration that you must complete for the parameter to apply.

You should have already configured a data item that has parameters.

Role required: admin

1. In Studio, in your mobile application, navigate to Mobile Studio > Applications > <Name of your application>.
2. Click Create a new applet or open an existing applet.
3. In the Data Item field, select the parameterized data item you created.
4. In the Parameter setting section that appears, under UI Parameters > User Input, click the row that contains the parameter label and type.
5. In the User Input Parameter Definition window, complete the following fields as needed. Some fields only appear when you select a specific input type.

Use the fields on this form to determine how the user interacts with the UI.

User Input Parameter Definition fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for the UI parameter. You can have multiple UI parameters with the same name so make sure you choose something you can easily discover later. The name you select appears in the mobile app. For this example, use Priority as the name.</td>
</tr>
<tr>
<td>Input type</td>
<td>The UI used to complete the variable. For example, if you have a parameter for the Assigned to field, select List so that users have a list of users in that field to search for. Choose from one of the following options.</td>
</tr>
<tr>
<td></td>
<td>- Text: Provides a simple text field. This option works best for fields that require free text, such as work notes or resolution details. Text is the default type.</td>
</tr>
<tr>
<td></td>
<td>- List: Opens a list for the user to select from. This option works well for reference fields that require specific information.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The list input returns a maximum of 1000 results. For references that require more than 1000 results, use the Searchlist input type.</td>
</tr>
<tr>
<td></td>
<td>- SearchList: Provides a search bar so that users can search in a list.</td>
</tr>
<tr>
<td></td>
<td>- QR/Barcode: Provides the option to search by QRC or barcode.</td>
</tr>
<tr>
<td>Table name</td>
<td>This field only appears if you select List as the input type. The table for the field you want to create a UI parameter for. For example, Incident.</td>
</tr>
<tr>
<td>Field name</td>
<td>This field only appears if you select List as the input type. The field name you want to create for the UI parameter. For example, Priority.</td>
</tr>
<tr>
<td>Input style</td>
<td>How the user interacts from the UI. Choose from inline or popup. For this example, choose Inline.</td>
</tr>
</tbody>
</table>
Default value type | The value that appears by default in the UI field. Select one of the following options.
--- | ---
None: There is no default text. This works well for a list input type.
Manual: An additional field appears for you to enter a default term. For example, Search for a field. The manual default works well for search or text input types.

Mandatory | Whether or not the user is required to enter information for that field. For this example, leave this check box cleared.

Placeholder | Text that appears below the field type. This option does not appear if you have a default value selected.

6. In the Screen UI Parameter Mapping section, make sure the following fields are completed with the correct values.
   - Data Item Parameter: The name of the parameterized data item you created. For example, Priority.
   - UI Parameter: The name of the UI parameter in the next section of the Parameter settings section. For example, Priority.

7. In the User Input Parameter Definition, click Save.
8. Click Save.
9. Optional: If you want the field on the mobile screen to be automatically populated with a value, configure the autofill parameters.
   a) In the Autofill Parameters section, click the add icon.

   ![Add icon]

   b) In the Name field, enter a name for the autofill parameter.

   c) In the Input type field, select from the following options.
      - GPS location: Automatically inputs the location of the mobile device.
      - Date: Automatically inputs the current date for the mobile device.
      - User: Automatically inputs the currently logged in user.

   d) Click Save.

10. Complete any additional applet fields as needed. For more information on creating an applet, see Create an applet.
11. Click Save.

**Configure a relationship data item**

Configure relationship data items created by Studio for embedded lists.

Role required: admin

When you create an embedded list, Studio automatically creates relationship data items. You can customize these existing data items to control what information appears in your embedded lists. For more information on embedded lists in mobile, see Configure an embedded list for a form applet.

1. In Studio, in your scoped application, navigate to Mobile Studio > Data Items.
2. Open the data item you want to modify.
3. Modify the fields as needed.
Data item fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A title for the data item. You can have multiple data items with the same name. Make sure that this name is unique so that you can find it easily.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional additional details about the data item</td>
</tr>
<tr>
<td>Table</td>
<td>Table you want the data item to pull information from. The records queried in the data item are records from this table. Note: Custom tables are not available by default. You can change this behavior by modifying the subscription.custom_table.enforce_entitlement system property. For details on making this change, see Allow or restrict access to custom tables in mobile data items.</td>
</tr>
<tr>
<td>Condition type</td>
<td>Determines what type of condition your data item uses. Select from: Declarative Use a declarative condition to create conditions for the data item using the condition builder. Scripted Use a script to determine the conditions of your data item. A text window to enter a script appears below this field when you select this option. Append Encoded Query Use this option only when creating data items for your chart applets. For details on that data item type, see Configure an encoded query data item.</td>
</tr>
<tr>
<td>Application</td>
<td>The scoped application for the data item. This field is automatically populated.</td>
</tr>
<tr>
<td>Query Condition</td>
<td>Set of conditions for the data item to conform to. You can create conditions using the condition builder.</td>
</tr>
<tr>
<td>Group by</td>
<td></td>
</tr>
<tr>
<td>Tables related to</td>
<td>Table from the form header. For example, for a list of Task SLA records embedded into a task form, the Tables related to value would be the Task table.</td>
</tr>
<tr>
<td>Relationship</td>
<td>The relationship between the table in the Table field and the table in the Table related to field.</td>
</tr>
</tbody>
</table>
Configure an encoded query data item

Configure an encoded query data items to query data for chart applets.

Role required: admin

An encoded query data item is a parametrized data item used specifically for chart applets.

1. In Studio, navigate to Mobile Studio > Data Items, and click the popout icon.

2. In the Data Items tab, click Create New.
Data item form in Studio

3. Complete the following fields as needed.

Data item fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A title for the data item. You can have multiple data items with the same name. Make sure that this name is unique so that you can find it easily.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional additional details about the data item</td>
</tr>
<tr>
<td>Table</td>
<td>The name of the table you want the data item to pull information from.</td>
</tr>
<tr>
<td>Condition type</td>
<td>Determines what type of condition your data item uses. For an encoded query data item, select Append Encoded Query.</td>
</tr>
<tr>
<td>Append Encoded Query</td>
<td>This field only appears after you select Append Encoded Query in the Condition type field. Leave this field blank. You use this field in later steps.</td>
</tr>
</tbody>
</table>

4. Click Save.
   After you click **Save**, the Parameter Definition section appears.
5. In the Parameter Definition section, click the add icon (➕).

6. In the **Name** field, enter a name for your parameter. For example, **URL**.

7. In the **Type** field, select **String**.
Parameter definition pop-up

8. Click Save.
9. In the **Append Encoded Query** field, click the reference value icon, and select the parameter you created in the previous steps.

```
Condition Type: Append Encoded Query
```

Append Encoded Query field

10. Click Save.
For more information on how parameters pass values between applets, see Parametrized list screens.

Configure a group by data item

Configure a data item for a grouped list applet.

Role required: admin

1. In Studio, navigate to Mobile Studio > Data Items, and click the pop-out icon that appears to the right of Data Items.
2. Click Create New.
3. In the New Data Item form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the data item.</td>
</tr>
<tr>
<td>Table</td>
<td>Table where the data item gets its data.</td>
</tr>
<tr>
<td>Description</td>
<td>Description used to identify your data item.</td>
</tr>
<tr>
<td>Condition Type</td>
<td>Whether the conditions for the data item are declarative, scripted, or use an encoded query. For a grouped list, select either Declarative or Scripted.</td>
</tr>
</tbody>
</table>

4. In the Group Configuration section, use the Group By field to select the field you want to use to group the records in your list.
5. Click Save.
To see a walk-through example of steps to create a grouped list, see Tutorial: Configure a list of grouped incidents

Allow or restrict access to custom tables in mobile data items

Use system properties to control whether custom tables are available when creating or modifying data items.

Role required: admin

1. Type sys_properties.list in the Application Navigator to open a list of your system properties.
2. Find the subscription.custom_table.enforce_entitlement system property.
3. To permit access to custom tables for data items, set the Value field to false, otherwise, set the value to true.
This system property is in the global scope. If you are not in the global scope, you see a prompt at the top of the page. Click the here link to edit the property.

4. Click Update.

Mobile dashboard preview

Dashboard previews display a preview of reports and Performance Analytics widgets in your applet launcher. Mobile dashboard previews display reports or Performance Analytics previews within an applet launcher in chart UI sections. Adding reports and Performance Analytics widgets helps users more easily identify trends and turning points through indicator scores and visual representation. For more details, see Configure chart UI sections.

Reporting charts

Dashboard previews support the follow reporting charts:

- Area
- Bar (vertical bar chart)
- Donut
- Line
- Single score
- Spline
- Step line

Performance Analytics widgets

Dashboard previews support the latest score Performance Analytics visualizations. For details about this chart, see Displaying Performance Analytics charts for mobile dashboard.

Create a mobile dashboard preview

Use a dashboard preview to display reports and user experience Performance Analytics widgets on your applet launcher pages.

Role required: admin

1. Navigate to System Applications > Studio.
2. Select your application.
3. In the Application Explorer, navigate to Mobile Studio > Dashboard Previews, and click the popout icon.
4. Click Create New.
   A new Studio tab opens with a dashboard preview form.
5. On the form, fill in the fields.
## Dashboard preview form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name for your dashboard preview.</td>
</tr>
<tr>
<td>Type</td>
<td>Determine whether your preview should display a <a href="#">Report</a> or <a href="#">Performance Analytics</a> widget.</td>
</tr>
<tr>
<td>Report</td>
<td>Report to be displayed in your preview. This field appears only when the <strong>Type</strong> field is set to <strong>Report</strong>.</td>
</tr>
<tr>
<td></td>
<td>Click the arrow to display a list of all reports hosted on the instance. If you select a report not supported for mobile, when you open the applet launcher a message is displayed stating that the content cannot be shown.</td>
</tr>
<tr>
<td></td>
<td>For a list of supported charts, see <a href="#">Mobile dashboard preview</a>. The displayed report contains the settings and configurations as defined in the instance.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Although pie charts configured on the instance are also supported in mobile, the data is displayed in a donut chart format.</td>
</tr>
<tr>
<td>Widget</td>
<td>Performance Analytics widget to be displayed in your chart preview. This field appears only when <strong>Performance Analytics</strong> is selected from the <strong>Type</strong> field.</td>
</tr>
<tr>
<td></td>
<td>Click the arrow to display a list of all widgets hosted on the instance. If you select a Performance Analytics widget not supported for mobile, when you open the applet launcher a message is displayed stating that the content cannot be shown.</td>
</tr>
<tr>
<td></td>
<td>For a list of supported charts, see <a href="#">Mobile dashboard preview</a>. The Performance Analytics widget contains the settings and configurations as defined in the instance.</td>
</tr>
<tr>
<td>Auto-create associated Navigation and Applet</td>
<td>Select this option to enable the automatic creation of chart and list applets. These screens display when users tap the chart preview to access additional information.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> To manually create a chart applet and a navigation function, do not select this check box. For more information, see <a href="#">Manually configure chart applets for dashboard preview</a>.</td>
</tr>
<tr>
<td>Navigation</td>
<td>Navigation function to use when a user taps the chart preview. This field is not displayed when the <strong>Auto-create associated Navigation and Applet</strong> field is selected.</td>
</tr>
</tbody>
</table>

6. **Click Save.**
Add your dashboard preview to a chart UI section in an applet launcher. For more information, see *Configure chart UI sections*.

**Manually configure chart applets for dashboard preview**

Manually define the chart screen to be displayed and the navigation function to use when your users tap on a dashboard preview. Use this configuration option if you are not using automatic creation of associated navigation and applets.

Complete the procedure detailed in *Create a mobile dashboard preview*. Do not select the field *Auto-create associated navigation and applet*, which enables automatic creation of associated navigation and applets.

Role required: admin

1. Navigate to *System Applications > Studio*.
2. Select your application.
3. In the Application Explorer, navigate to *Mobile Studio > Applets* and select *Applets*.
4. Click the popout icon
   ( ) that appears to the right of *Applets*.
5. In the applets list, click *Create an Applet*.
6. On the form, fill in the fields.

   **Create an applet form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the applet, which appears on a tile in the application.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the applet.</td>
</tr>
<tr>
<td>Icon</td>
<td>The icon image and color for your applet, which is displayed in the main Applets page.</td>
</tr>
<tr>
<td>Available offline</td>
<td>Option for making the applet available to users in offline mode.</td>
</tr>
</tbody>
</table>

7. From *Choose a template*, select the chart icon.
8. Click *Create new*.
9. On the *Date and Field* tab of the New Applet form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Type of chart displayed after tapping on the chart section in the applet launcher, either Report or Performance Analytics.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Report</td>
<td>Report to be displayed in your preview. This field appears only when <strong>Report</strong> is selected from the <strong>Type</strong> field. Click the arrow to display a list of all reports hosted on the instance. If you select a report not supported for mobile, when you open the applet launcher a message is displayed stating that the content cannot be shown. For a list of supported charts, see <em>Mobile dashboard preview</em>. The displayed report contains the settings and configurations defined in the instance. Note: Although pie charts configured on the instance are also supported in mobile, the data is displayed in a donut chart format.</td>
</tr>
<tr>
<td>Widget</td>
<td>Performance Analytics widget to be displayed in your chart preview. This field appears only when <strong>Performance Analytics</strong> is selected from the <strong>Type</strong> field. Click the arrow to display a list of all widgets hosted on the instance. If you select a Performance Analytics widget not supported for mobile, when you open the applet launcher a message is displayed stating that the content cannot be shown. For a list of supported charts, see <em>Mobile dashboard preview</em>. The Performance Analytics widget contains the settings and configurations defined in the instance.</td>
</tr>
<tr>
<td>Set the default view point to the most recent</td>
<td>Area of the report to highlight when the user opens the chart screen. This option changes depending on the report you select. For time series reports, the option <strong>Highlight most recent data</strong> means that the focus point is at the end of the chart. Clear the check box to set the focus point at the start of the chart. For bar reports, the option <strong>Highlight right-hand bar</strong> means that the focus is on the last bar in the chart. Clear the check box to set the focus on the first bar of the chart.</td>
</tr>
</tbody>
</table>

10. Click **Save**.
11. On the **Functions** tab of the New Applet form, add a navigation button by clicking the add button ( ). This navigation function is required to navigate to the specified chart screen.
12. In the Chart Screen Header Function screen, from the two lists, select **Navigation** as the function type and then select a function.
13. Click **Save**.
Add your dashboard preview to a chart UI section in an applet launcher. For more information, see *Configure chart UI sections*.

**Configure chart view for accessibility**

Give users the option to change chart views from color segments to black and white patterns. This option can be used for accessibility purposes.

Role required: admin

To grant users the option to view charts in black and white patterns or colored segments, for all chart types. Make the **Enable data visualization patterns for charts** button available so users can select the button to view charts as black and white patterns. Alternatively, users who do not select the button can view charts as colored segments.

To display the **Enable data visualization patterns for charts** button, the system property **glide.sg.chart_accessibility** must be set to true.

1. Type `sys_properties.list` in the Filter Navigator.
2. Click **New**, and then enter the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td><code>glide.sg.chart_accessibility</code></td>
</tr>
<tr>
<td>Type</td>
<td>true</td>
</tr>
<tr>
<td>Value</td>
<td>true</td>
</tr>
</tbody>
</table>

Users can select the display option best suited for their requirements.
Comparison of charts with and without accessibility option enabled
Applets

Learn how to use applets in ServiceNow mobile. Applets are made up of one or more screens. The screens have configurable components, conditional formatting, sorting order, and filters.
Applet types

You can choose one of the available applet types within Studio. Each type provides one or more screens, which your users see when they select an applet. The template may include other optional screens that your users can access for additional information. All screens display information that is based on records in a data item.

List

Use list applets to display a list of records queried from a data item. List applets can be configured together with a form screen. When both screens are configured together, your users can tap a record on the list screen to display the form screen for that record.

For details on List applets, see List screen.
Create an applet

Create an applet in the application to contain a specific type of screen. For example, you can create a list or a map.

Role required: admin

Applets are templates for a series of screens. Depending on the template that you select, different options are available.

**Warning:** You can also configure individual screens in the Now Platform web-based UI. Unlike Studio, the Now Platform doesn't prevent you from using configurations that might break the app.

1. Navigate to System Applications > Studio to open Studio.
2. Select a scoped application where you want to create your applet.
   - If you have not created an application, you can create one by using the Create Application button. For information on creating a scoped application for mobile, see Create a mobile application using Guided Application Creator.
3. In Application Explorer, navigate to Mobile Studio > Applets and select Applets.
4. Click the pop-out icon that appears to the right of Applets.
5. In the applet list, click Create an applet.
   - The Create an Applet window appears on the screen.
6. In the Name field, enter a name for the applet.
   - The applet name appears on a tile in the application on a mobile device.
7. Click the Icon field to configure an icon for your applet.
   a) In the Icons tab, select an image to use as your icon.
   b) In the Colors tab, select a color for your icon. The instance automatically selects a contrasting background color to ensure visibility for your icon.
<table>
<thead>
<tr>
<th>Color</th>
<th>Applets</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dark Gray</td>
<td>#33E3F</td>
</tr>
<tr>
<td>Acapulco</td>
<td>#81B5A1</td>
</tr>
<tr>
<td>Sea Green</td>
<td>#54AC98</td>
</tr>
<tr>
<td>Shamrock</td>
<td>#48C891</td>
</tr>
<tr>
<td>Carmine</td>
<td>#E65651</td>
</tr>
<tr>
<td>Earls Green</td>
<td>#C3B137</td>
</tr>
<tr>
<td>Tan Hide</td>
<td>#FF8C3E</td>
</tr>
<tr>
<td>Chetwode Blue</td>
<td>#7070B3</td>
</tr>
<tr>
<td>Seagull</td>
<td>#6ABECF</td>
</tr>
<tr>
<td>Concord</td>
<td>#7D7D7D</td>
</tr>
</tbody>
</table>

Available icon colors with contrasting background colors

8. From the list of icons, select an icon to appear in the applet.
9. Optional: In the **Description** field, enter a description for the applet.

**Note:** You can change the applet name, description, icon, offline mode availability, and form screen availability after you create the applet. Just click the **Properties** button in the applet tab header.

10. Select an applet template.
Depending on the applet that you select, different screen configuration options appear.

11. Optional: Select the **Hide filter** check box to remove the filter option from your applet.
This option appears only when the list, employee directory, map, or group list template is selected.
List applet with and without the filter option

12. Optional: Select the check box next to the optional screen to include it in your applet.
Some templates have the option to include more than one screen. For example, a list applet can optionally include a form applet.
If you decide you want to add or remove the optional screen later, you can do so by clicking **Properties** when viewing your application in Studio.

13. Click **Create New**.

What you configure for the applet varies depends on the template that you select. For more information on how to configure a specific applet type, navigate to the topic that is specific to that applet.

**Copy an applet**

Use Studio to create a copy of a list or employee directory applet.

Role required: admin

1. In Studio, navigate to **Mobile Studio > Applets** and select **Applets**.
2. Open an existing list or employee directory applet.
3. Click the **Make a Copy** button.

**Note:** The **Make a Copy** button is disabled if the applet has unsaved changes.

4. In the Make a Copy pop-up, enter a name for the new applet in the **Applet Name** field.
5. Click **Save**.

The instance creates a copy of your applet, and duplicate versions of all necessary components, aside from the data item. New components are named with the name entered in the **Applet Name** field, with an appended suffix for the specific component type.
List screen

Use a list to display a list of records. Records in lists display in a card format, showing a limited selection of the information in the record.
Use a list applet to see records in a list view. Within a list applet, each record has an item view. This item view is a template which contains a selected few fields that display as a preview for each record. You can configure the item view in studio when creating a list applet.

Use the optional features available with list screens to improve your users experience.

**Form screen**

Configure your list screen applet to include a form screen. This configuration automatically displays a form view of a record when your users tap it in the list screen. Learn more about form screens in *Form screen*.

**List filters**

Mobile list filters enable your users to find what they need in the records on your instance. For more detail on filters, see *Mobile list filters*.

**Dynamic screen name**

Configure your list screen applet to inherit the name of a previous screen selection or form field and display it as the header on a subsequent screen. For more information, see *Configure a dynamic screen name for a list screen*.

**Functions**

Functions determine which actions users can perform in your mobile apps. For more detail on functions, see *Mobile functions*. 
Configure lists with forms

Configure your list with a form so your users can tap list items to view details of the records on your list. For details on list configuration, see *List applet configuration*. 
Parametrized list applets

Use parameters to pass information into your list applets. For example, when a user navigates taps on a group in a group list applet, a list containing records in the group is displayed. The list of records in the group is a parametrized list, which accepts the value of the group your user selected as a parameter.

To learn about parametrized list configuration, see *Parametrized list screens*. 
List screen functions

Add functions to your list applets to give your users the ability to take actions directly on a form screen. List applets support both swipe and top menu functions. For more information on functions, see Mobile functions.

Swipe functions
Use swipe functions to act on a specific record on the list. You can configure one or more actions to appear when a user swipes on a record. You can also configure a different set of actions depending on whether a user swipes to the left or to the right.

Top menu functions
Use top menu actions for less frequently used actions, or actions that apply to the list rather than individual records, such as Approve all action.
List filters tbd

Dynamic screen name

Use dynamic screen name functionality to let users to easily identify the screen or field which they are currently viewing. You can configure a screen to inherit a name from a previous screen selection. These dynamic names can be used with grouped lists, where the selected group name becomes the header on a subsequent screen. Alternatively, a
field in a form can be used as a header in a different screen. For more information, see Configure a dynamic screen name for a list screen.

**Display name from a redirection parameter**

Use a dynamic screen name to inherit a name from a previous screen selection. In the example, the screen name is coming from a list which is grouped by category from an applet launcher.

**Display information from field previous record**

Use dynamic form screen names to display information from a specific record in a table. The form screen name can come from any field in a record. In the example, the screen name is coming from the number field.
List applet configuration

Configure a list applet so that your users can see a list of records from their mobile app. To access a record from the list, a user just has to tap the record name.

Role required: admin

1. Navigate to **System Applications** > **Studio** to open Studio.
2. Select a scoped application where you want to create your applet.
   
   If you have not created an application, you can create one by using the **Create Application** button. For information on creating a scoped application for mobile, see *Create a mobile application using Guided Application Creator*.
3. In Application Explorer, navigate to **Mobile Studio** > **Applets** and select **Applets**.
4. Click the pop-out icon

5. In the applet list, click **Create an applet**.
   
   The **Create an Applet** window appears on the screen.
Create an Applet window

6. In the Create an Applet window. Enter a name and description for your list applet.
7. Click the image next to the Icon field to select a color and image for your list applet icon. If you add your list applet to an applet launcher, this icon displays for the applet.
8. To prevent the list from displaying a list filter, enable Hide filter.
9. In the Choose a template field, select List.
10. Optional: To the right of the Choose a template, enable the checkbox next to Form to include a form screen in your applet. This form displays record information when your users tap a record on the list.
11. Click Create New.

The Create an Applet window closes, and a new Studio tab opens for your list applet. If you selected the form option in the previous steps, you can also see a tab to configure your form.
12. In the **List Screen** tab, in the **Data item** field, select a data item from the list.

If the Data Item list is disabled, no data items have been created. To create a data item, click the plus icon (➕) next to the list to open a **New Data Item** tab. In the **New Data Item** tab, create your data item. For more information on creating a data item, see *Configure a standard data item*.
13. Optional: Select the header pattern that you want to use for the list view and the details screen view by clicking **Change Pattern**.

14. In the field configuration section, move fields to the List item fields list by double-clicking a field, or selecting a field and clicking the add or remove buttons.

   **Note:** If you change the pattern or the data item, any existing header configuration is deleted.

If the pattern that you choose includes an image field that is indicated by a circle, select an image type field. For example, select the **Caller > Avatar** field. Make sure that you select the image field, or any other dot-walked fields, before you select the parent field.

---

**Header options**

**Note:** Although ServiceNow does not support pattern modification, you can change the JSON that defines the pattern layout. If you modify a pattern record, Studio may not recognize the pattern. In this case, Studio displays a warning and a link to the item view [sys_sg_item_view] record, which contains the JSON for the selected pattern. A pattern that is not recognizable may still function as expected in the mobile app but is not editable in Studio.
15. Click Save.
List can be configured to include a form screen when they are created. If your list includes a form screen, click the Form Screen tab to configure that screen. For details on how to configure form screens, see Configure a form applet for use with a list applet.

You can also configure your lists to navigate to screens other than forms. For an example of this, see Tutorial: Configure navigation from a list applet to another list applet.

List segment configuration

Create list segments so that your mobile app users can see multiple lists within the same list applet in a tabbed format.

Role required: admin

1. Navigate to System Mobile > Applets.
2. Open the applet record where you want to create a new segment.
   When you create a list applet in Studio, the instance automatically creates a form applet with the same name. Add the Class field to the list so that you can see which one is the list applet.
3. In the Item Stream Segments list, click Insert a new row.
4. Enter a name for your new segment.
5. Select Ascending or Descending as the sort order in the Interleave Order Direction field.
6. In the Order field, enter a numerical value.
   Segments display from the lowest to the highest value.
7. Clear the Hide filters option to hide the filter options for your lists.
   This option hides list filters for all segments in your list applet.
8. Right-click the form header and select Save.
9. To open the item stream segment record, click the name of your new segment.
10. In the Item Stream M2M Segments list, click Insert a new row.
11. Select the item stream list that you want to add to your segment.
12. Click Update.

Configure a form applet for use with a list applet

Use a form applet as part of a list so that your users can see the details of a record from their mobile app.

Role required: admin

Your instance creates form applets automatically for calendar, list, and map applets. You can also create your own form applets manually.

1. Navigate to System Applications > Studio to open Studio.
2. Select a scoped application that contains your list applet.
3. In Application Explorer, navigate to Mobile Studio > Applets and select Applets.
4. Select the applet where you want to configure a form.

Note: List applets have the option to include a form screen when you create them. Select a list applet where you used this option.
5. In your applet configuration form, select the **From Screen** tab.
6. Optional: To select the header pattern that you want to use for the list view and the details screen view, click **Change Pattern**.

**Note:** If you change the pattern or the selected table, any existing header configuration is deleted.

If the pattern that you choose includes an image field that is indicated by a circle, select an image type field. For example, select the **Caller > Avatar** field. Make sure that you select the image field, or any other dot-walked fields, before you select the parent field.

![Diagram of header options]

**Warning:** Although ServiceNow does not support pattern modification, you can change the JSON that defines the pattern layout. If you modify a pattern record, Studio may not recognize the pattern. In this case, Studio displays a warning and a link to the item view [sys_sg_item_view] record, which contains the JSON for the selected pattern. A pattern that is not recognizable may still function as expected in the mobile app, but it is not editable in Studio.

7. Click **Body**.
8. Move fields from the All fields list to the Selected fields list by double-clicking a field, or selecting a field and clicking the add or remove buttons.
9. Optional: Select the **Attachment list** option below the Selected fields list to show a list of attachments at the bottom of the form. For more detail on attachments in mobile, see *Manage attachments on details screens*.

10. Click **Save**.

**Parametrized list screens**

Learn how to use parameters to pass information into a list applet.

Use parameters to pass information into a list applet. For example, a user views a grouped list showing records by grouped by priority. When a user selects a priority, a list containing records matching that priority displays. This second list needs to know which priority the user selected to display the appropriate records. The system accomplishes this task by passing a parameter. The first list stores the selected priority value in a parameter. The second list uses the information in that parameter to display the appropriate records.
Incidents grouped by Priority

<table>
<thead>
<tr>
<th>Priority</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>27</td>
</tr>
<tr>
<td>High</td>
<td>4</td>
</tr>
<tr>
<td>Moderate</td>
<td>12</td>
</tr>
<tr>
<td>Low</td>
<td>5</td>
</tr>
<tr>
<td>Planning</td>
<td>17</td>
</tr>
</tbody>
</table>

Incident list with a high priority selected

- **Trouble getting to Oregon mail server**
  - Caller: Bud Richman
  - Urgency: 3 - Low
  - Incident ID: INC0000039

- **Can't read email**
  - Caller: Fred Luddy
  - Urgency: 1 - High
  - Incident ID: INC0000001

- **Network file shares access issue**
  - Caller: Fred Luddy
  - Urgency: 1 - High
  - Incident ID: INC0000002

- **Wireless access is down in my area**
  - Caller: Joe Employee
  - Urgency: 1 - High
  - Incident ID: INC0000003

- **Network**
  - Incident ID: INC0000049
Configure a parametrized list applet

Configure a list screen to query a user for a parameter, which the applet uses to filter the records that appear on the screen.

Role required: admin

Creating a parametrized list is similar to creating a standard list. In addition to the steps used to create a standard list, you must create:

- A parametrized data item for your list. This data item uses your parameter to filter what data is displayed in your list.
- A parameter on your list applet. This parameter stores the value passed into your list.
- A screen UI parameter mapping to associate the parameter in your list with the parameter in your data item.

1. Navigate to System Applications > Studio to open Studio.
2. Select a scoped application where you want to create your applet.
   If you have not created an application, you can create one by using the Create Application button. For information on creating a scoped application for mobile, see Create a mobile application using Guided Application Creator.
3. In Application Explorer, navigate to Mobile Studio > Applets and select Applets.
4. Click the pop-out icon that appears to the right of Applets.
5. In the applet list, click Create an applet.
   The Create an Applet window appears on the screen.
Create an Applet window

6. In the **Create an Applet** window. Enter a name and description for your list applet.
7. Click the image next to the **Icon** field to select a color and image for your list applet icon. If you add your list applet to an applet launcher, this icon displays for the applet.
8. To prevent the list from displaying a list filter, enable **Hide filter**.
9. In the **Choose a template** field, select **List**.
10. Optional: To the right of the **Choose a template**, enable the check box next to **Form** to include a form screen in your applet. This form displays record information when your users tap a record on the list.
11. Click **Create New**.

The **Create an Applet** window closes, and a new Studio tab opens for your list applet. If you selected the form option in the previous steps, you can also see a tab to configure your form.
12. Click the plus icon to the right of the **Data Item** field.
A new Studio tab opens to configure a new data item.

13. On the new data item form, fill in the fields.
New Data Item form

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the data item.</td>
</tr>
<tr>
<td>Table</td>
<td>Table where the data item gets its data.</td>
</tr>
<tr>
<td>Description</td>
<td>Description used to identify your data item.</td>
</tr>
<tr>
<td>Condition Type</td>
<td>Whether the conditions for the data item are declarative, scripted, or use an encoded query. Leave this field at its default setting.</td>
</tr>
</tbody>
</table>

14. Click **Save**.
15. In the **Parameter Definition** section, click the plus icon
   to create a new parameter.
16. In the **Parameter Definition** pop-up, enter a name for your parameter in the **Name** field.
17. Click **Save**.
18. In the **Query condition field**, create a query that uses your parameter to filter your records. To access your parameter in the query condition, click the reference value icon.
19. Click **Save**.
You have a parametrized data item to use in your parametrized list. This example shows a parametrized data item that uses a parameter named Priority to filter the data items for a priority matching that parameter's value.

**Note:** For more detail on available options when creating parametrized data items, see *Configure a parametrized data item.*
20. In Studio, navigate back to the tab containing your list applet.
21. In the **Data Item** field, select the parametrized data item you created in the previous steps.
22. Click **Save**.
23. In the **User Input Parameter Definition** section, click the plus icon (➕) to create a new parameter.
24. In the **User Input** form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for your UI parameter.</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Input type</td>
<td>How your users enter a value for this parameter. Select from the following options:</td>
</tr>
<tr>
<td></td>
<td>• Text</td>
</tr>
<tr>
<td></td>
<td>• Choice List</td>
</tr>
<tr>
<td></td>
<td>• Search List</td>
</tr>
<tr>
<td></td>
<td>• QR/Barcode</td>
</tr>
<tr>
<td>Table name</td>
<td>Table used for the choice list where users select a parameter value. This field is visible only when the Input type field is set to Choice List or Search list.</td>
</tr>
<tr>
<td>Field name</td>
<td>The field used for the choice list where users select a parameter value. This field is visible only when the Input type field is set to Choice List or Search list.</td>
</tr>
<tr>
<td>Default value</td>
<td>Default value for your parameter. This field is visible only when the Default value type field is set to Manual.</td>
</tr>
<tr>
<td>Input style</td>
<td>Input style for your parameter. Select from Inline or Popup.</td>
</tr>
<tr>
<td>Default value type</td>
<td>Whether the parameter has a default value. Select None to have no default value, or Manual to enter a manual value in the Default value field.</td>
</tr>
<tr>
<td>Mandatory</td>
<td>Whether user input for the parameter is mandatory.</td>
</tr>
<tr>
<td>Placeholder text</td>
<td>Text that appears in the parameter entry field before the users enters a value.</td>
</tr>
<tr>
<td>Multi-select</td>
<td>Whether the user can select multiple values from the choice list. This field is visible only when the Input type field is set to Choice List.</td>
</tr>
<tr>
<td>Search type</td>
<td>The type of search used when finding a parameter value. This field is visible only when the Input type field is set to Search List.</td>
</tr>
</tbody>
</table>

25. Click Save.  
   Your parameter is created, and the User Input form closes.  
26. In the Screen UI Parameter Mapping section, click the plus icon (+) to create a new parameter mapping.  
   The Screen UI Parameter Mapping form displays.  
27. In the Item Parameter field, select the parameter you created in your data item.  
28. In the UI Parameter field, select the parameter you created in your parametrized list.  
29. Click Save.  
   The parameter in your list is associated with the parameter in your data item. When a user accesses this list, the applet prompts the user for a value for the parameter. The data item uses that value to filter the records displayed in the list.  

For an example demonstrating how to use parameters to create a grouped list, see Tutorial: Configure a list of grouped incidents.
Mobile list filters

Use mobile list filters to enable your users to find what they need in the records on your instance.

Filter options in mobile lists

Choose from the following options when configuring your mobile lists:

Use the automatically created default filters
By default, the instance automatically creates a filter. This filter is based on the fields selected in your list pattern. Users can filter the list based on any of these fields.

Create a custom filtering experience
Manually create a filter experience for your users. Custom filters can include fields that are not in the list pattern.

Disable filtering
When filtering is disabled, users are unable to filter records.

Default filters

Default filters are automatically created based on the fields added to the pattern in a list applet. No further configuration is needed. Your users have access to filter options using these fields.

Custom filters

If the default filters do not fit your requirements, you can create a customer filter experience. Administrators can manually select the fields available to the list filter, as well as enable keyword filtering. For more details on creating custom filters, see Configure a custom list filter.

Disabling filters

To disable filters on a list, open the list applet record, click the Filter tab, and disable the Show or hide filters for this list screen option.
Configure a custom list filter
Manually create a filter experience for your users. Custom filters can include fields that are not in the list pattern.

Role required: admin

1. In Studio, navigate to Mobile Studio > Applets, and open the list applet where you want to configure a custom filter.
2. In the list applet form, click the Filters tab.
3. If disabled, enable the Show or hide filters for this list screen field.
4. Click Set filters manually.
   When Set filters manually is enabled, the default list of filter fields is cleared.
5. Optional: To enable keyword filtering on your list, enable the Enable Keyword filter field.
   Studio creates a keyword filter and adds it to your filter list.
6. Click the plus icon to add a field to your list filter.

7. In the Add Filter pop-up, fill in the fields as needed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Field</td>
<td>The field used in this filter item. The data item used by the list determines which fields are available.</td>
</tr>
<tr>
<td>Filter Label</td>
<td>The label used for this filter field. This field defaults to the name of the field selected in the Data Field field.</td>
</tr>
</tbody>
</table>
Operator Selection | The operator type used by the filter. Some fields support multiple operator types.  

Filter Type | Filter type used by this filter item. This value is automatically determined by the value in the Operator Selection field.

8. Click Save.
9. Add additional filters by repeating steps 6 through 9 as needed.
10. When you have completed adding fields to your filter, click Save in the list applet form.

Configure a dynamic screen name for a list screen

Configure a screen to dynamically inherit a name from a parametrized list applet. This setup enables users to view a passed parameter as the screen name instead of the applet record name.
Role required: admin

1. In the web-based UI, enter sys_sg_screen.list in the filter navigator.
2. On the Applets form, select a list screen to inherit the dynamic screen name.
3. Apply an existing UI parameter in the UI parameter related list.
   a) If the UI parameters, Screen UI element mappings, and Source and UI element related lists are not displayed, add them by clicking the Additional actions icon. Then, select Configure > Related Lists, and then select the required related lists.
   b) Click the UI parameters tab.
   c) Select an existing UI parameter.
   d) On the form, fill in the fields.

   **UI parameter element mapping form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter type</td>
<td>Screen</td>
</tr>
<tr>
<td>Input source</td>
<td>User input</td>
</tr>
<tr>
<td>Input type</td>
<td>Text</td>
</tr>
</tbody>
</table>

   e) Click Submit.

4. Define a UI element to serve as the location point of the dynamic screen name.

   **Note:** This step is a one-time configuration. Once you create the UI element, a new record for the screen title location is not required.

   a) In the web-based UI, enter sys_sg_ui_element.list in the filter navigator.
   b) Click New to create a new UI element.
c) On the form, fill in the fields.

**UI elements form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>Location where the dynamic screen name displays. Select <strong>Title</strong>.</td>
</tr>
<tr>
<td>Name</td>
<td>Name of the UI element.</td>
</tr>
</tbody>
</table>

d) Click **Submit**.

5. Create a screen UI element and map it to the screen type.
   a) Click the **Screen UI element mappings** tab.
   b) Click **New**.
   c) On the form, fill in the fields.

**Screen UI element mapping form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Screen</td>
<td>Auto-populated with the selected screen, which inherits the dynamic screen parameter.</td>
</tr>
<tr>
<td>UI Element</td>
<td>The UI element to be configured.</td>
</tr>
</tbody>
</table>

d) Click **Submit**.

6. Map the screen UI element with the UI parameter.
   a) Click the **Source and UI element** tab.
   b) Click **New**.
   c) On the form, fill in the fields.

**Source UI element mapping form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UI Element</td>
<td>The UI element to display the dynamic screen name.</td>
</tr>
<tr>
<td>Source Table</td>
<td>The UI Parameter table.</td>
</tr>
</tbody>
</table>

d) Click **Submit**.

The screen name dynamically inherits a value from the redirection parameter.
**Form screen**

Form screens display content for a specific single record.

Use form applets to display the content of a single record. You can configure functions on form applets to allow users to make edits and take actions. Form screen supports top menu functions, footer functions, and smart buttons.

When a form screen is created, a details screen is configured by default. Add any of the four additional segments types listed below as needed.

Within a form screen, you can control the embedded segments. There are four types of embedded screen segments available for use when configuring a form screen through Studio.

**Form configuration screen**

Use the form configuration screen in Studio to control the content and appearance of your form applets.
The form applet configuration screen in Studio
At the top of the form configuration page are the **Overall Settings** and **Body** tabs.

**Overall Settings**

Use the **Overall Settings** tab to define the following elements of your form applet.

- Use the **Data Item** field to define which data item your applet uses. You may select an existing data item or create a new one using the plus icon (+).
- Use the **Field Configurations section** to determine what fields from your record display in the applet.
- Click the **Change Pattern** button to view the available patterns and select how your selected fields appear on the form.
- Use the **UI Style Configurations** section to define the font and background color of fields in your applet.
- The top menu functions available on your form (shown under the **Functions** section.)

**Body**

Use the **Body** tab to define the elements in the form segments on your form applet such as:

- Use the **Modify Segment** button to define which screen segments appear on your form. See the section below for details on each screen segment.
- The **Field Functions** available on your form.
- Define which footer
Form screen segments

Details screen segment

Use a details screen to display specific screen fields from the record. On a form, you can see the same item view/pattern that was configured for the list item. Embed a details screen in your form when you want to display more info about the record.

For more information on detail screens, see Configure a details screen for a form applet.

Activity stream screen segment

Use an activity stream screen to display work notes, comments, and/or attachments for the record. Your users can configure whether all three of these are visible, or only a select few are visible.

To configure an activity stream screen segment, see Configure an activity stream screen for a form applet.
Configure a parametrized form applet

Configure a form screen to query a user for a parameter, which the applet uses to determine the record that appears on the screen.

Role required: admin

1. Navigate to System Applications > Studio to open Studio.
2. Select a scoped application where you want to create your applet.
   If you have not created an application, you can create one by using the Create Application button. For information on creating a scoped application for mobile, see Create a mobile application using Guided Application Creator.
3. In Application Explorer, navigate to Mobile Studio > Applets and select Applets.
4. Click the pop-out icon that appears to the right of Applets.
5. In the applet list, click Create an applet.
   The Create an Applet window appears on the screen.
6. In the **Create an Applet** window. Enter a name and description for your form applet.
7. Click the image next to the **Icon** field to select a color and image for your form applet icon. If you add your form applet to an applet launcher, this icon displays for the applet.
8. From the Choose a template section of the form, select the **Form** template.
   (Optional) You can also modify an existing form that is part of a calendar, list, or map applet. In Studio open a calendar, list, or map applet for which you
want to configure a form. Then, click the Form Screen tab to view your form configuration.

9. Click Create New.
   The Create an Applet window closes, and a new Studio tab opens for your form applet.

10. To see the field configuration for your form screen, click Data and Fields.

11. Click the plus icon

    to the right of the Data Item field.
   A new Studio tab opens to configure a new data item.

12. On the new data item form, fill in the fields.

    New Data Item form

    | Field            | Value                                |
    |------------------|--------------------------------------|
    | Name             | Name of the data item.               |
    | Table            | Table where the data item gets its data. |
    | Description      | Description used to identify your data item. |
    | Condition Type   | Whether the conditions for the data item are declarative, scripted, or use an encoded query. Leave this field at its default setting. |

13. Click Save.

14. In the Parameter Definition section, click the plus icon

    )

    to create a new parameter.

15. In the Parameter Definition pop-up, enter a name for your parameter in the Name field.

16. Click Save.

17. In the Query condition field, create a query that uses your parameter to filter your records. To access your parameter in the query condition, click the reference value icon

    ).

18. Click Save.
   You have a parametrized data item to use in your parametrized form. This example shows a parametrized data item that uses a parameter named Number to filter the data item a record with a specific number.

   Note: For more detail on available options when creating parametrized data items, see Configure a parametrized data item.
19. In Studio, navigate back to the tab containing your form applet.
20. In the Data Item field, select the parametrized data item you created in the previous steps.
21. Click Save.
22. In the User Input Parameter Definition section, click the plus icon (➕) to create a new parameter.
23. In the User Input form, fill in the fields.

**User Input form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for your UI parameter.</td>
</tr>
<tr>
<td>Input type</td>
<td>How your users enter a value for this parameter. Select from the following options:</td>
</tr>
<tr>
<td></td>
<td>• Text</td>
</tr>
<tr>
<td></td>
<td>• Choice List</td>
</tr>
<tr>
<td></td>
<td>• Search List</td>
</tr>
<tr>
<td></td>
<td>• QR/Barcode</td>
</tr>
</tbody>
</table>
### ServiceNow Mobile

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table name</td>
<td>Table used for the choice list where users select a parameter value. This field is visible only when the <strong>Input type</strong> field is set to <em>Choice List</em> or <em>Search list</em>.</td>
</tr>
<tr>
<td>Field name</td>
<td>The field used for the choice list where users select a parameter value. This field is visible only when the <strong>Input type</strong> field is set to <em>Choice List</em> or <em>Search list</em>.</td>
</tr>
<tr>
<td>Default value</td>
<td>Default value for your parameter. This field is visible only when the <strong>Default value type</strong> field is set to <em>Manual</em>.</td>
</tr>
<tr>
<td>Input style</td>
<td>Input style for your parameter. Select from <em>Inline</em> or <em>Popup</em>.</td>
</tr>
<tr>
<td>Default value type</td>
<td>Whether the parameter has a default value. Select <em>None</em> to have no default value, or <em>Manual</em> to enter a manual value in the <strong>Default value</strong> field.</td>
</tr>
<tr>
<td>Mandatory</td>
<td>Whether user input for the parameter is mandatory.</td>
</tr>
<tr>
<td>Placeholder text</td>
<td>Text that appears in the parameter entry field before the users enters a value.</td>
</tr>
<tr>
<td>Multi-select</td>
<td>Whether the user can select multiple values from the choice list. This field is visible only when the <strong>Input type</strong> field is set to <em>Choice List</em>.</td>
</tr>
<tr>
<td>Search type</td>
<td>The type of search used when finding a parameter value. This field is visible only when the <strong>Input type</strong> field is set to <em>Search List</em>.</td>
</tr>
</tbody>
</table>

24. Click *Save*.  
Your parameter is created, and the **User Input** form closes.  
25. In the **Screen UI Parameter Mapping** section, click the plus icon (+) to create a new parameter mapping.  
The **Screen UI Parameter Mapping** form displays.  
26. In the **Item Parameter** field, select the parameter you created in your data item.  
27. In the **UI Parameter** field, select the parameter you created in your parametrized form.  
28. Click *Save*.  
The parameter in your form is associated with the parameter in your data item. When a user accesses this form, the applet prompts the user for a value for the parameter. The data item uses that value to filter the record displayed in the form.  
29. Optional: To select the header pattern that you want to use for the list view and the details screen view, click *Change Pattern*.  

**Note:** If you change the pattern or the selected table, any existing header configuration is deleted.

If the pattern that you choose includes an image field, which is indicated by a circle, select an image type field. For example, select the **Caller > Avatar** field. Make sure that you select the image field, or any other dot-walked fields, before you select the parent field.

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
**Warning:** Although pattern modification is not supported, you can change the JSON that defines the pattern layout. If you modify a pattern record, Studio may not recognize the pattern. In this case, Studio displays a warning and a link to the item view [sys_sg_item_view] record, which contains the JSON for the selected pattern. A pattern that is not recognizable may still function as expected in the mobile app, but it is not editable in Studio.

30. **Click Body.**
31. **Move fields from the All fields list to the Selected fields list** by double-clicking a field. You can also select a field and click the add or remove buttons.
32. **Optional:** Select the Attachment list option below the Selected fields list to show a list of attachments at the bottom of the form. For more detail on attachments in mobile, see Manage attachments on details screens.
33. **Click Save.**
You have a completed parametrized form. When a user accesses the form screen, the form prompts the user for a value, and stores that value in the parameter. The value passes from the form parameter to the data item parameter, which defines which record appears on the form.
Configure a details screen for a form applet

Create a details screen to display record fields on your form.

Role required: admin

1. In Studio, navigate to Mobile Studio > Applets, and select the form applet you want to configure with an activity stream.

   Calendar, map, and list applets also include a form screen. You can add a related list to these screens by open of these applets and selecting the Form Screen tab.

2. Click Body.
Form Test
Applet | Form

Form Screen

Overall Settings  Body

Details

* Body Fields

All fields

Search

All Fields

Active

Activity due
Actual end
Actual start
Additional assignee list
Additional comments
Approval
Approval history
Approval set

Selected fields

Search

Number

Short description

Show attachment list
3. If you have more than one screen segment configured for your form app, click the Details button to configure fields for the details screen.

4. Move fields from the All fields list to the Selected fields list by double-clicking a field, or selecting a field and clicking the add or remove buttons.

5. Optional: Select the Attachment list option below the Selected fields list to show a list of attachments at the bottom of the form. For more detail on attachments in mobile see Manage attachments on details screens.

6. Click Save.

Manage attachments on details screens
Add, remove, rename, and view attachments in your mobile form applets.

Enable attachments on your mobile forms

You can enable attachments on a mobile form. In Studio, open a form applet, click the Body tab, and then click the Details button. Select the Show attachment list check box on the Form Screen tab of the Incidents page.
For details on the form configuration, see *Configure a form applet for use with a list applet*. 
View the attachment list on your form screens

When enabled by an administrator, the attachment list appears at the bottom of the Details tab of your mobile form screens. The attachment list shows a preview of each attachment that is associated with the current record. If there are no attachments, you see "No attachment" instead of the attachment list.

Tap an attachment to show a preview of the attachment.
San Diego Office Policy


© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Manage attachments

To delete an attachment, tap the Delete icon, and then click Yes in the confirmation pop-up window.
To rename an attachment, tap the **Rename** icon. Then, in the **Edit name** field, enter a name.

**Note:** The ability to perform these actions are controlled by the Access Control List rules on your instance.

*Manage functions on details screens*
Add functions to your details screens to enable your users to take actions or view additional information.
Add functions to your details screens

You can add field and footer functions to the details screen in your form applets in Studio. To add functions, open a form applet, click the **Body** tab, and then click the **Details** button.

### Add a field function

To add a field function, click the plus icon (⊕) in the **Field Functions** section. In the pop-up window, use the **Field** field to select a field from your form to associate with a function. Then, select a function in the **Function** field.
Add a footer function

To add a footer function, click the plus icon in the Footer Functions section. In the pop-up window, use the Label field to enter a name for your footer button. Then, select a function in the Function field. Use the Button Emphasis field to select an appearance for your footer function button.
For information on creating functions to add to your applets, see Mobile functions.

For more information on placing functions in your applets, see Associate a function with a location in the app.

Screen fields
Screen fields improve the usability of your form applets. With screen fields, your users can view and edit attachments, view and complete checklist items, or display a field without a label.

When configuring a details screen, you select fields to display on your form. When you save the details screen, the instance creates a Screen Field [sys_sg_screen_field] each selected field. You can modify these records to change the screen field type. Change the screen field type to change how your information is presented, and improve the appearance and usability of the applet for your users.

Screen field types

**Text**

Use a text screen field to display text on your form. This text can come from any field on the current record. The text field type has a **Value only** option, which displays the field without a label. In this example, the **Description** field is shown without a label.
Percentage

Use a percentage screen field to display a percentage value on your form. This value comes from a field on the current record. Percentages are determined by using a decimal value. For example, a value of 0.35 displays in a percentage field as 35%. This field type has a Value only option, which displays the field without a label.
Image

Use an image screen field to display an image on your form. The value for this field comes from an image field or field that contains the sys_ID of an image in the Attachment [sys_attachment] table. Users can tap an image to open a preview screen to display the full image.
Attachment

Use an attachment screen field to display all the attachments of a record. Buttons to rename or delete attachments display automatically for your users. Users can tap a button to modify attachments.
**Video**

Use a video screen field so that your users can watch a video in your form. The video screen field gets its value from a field on your current record and contains a link to a video. The video must be externally hosted. Users can see the video in a new window when they tap **Play Video**.
**File**

Use a file screen field to display a PDF file within your form. The PDF displays as a preview on your form. Users can tap the preview icon to open it in a preview screen where they can scroll and zoom on the PDF document. The file screen field gets its value from a field on the current record. The value must be the sys_ID of an attachment [sys_attachment] record on your instance that contains a PDF file.
HTML

Use an HTML screen field to display HTML content within your form. The HTML screen field can get its value from an HTML field in your current record or a string field that contains an HTML code.
Checklist

Use checklist screen fields to display all the checklist items that are associated with a record. Users can tap individual checklist items to mark them as complete or incomplete.
**Date**

Use a date screen field to display a date value on your form. This value comes from a date or date/time field on the current record. Dates are displayed in the format that are defined by your instance. This field type has a *Value only* option, which displays the value of the field without the field label. If the field is configured with an action function, users can change the values of these fields. For more information, see *Configure an action function*. 

---

**Opened by**

System Administrator

**Opened**

2019-01-30

**Caller**

Bow Ruggeri

**Project Completion**

35 %

**Created**

2019-02-20
Stage

Use stage fields to display a read-only approval or completion status of requested items and services. Stage fields are displayed under the following conditions:

- A stage field is added on a form for Requested items [sc_req_item] records.
- A stage field is added on a form for Catalog item [sc_cat_item] records, as long the stage names and statuses are configured using one of the following flow types:
  - Flow Designer
  - Workflow [wf_workflow]
  - Execution Plan [sc_cat_item_delivery_plan]
- A stage field is added on a form for any other tables if the stage names and statuses are configured using Workflow [wf_workflow].

Add screen fields to your form screen

Add screen fields to improve the usability of your form applets. When you add a screen field, you can change how information is shown in your form or you can provide access to additional elements, such as attachments, videos, or links.

Role required: admin

1. Navigate to System Mobile > Applets.
2. Open the form applet that you want to add an enhancement to.
   You can see form applets by filtering where Class is Form Screen.
3. In the Form Screen Segments list, click Details.
4. In the Screen fields tab, click New.
5. On the form, fill in the fields.
Form screen form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Type of screen field. For a list of types and a description of how they are used, see Screen fields.</td>
</tr>
<tr>
<td>Application</td>
<td>Application where this field is used. This field is automatically filled.</td>
</tr>
<tr>
<td>Field Name</td>
<td>Text that appears as a field label. This option is only available for Attachments list fields. All other field types use the same label as the selected Form Field.</td>
</tr>
<tr>
<td>Form Field</td>
<td>Table field that this screen field uses as a data source. Some field types require a specific type of value. These requirements are described in Screen fields.</td>
</tr>
<tr>
<td>Hidden</td>
<td>Option that you can select to prevent this field from showing on the form applet.</td>
</tr>
<tr>
<td>Order</td>
<td>Order in which this field appears. Fields appear on the form applet from the lowest to the highest value.</td>
</tr>
<tr>
<td>Value Only</td>
<td>Value of the field without the field label.</td>
</tr>
<tr>
<td>Screen</td>
<td>Screen where this field is used. This field is automatically filled.</td>
</tr>
</tbody>
</table>

6. Click Submit.

Add a video screen field to a form screen

Learn how to use a video screen field to view embedded videos in your mobile applets.

Role required: admin

1. Navigate to System Mobile > Applets.
2. Open the form applet that you want to add an enhancement to.
   You can see form applets by filtering where Class is Form Screen.
3. In the Form Screen Segments list, click Details.
4. In the Screen fields tab, click New.
5. In the Type field, select Video.
6. In the Form field field, select a field from your table. The value of the field you select is used by the instance as a URL to an embedded video. For example, you can use the text https://www.youtube.com/watch?v=7YNG0fhp7yGE&t=95s to embed a video from YouTube.
   Only externally hosted videos are supported. There are no supported methods for displaying videos on your instance.
7. Click Submit.

Configure an activity stream screen for a form applet

Configure an activity stream on your form to give your users access to comments, work notes, and attachments relating to the record they are viewing.

Role required: admin

1. In Studio, navigate to Mobile Studio > Applets, and select the form applet you want to configure with an activity stream.
Calendar, map, and list applets also include a form screen. You can add a related list to these screens by open of these applets and selecting the **Form Screen** tab.

2. Click **Body** to open the body configuration for your applet.
3. Click the **Modify Segments** button to open the **Modify Segments** window.
4. Click the **Activity Stream** slider to enable the activity stream, then click **Save**. Clicking save closes the **Modify Segments** window.
5. In the **Body** tab, click the **Activity Stream** button.
6. Click the sliders next to **Add Comments**, **Add Work Notes**, or **Add Attachments** to add those elements to your form.
7. In the upper right corner of your screen, click **Save**.

Your form applet includes an **Activity** tab. Your users can tap this tab to view comments, work notes, and attachments relating to the record.
My work  Breached incidents

Latest OS update is failing to install

State  In Progress

Caller Alene Rabeck

Details  Activity  Related

2020-04-13

Field changes

S  System  7:12 AM

Assigned to  Kerry
Impact  1 - High
Incident state  In Progress
Opened by
Priority  2 - High
Configure a related list screen for a form applet

Create a related list screen to give your users easy access to records related to the record they are already viewing.

A related list appears in a tab on your mobile form, and displays a list records relating to the record on the form. For example, if you are viewing a problem record, you may want your related list to display all incident records associated with that problem record.

**Note:** Supported relationships are defined by reference field between two tables. Custom relationships created using the Relationships [sys_relationship] table or scripted relationships are not supported with this component.

Creating a related list involves three tasks:

1. Create a parametrized data item. This data item uses a parameter include only items related to your current record.
2. Create a list that displays the related records from your data item. This is the list that appears in the related list tab on your form.
3. Update your form applet to display related lists, and define the relationship between the current record and the related list.

Create a parametrized data item for your related list

Create a parametrized data item to contain the records that display in your related list.

Role required: admin

1. In Studio, Navigate to **Mobile Studio > Data Items**.
2. Click the pop-out icon that appears to the right of **Data Items**.
3. In the Data items screen, click **Create New**.
4. In the **Name** field, enter a name for your data item.
5. In the **Table** field, select the table for the records you want to appear in your related list. Remember that this table is for the record in the related list rather than the record on your form. For example, if you want to display a list of incidents related to a problem record, you should select **Incident [incident]** rather than **Problem [problem]**.
6. Click **Save**.
7. In the Parameter Definition section, click the add button.
8. In the **Parameter Definition** pop-up, enter a name for your parameter. Enter a name that matches your field name to make identifying your parameter easier.
9. In the **Type** field, select a data type for your parameter.
   - **String**: Use the String type for list parameters, such as priority or state, or for reference fields, such as **Assigned to** or **Caller**.
   - **Integer**: Use the integer type for a parameter containing an integer value.
   - **Decimal**: Use the decimal type for a parameter containing an decimal value.
   - **Boolean**: Use the boolean type for a parameter containing a boolean (true or false) value.
   - **DateTime**: Use the datetime type for a parameter containing a date and time.
   - **Date**: Use the date type for a parameter containing a date only.
10. Click **Save**.
The parameter is a variable that will contain a value from the record on your form. You will use this parameter to define what records are contained in your data item by using the parameter in your data item conditions.

11. In the condition section of your data item, create a condition using the parameter you created in steps 7 through 10. You can access your parameters in the condition builder using the reference value button.

12. Click Save.

In this example, you have a form that displays a problem record. On the form, you want to display a related list showing incidents associated with that problem. Since the records in the related list are incidents, the data item uses the Incident [incident] table.

For the data item to show only incidents related to the problem on the form, we need to create a parameter for that information. To make it easy to identify, the parameter name is Problem.

In the condition field, create a condition for the Problem field on incident records, and use the parameter as the value. You select the parameter using the reference value button.
Create a list screen to use as a related list

Create a related list using your parametrized data item. This list will appear for your users when they select the related list tab on their form screen.

Role required: admin

1. In Studio, navigate to **Mobile Studio > Applets**.
2. Click the pop-out icon ( ) that appears to the right of **Applets**.
3. Click **Create an Applet**.
4. In the **Create an Applet** pop-up, select a name and description for your related list.
5. Select the **List** template.
6. Click **Create New**.
7. In the **Data Item** field, select the parametrized data item you created in the previous steps.
8. In the Field Configurations section. Select the fields you want to display on your related list for each record.
9. Click **Save**.
10. In the UI Parameter section, click the add button to add a UI parameter for your related list.
11. In the **User Input** pop-up, enter a name for UI Parameter in the **Name** field.
12. Click **Save**.

Continuing the preceding example, you create a new applet to display incidents for your problem record. This applet uses the data item created in the previous steps. This applet needs a parameter to contain the problem record from your problem form. To make it easy to identify, the parameter name is also named **Problem**. In the UI Parameter Mapping section, you create a mapping between the data item parameter and the applet parameter, so the value can pass between them.
Update your form to display a related list
Update your mobile form to display a list of related records.

Role required: admin

1. In Studio, Navigate to Mobile Studio > Applets, and open the form applet where you want to display a related list.
   In some cases, the form applet where you want to display your related list is part of a list applet. In that case, open your list applet, and open the Form Screen tab to configure the form.
2. In the Body section of your form screen, click Modify Segments in the upper right corner.
3. In the Modify Segments pop-up, select Related List.
4. Click **Save**.
5. In the form screen, click **Related List**.
   The **Related List** appears next to the **Details** button after you enabled related lists in the previous steps.
6. Optional: Select **Show related list count** to display a count of records in your related list.
7. Click the add button
   to create a relationship. This relationship is what connects your form to the related list.
8. In the **Related List** pop-up, fill in the fields.
### Related list fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relationship</td>
<td>Select a relationship between the current form and the related list.</td>
</tr>
<tr>
<td></td>
<td>For example, if you are in a form for problem records, and want to</td>
</tr>
<tr>
<td></td>
<td>display a related list of incidents, you would choose <strong>incident-&gt;problem</strong>.</td>
</tr>
</tbody>
</table>

**Note:**

| Destination screen | Select the list applet you created in previous steps. This list    |
|                   | displays when the user selects the related list tab on their form screen. |

9. In the **Related List** pop-up, click **Save**.

10. On the form screen, click **Save**.

Your form can display a tab for your related list. Your users can tap this tab to see records on that list.
Continuing the preceding example, the problem form must have a related list. In the **Related List** pop-up, you select the **Incident->Problem** relationship. In the **Destination Screen**, you select the related incident list created in the previous steps. After logging out and back in again, you now have a related list on your problem form, which displays a list of incidents associated with that problem.
Configure an embedded list for a form applet

Use an embedded list to place a list of information that is related to your current record within a segment on your form applet so that your users don't have to navigate to a related list.
Use embedded lists to display lists of related information in an easily accessible form segment rather than having your users navigate away to a related list. For example, you could add a list of part records to your work order form.

To create an embedded list on a form, you will first need to perform the following tasks:

**Create a data item to contain the data for your embedded list**

You need a data item to store the data that appears in your embedded list. To create a data item, follow the process shown in [Configure a standard data item](#).

**Create a list applet using that data item**

You need to configure a list to embed into your form. You can create this list using the process detailed in [List applet configuration](#).

**Embed your list into your form**

Use the steps below to embed your list into a form screen.

---

**Configure an embedded list for a form screen**

Role required: admin

1. In Studio, open navigate to **Mobile Studio > Applets** and select the form applet where you want to add your embedded list.
2. In the applet configuration form, click **Body**.
3. Click **Modify Segments**.
4. In the modify segments popup, click to enable **Embedded Screen**.
5. Click Save.

Your form applet has an Embedded Screen tab.
6. Click the **Embedded Screen** button to open the embedded screen configuration options.

7. Click the plus icon

   ![Plus Icon](image)

   to add your embedded list.

8. In the **Embedded List** popup, select a relationship for your embedded list in the **Relationship** field.

9. In the **Destination Screen** field, select the list applet you created to use as an embedded list.

10. Click **Save**.

    After clicking **Save**, you will see a warning stating that the data item is being converted to a relationship data item, which can only be used for embedded lists.
11. Click **Ok** to close the warning popup.

Your form is configured with an embedded list. Your user can access this list on the form by tapping the tab with the name of this list.

To learn more about relationship data items used for embedded lists, see *Configure a relationship data item*.

**Configure a dynamic screen name for a form screen**

Configure a form screen to dynamically inherit a name from a field in a previous record. This setup enables users to view a single specified field as the screen name instead of the applet record name.

Role required: admin

1. In the web-based UI, enter `sys_sg_screen.list` in the filter navigator.
2. Select a form screen to inherit the dynamic screen name.
3. Create a UI parameter in the UI parameter related list.
   a) If the **UI parameters**, **Screen UI element mappings**, and **Source and UI element** related lists are not displayed, add them by clicking the Additional actions icon (⋮), selecting **Configure > Related Lists**, and then selecting the required related lists.
   b) Click the **UI parameters** tab.
   c) Either select an existing UI parameter or click **New** to configure a new UI parameter with specific values.
   d) On the form, fill in the fields.
UI parameter element mapping form

<table>
<thead>
<tr>
<th>Field</th>
<th>Value / Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter type</td>
<td>Screen</td>
</tr>
<tr>
<td>Input source</td>
<td>Auto fill</td>
</tr>
<tr>
<td>Input type</td>
<td>Source field</td>
</tr>
<tr>
<td>Button parent table</td>
<td>The same table as listed in the form screen.</td>
</tr>
<tr>
<td>Source field</td>
<td>The field to display for the screen name.</td>
</tr>
</tbody>
</table>

e) Click **Submit**.

### 4. Define a UI element to serve as the location point of the dynamic screen name.

**Note:** This step is a one-time configuration. Once you create the UI element a new record for the screen title location is not required.

a) In the web-based UI, enter `sys_sg_ui_element.list` in the filter navigator.

b) Click **New**.

c) On the form, fill in the fields.

**UI elements form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>Location where the dynamic screen name displays. Select <strong>Title</strong>.</td>
</tr>
<tr>
<td>Name</td>
<td>Name of the UI element.</td>
</tr>
</tbody>
</table>

d) Click **Submit**.

### 5. Create a screen UI element and map it to the screen type.

a) Click the **Screen UI element mappings** tab.

b) Click **New**.

c) On the form, fill in the fields.

**Screen UI element mapping form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Screen</td>
<td>Auto-populated with the selected screen, which inherits the dynamic screen parameter.</td>
</tr>
<tr>
<td>UI Element</td>
<td>The UI element to be configured.</td>
</tr>
</tbody>
</table>

d) Click **Submit**.

### 6. Map the screen UI element with the UI parameter.

a) Click the **Source and UI element** tab.

b) Click **New**.
c) On the form, fill in the fields.

Source UI element mapping form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UI Element</td>
<td>The UI element to display the dynamic screen name.</td>
</tr>
<tr>
<td>Source Table</td>
<td>The UI Parameter table.</td>
</tr>
</tbody>
</table>

d) Click Submit.

The screen name dynamically inherits the value from a defined field in an existing record. In the graphic, the screen name comes from the number field.

**Grouped list screen**

Grouped list applets display a list of records that are grouped by a specific field.
Use a grouped list to provide a view of a list of items grouped by a field. Groups lists can help present a large number of records in an easily navigable format. For example, you can display a list of incidents grouped by category or priority. You can configure grouped lists to navigate to a second list showing the records in the selected group.

**Configure a group list screen**

Create a group list to provide a view of a list of records grouped by a field you specify.

Before you can create a group list applet, you must create a data item with a group configuration. For more information on configuring a data item, see Configure a standard data item, and note the optional steps for setting the Group by field. After creating the data item for the grouped by applet, leave the data item tab open so that you can access it later.

Role required: admin

1. Navigate to **System Applications > Studio** to open Studio.
2. Select a scoped application where you want to create your applet.

If you have not created an application, you can create one by using the Create Application button. For information on creating a scoped application for mobile, see Create a mobile application using Guided Application Creator.
3. In Application Explorer, navigate to Mobile Studio > Applets and select Applets.
4. Click the pop-out icon
   ![Applets icon]
   that appears to the right of Applets.
5. In the applet list, click Create an applet.
   The Create an Applet window appears on the screen.

Create an Applet window

6. In the Create an Applet window. Enter an name and description for your group list applet.
7. Click the image next to the Icon field to select a color and image for your list applet icon. If you add your group list applet to an applet launcher, this icon displays for the applet.
8. In the Choose and template section, select Group List.
9. Click **Create New**.
   The **Create an Applet** window closes, and a new Studio tab opens for your group applet.

10. In the **Data Item** field, select a data item.

    **Note:** If the **Data item** field list is empty, you may have no data items have been created with a group configuration.

    To create a data item, click the plus icon (+) next to the list.

11. Click **Save**.

    You now have an applet displaying a list of records, grouped by the field you defined in your data item.
<table>
<thead>
<tr>
<th>Priority</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Critical</td>
<td>27</td>
</tr>
<tr>
<td>High</td>
<td>4</td>
</tr>
<tr>
<td>Moderate</td>
<td>12</td>
</tr>
<tr>
<td>Low</td>
<td>5</td>
</tr>
<tr>
<td>Planning</td>
<td>17</td>
</tr>
</tbody>
</table>

**Group List applet**

*Tutorial: Configure a list of grouped incidents*

Use this example to create a list of incident records, grouped by priority.

This example creates a grouped list as shown in the example image. Tapping on a priority displays a list of incidents matching that priority, which in turn can be tapped to display the incident record.
Create a data item for a grouped list
Create a data item that has a group configuration to use with your grouped list applet.

Role required: admin

Before you create an applet for your grouped list, you need a data item that has been created with a group configuration.

1. In Studio, navigate to Mobile Studio > Data Items, and click the pop-out icon that appears to the right of Data Items.
2. Click Create New.
3. In the New Data Item form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the data item. For this example, enter Incidents by Priority</td>
</tr>
<tr>
<td>Table</td>
<td>Table where the data item gets its data. Select Incident [incident].</td>
</tr>
<tr>
<td>Description</td>
<td>Description used to identify your data item.</td>
</tr>
<tr>
<td>Condition Type</td>
<td>Whether the conditions for the data item are declarative, scripted, or use an encoded query. Leave this field at its default setting.</td>
</tr>
</tbody>
</table>

4. In the Group Configuration section, set the Group By field to Priority
5. Click Save.
**Incidents by Priority**

Data Item

**Properties**

- **Name**: Incidents by Priority
- **Description**: Incidents grouped by Priority
- **Condition Type**: Declarative

**Parameter Definition**

**Parameter name** | **Parameter Type**
--- | ---
No data

**Group Configuration**

Once configured, this data item can also be used in Group by template.

**Group By**: Priority

**Completed data item for a grouped list**

You now have a data item for the incident table, grouped by priority. In the next steps, you use this data item to create the grouped list applet.

**Create a grouped list applet**

Create an applet using the grouped list template. This list displays your incident priorities, and the number of records matching each priority.

**Role required**: admin

1. In Studio, navigate to **Mobile Studio > Applets**, and click the pop-out icon that appears to the right of **Applets**.
2. Click **Create an Applet**.
3. In the **Create an Applet** form, fill in the fields.

**Create an Applet form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of your applet. For this example, enter <strong>Incidents by Priority</strong>.</td>
</tr>
<tr>
<td>Icon</td>
<td>Icon used by your applet. Select any icon.</td>
</tr>
<tr>
<td>Description</td>
<td>Description used to identify your applet.</td>
</tr>
<tr>
<td>Choose a template</td>
<td>Template used for your applet. Select <strong>Grouped List</strong>.</td>
</tr>
</tbody>
</table>

4. Click **Create new**.
   The create an applet pop-up closes, and you are directed to your new applet's form in a studio tab.
5. In the **Data Item** field, select the **Incidents by Priority** data item you created in the previous steps.
6. Click **Save**.
You now have an applet that displays your incident priorities, and a count of records matching each priority. In the next steps you create a navigation function to display a list of records matching the priority when a user taps on one of these priorities.

Create a list applet and data item

Create an applet using a list template. This list displays all the incident records in a matching priority when your user taps a priority in the previous list. You also create a new data item containing all the records matching a specific priority.
Role required: admin

1. In Studio, navigate to Mobile Studio > Applets, and click the pop-out icon that appears to the right of Applets.

2. Click Create an Applet.

3. In the Create an Applet form, fill in the fields.

   Create an Applet form

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of your applet. For this example, enter Incidents matching Priority.</td>
</tr>
<tr>
<td>Icon</td>
<td>Icon used by your applet. Select any icon.</td>
</tr>
<tr>
<td>Description</td>
<td>Description used to identify your applet.</td>
</tr>
<tr>
<td>Choose a template</td>
<td>Template used for your applet. Select List.</td>
</tr>
</tbody>
</table>

4. Click Create new
The create an applet pop-up closes, and you are directed to your new applet's form in a studio tab.

5. Click the plus icon to the right of the Data Item field.

6. On the new data item form, fill in the fields.

   New Data Item form

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the data item. For this example, enter Incidents matching Priority</td>
</tr>
<tr>
<td>Table</td>
<td>Table where the data item gets its data. Select Incident [incident].</td>
</tr>
<tr>
<td>Description</td>
<td>Description used to identify your data item.</td>
</tr>
<tr>
<td>Condition Type</td>
<td>Whether the conditions for the data item are declarative, scripted, or use an encoded query. Leave this field at its default setting.</td>
</tr>
</tbody>
</table>

7. Click Save.

8. In the Parameter Definition section, click the plus icon to create a new parameter.

9. In the Parameter Definition pop-up, enter Group priority in the Name field.

10. Click Save.

11. In the Query condition field select <Priority> <is> , and then click the reference value icon to select the Group priority parameter created in the previous steps.
Incidents matching priority data item

12. Return to the Studio tab for your Incidents matching priority applet.
13. In the Data Item field, select the Incidents matching priority data item you created in the previous steps.
14. In the Field Configurations section, select the fields you want to include in your incident list.
Example field configuration for the list applet

15. Click the **Form Screen** tab at the top of the form.

16. Click the **Replicate from primary** button

   ( )

   to automatically add fields from your list to the form.

17. Click **Save**.
You now have a list applet to display all incident records with the priority that your users select from the group list you created earlier. In the final set of steps, you will create a navigation function that handles the transition from the grouped list applet to the list applet.

Create a navigation function for your grouped list.
create a navigation function that handles the transition from the grouped list applet to the list applet containing records of a specific priority.

Role required: admin

1. In Studio, open the tab for your **Incidents by Priority** applet. This is the first applet you created to display your records grouped by priority.

2. Click **Functions** to open the functions section of the form.

3. Click the plus icon to create a new list item function.

4. In the list item function pop-up, click the plus icon to create a new function.

5. In the new navigation form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of your function. For this example, enter Grouped list navigation.</td>
</tr>
<tr>
<td>Destination Type</td>
<td>Whether the navigation will direct to an applet or applet launcher. Select Applet</td>
</tr>
<tr>
<td>Description</td>
<td>Description of your function.</td>
</tr>
<tr>
<td>Destination</td>
<td>The applet or applet launcher your function will direct to. Select the Incidents matching priority applet you created in previous steps.</td>
</tr>
<tr>
<td>Context</td>
<td>The context for the function. Select Global.</td>
</tr>
<tr>
<td>Table</td>
<td>The table used for display conditions. Select Incident [incident]</td>
</tr>
</tbody>
</table>
6. Click Save.

7. Return to the **Incidents by Priority** tab, and select your new navigation function.

8. Click Save.

**Employee directory screen**

Employee directory screen display user record information in ServiceNow mobile apps.
Use an employee directory screen to present users with information on user records on your instance. You can design your employee directory screen with functions to allow users to communicate via phone or email, and navigate to their address using their mobile device's navigation functionality.

Configure an employee directory applet

Use an employee directory applet to provide a list of employees.

Role required: admin

1. Navigate to System Applications > Studio to open Studio.
2. Select a scoped application where you want to create your applet.
   If you have not created an application, you can create one by using the Create Application button. For information on creating a scoped application for mobile, see Create a mobile application using Guided Application Creator.
3. In Application Explorer, navigate to Mobile Studio > Applets and select Applets.
4. Click the pop-out icon that appears to the right of **Applets**.

5. In the applet list, click **Create an applet**.
   The **Create an Applet** window appears on the screen.

   ![Create an Applet window](image)

   - **Define the properties of the applet**
     - **Name**
     - **Icon**
     - **Description**
     - **Hide filter**

   - **Choose a template**
     - **List**
     - **Form**
     - **Employee Directory**
     - **Map**
     - **Group List**
     - **Calendar**
     - **URL**
     - **Chart**

   - **Choose and template section**, select **Employee Directory**.

6. In the **Create an Applet** window. Enter an name and description for your employee directory applet.

7. Click the image next to the **Icon** field to select a color and image for your employee directory applet icon. If you add your combined list applet to an applet launcher, this icon displays for the applet.

8. In the **Choose and template** section, select **Employee Directory**.

9. Click **Create New**.
The Create an Applet window closes, and a new Studio tab opens for your employee directory applet.

10. From the data item list, select a data item.
    If the list is disabled, no data items have been created. To create a data item, click the plus icon next to the list.
    ( ). For more information on creating a data item, see Configure a standard data item. For the employee directory applet, make sure you create a data item with a list of users, for example, from the Users [sys_user] table.

11. Move fields from the All fields list to the List item fields list by double-clicking a field, or selecting a field and clicking the add or remove buttons. Use an image field where the item preview shows an image (E1).

![Employee list item]

12. Click the Profile Screen tab.

13. In the Body section, add fields such as phone number or email. You can associate smart buttons with fields in the body section that allow users to open other native apps on a mobile device, such as an email client.
   You can also use the Replicate from primary button to copy the same fields you used in the Employee Directory Screen.

14. Click Save.

You may want to include functions in your screen to allow users to communicate with or navigate to people in your user records. Learn more about mobile functions in Mobile functions.

**Map screen**

Map screens display a map with locations that are associated to the records in a data item.
Use a map screen when you need your users be aware of the locations related to the records in your data item. For example, map screens can show your users where their assets are located, or which job locations they need to travel to.

**Configure a map applet**

Use Studio to configure a map applet so that your users can see location-based information on a map from their mobile app.

Role required: admin

When you create a data item for a map screen, use information that can be plotted on a map. The map screen includes a card list to display records that are shown on the map.

1. Navigate to **System Applications > Studio** to open Studio.
2. Select a scoped application where you want to create your applet.
   
   If you have not created an application, you can create one by using the **Create Application** button. For information on creating a scoped application for mobile, see **Create a mobile application using Guided Application Creator**.

3. In Application Explorer, navigate to **Mobile Studio > Applets** and select **Applets**.
4. Click the pop-out icon that appears to the right of Applets.

5. In the applet list, click Create an applet. The Create an Applet window appears on the screen.

   Create an Applet window

6. In the Create an Applet window, enter a name and description for your map applet.

7. Click the image next to the Icon field to select a color and image for your map applet icon. If you add your map applet to an applet launcher, this icon displays for the applet.

8. In the Choose and template section, select Map.

9. Click Create New.
The Create an Applet window closes, and a new Studio tab opens for your map applet.

10. In the **Data Item** field of the map template, select a data item.

If the list is disabled, no data items have been created. To create a data item, click the plus icon next to the list. For more information on creating a data item, see *Configure a standard data item*.

11. In the **Location** field, select a field to use as the record's location.

This field must be a reference to a Location [cmn_location] record.

12. Move fields to the Map item fields list by double-clicking a field or selecting a field and clicking the add or remove buttons.

The selected fields appear in the item cards that appear below the map. Use the pattern mapping preview to see how the fields appear. Add at least one field to the header fields for the primary screen and the details screen.

Map screens require that specific field types are present to display information correctly. The E4 location requires a reference to an item in the Location [cmn_location] table, such as an address or a city.

13. Click **Save**.

The configured map screen displays information from your data item plotted on a map, along with a card list to display details on each item.
Configure the form screen so that you can give your users a more detailed view of the records from your data item. The map applet automatically includes a form screen, which displays when a user taps on a card from the map screen. For details on how to configure form screens, see Configure a form applet for use with a list applet.

Change the pin colors for locations in your map applets
Learn how to change pin colors to indicate locations on your map applets.

Role required: admin

1. In the web-based UI, navigate to System Mobile > Applets.
2. In the Item Stream Segments tab, open the item stream segment record.
3. In the Item Stream M2M Segments list on the item stream segment form, open the item stream record.
4. To the right of the Item Stream field on the item stream M2M segment form, click the reference icon
   and then click Open Record in the pop-up window.
5. In the Pin color field, enter a color value in hexadecimal format for your map pin.
6. Click Update.

Configure map filters
Use mobile map filters to enable your users to find what they need in the records on your instance.
You can configure filters for mobile map applets using the same process used to configure filters for a mobile list. For
details on these options, and steps to create a custom filter, see Mobile list filters.

Calendar screen
Calendar screens display a calendar interface, and records associated with the selected date.
Use a calendar to display records when the dates relating to those records are relevant. For example, displaying when your tasks are due, or when important events take place.

The calendar screen displays a calendar interface. Each date on the screen displays an indicator below any date that has records associated to it.

Below the calendar, your users can see a list of records associated to the date selected in the calendar. You can configure the calendar app to display these records in a form screen when the users taps them.

Configure a calendar applet

Configure the calendar applet so that your users can see their scheduled tasks in a calendar view in your mobile apps.

Create the data item and calendar applet
Create the data item and calendar applet in Studio.

Role required: admin

1. Create a data item for your calendar applet. To create this applet, follow the steps detailed in Configure a standard data item.

You now have a data item to use with your calendar applet. Leave the tab for this data item open in Studio, as you will need to return to this tab in later steps.
2. In Application Explorer, navigate to Mobile Studio > Applets and select Applets.
3. Click the pop-out icon that appears to the right of Applets.
4. In the applet list, click Create an applet.
   The Create an Applet window appears on the screen.

   Create an Applet window

   5. In the Create an Applet window, enter a name and description for your calendar applet.
6. Click the image next to the Icon field to select a color and image for your calendar applet icon. If you add your calendar applet to an applet launcher, this icon displays for the applet.
7. In the Choose and template section, select Calendar.
8. Click **Create New**.
   The **Create an Applet** window closes, and a new Studio tab opens for your calendar applet.

9. After creating the applet, you can see the calendar applet form. Click **Submit** to save the record.
   Saving the record displays the record's related lists, which you need to complete the calendar applet setup. Leave
   the tab for this applet open in Studio, as you will need to return to this form in later steps.

*Configure a list and form for your calendar*

Configure list and form applets to display the records associated with your calendar dates.

Role required: admin

1. In the applet list, click **Create an applet**.
   The **Create an Applet** window appears on the screen.
Create an Applet window

2. In the Create an Applet window, enter a name and description for your list applet.
3. In the Choose and template section, select List.
4. To the right of the Choose and template section, enable the Form check box.
5. Click Create New.
   - The Create an Applet window closes, and a new Studio tab opens for your list applet.
6. In the Data Item field, use the data item you created in previous steps.
7. In the Field Configuration section, add the fields you want in your list to the Selected Fields list.
8. Click Save.
9. Click the Form Screen tab to open the form screen configuration options.
10. Click the **Replicate from primary** button

   ![Replicate from primary button]

   to copy the fields from your list to your form.

11. Click **Save**.

*Create an item stream segments*

Create an item stream segment and many-to-many segment for your calendar applet.

**Role required: admin**

1. Return to the tab containing your calendar applet. If this tab is no longer open you can open **Mobile Studio > Applets** and open your calendar applet.
2. In the calendar applet tab, open the **Item Stream Segments** related list, and click **New**.
3. In the **Name** field, enter a name for your item stream segment.
4. In the **Screen** field, select your calendar applet.
5. Click **Save**.
6. In your instance, outside Studio, enter *sys_sg_time_span_item_stream.list* in the filter navigator.
7. In the time span item streams list, click **New**.
8. In the **Time span item stream** form, fill in the fields as needed.

<table>
<thead>
<tr>
<th><strong>Time span item stream form</strong></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Field</strong></td>
<td><strong>Value</strong></td>
</tr>
<tr>
<td>Name</td>
<td>Name of your item stream</td>
</tr>
<tr>
<td>Application</td>
<td>Current scoped application. This field is automatically populated.</td>
</tr>
<tr>
<td>Data item</td>
<td>Data item used by the item stream. Use the data item you created in the previous steps.</td>
</tr>
<tr>
<td>Table</td>
<td>Table used by the item stream. Select the same table used by your data item.</td>
</tr>
<tr>
<td>Interleaved field</td>
<td>Select the field you want to use to interleave your data item.</td>
</tr>
<tr>
<td>Start Date Field</td>
<td>The date field on the table the calendar applet uses for start dates. For example the <strong>Opened</strong> field on an incident record.</td>
</tr>
<tr>
<td>End Date Field</td>
<td>The date field on the table the calendar applet uses for end dates. For example the <strong>Closed</strong> field on an incident record.</td>
</tr>
</tbody>
</table>

9. Click **Submit**.

10. Return to Studio, and open the tab for your item stream segment.

11. In the **Item Stream M2M Segments** list, click the plus (+) button.

12. In the **Item Stream** field, enter the name of the time span item stream record you created in steps 6 through 9.

13. Right-click the header of the form and select **Save**.

14. In the **Item Stream M2M Segments**, double click the name of the time span item stream record to open that record.

15. In the **Item Stream M2M Master Items** list, click **New**.

16. On the **Item Stream M2M Master Item** form, enter the master item for the list applet you created in the **Master Item** field. You can find the master item using the name of the list you created.

17. Click **Submit**.
Configure an item view for your calendar applet

Configure the item view for your calendar applet to define the appearance of the list items under your calendar.

Role required: admin

1. In your instance, outside Studio navigate to System Mobile > Applets
2. Open the record for the list applet you created in the previous steps.
3. In the applet form, click the name of the record in the Item Stream Segments list to open the item stream segment record.
4. In the item stream segment record, click the record name in the Item Stream M2M Segments form to open the master item record.
5. In the master item form, click the reference icon next to the Item view field to open the item view record.
6. In the item view record, replace the contents of the Item view JSON field with the following JSON:

```json
{
    "Id":28090,
    "Type":"ViewGroup",
    "Orientation":"Horizontal",
    "Alignment":"Center",
    "Distribution":"Auto",
    "Margin":{
        "Top":16,
        "Bottom":16
    },
    "Children":[
        {
            "Id":12259,
            "Type":"ViewGroup",
            "Orientation":"Vertical",
            "Alignment":"Left",
            "Distribution":"Auto",
            "Children":[
                {
                    "Type":"Text",
                    "Id":41927,
                    "Margin":{
                        "Top":3
                    },
                    "CellId":"opened_at",
                    "TextColor":"#293e40",
                    "TextAlignment":"Left",
                    "MaxLines":1,
                    "Font":{
                        "Weight":"regular",
                        "Size":12
                    },
                    "Width":60,
                    "Text":"
                },
                {
                    "Type":"Text",
                    "Id":41928,
                    "Width":60,
                    "Margin":{
                        "Top":9
                    },
                    "Text":"
                }
            ]
        }
    ]
}
```
null
"Id":12261,
"Margin":{
  
}
]
}

**Note:** The formatted JSON here uses the **opened_at**, **closed_at**, and **short_description** fields. If you intend to use different fields in your calendar list, replace these values in the above JSON with the fields you want to display.

7. Right-click the form header and select click **Save** in the context menu.

You applet is configured and ready to use in a mobile app.

**Browser screen**

Browser applets open an external URL or a relative URL within your instance.
Use a browser applet to Open a URL from within the ServiceNow app. You can configure relative URLs to open pages within the ServiceNow platform, or an external link. For example, a user can see a knowledge article on the instance via Service Portal.

Relative URLs that direct your users to open pages within your instance display within mobile app. URLs that open external links open the link in the default browser of the user's mobile device.

Configure a URL applet

Use a URL applet to open a URL from within a ServiceNow mobile application. You can configure relative URLs to open pages within the ServiceNow platform.

Role required: admin

1. Navigate to System Applications > Studio to open Studio.
2. Select a scoped application where you want to create your applet.

If you have not created an application, you can create one by using the Create Application button. For information on creating a scoped application for mobile, see Create a mobile application using Guided Application Creator.
3. In Application Explorer, navigate to Mobile Studio > Applets and select Applets.

4. Click the pop-out icon that appears to the right of Applets.

5. In the applet list, click Create an applet. The Create an Applet window appears on the screen.

6. In the Create an Applet window. Enter an name and description for your URL applet.
7. Click the image next to the **Icon** field to select a color and image for your URL applet icon. If you add your URL applet to an applet launcher, this icon displays for the applet.

8. In the **Choose and template** section, select **URL**.

9. Click **Create New**.

The **Create an Applet** window closes, and a new Studio tab opens for your URL applet.

10. Complete the following fields as needed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of your screen. This name appears as a tile in the mobile application.</td>
</tr>
<tr>
<td>Hidden</td>
<td>Whether or not the screen is accessible from the application homepage. You may have screens that you only want users to access from a particular field, in which case you can hide the screen from the homepage.</td>
</tr>
<tr>
<td>Description</td>
<td>Additional information about the screen.</td>
</tr>
<tr>
<td>URL</td>
<td>The URL you want the send to user to.</td>
</tr>
<tr>
<td>Open in external browser</td>
<td>Whether or not the new page opens within the ServiceNow app or in an external browser application on the mobile device.</td>
</tr>
<tr>
<td>Relative URL</td>
<td>Whether or not the URL uses a long address including the https:// or a short URL appended to the instance name. For example, if you wanted to send users to a knowledge base within an instance, you could type <code>$knowledge</code> as the URL. When relative URL is select, the system automatically appends whatever you type in this field to the instance URL so the screen routes to, for example, <code>https://&lt;instance_name&gt;.service-now.com/$knowledge</code>. You can use the relative URL option to route users to the mobile Service Portal. You can use the relative URL to link users directly to a query by including the query URL. For example, if you search for <code>server</code> in a knowledge base, <code>query=server</code> is included in the URL. The relative URL to link a user to a knowledge search for server would be <code>$knowledge?query=server</code>.</td>
</tr>
</tbody>
</table>

11. Click **Save**.

**Chart screen**

Chart screens display an interactive view of a report or performance analytics widget.
Use chart sections to display time series or single score reporting charts, and performance analytics scores. Charts give your managers or owners indications on trends or items which require their attention.

The chart screen supports the following report types:
- Time Series
- Bar
- Pie

To use a chart screen, you will first need to create a dashboard preview that will link to your chart screen. For details on dashboard previews, see Mobile dashboard preview.

Configure a chart applet for a report

Configure a chart applet to give your users access to pre-configured mobile dashboard views for single score reports

Role required: admin

You must have an existing report to display in your applet. If you have not yet configured a report, see Getting started with reports.

1. Navigate to System Applications > Studio to open Studio.
2. Select a scoped application where you want to create your applet.
If you have not created an application, you can create one by using the Create Application button. For information on creating a scoped application for mobile, see Create a mobile application using Guided Application Creator.

3. In Application Explorer, navigate to Mobile Studio > Applets and select Applets.
4. Click the pop-out icon that appears to the right of Applets.
5. In the applet list, click Create an applet.
   The Create an Applet window appears on the screen.
6. In the Create an Applet window. Enter a name and description for your chart applet.
7. Click the image next to the Icon field to select a color and image for your chart applet icon. If you add your chart applet to an applet launcher, this icon displays for the applet.
8. In the Choose and template section, select Chart.
9. Click Create New.
   The Create an Applet window closes, and a new Studio tab opens for your chart applet.
10. In the Chart Screen tab, fill in the fields as needed.
Chart screen fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Whether the chart displays a report or performance analytics chart. Choose <strong>Report</strong> to configure a single score report chart.</td>
</tr>
<tr>
<td>Report</td>
<td>The report to display in the chart applet. Select report from the list. This list is automatically filtered for single score reports on your instance.</td>
</tr>
<tr>
<td>Set the default view point to most recent</td>
<td>Enable to display the most recent data as the default view point.</td>
</tr>
</tbody>
</table>

11. Optional: In the **Functions** tab, click the **Add** button
to add global functions for your chart applet. These functions appear in the top menu of your chart applet.

12. Click **Save**.

Configure a navigation function to direct your users from a dashboard preview to your chart applet. For details on this process, see *Configure a navigation to a chart applet*.

**Configure a chart applet for a Performance Analytics widget**

Configure a chart applet to give your users access to pre-configured mobile dashboard views of a Performance Analytics widget.

Role required: admin

1. Navigate to **System Applications** > **Studio** to open Studio.
2. Select a scoped application where you want to create your applet.
   If you have not created an application, you can create one by using the **Create Application** button. For information on creating a scoped application for mobile, see *Create a mobile application using Guided Application Creator*.
3. In Application Explorer, navigate to **Mobile Studio** > **Applets** and select **Applets**.
4. Click the pop-out icon
   that appears to the right of **Applets**.
5. In the applet list, click **Create an applet**.
The **Create an Applet** window appears on the screen.
6. In the **Create an Applet** window, enter a name and description for your chart applet.
7. Click the image next to the **Icon** field to select a color and image for your chart applet icon. If you add your chart applet to an applet launcher, this icon displays for the applet.
8. In the **Choose and template** section, select **Chart**.
9. Click **Create New**.
   - The **Create an Applet** window closes, and a new Studio tab opens for your chart applet.
10. In the **Chart Screen** tab, fill in the fields as needed.
Chart screen fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Whether the chart displays a report or Performance Analytics chart. Choose Performance Analytics to configure a performance analytics chart.</td>
</tr>
<tr>
<td>Widget</td>
<td>The Performance Analytics widget to display in your chart applet.</td>
</tr>
<tr>
<td>Set the default view point to most recent</td>
<td>Enable to display the most recent data as the default view point.</td>
</tr>
</tbody>
</table>

11. Optional: In the **Functions** tab, click the **Add** button

(+) to add global functions for your chart applet. These functions appear in the top menu of your chart applet.

12. Click **Save**.

Configure a navigation to a chart applet

Configure a navigation function to direct your users to your chart applet.

Role required: admin

These steps detail instruction for creating a navigation from a dashboard preview to a chart applet. To make use of the navigation function, you will need to have configured an applet launcher with a dashboard preview. For details on that process, see [Create a mobile dashboard preview](#).

1. In Studio, navigate to **Mobile Studio > Functions > Navigations**.
2. Click the pop-out icon

(+) that appears to the right of **Navigations**.
3. In the **Navigations** list, click **Create New**.
4. On the **New Navigation** form, fill in the fields as needed.

**Navigation form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of your navigation function</td>
</tr>
<tr>
<td>Destination Type</td>
<td>The type of record your navigation destination. Since you are creating a navigation to a chart applet, select <strong>Applet</strong>.</td>
</tr>
<tr>
<td>Description</td>
<td>A unique description for your navigation function.</td>
</tr>
<tr>
<td>Destination</td>
<td>The applet or applet launcher to use for your navigation destination. Select the chart applet you created in previous steps.</td>
</tr>
<tr>
<td>Context</td>
<td>Whether the function uses the global or record context. For this example, select global.</td>
</tr>
</tbody>
</table>

5. Click **Save**.
Add a navigation function to a chart preview

Configure your chart preview to use a navigation function. This function directs your users to your chart applet from the chart preview on your applet launcher.

Role required: admin

These steps detail instruction for adding your navigation function to an existing dashboard preview. To make use of the navigation function, you will need to have configured an applet launcher with a dashboard preview. For details on that process, see Create a mobile dashboard preview.

1. In Studio, navigate to Mobile Studio > Dashboard Previews.
2. Open the dashboard preview where you want to add your navigation.
3. In the Navigation field, select the navigation function you created in the previous steps.
4. Click Save.

You can now tap the chart preview in your applet launcher to navigate to your chart applet.

Configure a navigation from a chart to a list applet

Configure a navigation to allow your users to navigate to a list of records from your chart applet.

To create your navigation, you need to create the following components:

• Configure a parametrized list and form applet.
• Configure a navigation function to navigate from your chart to your parametrized list and form.
• Associate your navigation function to your chart applet.

These steps assume you have an existing chart applet you want to configure. If you have not yet created a chart applet, you can find details on creating charts at Configure a chart applet for a report.

Create a parametrized list for your chart

Create the list and form you users see when they tap on your chart applet.

Role required: admin

Creating a parametrized list is similar to creating a standard list. In addition to the steps used to create a standard list, you must create

• A parametrized data item for your list. This data item uses your parameter to filter what data is displayed in your list. In this case, the records from your chart applet
• A parameter on your list applet. This parameter stores the value passed into your list.
• A screen UI parameter mapping to associate the parameter in your list with the parameter in your data item.

1. Navigate to System Applications > Studio to open Studio.
2. Select a scoped application where you want to create your applet.
3. In Application Explorer, navigate to Mobile Studio > Applets and select Applets.
4. Click the pop-out icon that appears to the right of Applets.
5. In the applet list, click Create an applet.
The Create an Applet window appears on the screen.
6. In the **Create an Applet** window. Enter a name and description for your list applet.

7. Click the image next to the **Icon** field to select a color and image for your list applet icon. If you add your list applet to an applet launcher, this icon displays for the applet.

8. To prevent the list from displaying a list filter, enable **Hide filter**.

9. In the **Choose a template** field, select **List**.

10. Optional: To the right of the **Choose a template**, enable the check box next to **Form** to include a form screen in your applet. This form displays record information when your users tap a record on the list.

11. Click **Create New**.

   The **Create an Applet** window closes, and a new Studio tab opens for your list applet. If you selected the form option in the previous steps, you can also see a tab to configure your form.
12. Click the plus icon to the right of the **Data Item** field.

   A new Studio tab opens to configure a new data item.

13. On the new data item form, fill in the fields.
New Data Item form

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the data item.</td>
</tr>
<tr>
<td>Table</td>
<td>Table where the data item gets its data. This table should be the same one as the records on your chart.</td>
</tr>
<tr>
<td>Description</td>
<td>Description used to identify your data item.</td>
</tr>
<tr>
<td>Condition Type</td>
<td>Whether the conditions for the data item are declarative, scripted, or use an encoded query. In this case, select URL.</td>
</tr>
</tbody>
</table>

14. Click Save.
15. In the Parameter Definition section, click the plus icon
   ( ) to create a new parameter.
16. In the Parameter Definition pop-up, enter a name for your parameter in the Name field.
17. Click Save.
18. Next to the Append Encoded Query field, click the reference value icon
   ( ).
19. Select the
20. Click Save.
   You have a parametrized data item to use in your parametrized list. The condition type is set to Encoded Query, which in this case, is a URL that the data item receives from the chart applet. You configure this in the next steps.
21. In Studio, navigate back to the tab containing your list applet.

22. In the **Data Item** field, select the parametrized data item you created in the previous steps.

23. Click **Save**.  
The list record is saved. Because the record is saved using a parametrized data item, the field you need to configure parameters are added to the form.

24. In the **Autofill Parameter Definition** section, click the plus icon ![+] to create a new parameter.

25. In the **Name** field, enter a name for your parameter. For example **URL**.

26. In the **Input Type** field, select **Instance relative URL**.

27. Click **Save**.

28. In the **Screen UI Parameter Mapping** section, click the plus icon ![+].

29. In the **Item Parameter** field, select the parameter from the data item. Since this list is drawn from that data item, there should be only one option.

30. In the **UI Parameter field**, select the UI Parameter you created in the preceding steps. Since this list is drawn from that record, there should be only one option.
31. Click Save.

Use the **Field Configurations** sections in the **List Screen** and **Form Screen** tabs to configure which fields you want to appear in your list and form screens. For more detail on these configurations, see [List applet configuration](#).

### Create a navigation function for your chart applet

Create a navigation function to direct your users from the chart applet to the parametrized list.

**Role required:** admin

1. In Studio, navigate to **Mobile Studio > Functions > Navigations**.
2. Click the pop-out icon.
3. In the **Navigations** list, click **Create New**.
4. In the **New Navigation** form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for your navigation function</td>
</tr>
<tr>
<td>Destination Type</td>
<td>Whether the function will navigate to an applet or applet launcher. In this case, select <strong>Applet</strong></td>
</tr>
<tr>
<td>Description</td>
<td>Description for your navigation function</td>
</tr>
<tr>
<td>Destination</td>
<td>Select the list you created in the previous steps</td>
</tr>
<tr>
<td>Context</td>
<td>Whether the function uses the global or record context. In this case, select <strong>Global</strong>.</td>
</tr>
</tbody>
</table>

5. Click Save.

### Assign the navigation function to the chart applet

Assign your navigation function to your chart applet so your users can tap the chart to access the list of records.

**Role required:** admin

1. In Studio, navigate to **Mobile Studio > Applets**.
2. Open your chart applet.
3. In the chart applet tab, click **Functions**.
4. Click the add button to add a function to your applet.
5. In the **function** field, select the navigation function you created in previous steps.
6. Click Save.
7. In the chart applet tab, click Save.

### Mobile fetch types

Fetch type settings determine when data is loaded in your applets. Change your fetch type to optimize load time performance for your applets.
### Mobile fetch types

<table>
<thead>
<tr>
<th>Fetch type</th>
<th>Description</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prefetch</td>
<td>Prefetch is the default fetch type for new applets, except form and details. This option pre-loads form applet data while when your user accesses a list, calendar, or form applet. Lists may take longer to load, but form load time is faster.</td>
<td>This fetch type is the default for most applets. Use this fetch type when the form and form segments do not take much additional time to load.</td>
</tr>
<tr>
<td>Background</td>
<td>The app makes a background network request to load embedded screens or form segments. Embedded screens and form segments load instantly once the background request completes.</td>
<td>Use this fetch type when an applet is not the first loaded, but one your users are likely to navigate to. For example, a related list associated with a form.</td>
</tr>
<tr>
<td>On Demand</td>
<td>The app makes a network request to load the app only when your users navigate to it.</td>
<td>Use this fetch type when a screen is not expected to be used often.</td>
</tr>
<tr>
<td>Dynamic</td>
<td>The screens for the first 10 rows load as described in the <strong>prefetch</strong> type. After 10 first rows, the app loads screens as defined in the <strong>on demand</strong> fetch type. This option is the default for form and details screens. You can change the number of rows loaded with <strong>prefetch</strong> by changing the value of the <strong>Dynamic prefetch count</strong> field.</td>
<td>Use this fetch type when large lists load too slowly using the <strong>prefetch</strong> fetch type.</td>
</tr>
</tbody>
</table>

### Change the fetch type for an applet

Change the fetch type of an applet to change when the app loads its data.

Role required: admin

Applet fetch types typically do not need to be changed. Be sure to test fetch type changes thoroughly before using in a production environment, as fetch type changes can have a significant affect on performance.

1. Navigate to **System Mobile > Applets**.
2. Open an applet.
3. In the **Fetch Type** field, select a fetch type.
   
   If you are in the global scope, the record appear read-only. In this case, you can temporarily switch to the appropriate scope by clicking on the banner notification that appears at the top of the record.

4. Click **Update**.
Mobile styles

Use mobile style elements to change the appearance of your mobile applications.

**Item views**

Use item views to show a summary of a record in an easy to read format. Item views are the patterns that define the appearance and location of fields in an applet. Item views are specific to a screen, each screen has its own unique item view record.

Learn more about item views at [Item views](https:// servicenow.com/).

**UI styles**

Use UI styles to dynamically change item view elements such as background color or font color. You can use UI styles to show or hide item view elements, and add icons next to an element within an item view.

For more detail on UI styles, see [Mobile UI styles](https:// servicenow.com/).

**Screen UI policies**

Use screen UI policies to control which field is mandatory or visible on a details screen or parameter screen. The conditions you define in these policies control when the policy applies. By creating screen UI policies, you can improve readability on your screens, giving your users the information they must do their tasks more efficiently.

In the example to the right, a UI style hides the **Impact** field based on conditions defined by an administrator.

For more information on screen UI policies, see [Screen UI policies](https:// servicenow.com/).
Item views

Item views are patterns that determine the formatting and appearance for fields in your applets.

Item views are the patterns that appear within the app. Item views are specific to a screen. Each screen will have its own unique item view record. The purpose of an item view to so show a summary and support a nice UI for a record. Item views use JSON to determine the configuration for fields in your applet.
Using item views

Item views provide the formatting for your form, list and calendar applets. When you select a template during applet creation, you are choosing the item view for the applet. Applets are configured using Studio in most cases. Learn more about creating applets in Applets.

Modifying item view records

Item views exist as records in your instance on the Item View [sys_sg_item_view] table. Each item view record contains an item view JSON field that contains a JSON. This JSON defines a pattern to control how data appears on the screen. The instance automatically creates this JSON when you select an applet template while designing applets in Studio.

While it is possible to manually create a pattern by either creating a new record or modifying an existing record on the Item View [sys_sg_item_view] table, consider using the pre-configured options in Studio, which have been tested to ensure a consistent experience.

When creating custom patterns, ensure functionality for all your users by testing on iOS and Android, as well as tablets, and any other devices that might be used. ServiceNow does not currently have any guidance around these custom patterns.

Mobile UI styles

UI styles define the font and background color of fields in your applets. You can associate one or more UI styles to an item view record to change the appearance of the fields in that item view. Each UI style applies to a single table. You and only applies under conditions you set within the UI style record.

Create UI Styles

Create and modify your UI Styles using the mobile Studio environment. In a UI style, you select a field from your selected table, and define a font and background color for that field. Use may include conditions to determine when a style is applied. For example, you may want to use a colored background on the Priority field to highlight when a record is high priority.

Each UI style record defines the colors for a single field, however multiple UI styles can be added to an item view to customize the appearance of multiple fields.

Learn more about creating UI styles at Create mobile UI styles.

Create mobile UI styles

Create UI styles to apple to fields in the mobile app.

Create an app and applets before configuring UI styles.

Role required: admin

1. Navigate to System Applications > Studio.
2. From the Select Application window, select the application containing your app.
3. In the Application explorer on the left edge of the screen, click UI Styles, and then click the pop-out icon that appears to the right of the text.
4. Click New.
5. Complete the following fields as needed.
## UI Style fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A name for the UI style. Choose a name that is easily identifiable. You use this name to apply the style to a specific field in an applet.</td>
</tr>
<tr>
<td>Table</td>
<td>The name of the table containing the field you want to add a style to. Select the table matching the table for the applet where you intend to apply this style.</td>
</tr>
<tr>
<td>Condition</td>
<td>The conditions in which you want the field style to appear. For example, you can configure the State field to appear blue, but only when State is Assigned.</td>
</tr>
<tr>
<td>Use item view elements</td>
<td>Enable to select an item view and use the elements from that record.</td>
</tr>
<tr>
<td>Item view</td>
<td>Select an item view. This field appears when the Use item view elements option is selected.</td>
</tr>
<tr>
<td>Item view element</td>
<td>This field appears when the Use item view elements option is selected. The field contains a list of the elements from the item view selected in the Item view field. These elements represent the JSON elements found in the selected Item view record.</td>
</tr>
<tr>
<td>Field name</td>
<td>The name of the field you want to add style to. Only fields available on the table you selected appear in this list.</td>
</tr>
<tr>
<td>Style</td>
<td>The style field supports multiple entries. Each entry has a Name and Value elements. Click the Add icon to add a new entry to the style field or the remove icon to remove existing elements. See the following table for a list of the available UI style names.</td>
</tr>
</tbody>
</table>

*Note: For styles with color values, make sure to include the # symbol. For example, use #FFFFFF for white rather than FFFFFF.*

| Active                             | Whether the UI style is active.                                                                                                         |

## Available UI styles

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>font_color</td>
<td>Use this style to change the text color for a field. The value for this style is a color in hexadecimal format.</td>
</tr>
<tr>
<td>background_color</td>
<td>Use this style to change the background color for a field. The value for this style is a color in hexadecimal format.</td>
</tr>
</tbody>
</table>
### Name | Description
--- | ---
`is_hidden` | Use this style to hide or reveal a field. The value of this style must be `true` or `false`.
`text_decorator_icon` | Use this style to add an icon to the left edge of your field. The value of this style must be the Sys ID of a record on the Icon [sys_sg_icon] table.

6. Click **Submit**.
7. Repeat this process for each condition you want the UI style to appear for. For example, if you want the **State** field to have a different color for each value, create a UI style for each state value.

### Suggested UI styles

Listed here are suggested UI styles for your instance. These styles are WCAG compliant, and are created with high contrast backgrounds to make the labels easily readable for all users.

#### High contrast label styles

<table>
<thead>
<tr>
<th>Label</th>
<th>Background color</th>
<th>Text color</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - Critical</td>
<td>Sweet Pink #FFA4A3</td>
<td>Cherrywood #64201A</td>
</tr>
<tr>
<td>2 - High</td>
<td>Apricot #FBD0B3</td>
<td>Jacko Bean #2F1F0A</td>
</tr>
<tr>
<td>3 - Moderate</td>
<td>Periwinkle #C9C9ED</td>
<td>Gulf Blue #38385B</td>
</tr>
<tr>
<td>4 - Low</td>
<td>Breeze #B5DDE5</td>
<td>Tiber #1A424B</td>
</tr>
<tr>
<td>5 - Planning</td>
<td>Zumthor #D1D6D8</td>
<td>Gable Green #293E40</td>
</tr>
<tr>
<td>Label</td>
<td>Drover #F8F2B4</td>
<td>Jacko Bean #2F1F0A</td>
</tr>
<tr>
<td>Label</td>
<td>Cruise #B5EBD4</td>
<td>Sherwood Green #1E4335</td>
</tr>
<tr>
<td>Label</td>
<td>Iceberg #CFDFD7</td>
<td>Sherwood Green #1E4335</td>
</tr>
<tr>
<td>Label</td>
<td>Marble #D0E6E6</td>
<td>Gable Green #293E40</td>
</tr>
<tr>
<td>Label</td>
<td>Seashell #F1F1F1</td>
<td>Gable Green #293E40</td>
</tr>
</tbody>
</table>
Add the field style to the field in an applet.

Create text decorator icons

Role required: admin

1. On your instance, navigate to System Mobile > Mobile UI Styles.
2. In the UI Styles list, click the New button.
3. In the UI Style form, fill in the fields as needed.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of your UI style</td>
</tr>
<tr>
<td>Table</td>
<td>Table to which the UI style applies. Use the same table as the applet where you apply your style.</td>
</tr>
<tr>
<td>Condition</td>
<td>Condition under which the icon is visible. Leave the condition field blank to apply the icon to all records.</td>
</tr>
<tr>
<td>Use Item View elements</td>
<td>Whether the item view elements are used. Enable this checkbox to create a text decorator icon.</td>
</tr>
<tr>
<td>Item view</td>
<td>The item view where the icon appears. Use the same item view as the applet where you want to see your icon.</td>
</tr>
</tbody>
</table>

### Conditional formatting styles

<table>
<thead>
<tr>
<th>Label</th>
<th>Background color</th>
<th>Text color</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample</td>
<td>Transparent #FFFFFF00</td>
<td>Mahogany #C83C36</td>
</tr>
<tr>
<td>Sample</td>
<td>Transparent #FFFFFF00</td>
<td>Tree Poppy #DC7C40</td>
</tr>
<tr>
<td>Sample</td>
<td>Transparent #FFFFFF00</td>
<td>Chetwode Blue #7070B3</td>
</tr>
<tr>
<td>Sample</td>
<td>Transparent #FFFFFF00</td>
<td>Fountain Blue #5297C4</td>
</tr>
<tr>
<td>Sample</td>
<td>Transparent #FFFFFF00</td>
<td>Oslo Gray #7D8A8C</td>
</tr>
<tr>
<td>Sample</td>
<td>Transparent #FFFFFF00</td>
<td>Earls Green #B09F2C</td>
</tr>
<tr>
<td>Sample</td>
<td>Transparent #FFFFFF00</td>
<td>Sea Green #54AC98</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
<td></td>
</tr>
<tr>
<td>---------------------</td>
<td>----------------------------------------------------------------------</td>
<td></td>
</tr>
<tr>
<td>Item view element</td>
<td>The element of the item view where you want to see your icon.</td>
<td></td>
</tr>
<tr>
<td>Style</td>
<td>The style where you define your icon. For a text decorator icon, use</td>
<td></td>
</tr>
<tr>
<td></td>
<td>text_decorator_icon in the first box, and the Sys_id of the icon in</td>
<td></td>
</tr>
<tr>
<td></td>
<td>the second box.</td>
<td></td>
</tr>
</tbody>
</table>

**Note:** You can find your available icons on the Icon [sys_sg_icon] table. You can right-click any icon on this list and select Copy sys_id from the context menu to copy the sys_id.

4. Click Submit.

This example demonstrates how to apply a text decorator icon to all records on an incident list in the New state.

This example applies to the **State** element of the **Incident List Screen Item View** UI style. The condition field has been set, so the icon only appears for records in the New state. The icon is located on the **Icon** [sys_sg_icon] table.
Incidents

- New

Add memory to laptop
- Avatar
- Due date: 7 days from now

- On Hold

Hack a feature
- Avatar
- Due date

- New

The team file share is not accessible
- Avatar
- Due date

- In Progress

My desk phone stopped working
- Avatar
Add a UI style to an applet

After you create a UI style for a field, you must add it to an applet.

Role required: admin

The process to add UI styles to your applets is different depending on whether the applet is using a pattern from the pattern set 1 or pattern set 2.

Apply a UI style to a pattern from set 1

Use these steps to apply a UI style if you have chosen a pattern from set 1.

Role required: admin

1. In Studio, navigate to Mobile Studio > Applets, then click the expand arrow to see the list of applets for your application.
2. Create an applet or open an existing applet where you want to add UI styles.
3. From the applet configuration tab, add the field you want to add styles to, to the primary screen or details screen header.
   Only certain field locations in the header support UI styles. Make sure you add the field to a position that allows UI styles. If the field is not in a position that allows UI styles, the UI Style Configuration section does not display any options.

   For example, in the following header preview, only the field positioned at E1 allows UI styles. In the UI Style Configuration section, only the E1 field is available, and selection is disabled.
4. In the UI Style Configuration section, move styles from the All Styles list to the Selected Styles list to apply these styles. The table selected for the UI style must match the table for the data item associated with your applet. Only UI styles that meet this criteria appear in the All Styles section of the UI Style Configuration section.

5. Click Save.

**Apply a UI style to a pattern from set 2**

Use these steps to apply a UI style if you have chosen a pattern from set 2.

Role required: admin

1. In Studio, navigate to Mobile Studio > Applets, then click the expand arrow to see the list of applets for your application.

2. Create an applet or open an existing applet where you want to add UI styles.

3. Optional: If you have not yet created a UI style for this pattern, you can create one within Studio by clicking the Create & Edit UI Styles button.

4. Click Map UI Styles to open the item view record.
5. **On the UI Styles field**, click the lock icon

![Lock Icon]

to open and display the list of UI styles assigned to the item view.
6. On the UI Styles field, click the **Add/Remove multiple** icon to open a list of UI styles.

7. Move UI styles from the list on the left to the list on the right to add them to your applet.

8. Click **Save** when you are finished adding UI styles. You are directed back to the item view record after clicking **Save**.
9. Click **Updated** in the item view record to save your changes.

**Screen UI policies**

Use screen UI policies to control which field is mandatory or visible on a mobile app screen, depending on the conditions that you define for the policy. By creating screen UI policies, you can improve readability on your screens, giving your users the information they must do their tasks more efficiently.

Screen UI policies are similar to the UI policies that are used on forms in the instance, but screen UI policies are designed for the mobile app. Policies contain a set of conditions that you can use to determine when the policy applies. When a screen UI policy is triggered by a condition that you defined for the policy, all UI policy rules that are associated with that policy are applied.

**Differences from UI policies for forms**

Screen UI policies are similar to the UI policies that are used on forms with the following exceptions:

- Screen UI policies cannot control whether a field is read-only.
- Screen UI policy conditions cannot be scripted.
- Reference and date/time field types cannot be used in screen UI policy conditions.

**Create a screen UI policy for the mobile app**

Create screen UI policies to dynamically change which fields are mandatory or visible on screens in the ServiceNow Agent mobile app.

Role required: `ui_policy_admin` or `admin`

A screen UI policy condition evaluates all fields on a table even if they are not visible on the screen.

1. Navigate to **System Applications > Studio**.
2. In the Select Application pop-up window, select your mobile application.
3. In the Application Explorer pane, point to **Mobile Studio > UI Policies** and click the pop-out icon that appears.
4. On the Screen UI Policies list, click **New** to create a new screen UI policy.
5. On the form, fill in the fields.

**Screen UI Policy form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the screen UI policy.</td>
</tr>
<tr>
<td>Short description</td>
<td>Short summary of the screen UI policy.</td>
</tr>
<tr>
<td>Order</td>
<td>Processing sequence, from the lowest to the highest number. If two policies conflict, the screen UI policy with the higher number executes. For inherited UI policies, the extended (child) table UI policies are executed first. Next, the base table UI policies are executed from the lowest to the highest specified value.</td>
</tr>
<tr>
<td>Table</td>
<td>Table for the screen that you want to modify.</td>
</tr>
</tbody>
</table>
### Field Description

**Condition**

Conditions that must be fulfilled to apply this screen UI policy. Conditions are built with the condition builder. For details about this tool, see [Condition builder](#).

Conditions are only rechecked if a user manually changes a field on a screen. If the change is made by a UI action, context menu action, or through the list editor, it is not evaluated.

**Reverse if false**

Check box for specifying that the screen UI policy action should be reversed when the conditions of its policy evaluate as false. When the conditions are true, actions are taken and when they change back to false, the actions are reversed (undone).

**Active**

Check box for enabling the screen UI policy.

---

#### UI Policy condition support

<table>
<thead>
<tr>
<th>Supported UI policy conditions</th>
<th>Unsupported UI policy conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Equal and Not Equal (=, !=)</td>
<td>• starts with</td>
</tr>
<tr>
<td>• ISEMPTY, ISNOTEMPTY</td>
<td>• ends with</td>
</tr>
<tr>
<td>• Less than (or equals)</td>
<td>• is anything</td>
</tr>
<tr>
<td></td>
<td>• is different</td>
</tr>
<tr>
<td></td>
<td>• is empty string</td>
</tr>
<tr>
<td></td>
<td>• is one of, is not one of</td>
</tr>
<tr>
<td></td>
<td>• Conditions on DateTime fields</td>
</tr>
<tr>
<td></td>
<td>• &quot;Read only&quot; UI policies</td>
</tr>
</tbody>
</table>

---

**Use a screen UI policy when the state field is not On Hold**

This example shows how you can configure a screen UI policy to apply on screens using data from the Incident [incident] table. You can build a condition that checks the values of fields on this table, in this case, the **State** field. When **[State] [is not] [On Hold]**, the UI policy applies the associated UI policy rules. When you select the **Reverse if false** check box, a field made invisible when the state field is not **On Hold** is made visible when the state is **On Hold**.
Create UI policy rules in the related list. The screen UI policy applies these rules when the data on a screen matches the policy conditions. For details on creating these rules, see Create a UI policy rule.

Create a UI policy rule

Create UI policy rules to control whether the fields that your users see are visible or mandatory with the ServiceNow Agent mobile app, according to the conditions in your screen UI policy actions.

Role required: ui_policy_admin or admin

1. Navigate to System Applications > Studio.
2. In the Select Application pop-up window, select your mobile application.
3. In the Application Explorer pane, point to Mobile Studio > UI Policies and click the pop-out icon that appears.
4. Select the UI policy that you want to create rules for.
5. In the UI Policy Rules related list, click New.
6. On the form, fill in the fields.
**UI Policy Rule form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UI policy</td>
<td>Screen UI policy that this rule is associated with. The field is automatically populated when you create rules from the UI policy rules related list on a screen UI policy form.</td>
</tr>
<tr>
<td>Field name</td>
<td>Name of the field to apply the rule to.</td>
</tr>
<tr>
<td>Visible</td>
<td>Options that change the visibility of a field on a screen. If the field is not available on the screen, you cannot configure a rule to make it appear.</td>
</tr>
<tr>
<td></td>
<td>• Leave Alone: Rule that has no effect on the visibility of this field.</td>
</tr>
<tr>
<td></td>
<td>• True: Rule that makes this field visible.</td>
</tr>
<tr>
<td></td>
<td>• False: Rule that hides this field on the screen.</td>
</tr>
<tr>
<td>Mandatory</td>
<td>Options that change whether the field is mandatory.</td>
</tr>
<tr>
<td></td>
<td>• Leave Alone: Rule that has no effect on this field.</td>
</tr>
<tr>
<td></td>
<td>• True: Rule that makes this field mandatory.</td>
</tr>
<tr>
<td></td>
<td>• False: Rule that makes this field non-mandatory.</td>
</tr>
</tbody>
</table>

**Use a UI policy rule to hide a field**

This example shows how you can control the visibility of the **On hold reason** field. **On hold reason** is selected in the **Field name** field, and the **Visible** field is set to **False**. If the screen UI policy conditions are met, the **On hold reason** field is hidden on the screen.
Mobile icons

Use icons to provide unique visual identifiers for navigation tabs and quick actions in your mobile applications.

Icons for navigation tabs and quick actions

Applets and applet launchers have a static selection of icons. However, navigation tabs use customizable icons that are stored on the icon [sys_sg_icon] table.

Icon records

You can see existing mobile icon records on the icon [sys_sg_icon] table by entering sys_sg_icon.list in the navigation filter of your instance. The fields on the icon form define the appearance of your icon.
For quick actions, you can select an icon using the icon picker in Studio.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the icon.</td>
</tr>
<tr>
<td>Icon</td>
<td>Icon field that consists of one or more key and value pairs. Each value defines the appearance of your icon. See the following table for descriptions of these keys.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of icon</td>
</tr>
<tr>
<td></td>
<td>• Font: Use font type for icons that are used in quick actions.</td>
</tr>
<tr>
<td></td>
<td>• Image: Use image type for icons that are used in navigation bar tabs. Image type icons do not use <code>BackgroundColor</code>, <code>FontColor</code>, or <code>Shape</code> values in the <code>Icon</code> field.</td>
</tr>
</tbody>
</table>

**Icon field keys**

<table>
<thead>
<tr>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>BackgroundColor</code></td>
<td>Background color for the icon. The value must be a valid web color. For example, enter <code>#ffffff</code> for a white background.</td>
</tr>
<tr>
<td><code>FontColor</code></td>
<td>Color for the icon image. The value must be a valid web color. For example, enter <code>#000000</code> for a black image.</td>
</tr>
</tbody>
</table>
### Navigation bar icons

Navigation bar icons consist of a name and an icon value. Your instance theme determines the color of the navigation bar icons and background. The images available for use in the navigation bar icons are displayed in the table in the next section.

You can create icons for your navigation tabs using any of the images that are listed in this table. Depending on the plugins that you activated on your instance, some icon records in the icon [sys_sg_icon] table may be using some of these images. By default, your instance doesn't have icon records for each image.

To see a list of existing navigation bar icons on your instance:

1. In the navigation filter, enter `sys_sg_icon.list` to open a list of icon records.
2. Filter your list for `<Type><is><Image>`.

**Note:** You can use existing icons in your custom applications, but do not change or rename existing icons. These icons may be used in other applications in your mobile environment.

For more information on configuring navigation bars, see *Configure the navigation bar*.

### Available navigation bar icons

<table>
<thead>
<tr>
<th>Icon</th>
<th>Name</th>
<th>Icon</th>
<th>Name</th>
<th>Icon</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Cart" /></td>
<td>Cart</td>
<td><img src="image" alt="Comment" /></td>
<td>Comment</td>
<td><img src="image" alt="Inventory" /></td>
<td>Inventory</td>
</tr>
<tr>
<td>Icon</td>
<td>Name</td>
<td>Icon</td>
<td>Name</td>
<td>Icon</td>
<td>Name</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td><img src="#" alt="Home" /></td>
<td>Home</td>
<td><img src="#" alt="Map" /></td>
<td>Map</td>
<td><img src="#" alt="Bag" /></td>
<td>Bag</td>
</tr>
<tr>
<td><img src="#" alt="User" /></td>
<td>User</td>
<td><img src="#" alt="Request" /></td>
<td>Request</td>
<td><img src="#" alt="Group" /></td>
<td>Group</td>
</tr>
<tr>
<td><img src="#" alt="Hardware" /></td>
<td>Hardware</td>
<td><img src="#" alt="Chart Bar" /></td>
<td>Chart Bar</td>
<td><img src="#" alt="Clock" /></td>
<td>Clock</td>
</tr>
<tr>
<td><img src="#" alt="Heart" /></td>
<td>Heart</td>
<td><img src="#" alt="Megaphone" /></td>
<td>Megaphone</td>
<td><img src="#" alt="Project Status" /></td>
<td>Project Status</td>
</tr>
<tr>
<td><img src="#" alt="Star" /></td>
<td>Star</td>
<td><img src="#" alt="Knowledge" /></td>
<td>Knowledge</td>
<td><img src="#" alt="Agile" /></td>
<td>Agile</td>
</tr>
<tr>
<td><img src="#" alt="File" /></td>
<td>File</td>
<td><img src="#" alt="Wrench" /></td>
<td>Wrench</td>
<td><img src="#" alt="Clipboard" /></td>
<td>Clipboard</td>
</tr>
<tr>
<td><img src="#" alt="Calendar" /></td>
<td>Calendar</td>
<td><img src="#" alt="Bell" /></td>
<td>Bell</td>
<td><img src="#" alt="Exclamation" /></td>
<td>Exclamation</td>
</tr>
</tbody>
</table>
Quick action icons have configurable name, icon, and color values. The images that are available for use in quick action icons appear in the table in the next section.

Select a quick action icon using the icon picker when you create a quick action.
For more details on quick action configuration, see *Quick actions*.

**Available quick action icons**

<table>
<thead>
<tr>
<th>Icon</th>
<th>Name</th>
<th>Icon</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="https://via.placeholder.com/40" alt="User icon" /></td>
<td>User</td>
<td><img src="https://via.placeholder.com/40" alt="Check Circle icon" /></td>
<td>Check Circle</td>
</tr>
<tr>
<td><img src="https://via.placeholder.com/40" alt="Group icon" /></td>
<td>Group</td>
<td><img src="https://via.placeholder.com/40" alt="Edit 2 icon" /></td>
<td>Edit 2</td>
</tr>
<tr>
<td><img src="https://via.placeholder.com/40" alt="Add User icon" /></td>
<td>Add User</td>
<td><img src="https://via.placeholder.com/40" alt="Wrench icon" /></td>
<td>Wrench</td>
</tr>
<tr>
<td>Icon</td>
<td>Name</td>
<td>Icon</td>
<td>Name</td>
</tr>
<tr>
<td>------</td>
<td>-------</td>
<td>------</td>
<td>-------</td>
</tr>
<tr>
<td><img src="image" alt="Cart" /></td>
<td>Cart</td>
<td><img src="image" alt="Phone" /></td>
<td>Phone</td>
</tr>
<tr>
<td><img src="image" alt="Flag" /></td>
<td>Flag</td>
<td><img src="image" alt="Hardware" /></td>
<td>Hardware</td>
</tr>
<tr>
<td><img src="image" alt="Star" /></td>
<td>Star</td>
<td><img src="image" alt="Mail" /></td>
<td>Mail</td>
</tr>
<tr>
<td><img src="image" alt="Comment" /></td>
<td>Comment</td>
<td><img src="image" alt="Calendar" /></td>
<td>Calendar</td>
</tr>
<tr>
<td><img src="image" alt="Comments" /></td>
<td>Comments</td>
<td><img src="image" alt="Map Pin" /></td>
<td>Map Pin</td>
</tr>
<tr>
<td><img src="image" alt="Bag" /></td>
<td>Bag</td>
<td><img src="image" alt="Megaphone" /></td>
<td>Megaphone</td>
</tr>
<tr>
<td>Icon</td>
<td>Name</td>
<td>Icon</td>
<td>Name</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td><img src="image" alt="Headphones Mic" /></td>
<td>Headphones Mic</td>
<td><img src="image" alt="Chart Bar" /></td>
<td>Chart Bar</td>
</tr>
<tr>
<td><img src="image" alt="Copy" /></td>
<td>Copy</td>
<td><img src="image" alt="Knowledge" /></td>
<td>Knowledge</td>
</tr>
<tr>
<td><img src="image" alt="Clipboard" /></td>
<td>Clipboard</td>
<td><img src="image" alt="Cloud" /></td>
<td>Cloud</td>
</tr>
<tr>
<td><img src="image" alt="Clipboard Exclamation" /></td>
<td>Clipboard Exclamation</td>
<td><img src="image" alt="Heart" /></td>
<td>Heart</td>
</tr>
<tr>
<td><img src="image" alt="Web Link" /></td>
<td>Web Link</td>
<td><img src="image" alt="Trash" /></td>
<td>Trash</td>
</tr>
<tr>
<td><img src="image" alt="Information" /></td>
<td>Information</td>
<td><img src="image" alt="File" /></td>
<td>File</td>
</tr>
</tbody>
</table>

Create an icon for a mobile navigation tab

Create an icon for a navigation tab in your mobile applications to help your user quickly identify their apps.
Role required: admin

1. In the navigation filter, enter `sys_sg_icon.list` to open a list of Icon [sys_sg_icon] records.
2. Click New.
3. On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name for your icon. You may want to include something to identify the icon as a navigation tab icon. For example, you can add <code>–nav</code> to the end of the name.</td>
</tr>
<tr>
<td>Icon</td>
<td>Text fields within the Icon field that you can use to define key and value pairs that control the appearance of the icon. In the first field, enter Name. In the second field, enter the name of the icon that you want to use. For a list of the available icons for navigation bar tabs, see the reference in the preceding section.</td>
</tr>
<tr>
<td>Type</td>
<td>Icon type. Select Image.</td>
</tr>
</tbody>
</table>

4. Click Submit.
5. Navigate to System Applications > Studio.
6. Open your mobile application.
7. In the Application Explorer, navigate to Mobile Studio > Application Menu, and select the mobile app where you want to use your icon.
8. In the Navigation tabs section of the Navigation form, click the navigation tab where you want to use your icon.
9. In the Tab form, update the Icon field with your new icon.
10. Click Update.

Note: Changes made to the navigation bar are not visible to your users until they have logged out and logged back into the mobile app.

Mobile functions

Configure functions in Studio to determine which actions users can perform in the mobile app.
Function types

**Actions**

Use action functions to change data, such as assigning a task to yourself or adding a comment to a record. Action functions require a write-back action item to operate. Configure your actions with input parameters to include user input in the changes you make. For more detail on this function type, see *Action functions*.

**Navigations**

Use navigation functions to transition from your current screen to another applet or applet launcher. For example, opening a record from a list, or moving from an employee user profile screen to a manager user profile screen. For more detail on this function type, see *Navigation functions*. 
Function context

When you create an action, you must choose whether the function context is record or global. Choose a context based on what you want to function to do.

Record context

Use record context for functions that use the current record. For example, a function on a form used to update or edit the record would use record context.

Global context

Use global context if the function is not required to act on an existing record. For example, a function to create a new record on a table, or global actions that appear in an applet launcher.

Function locations

For each function you create for an application, you must associate it with a specific location. You can associate most functions with a top menu, a swipe, or a specific field. For details on this configuration, see Associate a function with a location in the app.

Carried parameters

Parameters are a way of creating a variable or a placeholder that is waiting for input from either the user or the database. The variable then queries the database or the user for more information. For examples detailing how to use parameters in your functions, see Mobile parameter tutorials.

Action functions

Action functions make changes to records on your instance.

Use action functions to perform tasks such as assigning a task to yourself or adding a comment to a record. Action items can optionally include an input parameter, which collects input from your user to update a record.

Action items

Each action function includes an action item, which defines what the action function is and how it works. Learn more about action items at Configure an action item.
Action function types

Non-parametrized action function

Actions functions allow the user to change something in the database. For example, making an update or adding a comment to a record requires an action function. Use non-parametrized action function in situations where no user input is required.

For details on configuring these actions, see Configure an action function.

Parametrized action function

Use parametrized action functions to request additional information from your end users when they interact with an action. For example, you can allow your end users to provide a reason for rejecting an approval by adding a text input parameter to a Reject button. When you create an action item that has parameters, you must also configure the UI parameters for the action function.

For details on configuring these actions, see Create an action function with parameters.

Chat launcher action function

Create a contextual link in your mobile application so that your users can connect to Virtual Agent or Live Agent, and receive information that pertains to their issues. By using a contextual link, your users automatically receive the information that you define about the relevant record.

For details on configuring these actions, see Configure a contextual link to Virtual Agent.

Note: Connect chat is not supported in the native chat launcher.
Configure an action item

For an action function to work, you must create an action item to associate with the action function. Action items define what the action function is and how it works.

Most action items use parameters.

Use action items to define what an action function does when a user uses that function. The following steps detail creating an action without parameters. To create a parametrized action item, see Configure an action item with parameters.

Note: Mobile apps cannot perform any actions that the platform cannot perform. For example, if you use ACLs to prevent a user from closing an incident without adding a resolution code and notes, the user cannot close an incident in the app without the same requirements. Keep this in minds when creating actions, so that you can add the correct parameters.

Role required: admin

1. In Studio, navigate to Mobile Studio > Action Items.
2. Click the pop out icon to open the Actions items list in a tab.
3. Click Create a new action item.
4. Complete the following fields as needed.

Action item fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A name for the action item. You can have multiple action items with the same name. Make sure you choose a name that is easily identifiable.</td>
</tr>
<tr>
<td>Description</td>
<td>More information to help you identify the action item.</td>
</tr>
<tr>
<td>Type</td>
<td>The kind of action item. Choose from the following:</td>
</tr>
<tr>
<td></td>
<td>• New</td>
</tr>
<tr>
<td></td>
<td>• Update</td>
</tr>
<tr>
<td></td>
<td>• Delete</td>
</tr>
<tr>
<td></td>
<td>• Script</td>
</tr>
<tr>
<td></td>
<td>Different fields appear on the action item form depending on the type of action you select.</td>
</tr>
<tr>
<td>Table</td>
<td>The table the action item applies to, for example, Incident.</td>
</tr>
<tr>
<td>Execution Script</td>
<td>The script executed by the action. This field only appears if you select Script as the type. For more information, see the example below.</td>
</tr>
<tr>
<td>Use current record as condition</td>
<td>Whether you want a separate set of query conditions for the action item. If selected, the Query conditions field is disabled. For update or delete actions, you must define the record you are updating or deleting by providing a sys ID. Marking Use current record as condition as true allows you to do this without creating a parameter.</td>
</tr>
<tr>
<td>Query Condition</td>
<td>Filter conditions that apply to the action item.</td>
</tr>
</tbody>
</table>
Field | Description
--- | ---
Set field values | Determine the field values for an action. For example, if you want to create an action that updates an incident with a state of Resolved, you may the field values for State = Resolved. You can also create parametrized items to pass into the field value.

5. Click **Submit**.

The following example uses a script to assign a task to the current user, using the SMTask object. The first `if` statement checks to see that the input is a valid `wm_task` record and ends the script if it is not. The second `if` statement contains code that assigns the task to the current user, if the user has permission, as determined by the `canAssignToSelf` method. This action was done as a script rather than an update so that these checks could be included.

```javascript
(function WriteBackAction(input) {
  var smTask = new global.SMTask();
  var wotGR = new GlideRecord("wm_task");
  if (!wotGR.get(input.sys_id)) {
    gs.error("wot_assign_to_me write-back action - failed to find work order task");
    gs.addErrorMessage(gs.getMessage("Task assignment failed."));
    return;
  }
  if (smTask.canAssignToSelf(wotGR))
    smTask.assignToMe(gs.getUserID(), input.sys_id);
  else
    gs.addErrorMessage(gs.getMessage("Not a valid task assignment."));
})(input);
```

If you use parameters for the action item, you can call them in the script. The call in the script must match the parameter name exactly. For example, if the parameter name is `wb_wot_reject_work_note`, you can call it in the script using `gr.work_notes = input.wb_wot_reject_work_note;`.

Associate the action item with an **action function**.

**Configure an action item with parameters**

Parameters determine the information you are passing into the action to ensure you are changing the correct record and to enforce required fields as needed. Create an action item with parameters to define the changes being made to an action and how the changes get made.

Use action items to define what an action function does when a user uses that function. The following steps detail creating an action with parameters. To create a non-parametrized action item, see **Configure an action item**.

1. In Studio, in a mobile application, navigate to **Mobile Studio > Action Items**.
2. Click **Create a new action item** or select an existing action item to add parameters to.
   For more information on creating an action item, see **Configure an action item**.
3. In the Item Parameters related list, click **New**.
4. In the Name field, enter a name for the action item.
   You can have multiple parameters with the same name, so choose a name that you can distinguish easily.
5. From the Type list, select the type of parameter. The type determines how the user interacts with the mobile UI. For example, a type of Decimal or Integer tells the mobile device to open a numbers-only keypad. Select from the following types.
• String: Uses a full keyboard for input. Use the String type for list parameters, such as priority or state, or for reference fields, such as assigned to or caller.
• Integer: Uses a numbers-only keypad for input
• Decimal: Uses a numbers-only keypad for input
• Boolean: Uses a true or false selection option

**Note:** Making a Boolean mandatory has no effect. Boolean fields are always considered to have a value. A selected check box has a value of true and an unselected check box has a value of false. Either of these values satisfies the requirement of a mandatory field.

• DateTime: Uses a calendar with an exact time selector
• Date: Uses a calendar

6. In the Item Parameter tab, click **Save**.
7. In the Action Item tab, click the contextual reference value icon

( )

to add the item parameter you created as a condition in the condition builder for the action item.
8. Click **Update**.

**Configure an action function**

Actions functions allow the user to change something in the database. For example, making an update or adding a comment to a record requires an action function.

Before creating an action function, create an action item.

Role required: admin

1. Open Studio, and open the application where you want to add an action function.
2. In Studio, navigate to **Mobile Studio > Functions > Actions**.
3. Click the pop out icon

   ![Icon](image)

   to open the Actions list in a tab.
4. In the upper right corner of the list, click **Create new**.
5. Complete the fields in the following table as needed.

**Action function fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the action. Choose a name that is easy to identify.</td>
</tr>
<tr>
<td>Description</td>
<td>Additional information about the action.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of action. This field automatically populates with <strong>Action item</strong>.</td>
</tr>
<tr>
<td>Context</td>
<td>The level to apply the action to.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Record</strong>: Applies an action at the record level.</td>
</tr>
<tr>
<td></td>
<td>For example, use this option to create a button that changes the state of</td>
</tr>
<tr>
<td></td>
<td>a record. You can set the context to Record for actions on a particular</td>
</tr>
<tr>
<td></td>
<td>field, a particular record, or swipe actions. If the action function</td>
</tr>
<tr>
<td></td>
<td>includes a UI parameter with the <strong>Input source</strong> field set to <strong>Auto fill</strong>, you must specify the table in the <strong>Table</strong> field.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Global</strong>: Applies an action at the global, or list level.</td>
</tr>
<tr>
<td></td>
<td>For example, use this option to add a button that creates a record. You</td>
</tr>
<tr>
<td></td>
<td>can only set the context to Global for actions in the list menu, or swipe</td>
</tr>
<tr>
<td></td>
<td>actions. Do not use <strong>Global</strong> for actions that are in the context of a</td>
</tr>
<tr>
<td></td>
<td>record, or actions that include a UI parameter with the <strong>Input source</strong></td>
</tr>
<tr>
<td></td>
<td>field set to <strong>Auto fill</strong>.</td>
</tr>
<tr>
<td></td>
<td>For more information on button locations, see <strong>Associate a function with a location in the app</strong>.</td>
</tr>
<tr>
<td>Action item</td>
<td>Action item to associate with the action. For more information on action items, see <strong>Configure an action item</strong>.</td>
</tr>
<tr>
<td>Jump to previous screen</td>
<td>Redirects the user to the previous screen after completing the action.</td>
</tr>
<tr>
<td>Show refresh on previous screen</td>
<td>Shows a <strong>New Update</strong> message after the user completes the action and redirects back to the previous screen.</td>
</tr>
<tr>
<td>Allow images upload</td>
<td>Allows an image upload. Images are stored in the attachment [sys_attachment] table.</td>
</tr>
<tr>
<td>Show signature field</td>
<td>Requires a user signature before submitting the action. The signature field allows the user to sign with their finger or type their name. If <strong>Allow images upload</strong> is selected, an image of the signature is sent to the Signature Images [signature_image] table when the action executes. When this option is selected, the signature form displays on a separate page. To overlay the signature form, use the <strong>Signature</strong> option in the Preconditions field.</td>
</tr>
<tr>
<td>Use Overlay</td>
<td>Overlays a text input parameter on the current details screen. Must have exactly one text input parameter defined for the action. If this option is not defined, the input parameter displays on a separate screen.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Preconditions</td>
<td>Select an option to require user confirmation before submitting the action.</td>
</tr>
<tr>
<td></td>
<td>• <strong>None</strong>: Do not require user confirmation. This option is the default.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Signature</strong>: Require a user signature before submitting the action. The signature field allows the user to sign with their finger or to type their name. If <strong>Allow images upload</strong> is selected, an image of the signature is sent to the Signature Images [signature_image] table when the action executes. When this option is selected, the signature form overlays the current screen.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Credentials</strong>: Require a user to input their credentials before submitting the action. This option is only supported with local authentication.</td>
</tr>
</tbody>
</table>

**Condition tab fields**

<table>
<thead>
<tr>
<th>Condition Type</th>
<th>Type of condition to use.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• <strong>Declarative</strong>: Adds a condition builder to the form. For more information on using the condition builder, see <a href="#">condition builder</a>.</td>
</tr>
<tr>
<td></td>
<td>• <strong>Script</strong>: Adds a script condition field to the form.</td>
</tr>
</tbody>
</table>

**Note**: Only displays if **Context** is **Record**.

<table>
<thead>
<tr>
<th>Table</th>
<th>The table that the action applies to.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Condition</td>
<td>Conditions that must be met to use the action. For example, prevent users from resolving an incident that is in a state of closed, resolved, or canceled.</td>
</tr>
<tr>
<td>Roles</td>
<td>Limit user access to an action by role.</td>
</tr>
</tbody>
</table>

**Messages tab fields**

<table>
<thead>
<tr>
<th>Show confirmation message</th>
<th>Displays a confirmation message to verify whether user wants to continue with the action.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Confirmation message</td>
<td>The confirmation message to display.</td>
</tr>
</tbody>
</table>

**Note**: Only displays if **Show confirmation message** is selected.

<table>
<thead>
<tr>
<th>Confirm label</th>
<th>Label of the button to confirm the action.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cancel label</td>
<td>Label of the button to cancel the action.</td>
</tr>
</tbody>
</table>

**Note**: Only displays if **Show confirmation message** is selected.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Success message</td>
<td>Text for a success confirmation message. Use curly braces to create a dynamic success message. For example, <code>{{number}} has successfully updated</code>.</td>
</tr>
<tr>
<td>Failure message</td>
<td>Text for a failure confirmation message. Use curly braces to create a dynamic failure message. For example, <code>{{number}} could not be updated</code>.</td>
</tr>
</tbody>
</table>

**Acknowledgment Messages tab fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Show acknowledgment text</td>
<td>Require user acknowledgment before submitting the action. If this field is enabled, the <strong>Context</strong> field must be <strong>Record</strong>.</td>
</tr>
<tr>
<td>Acknowledgment field</td>
<td>Field that includes the acknowledgment text you want to display to the user. Select the table for the field in the <strong>Condition</strong> tab. Only String fields are supported.</td>
</tr>
<tr>
<td>Confirm label</td>
<td>Label for the confirmation button that appears below the acknowledgment text. The default value is <strong>I Agree</strong>.</td>
</tr>
</tbody>
</table>

After you create an action function, you must associate it with a specific location in the mobile app. You can add an action function to a top menu, a swipe action, or to a specific field. For more information on associating the action function to a location, see Associate a function with a location in the app.

**Create an action function with parameters**

Request additional information from your end users when they interact with an action. For example, you can allow your end users to provide a reason for rejecting an approval by adding a text input parameter to a Reject button. When you create an action item that has parameters, you must also configure the UI parameters for the action function.

The UI parameter fields do not appear on the action function unless you have associated an action item with parameters to the action function.

1. In Studio, in a mobile application, navigate to **Mobile Studio > Functions > Actions**.
2. Complete the action function fields as needed. In the action item field, include an action item with parameters. For more information on completing the fields for the action function, see Configure an action function.
3. In the UI Parameters related list, click **New**.
4. Complete the following fields as needed.
UI Parameter fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for the UI parameter. You can have multiple UI parameters with the same name so make sure that you choose something you can easily discover later. If the parameter requires user input, use a user-friendly name.</td>
</tr>
<tr>
<td>Parameter type</td>
<td>Choose Screen or Button.</td>
</tr>
<tr>
<td>Button</td>
<td>Choose a button to associate with this parameter. This field only appears when the Parameter type field is set to Button.</td>
</tr>
<tr>
<td>Screen</td>
<td>Choose a screen to associate with this parameter. This field only appears when the Parameter type field is set to Screen.</td>
</tr>
<tr>
<td>Mandatory</td>
<td>Whether the user is required to enter information for that field.</td>
</tr>
<tr>
<td>Order</td>
<td></td>
</tr>
<tr>
<td>Input source</td>
<td>The source for the parameter input. Choose User Input or Auto fill.</td>
</tr>
<tr>
<td>Field</td>
<td>Definition</td>
</tr>
<tr>
<td>----------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Input type</td>
<td>The UI used to complete the variable. The available options in this field depend on the choice entered in the <strong>Input source</strong> field. Choose from one of the following options.</td>
</tr>
</tbody>
</table>

**User input options:**

- **Text**
  - Provides a simple text field. This option works best for fields that require text, such as work notes or resolution details.

- **List**
  - Opens a list for the user to select from. This option works well for reference fields that require specific information.

  **Note:** The list input returns a maximum of 1000 results. For references that require more than 1000 results, use the Searchlist input type.

- **SearchList**
  - Provides a search bar so that users can search in a list. Select a reference field on a table for users to search on.

- **QR/Barcode**
  - Provides the option to search by QRC or barcode. Learn more about this feature in *Mobile bar-code scanning*.

**Auto fill options**

- **GPS Location**
  - Auto-fills with the mobile user's location when the action is used. For detail on this feature, see *Mobile location tracking configuration*.

- **Date**
  - Auto-fills with the date and time stamp of when the action is used.

- **Constant**
  - Auto-fills with a constant value. The **Constant value** field appears on the form when this choice is selected.

- **Source field**
  - Auto-fills from a specified field. The **Button parent table** and **Source field** fields appear on the form when this choice is selected.

- **User**
  - Auto-fills with the mobile user's user record.
<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input style</td>
<td>How the user interacts in the UI. Choose from inline or pop-up.</td>
</tr>
<tr>
<td>Constant value</td>
<td>Enter a static value to use for this parameter. This field only appears if you use the Constant input type.</td>
</tr>
<tr>
<td>Placeholder</td>
<td>Text that appears below the field type. This option does not appear if you have a default value selected.</td>
</tr>
<tr>
<td>Table name</td>
<td>The name of the table you want to pull information from. This field only appears if you use the List or the SearchList input type. Select a table that correlates with the action item. For example, if you create an action item that uses the incident table, select the incident table for the UI parameter as well.</td>
</tr>
<tr>
<td>Field name</td>
<td>The name of the field you want to pull information from. This field only appears if you use the List or the SearchList input type. Select a reference field from the table you selected. For example, select the Assign to field.</td>
</tr>
<tr>
<td>Default value type</td>
<td>The value that appears by default in the UI field. Select one of the following options.</td>
</tr>
<tr>
<td></td>
<td>• None: There is no default text. This option works well for a list input type.</td>
</tr>
<tr>
<td></td>
<td>• Manual: A field appears for you to enter a default term. For example, Search for a field. The manual default works well for search or text input types.</td>
</tr>
<tr>
<td></td>
<td>• Source field: Pulls in information from the field selected in the Field name section. If you select this option, a table name and field are required.</td>
</tr>
<tr>
<td>Button parent table</td>
<td>The parent table for the source field. This field only appears if you use the Source field input type.</td>
</tr>
<tr>
<td>Source field</td>
<td>The field used for the source field. This field only appears if you use the Source field input type.</td>
</tr>
<tr>
<td>Carried</td>
<td>Whether this parameter a carried parameter. Use carried parameters to move information between different screens and actions.</td>
</tr>
</tbody>
</table>

5. Click **Save**.
6. From the Action function, in the Action parameter mappings related list, click **New**.
   a) In the Button field, if the field is not completed already, enter the name of the action function.
   b) In the Item Parameter tab, search for the item parameter you created for the action item. For example, **Assignee**.
   c) In the UI parameter field, search for the name of the UI parameter you created.
   d) Click **Save**.

7. From the action function, click **Update**.

Associate the function with a specific applet. For more information on how to associate the action, see **Associate a function with a location in the app**.
Configure a contextual link to Virtual Agent

Use Studio to create a contextual link in your mobile application so that your users can connect to Virtual Agent and receive information that pertains to their issues. By using a contextual link, your users automatically receive the information that you define about the relevant record.

Role required: admin

Before you can create a virtual link to Virtual Agent feature, you must have the Glide Virtual Agent plugin (com.glide.cs.chatbot) installed on your instance. For more details, see Virtual Agent.

Perform the following steps in Studio.

1. Navigate to **Mobile Studio > Functions > Actions**.
2. Click the pop-out icon to open the Actions list in a tab.
3. In the upper right corner of the list, click **Create new**.
4. On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the action. Choose a name that is easy to identify.</td>
</tr>
<tr>
<td>Description</td>
<td>Descriptive information about the action.</td>
</tr>
<tr>
<td>Type</td>
<td>Action taken by this function. In this example, select Chat Launcher.</td>
</tr>
<tr>
<td>Context</td>
<td>Context of the action. Select Record.</td>
</tr>
</tbody>
</table>

5. Click **Submit**.

*Create UI parameters for your Virtual Agent link*

Create UI parameters to pass information from your record into Virtual Agent so that your users get the information they need for their issues.

Role required: admin

1. In the **UI parameters** tab of your action function, click **New**.
2. On the form, fill in the fields.
   - If a field isn't mentioned in the following table, use the default values for that field.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table</td>
<td>Table that matches the record where you will apply your button. For example, if you intend to create a link to Virtual Agent on an incident form, select the Incident [incident] table.</td>
</tr>
</tbody>
</table>
UI parameter form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name that describes the information that is being passed to Virtual Agent. For example, you might use <em>incident_caller</em> for a UI parameter that contains the caller for an incident record. Because these names are used in scripts, do not use spaces in the names.</td>
</tr>
<tr>
<td>Parameter type</td>
<td>Parameter type that can be a button or screen. Select <em>Button</em>.</td>
</tr>
<tr>
<td>Button</td>
<td>Action that you created in the previous steps.</td>
</tr>
<tr>
<td>Input source</td>
<td>Source where the UI parameter gets its value. Select <em>Auto fill</em>.</td>
</tr>
<tr>
<td>Input type</td>
<td>Location where the user inputs data for the parameter. Select <em>Source field</em>.</td>
</tr>
<tr>
<td>Parent button table</td>
<td>Table that is used in the <em>Table</em> field of the action that you created in the previous steps.</td>
</tr>
<tr>
<td>Source field</td>
<td>Field that contains the information that is being passed to Virtual Agent. For example, select <em>Caller</em> if you are passing the value of the <em>Caller</em> field on an incident form.</td>
</tr>
</tbody>
</table>

**Note:** The table and sys_id of your record are automatically passed to Virtual Agent, so you don’t need to create UI parameters for these values.

3. Click **Save**.
4. Optional: Repeat steps 1 through 3 to create additional UI parameters for any other information that you want to pass to Virtual Agent.

You can direct your mobile users to a specific Virtual Agent topic by creating a *search_text* UI parameter. You can create this parameter using the preceding steps, with the following changes:

UI parameter form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the UI parameter. For this example, enter <em>search_text</em>.</td>
</tr>
<tr>
<td>Input Type</td>
<td>Location where the user will input data for the parameter. For this example, select <em>Constant</em>.</td>
</tr>
<tr>
<td>Constant Value</td>
<td>Value that matches a keyword from one of your Virtual Agent conversations. When your users tap the chat button, they are directed to the matching topic.</td>
</tr>
</tbody>
</table>

The UI parameters that you created are available to your Virtual Agent topics within the *vaContext* object. You can access these parameters in Virtual Agent Designer by entering *vaContext.*., followed by the name of your UI parameter. For example, you can access the *incident_caller* UI parameter by entering *vaContext.* incident_caller.. For more details on using context variables in Virtual Agent, see [Virtual Agent scripts](#).
**Navigation functions**

Navigation functions transition users from their current screen to another applet or applet launcher.

Use navigation functions to transition from your current screen to another applet or applet launcher. For example, opening a record from a list, or moving from an employee user profile screen to a manager user profile screen. Navigation functions use the **Global** or **Record** context.
Navigation function contexts

Global Context
Use global context navigation functions in situations where the navigation does not depend on information from a record. For example, to navigate to a specific applet or applet launcher, you can use a global context navigation function. For examples of global context navigation configuration, see:

- Configure a navigation to an applet
- Configure navigation to an applet launcher

Record Context
Use record context navigation functions in situations where the navigation depends on information from a record. For example, you want to navigate from the Assigned to field in an incident record to the assignee’s user record. In this case, you would use a record context navigation. For an example of record context navigation configuration, see Tutorial: Configure navigation from a list applet to another list applet.

Configure a navigation to an applet
Navigation functions enable you to define simple ways for end users to navigate within the mobile platform, for example, navigating to a record from a field on another record.
Before you create a navigation function, you should have a source applet and a destination applet for your navigation.
Role required: admin
1. In Studio, navigate to Mobile Studio > Functions > Navigation.
2. Click the pop-out icon.
3. In the Navigations list, click Create New.
4. In the Properties section, in the Name field, type a name for the navigation.
   Because you can reuse navigations, use a name that you can easily identify.
5. In the Description field, provide a description for your navigation function.
6. In the Destination Type field, select whether you are navigating to an applet or applet launcher.
7. In the Destination field, select the applet you want to navigate to.
   Depending on the destination screen selected in your Destination field, you might need to include parameter settings.
8. In the Context field, select whether the action should be available for a specific or global context. For an applet, select Global.
9. Set the Available Offline option to determine whether the navigation function is available offline.
   The applet and the application containing the navigation function must be marked as available offline for the navigation function to work offline. For more information on offline mode, see Offline mode.
10. If you added a destination applet that has a parametrized data item, in the Parameter Setting section, update the redirection parameter fields.
   a) Click the value in the parameter name field.
   b) Complete the parameter settings.
   Note that some of the fields vary depending on the value you select for the Type field.
Redirection parameter form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Destination UI parameter</td>
<td>The name of the parameter you created for the data item. If the parameter Type value is User Input, use a user-friendly name.</td>
</tr>
<tr>
<td>Type</td>
<td>Source of information for the parameter. The available options are:</td>
</tr>
<tr>
<td></td>
<td>• Field: The parameter gets the information from a field on a table.</td>
</tr>
<tr>
<td></td>
<td>• Constant: The parameter uses a constant value.</td>
</tr>
<tr>
<td></td>
<td>• Source UI parameter</td>
</tr>
<tr>
<td>Source field</td>
<td>Source for the parameter. This field appears only if Field is the redirection parameter type. The table defined in the data item determines the available fields.</td>
</tr>
<tr>
<td>Constant value</td>
<td>Value to appear for the data item parameter. This field appears only if Constant is the redirection parameter type.</td>
</tr>
<tr>
<td>Source UI parameter</td>
<td>UI parameter for the source screen. This field appears only if Source UI Parameter is the redirection parameter Type value.</td>
</tr>
</tbody>
</table>

c) Click Save.

11. In the Advanced Configurations section, determine the display conditions and the roles required for the navigation function.
   For example, you could define a different function or action for the same button according to user roles.
   a) Select a table in the Display Conditions section.
   b) In the condition builder, select filter conditions to limit when the navigation function is available.
      For example, if you want users to be able to navigate to a problem from an incident form, you can configure the navigation function to appear only when the problem field has a value.
   c) Add the roles that can use the navigation function to the Select Roles section in the Roles Permission section.

12. Click Save.

After you create a navigation function, you must associate it with a specific location in the mobile app. For more information, see Associate a function with a location in the app.

Tutorial: Configure navigation from a list applet to another list applet

Configure a navigation function that directs your users to another list applet when they tap a record from a list.

When you create a list applet, you can include a form applet that lets your users display record details when they tap a record. However, you can instead use a navigation function to direct your users to any other type of applet when they tap a record from a list. In this example, you can see a navigation function that directs users to a second list.
You can create incident priorities that are displayed to your users in an applet or within an applet launcher. When a user taps a priority entry in the primary list, they are directed to a second list showing all incidents that match that priority. Users can then tap an item from the secondary list to see that record in a form screen.

To create an incident priority that you can display in an applet or within an applet launcher, you create:

- A primary list that shows your incident priorities.
- A secondary list that shows the incidents matching the selected priority.
- A navigation function to list the primary and secondary lists.

Create a primary list
Create your first list. This list displays the priorities for your incidents. Your users can tap on any priority to see records that match that priority.

Role required: admin

1. Navigate to **System Applications** > **Studio**.
2. Select a scoped application where you want to create your applet.
3. In the Application Explorer, navigate to **Mobile Studio** > **Applets** and select **Applets**.
4. Click the pop-out icon that appears to the right of **Applets**.
5. In the applet list, click **Create an applet**.
6. In the **Name** field, enter a name for your applet.
7. From the choose a template section, select List.
8. Deselect the option to include a form with this list. This list uses a navigation function to direct to another list.
9. Click Create New.
   The list applet is created. The configuration screen for your new applet displays in Studio.
10. In the Data Item field, click Add

      

      to create a new data item for your list.
11. In the Name field, enter a name for the data item.
12. In the Table field, select Incident [incident].
13. In the All of these conditions must be met section, enter \(<\text{Priority}\> <\text{is one of}> <1-\text{Critical}><2-\text{High}><3-\text{Moderate}>\).

   

   Data item conditions

15. Click Save.
16. Click the tab that contains your list applet to return to the applet configuration screen.
17. In the Data Item field, select the data item that you created in the previous steps.
18. Click Change Pattern.
19. Select pattern 11 from the available patterns.

   

   Pattern 11

20. Click Done.
21. In the Field Configurations section, select priority for your field.
22. Click Save. You have a list applet that shows an entry for each incident priority. Currently, nothing happens when you tap these entries. In the next steps, you'll create a second list to show the incidents that match a priority. Then, you'll create a navigation function to navigate to the list that shows priorities to the second list that shows the incidents matching that priority.
Create a secondary list

Create a second list of incidents to match the categories that your users selected in the primary list. This list uses a parametrized data set to receive the chosen priority from the primary list.

Role required: admin

1. In the Application Explorer, navigate to Mobile Studio > Applets and select Applets.
2. Click the pop-out icon that appears to the right of Applets.
3. In the applet list, click Create an applet.
4. In the Name field, enter a name for your applet.
5. Select List from the Choose a template section.
6. Select the option to include a form with this list.
   This list uses a navigation function to direct to another list.
   This second list shows your incidents. Selecting the form option lets your users see the incident form when they tap an incident.
7. Click Create New.
   The list applet is created. The configuration screen for your new applet appears in Studio.
8. In the Data Item field, click Add
   to create a new data item for your list.
9. In the Name field, enter a name for the data item.
10. In the Table field, select Incident [incident].
11. Click Save.
12. In the **Parameter Definition** section, click the add button
   
   ![Add Parameter Symbol]

   to add a new parameter.

   The Parameter Definition pop-up window appears.

13. In the **Name** field, enter **priority**.

14. In the **Type** field, select **String**.

15. Click **Save**.

   The Parameter Definition pop-up window closes.

16. In the **All of these conditions must be met** section, enter `<Priority> <is>`.

17. Click the reference value button
   
   ![Reference Value Button]

   and select **priority**.

18. Click **Save**.

   You have created a parametrized data item to use in your second list. This data item uses a parameter to pass the 
   priority from the first list that you created in previous steps.

19. Return to the configuration form for your second list.

20. In the **Data Item** field, select the parametrized data item that you created in the previous steps.

21. In the **User Input Parameter Definition** section, click the add button
   
   ![Add Parameter Symbol]

   to add a new parameter.

22. In the **Name** field, enter **priority**.

   Leave all other fields at their default value.

23. Click **Save**.

24. In the **Screen UI Parameter Mapping** section, click the add button
   
   ![Add Parameter Mapping Symbol]

   to add a new parameter mapping.

25. In the **Item parameter** field, select **priority**.

26. In the **UI parameter** field, select **priority**.

27. Click **Save**.

   The list is now configured to use the priority that you selected in the first list. This information passes between 
   lists using a parameter.

28. In the **Field Configurations** section, select the incident fields that you want to appear on the incident list.

29. At the top of the list configuration screen, click the **Form Screen** tab.

30. In the **Field Configurations** section, select the incident fields that you want to appear on the incident form.

31. Click **Save**.

   You now have a second list, which displays the incidents in the category that your user selected from the first 
   list. In the final steps, you'll create a navigation function that can handle the navigation between the two lists.

---

### Create a navigation function

Create a navigation function to connect your two lists. The navigation function handles the transition from one list to 
the next when the user taps a record on the list.

Role required: admin

1. In the Application Explorer, navigate to **Mobile Studio > Functions > Navigations**. 
2. Click the pop-out icon that appears to the right of Navigations.

3. Click Create New.

4. In the Name field, enter a name for your navigation function.

5. In the Destination Type field, select Applet.

6. In the Destination field, select the parametrized list that you created in previous steps.

7. In the Context field, select Record.

8. In the Table field, select Incident [incident].

9. In the Parameter Name section, click the parameter. There should be a single parameter named priority. The Redirection Parameter pop-up window appears.

10. In the Source field field, select Priority.

11. Click Save.

You have configured a navigation function to direct to your second list, and used the parameter settings to pass the priority field to your second list.

12. In the Application Explorer, return to Mobile Studio > Applets and select your first list applet.

13. Click the Functions tab to display the functions configuration section.

14. Under List Item Function, click the add button.

15. In the Function field, select the navigation function that you created in the previous steps.

16. Click Save.

Now that your navigation function is configured, you can tap a priority in your first list, and see a list of all incidents with that priority. You can also tap an incident in the second list to see the form for that incident.

Configure navigation to an applet launcher

Use a navigation function to direct your users to an applet launcher page. Use this navigation function in a quick action, footer button, or top menu.

Navigation functions direct your users from one screen to another. In this example, you create a navigation function to direct your users to an applet launcher. After you have created this function, you can put it to use. You can use your navigation function anywhere you can normally use functions. This example shows how to include this function in a screen's footer, a quick action, or a screen's top menu.

Create a navigation function

Create a navigation function to direct to an applet launcher.

Role required: admin

In the following steps you create a navigation function to direct your users to an existing application launcher. You will need to have an existing applet launcher to use as a destination for this function. For detail on creating applet launchers, see Applet launchers.

1. In Studio, navigate to Mobile Studio > Functions > Navigations.

2. Click the pop-out icon that appears to the right of Navigations.

3. In the Navigations list, click Create a new navigation.

4. On the navigation form, fill in the fields.
Navigation form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for your navigation function.</td>
</tr>
<tr>
<td>Destination Type</td>
<td>The destination type for this navigation. For this example, select <strong>Applet Launcher</strong>.</td>
</tr>
<tr>
<td>Description</td>
<td>Description for your navigation function.</td>
</tr>
<tr>
<td>Destination</td>
<td>The applet or applet launcher users will see after using this function.</td>
</tr>
<tr>
<td>Context</td>
<td>Whether the function applies at the record or table level. Select <strong>Global</strong>.</td>
</tr>
<tr>
<td>Available Offline</td>
<td>Whether the action will be available in offline mode.</td>
</tr>
</tbody>
</table>

5. Click **Save**.

*Use your navigation function as a quick action*

Use your navigation function as a quick action if users need to frequently access the launcher.

Role required: admin

Quick actions are appear on applet launcher pages, and serve as a quick way to provide shortcuts to commonly used applets and actions. Add your navigation function as a quick action to give your users an easily accessible shortcut to the launcher defined in your navigation function.

1. In Studio, navigate to **Mobile Studio > Applet Launchers**.
2. Click the pop-out icon that appears to the right of **Applet Launchers**.
3. Open an applet launcher where you want to add your quick action.
4. In the Quick Actions section, click the add button to create a new quick action.
5. In the **Create a Quick Action** pop-up. Fill in the fields.

Create a Quick Action form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>The label for this quick action. Use a value to inform your users of what the action does. In this case, you could use the name of the destination applet launcher.</td>
</tr>
<tr>
<td>Icon</td>
<td>The icon that appears for this quick action.</td>
</tr>
<tr>
<td>Function</td>
<td>The function used by this quick action. Select the function you created in previous steps.</td>
</tr>
</tbody>
</table>

6. Click **Save**.
7. In the applet launcher form, click **Save**.
Test the navigation using your mobile app. After completing the steps, you can see a quick action in your applet launcher with the label and icon you defined in these steps. Tapping the quick action will direct you to the applet launcher you defined in your navigation function.

*Use your navigation function as a top menu selection*

Use your navigation function as a top menu selection to give your users access to the application launcher from within an applet.

Role required: admin

Your users access the top menu function by tapping the icon in the upper right corner of an applet screen. In these steps you add your navigation function to a top menu function of an existing applet. If you have not configured any applets, you need to do that first. For detail on that process see *Create an applet*.

1. In Studio, navigate to Mobile Studio > Applets.
2. Click the pop-out icon
   
   that appears to the right of Applets.
3. Open an applet where you want to add your navigation function as a top menu action.
4. In the applet form, click Functions to access the applet's functions.
5. In the Top Menu Functions section of the form, click the add button

   to create a new top menu function.
6. In the Top Menu Function pop-up, Enter a label for your menu item in the Label field, and select the navigation function you created in earlier steps in the Function field.
7. Click Save.
8. On the applet form, click Save.

Test the top menu action using your mobile app. After completing the steps, you can see a menu icon in the upper right corner of your applet. Tapping the menu icon displays the items in the menu. Your navigation function appears in this list using the name you provided in the Label field in step 6.

*Use your navigation function as a footer function*

Footer functions enable your end users to take an action on a details segment of a form. You can use the navigation function you created to give your users access to the application launcher from within the details segment of your forms. Unlike the top menu function, the form footer button is visible at the bottom of the form.

Role required: admin

1. In Studio, navigate to Mobile Studio > Applets.
2. Click the pop-out icon

   that appears to the right of Applets.
3. Open an applet where you want to add your navigation function as a top menu action.
4. In the applet form, select the Body section.
   
   Click the Form Screen tab if you are accessing a form screen within another applet, such as a list.
5. **Note:** The applet you select should be a form applet, or another applet that contains a form screen, such as a list, calendar, or map applet.
6. In the applet form, select the Body section.
   
   Click the Form Screen tab if you are accessing a form screen within another applet, such as a list.
5. In the footer functions section at the bottom of the form, click the add button to create a new footer function.

6. In the <strong>Footer Function</strong> pop-up, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label</td>
<td>The label for this footer function. Use a value to inform your users of what the action does. In this case, you could use the name of the destination applet launcher.</td>
</tr>
<tr>
<td>Function</td>
<td>The function used by this footer function. Select the function you created in previous steps.</td>
</tr>
<tr>
<td>Button Emphasis</td>
<td>The button emphasis for the footer function. This selection affects the color of the button. For a navigation function, select &lt;strong&gt;Primary&lt;/strong&gt; or &lt;strong&gt;Secondary&lt;/strong&gt;.</td>
</tr>
</tbody>
</table>

7. Click <strong>Save</strong>.

8. In the applet form, click <strong>Save</strong>.

Test the footer function using your mobile app. After completing the steps, you can see a footer function button at the bottom of your form applet. Tap the button to navigate to the applet launcher you defined in the previous steps.

**Smart button functions**

Use smart buttons to interact with native applications on your mobile device, such as your phone, map, or email applications.
Smart buttons

Kerry
Service Desk Agent
IT

Business phone
+1 999-999-9999

Mobile phone
+1 999-999-9999

Email
kerry@servicenow.com

Location
540 North Dearborn Street, Chicago, IL
Use smart buttons to quickly perform actions you specify outside the app. These actions can include navigating to a location on a map, sending a text message or email to a contact, placing a phone call, or opening a URL in a browser. You can choose from any of the following options.

**Smart button types**

<table>
<thead>
<tr>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address</td>
<td>Use the <strong>Address</strong> smart button type to navigate to an address on your instance using your mobile device's navigation software. For example, you can configure a button to direct your field technician to their next work location.</td>
</tr>
<tr>
<td>Email</td>
<td>Use the <strong>Email</strong> smart button type to send an email. The recipient can be a static email address, or an email address stored on a table on your instance. For example, you can configure a button to send an email with a preset subject from within an incident record.</td>
</tr>
<tr>
<td>Phone</td>
<td>Use the <strong>Phone</strong> smart button type to place a call or send an SMS text message. As with email, the recipient can be a static, or a number stored on an instance record. For example, you can configure a button on an incident record to call your customer. You can configure your smart button to automatically use the phone number in your customer's user record.</td>
</tr>
<tr>
<td>URL</td>
<td>Use the <strong>URL</strong> smart button to navigate to a web address. For example, you can configure a button to open your company's website. URLs can be relative or external. Relative URLs display within the app, while external URLs open in the mobile device's default browser.</td>
</tr>
</tbody>
</table>

**Smart button context**

Context determines whether a smart button uses information in a record, or static information you define when creating the smart button.

**Record Context**

Use record context when you want to use information from the record where you have included your smart button. For example, you want to create a smart button to call the user listed in the **Caller** field of an incident. In this case you would select **Record** in the smart button's context field. The number used for the phone call is specified by selecting a table and field where the caller's phone number is stored. Using this method, the number called changes dynamically when accessing incidents with different callers.

**Global Context**

Use global context when you want to create a button that does not depend on information in the record. For example, you want to create a smart button that calls your company's support number. This number remains the
same no matter where the smart button is placed in your application. When creating a smart button with the global context, you have a **Phone Number** field where you can input the number.

For examples of configuring smart buttons, see *Configure a smart button*.

**Smart button advanced configurations**

Use advanced configurations to control when your button appears, based on conditions or roles. You may, for example want to hide an email button for records that have no email address, or display a URL link only to your admin users.

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display Conditions</td>
<td>Conditions under which the smart button is visible.</td>
</tr>
<tr>
<td>Roles Permission</td>
<td>Roles that can see the smart button. If no roles are selected, the button is visible to all users.</td>
</tr>
</tbody>
</table>

**Configure a smart button**

Smart buttons are actions that allow you to perform another action, such as sending an email, making a phone call, pulling up a location, or navigating directly to a URL.

Role required: admin

1. In Studio, navigate to **Mobile Studio > Functions > Smart Buttons**.
2. Click the pop out icon to open the Smart buttons list in a tab.
3. Click **Create New**.
4. In the new smart button form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for your smart button</td>
</tr>
<tr>
<td>Type</td>
<td>Type of smart button. Select from:</td>
</tr>
<tr>
<td></td>
<td>• Address</td>
</tr>
<tr>
<td></td>
<td>• Email</td>
</tr>
<tr>
<td></td>
<td>• Phone</td>
</tr>
<tr>
<td></td>
<td>• URL</td>
</tr>
<tr>
<td>Description</td>
<td>Unique description for your smart button to make it easy to identify.</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Context</td>
<td>Context for your smart button. Select from:</td>
</tr>
<tr>
<td></td>
<td>• Record</td>
</tr>
<tr>
<td></td>
<td>• Global</td>
</tr>
</tbody>
</table>

**Note:** Smart buttons using the Address type must use the Record context. The Context field becomes read only when you select that type.

Table | Table where you want to use your smart button.                        |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Fields</td>
<td>Field where you want to add your smart button. The table selected in the Table field determines which fields are available.</td>
</tr>
<tr>
<td>Phone Number</td>
<td>Phone number called by the smart button. This field is only visible when you select Phone in the Type field, and Global in the Context field.</td>
</tr>
</tbody>
</table>

**Note:** If you have selected Record in the context field, the phone number called is the number in the Fields field. Be sure to select a field that contains a phone number.

<table>
<thead>
<tr>
<th>Phone Type</th>
<th>Type of phone the smart button will communicate with. Select from:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• Cellular</td>
</tr>
<tr>
<td></td>
<td>• Land Line</td>
</tr>
<tr>
<td></td>
<td>This field is only visible when you select Phone in the Type field.</td>
</tr>
<tr>
<td>SMS</td>
<td>The SMS message sent by the smart button. This field is only visible when you select Cellular in the Phone Type field.</td>
</tr>
<tr>
<td>Email Address</td>
<td>Email address to which the email is sent. This field is only visible when you select Email in the Type field, and Global in the Context field.</td>
</tr>
</tbody>
</table>

**Note:** If you have selected Record in the context field, the email is sent to the email address in the Fields field. Be sure to select a field that contains an email address.

<table>
<thead>
<tr>
<th>Email Subject</th>
<th>Subject for the email message. This field is only visible when you select Email in the Type field.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Email Content</td>
<td>Content of the email message. This field is only visible when you select Email in the Type field.</td>
</tr>
<tr>
<td>URL Label</td>
<td>The visible label of your URL link. This field is only visible when you select URL in the Type field.</td>
</tr>
<tr>
<td>Field</td>
<td>Value</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>URL Link</td>
<td>The URL to which your smart button will navigate. This field is only</td>
</tr>
<tr>
<td></td>
<td>visible when you select Email in the Type field, and Global in the</td>
</tr>
<tr>
<td></td>
<td>Context field. Note: If you have selected Record in the context field,</td>
</tr>
<tr>
<td></td>
<td>the button will use the value in the Fields field for the URL. Be</td>
</tr>
<tr>
<td></td>
<td>sure to select a field that contains a URL.</td>
</tr>
</tbody>
</table>

5. Optional: In the Advanced Configuration section, create conditions to limit when the smart button appears. For example, if you are creating a Phone type of smart button, you could create a condition to require a value in the phone number field.

After you create a smart button function, you must associate it with a specific location in the mobile app. You can add a smart button function to a top menu, a swipe action, or to a specific field. For more information on associating the smart button function to a location, see Associate a function with a location in the app.

Configure a smart button using a parametrized URL

Use parametrization to include record specific information in your smart buttons.

Role required: admin

This example demonstrates how parameters are used to improve the functionality of smart buttons. In this case, the smart button provides a link to a list of knowledge articles. The button uses the short description of the current incident as that search criteria for the knowledge article list.

Watch this two-minute video to learn how to find a relative link in your ServiceNow instance.

1. Navigate to System Applications > Studio and open your mobile application.
2. Navigate to Mobile Studio > Functions > Smart Buttons and click the pop out icon to open the Smart buttons list in a tab.
3. Click New to create a new smart button.
4. Fill in the smart button properties as shown in the table.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>KB with Short Description</td>
</tr>
<tr>
<td>Type</td>
<td>URL</td>
</tr>
<tr>
<td>Relative URL</td>
<td>Checked</td>
</tr>
<tr>
<td>Open in external browser</td>
<td>Unchecked</td>
</tr>
<tr>
<td>Table</td>
<td>Incident [incident]</td>
</tr>
<tr>
<td>fields</td>
<td>Short Description</td>
</tr>
<tr>
<td>URL Label</td>
<td>Show KB</td>
</tr>
<tr>
<td>Relative Link</td>
<td>sp?id=search&amp;spa=1&amp;t=kb&amp;q={{short_description}}</td>
</tr>
</tbody>
</table>

5. Click Save.
6. In the web-based UI, navigate to **System Mobile > Functions**.

7. In the functions list, find and open the record for the smart button you created in step 4.

8. Uncheck **Take source value from field**.

9. Click **Update**.

10. Associate your new button to your app. For detail on this process, see *Associate a function with a location in the app*.

Tapping this button displays a list of knowledge base articles, using the short description for the incident as a search term.

**Associate a function with a location in the app**

For each function you create for an app, you must associate it with a specific location. You can associate most functions with a top menu, a swipe, or a specific field.
Quick action functions

Use the quick action item location to provide your users with a shortcut to an item or action in your mobile apps. Quick actions appear under a plus icon (++) at the bottom of the applet launcher page in Android. On iOS devices, quick actions appear under the ellipsis icon (…) in the applet launcher header.

Configure quick actions on your applet launcher form in Studio. For details on configuring quick actions, see Quick actions.
Top menu functions

Use top menu function location for less frequently used actions, and on forms where you have more actions than will fit conveniently in the footer.

1. Navigate to an applet that you want to add a navigation function to.
2. From a primary or details screen, in the screen configuration section, switch to the Functions tab.
3. In the Top Menu Functions section, click the Add icon.
4. In the Label field, type a name for the navigation function. This name appears in the top menu in the app.
5. In the Function field, click the type of function you want to add to the top menu, then select the name of the function you created.
6. Click Done.
List item functions

Use the list item function location to trigger a function when a user taps on a record in a list. This function location can for example, navigate a user to another applet when they tap a record.

1. Navigate to a list applet that you want to add a list item function.
2. Switch to the Functions tab.
3. In the List Item Functions section, click the Add icon.
4. In the Function field, click the type of function you want to add to the top menu, then select the name of the function you created.
5. Click Done.

For an example of using this function location to navigate from one list to another, see Tutorial: Configure navigation from a list applet to another list applet.
Swipe functions

The swipe function location only applies to primary screens that display a list. You cannot, for example, create a swipe function for a map screen.

1. Navigate to a list-type applet that you want to add a swipe function to.
2. From the primary screen, in the screen configuration section, switch to the Functions tab.
3. In the Swipe Functions section, click the Add icon.
4. In the Swipe Functions definitions window, in the Label field, type a name for the swipe function. This name appears when you swipe an item in a list.
5. In the Function field, click the type of function you want to add to the top menu, then select the name of the function you created.
6. In the Swipe direction field, select Left or Right, depending on which direction you want the swipe to appear.
7. Click Done.
Field functions enable your end users to change the value of a field. Field functions only apply to items on the details screen in the body displayed area.

1. In Studio, open a form screen that you want to add a field function to. Make sure that there are elements in the body display area that you can add field functions to.

2. In the Details Segment of the Body tab, add a function under **Field Functions**.

3. Select the field that you want to add the function to and select the function. This function is the behavior that the system performs when a user taps the field function.

4. Click **Save**.
Media section functions

You can associate a function to one of your media sections in an applet launcher. Use this function to navigate from your media to a applet or applet launcher, or URL.

Note: The Check in now text in the image represents the media section function.

1. In your instance, navigate to System Mobile > Applet Launchers.
2. Open the applet launcher containing your media section.
3. In the Body tab, find your media section in the Applet Launcher Sections list.
4. Click the title of your media section to open the media section record.
5. In the Function instance field, select the function you want to use in your media section.
6. Click Update.
Footer functions enable your end users to take an action on a details segment of a form.

1. In Studio, open a form screen that you want to add a footer function to.
2. In the Details Segment of the Body tab, add a function under Footer Functions.
3. Add a label and a function. This function is the behavior that the system performs when a user taps the footer function.
4. Add a color option. Choose from Primary, Destructive, or Secondary.
5. Click Save.
Mobile push notifications

Both administrators and users must do some configuration to set up push notifications in the mobile app.

To set up push notifications for the mobile app, you must perform the following steps:

• In Studio create an app and an applet to apply a push notification to.
• In the platform, configure the push notification message and the push notification.

Note: Push notifications are not supported on on-premise instances.
Push notifications structure

1. Admin creates an app and applet(s) in ServiceNow Studio

2. Admin creates/updates push messages and notifications for SN mobile app

3. Users install SN mobile app on their devices and update preferences for receiving push notifications on SN mobile app
Push notification layout

A layout of a push notification determines the information that appears in the mobile app. This layout is defined in a JSON payload within the Push Notification Message Contexts [sys_push_notif_msg_content] record. The layout defines the following properties, as shown in the example image:

- Status
- Identifier
- Description

To see an example of a JSON payload that is used to configure a notification, see Set up push notifications for mobile apps.

Mobile deep linking

You can use mobile notifications to link to navigate to specific content within a mobile app using a deeplink script. For details on this script API, see MobileDeepLinkGenerator - Global

Set up push notifications for mobile apps

Configure push notifications for ServiceNow mobile apps to keep your users informed.

Role required: admin

This procedure covers non-actionable push notifications. For details on actionable push notifications, see Configure actionable push notifications.

Note:

Apple is deprecating their legacy binary protocol, which will affect customers who use custom push notifications. Custom push notifications for customers with mobile branding will stop working as of November 2020. This deprecation only affects customers using New York and Orlando instances who have apps customized with Mobile publishing.

- New York customers using mobile publishing customized apps can prevent the push issue by upgrading to New York Patch 10.
• Orlando customers using mobile publishing customized apps can prevent the push issue by upgrading to Orlando Patch 5.

Paris customers are not affected.
For more information see KB0830082.

2. Click New.
3. In the push notification form, fill in the fields.

Push notification form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of your push notification.</td>
</tr>
<tr>
<td></td>
<td><img src="image" alt="Note: This name does not appear to users viewing the notification." /></td>
</tr>
<tr>
<td>Application</td>
<td>The scoped application for your notification. This field automatically uses the current application scope.</td>
</tr>
<tr>
<td>Actionable</td>
<td>Whether the push notification is actionable. Click to enable this field.</td>
</tr>
<tr>
<td></td>
<td><img src="image" alt="Note: For details on actionable push notifications, see Configure actionable push notifications." /></td>
</tr>
<tr>
<td>Screen</td>
<td>An optional screen associated with your notification. The notification uses this screen. This screen displays to a user when they tap on the notification.</td>
</tr>
<tr>
<td>Category</td>
<td>Category of your notification. This field is visible only when the Actionable field is enabled. This category defines which actions are available in the push notification. You will fill in this field in later steps.</td>
</tr>
</tbody>
</table>

4. Click Save.
   Your notification record is created. After saving, the Push Action Instances related list appears on the form.

Create message content for your notification

Create a record to determine what information the notification displays to your users.

Role required: admin

1. In your push notification record, click Create Push Message Content in the Related Links section.
   If you are not already in this record, Navigate to System Mobile > Mobile Push Notifications > Push Notifications, and open the notification you created in previous steps.
2. In the Push Notification Message Content form, fill in the fields.
Push Notification Message Content form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of your push message content record. This name is not visible to your users.</td>
</tr>
<tr>
<td>Application</td>
<td>Scoped application associated with this record. This field defaults to the current application.</td>
</tr>
<tr>
<td>Push app</td>
<td>Mobile app that uses your push notification.</td>
</tr>
<tr>
<td></td>
<td>• Select ServiceNow Mobile Application for the Mobile Agent app.</td>
</tr>
<tr>
<td></td>
<td>• Select ServiceNow Request Application for the Now Mobile app.</td>
</tr>
<tr>
<td></td>
<td>• Select ServiceNow Onboarding Mobile Application for the Mobile Onboarding app.</td>
</tr>
</tbody>
</table>

**Note:** If you are using a branded mobile application, select the record for the branded app. For example, select ServiceNow Onboarding Intune Mobile Application if you are using the Microsoft Intune branded application.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Push Message Generation</td>
<td>Script that generates the push notification. When you create message content using these steps, the instance automatically generates this script.</td>
</tr>
</tbody>
</table>

3. Click Submit.

**Create a standard notification**

Create a standard notification on your instance using the platform notifications.

Role required: admin

1. Navigate to **System Notification > Push > Create Push Notification**.
2. In the notification form, fill in the fields.

**Push notification form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of your Notification. This name is not visible to your users.</td>
</tr>
<tr>
<td>Active</td>
<td>Whether the notification is active. The instance does not send inactive notifications to your users.</td>
</tr>
<tr>
<td>Table</td>
<td>The table containing the records relating to your notification. For example, notifications about new incidents would use the Incident [incident] table.</td>
</tr>
<tr>
<td>Push Message Only</td>
<td>Whether the notification is a push message only. This field is read-only by default.</td>
</tr>
<tr>
<td>Category</td>
<td>The category for your notification. This field value is Uncategorized by default.</td>
</tr>
</tbody>
</table>

3. Right-click the form header and select Save.
4. Click to display the **What to Send** tab.
5. Click the lock icon
   ![Lock](image)
   next to the **Push Messages** field.
6. Click the reference icon
   ![Reference](image)
   to display the **Push Notification Messages** list.
7. Click **New**.
8. In the push notification message form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of your push notification message record. This name is not visible to your users.</td>
</tr>
<tr>
<td>Application</td>
<td>Scoped application associated with this record. This field defaults to the current application.</td>
</tr>
<tr>
<td>Push App</td>
<td>Mobile app that uses your push notification.</td>
</tr>
<tr>
<td>Table</td>
<td>The table containing the records relating to your notification. For example, a notification about new incidents would use the Incident [incident] table.</td>
</tr>
<tr>
<td>Push Message Content</td>
<td>Select the record for the push message content you created in the previous section.</td>
</tr>
<tr>
<td>Message</td>
<td>Text the user sees in the push notification. For example, An incident has been assigned to you.</td>
</tr>
</tbody>
</table>

9. Click **Submit**.
10. In the **Notification** form, click **Update**.

**Register the standard push notification**

Register your standard push notification in a push application to being using it in your mobile apps.

Role required: admin

The push application handles sending notifications to mobile devices. Add your new notification to the **ServiceNow Mobile Application** push application so your mobile app users start receiving this notification.

1. Switch to the **Global** scope.
2. Navigate to **System Notification > Push Application**.
3. Open the record for the push application you want to modify.
   - Open **ServiceNow Mobile Application** for the Mobile Agent app.
   - Open **ServiceNow Request Application** for the Now Mobile app.
   - Open **ServiceNow Onboarding Mobile Application** for the Mobile Onboarding app.
4. Right-click the record header and choose Configure > Related Lists from the menu.
5. Add Default Push Registration > Push App to the list on the right.
6. In the Push Default Registrations related list, click the New button to create a new Push Default Registration record.
7. In the Notification field, select the notification you created in the previous steps.
8. Click Submit.

Your push message is rendered on mobile devices that it is registered to. To see how push notifications appear to end users, see Mobile push notifications.

Note: Once you have created or modified a notification, users must log out and back in for the changes to take effect.

Configure push applications for iOS branded apps

Install an iOS push certificate to use push notifications on your branded ServiceNow mobile apps for iOS.

Role required: admin

To use push notifications on your branded ServiceNow mobile apps for iOS, you must upload your Apple certificates p12 file, and enter your key store password into your instance.

Note: These steps are not necessary if you are not using a custom branded mobile apps. For more detail on custom branding, see Request and publish a branded mobile app.

2. Open the record for the push application you want to modify.
   - Open ServiceNow Mobile Application for the Mobile Agent app.
   - Open ServiceNow Request Application for the Now Mobile app.
   - Open ServiceNow Onboarding Mobile Application for the Mobile Onboarding app.

Note: If you are using a branded mobile application, select the record for the branded app. For example, select ServiceNow Onboarding Intune Mobile Application if you are using the Microsoft Intune branded application.

3. In the Push field, change the value from REST API to Direct.
4. In the Apple tab, click the reference icon next to the Certificate field.
5. In the X.509 Certificates list pop-up, click New.
6. In the X.509 Certificates form, fill in the fields.
**X.509 Certificate form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name for your certificate.</td>
</tr>
<tr>
<td>Notify on expiration</td>
<td>User to receive a notification when the certificate expires.</td>
</tr>
<tr>
<td>Type</td>
<td>Certificate type. Enter <strong>PKCS12 Key Store</strong> in this field.</td>
</tr>
<tr>
<td>Key store password</td>
<td>Your Apple key store password.</td>
</tr>
<tr>
<td>Warn in days to expire</td>
<td>Number of days before certificate expiration to receive an expiration warning.</td>
</tr>
<tr>
<td>Short Description</td>
<td>Description for your certificate.</td>
</tr>
</tbody>
</table>

7. Click the attachment icon.


9. Click **Save**.

10. Optional: Re-open your X.509 Certificates record and click the **Validate Stores/Certificates** link at the bottom of the form to validate your certificates and stores.

11. On the push application form, click **Update**.

**Configure push applications for Android branded apps**

Enter your Google Firebase Cloud Messaging legacy server key in your push application record to use push notifications on your branded ServiceNow mobile apps for Google Android.

Role required: admin

To use push notifications on your branded ServiceNow mobile apps for Android, you must enter your Google Firebase Cloud Messaging legacy server key in your push application record.

**Note:** These steps are only necessary if you are using a custom branded mobile apps. For more detail on custom branding, see *Request and publish a branded mobile app*.

1. Navigate to **System Notification** > **Push** > **Push Application**.
2. Open the record labeled **ServiceNow Mobile Application**.
3. In the **Push** field, change the value from **REST API** to **Direct**.
4. In the **Google** tab, enter your Google Firebase Cloud Messaging legacy server key in the **API Key** field.
5. Click **Update**.

**Enable push notification categories in mobile settings**

Enable push notification categories so your users can enable or disable notifications by category.

Role required: admin

1. Type **sys_properties.list** in the Application Navigator.
2. In the system properties list, find and open the property with the name **com.glide.sg.notifications.management**.
3. In the **Value** field, enter `true`.
4. Click **Update**.

**Note:** If the property does not exist, click **New** and create a true/false property with the name in the previous step.

**Note:** After changing this property, users must log out and back in for the change to take effect.
Your user can now enable or disable notifications by category. The categories shown in the notification preferences screen are defined in the Notifications [sysevent_email_action] table. The screen to the left shows all the reference categories defined on this table. The screen to the right shows record matching the selected category.

After you have updated your system property, your users will be able to enable or disable notifications by the defined categories. For information on how users perform these tasks see Manage push notifications for mobile.

**Configure actionable push notifications**

Include actions with your push notifications. Users can perform push notification actions without opening the app.

Up to three actions can be associated with a push notification. These actions must refer to an existing mobile function. The following function types of actions are supported:

- Action item
- Navigation
- URL
- Chat launcher

**Creating actionable push notifications**

Create actionable push notifications using the following process:

**Create a push notification**

Create a push notification your users will see on their mobile devices.

**Add a push action category**
Select a push category to determine what actions your users can take in a notification.

**Create functions for each push action**
Create mobile function for each function in the selected action category. These actions perform tasks on your instance based on what the user selects in the notification.

**Map functions with push actions on the actionable push notification**
Associate functions to the actions in your notification so the instance uses the correct function for each action.

**Create push message content**
Create a record to determine what information the notification displays to your users.

**Create a standard notification**
Create a standard notification your instance using the platform notifications.

**Create an actionable push notification**
Create a push notification your users will see on their mobile devices.

Role required: admin

2. Click New.
3. In the push notification form, fill in the fields.

### Push notification form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of your push notification.</td>
</tr>
<tr>
<td></td>
<td><strong>Note</strong>: This name does not appear to users viewing the notification.</td>
</tr>
<tr>
<td>Application</td>
<td>The scoped application for your notification. This field automatically uses</td>
</tr>
<tr>
<td></td>
<td>the current application scope.</td>
</tr>
<tr>
<td>Actionable</td>
<td>Whether the push notification is actionable. Click to enable this field.</td>
</tr>
<tr>
<td>Screen</td>
<td>An optional screen associated with your notification. The notification uses</td>
</tr>
<tr>
<td></td>
<td>this screen. This screen displays to a user when they tap on the notification.</td>
</tr>
<tr>
<td>Category</td>
<td>Category of your notification. This field is visible only when the <strong>Actionable</strong> field is enabled. This category defines which actions are available in the push notification. You will fill in this field in later steps.</td>
</tr>
</tbody>
</table>

4. Click Save.
   Your notification record is created. After saving, the **Push Action Instances** related list appears on the form.

**Add a push action category**
Select a push category to determine what actions your users can take in a notification. This category defines which actions your users can take when viewing a notification.
Role required: admin

1. In the **Push Notification** record, select a category in the **Category** field. There are 12 base system categories to choose from on your instance.

### Push action categories

<table>
<thead>
<tr>
<th>Action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>accept-reject-back</td>
<td>Displays <strong>Accept</strong> and <strong>Reject</strong> buttons. Users can select an option without opening the app.</td>
</tr>
<tr>
<td>Accept-Reject-fore</td>
<td>Displays <strong>Accept</strong> and <strong>Reject</strong> buttons. The app opens when the user selects an action.</td>
</tr>
<tr>
<td>accept-rejectwcomments-back</td>
<td>Displays <strong>Accept</strong> and <strong>Reject</strong> buttons. Users see a text field to enter comments when they select the <strong>Reject</strong> option. Users can select an option without opening the app.</td>
</tr>
<tr>
<td>ack-escalate-ignore-back</td>
<td>Displays <strong>Acknowledge</strong>, <strong>Escalate</strong> and <strong>Ignore</strong> buttons. Users can select an option without opening the app.</td>
</tr>
<tr>
<td>approve-reject-back</td>
<td>Displays <strong>Approve</strong> and <strong>Reject</strong> buttons. Users can select an option without opening the app.</td>
</tr>
<tr>
<td>approve-rejectwcomments-back</td>
<td>Displays <strong>Approve</strong> and <strong>Reject</strong> buttons. Users see a text field to enter comments when they select the <strong>Reject</strong> option. Users can select an option without opening the app.</td>
</tr>
<tr>
<td>confirm-reschedule-cancel-back</td>
<td>Displays <strong>Confirm</strong>, <strong>Reschedule</strong> and <strong>Cancel</strong> buttons. Users can select an option without opening the app.</td>
</tr>
<tr>
<td>joinbridge-fore</td>
<td>Displays a <strong>Join Meeting</strong> button. Tapping this button opens the app.</td>
</tr>
<tr>
<td>PromoteWComments-RejectWComments-fore</td>
<td>Displays <strong>Promote</strong> and <strong>Reject</strong> buttons. Users see a text field to enter comments when they select either option.</td>
</tr>
<tr>
<td>queuerejoin-openincident-back</td>
<td>Displays <strong>Online Check-in</strong> and <strong>Open Incident</strong> buttons. Users can select an option without opening the app.</td>
</tr>
<tr>
<td>queuestay-queueleave-back</td>
<td>Displays <strong>Stay in Queue</strong> and <strong>Leave Queue</strong> buttons. Users can select an option without opening the app.</td>
</tr>
<tr>
<td>yes-no-back</td>
<td>Displays <strong>Yes</strong> and <strong>No</strong> buttons. Users can select an option without opening the app.</td>
</tr>
</tbody>
</table>

Your push notification displays buttons to your user in addition to the message content. In the next steps, you create functions that determine what happens when a users taps these actions.

2. Right-click the form header and click **Save**.

### Create functions for each push action

Create mobile function for each function in the selected action category. These actions perform tasks on your instance based on what the user selects in the notification.

Role required: admin

1. Navigate to **System Mobile > Functions**.
2. Click **New** to create a new function record.
3. Create a new function for one of your push actions. When selecting a function type in the **Type** field, you must select one of the type supported for actionable push notifications.
   - Action item
   - Navigation
   - URL
   - Chat launcher

   For information on function types, and detailed steps for creating functions, see *Mobile functions*.

4. Create additional functions for each push action.
   For example, if you have selected **ack-escalate-ignore-back** as your push notification category, your notification displays **Acknowledge**, **Escalate** and **Ignore** buttons. You need to create a function to determine the behavior of each of these buttons.

You have functions to perform actions for each of the buttons that display in your push notification. In the next steps, you will associate these functions to the notification actions so your functions trigger when users tap the buttons.

**Map functions with push actions on the actionable push notification**

Associate functions to the actions in your notification so the instance uses the correct function for each action.

Role required: admin

1. **System Mobile > Mobile Push Notifications > Push Notifications**, and open the notification you created in previous steps.

2. In the **Push Action Instances** related list, click New.

3. In the Push action instance form, fill in the fields.

   **Push action instance form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Push Notification</td>
<td>Push notification associated with this push action instance. This is automatically filled in with your push notification.</td>
</tr>
<tr>
<td>Application</td>
<td>Scoped application associated with this record. This is automatically filled in with the current application.</td>
</tr>
<tr>
<td>Push Action</td>
<td>Push action from your push notification record. Select one of the available actions.</td>
</tr>
<tr>
<td>Button</td>
<td>Function to associate with the push action. Select an action you created in previous steps. This action will trigger when a user uses the action listed in the <strong>Push Action</strong> field.</td>
</tr>
</tbody>
</table>

4. Click **Submit**.

   This example notification uses **accept-rejectwcomments-back** in the **Category** field. If you click the preview icon 
   ![preview_icon]
   you can see that this category uses two actions: **Accept** and **Reject**.
In the push action instance shown here, the Approve push action is selected in the Push Action field, and the Approve REQ /w comments function is selected in the Button field. The instance executes this function when a user taps the Approve button in their notification.
Create push message content

Create a record to determine what information the notification displays to your users.

Role required: admin

1. In your push notification record, click **Create Push Message Content** in the **Related Links** section.
   If you are not already in this record, Navigate to **System Mobile > Mobile Push Notifications > Push Notifications**, and open the notification you created in previous steps.

2. In the **Push Notification Message Content** form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of your push message content record. This name is not visible to your users.</td>
</tr>
<tr>
<td>Application</td>
<td>Scoped application associated with this record. This is automatically filled in with the current application.</td>
</tr>
<tr>
<td>Push app</td>
<td>Mobile app that uses your push notification.</td>
</tr>
<tr>
<td>Push Message Generation</td>
<td>Script that generates the push notification. When you create message content using these steps, this script is automatically generated.</td>
</tr>
</tbody>
</table>

3. Click **Submit**.

Create a standard notification

Create a standard notification on your instance using the platform notifications.

Role required: admin

1. Navigate to **System Notification > Push > Create Push Notification**.
2. In the notification form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of your Notification. This name is not visible to your users.</td>
</tr>
<tr>
<td>Active</td>
<td>Whether the notification is active. The instance does not send inactive notifications to your users.</td>
</tr>
<tr>
<td>Table</td>
<td>The table containing the records relating to your notification. For example, notifications about new incidents would use the Incident [incident] table.</td>
</tr>
<tr>
<td>Push Message Only</td>
<td>Whether the notification is a push message only. This field is read-only by default.</td>
</tr>
<tr>
<td>Category</td>
<td>The category for your notification. This field value is <strong>Uncategorized</strong> by default.</td>
</tr>
</tbody>
</table>

3. Right-click the form header and select **Save**.
4. Configure the push message for your standard notification.
   a) Click to display the **What to Send** tab.
b) Click the lock icon

next to the **Push Messages** field.

c) Click the reference icon

to display the **Push Notification Messages** list.

d) Click **New**.

e) In the push notification message form, fill in the fields.

**Push notification message form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of your push notification message record. This name is not visible to your users.</td>
</tr>
<tr>
<td>Application</td>
<td>Scoped application associated with this record. This field defaults to the current application.</td>
</tr>
<tr>
<td>Push App</td>
<td>Mobile app that uses your push notification.</td>
</tr>
<tr>
<td>Table</td>
<td>The table containing the records relating to your notification. For example, a notification about new incidents would use the Incident [incident] table.</td>
</tr>
<tr>
<td>Push Message Content</td>
<td>Select the record for the push message content you created in the previous section.</td>
</tr>
<tr>
<td>Message</td>
<td>Text the user sees in the push notification. For example, An incident has been assigned to you.</td>
</tr>
</tbody>
</table>

f) Click **Submit**.

5. In the **Notification** form, click **Update**.

**Siri shortcuts**

Configure Siri shortcuts to give your users quick access to common app functions on their iOS devices.

By default, your iOS users can use the following pre-configured shortcuts:

- Open a chat window
- Browser items and services
- Open my tasks
- Open my requests

**Configure a Siri shortcut**

Change the destination screen of your Siri shortcuts to direct your users to a different screen than the default.
Role required: admin

As an admin, you can change the destination screen of the three of default Siri shortcuts listed below. Creating new Siri shortcuts is currently not supported.

1. In the web-based UI, enter `sys_sg_screenShortcut.list` in the filter navigator.
2. Open the record for the shortcut you want to modify.

### Siri shortcuts

<table>
<thead>
<tr>
<th>Shortcut</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>com.servicenow.requestor.sc</code></td>
<td>Browse items and services</td>
</tr>
<tr>
<td><code>com.servicenow.requestor.tasks</code></td>
<td>Open my tasks</td>
</tr>
<tr>
<td><code>com.servicenow.requestor.request</code></td>
<td>Open my requests</td>
</tr>
</tbody>
</table>

**Note:** You can update the screen your user will navigate to when using a shortcut, however you cannot change the text of the shortcut.

**Note:** Open the record by clicking on the Shortcut field. Clicking on the Screen field will open the screen record rather than the shortcut record.

3. In the screen shortcut record, update the Screen field by clicking the reference icon and selecting an applet. This applet displays to your users when they access the associated Siri shortcut.

### Displaying campaigns on ServiceNow mobile

Use campaigns to deliver messages and important information to your users. You can set up and display a campaign on ServiceNow mobile.

Mobile campaigns enable you to share curated content using a scrolling list of images. You can mix any of three card types in the carousel: video, image, and text. At the bottom of the campaign cards, dots represent up to eight cards and numbers more than eight cards. A mobile campaign, with its carousel format, is most effective when displayed at the top of the mobile home page.

**Note:**

To set up a campaign to display on mobile devices, you must have the HR Service Content Delivery plugin [com.sn_content_delivery], installed. For more information, see Activate Content Delivery.

You create campaign content through Content Automation in HR Service Delivery. For more information, see Campaigns for HR Service Delivery and Configure mobile content.
Examples of a mobile campaign with an image, text, and video card.
Displaying mobile campaign process

Content for campaigns include videos, images (banners), and text cards.

You specify how to display the cards within a mobile campaign by first creating templates that provide information about the appearance of cards. Once you have defined the card appearances, you build and link components to use the data provided by the item views and display the content as a campaign on your mobile device.

This process involves multiple components contained in a content UI section:

- An item view for each type of content card, which serves as a template for the appearance of the three types of cards: video, image, and text.
- A master item for each type of content card, which contains the item view.
- An item stream that contains the data item, which collects the relevant content for the campaign, and the master item.
- An item stream container, which references the item stream.

Create appearance templates for cards for a mobile campaign

Create item view templates that contain information about the appearance of each type of card displayed in a mobile campaign.

Role required: admin

When creating cards, you need to assign an item view. Item views provide information about a card's appearance. Therefore, for each card style you want to use, you need a create a separate item view even if the card type is the same. For example, if you want some image cards to have light text on a dark background and other image cards to have dark text on a light background, you would need to create a separate item view for each of those styles.

1. In the web-based UI, enter `sys_sg_item_view.list` in the filter navigator to open a list of item views.
2. Click New.
3. On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the item view.</td>
</tr>
<tr>
<td>Short Description</td>
<td>Optional additional details.</td>
</tr>
<tr>
<td>Item view JSON</td>
<td>The configuration code you created for the card type. For more information, see Configure the details for the appearance of cards for a mobile campaign.</td>
</tr>
<tr>
<td>Table</td>
<td>sn_cd_content_mobile the HR Service Content Delivery table, which contains the content of the cards.</td>
</tr>
<tr>
<td>Dependency fields</td>
<td>Dependency fields taken from data entered in the Item view JSON field.</td>
</tr>
<tr>
<td>UI Styles</td>
<td>UI styles to apply to text cards. For information about how to configure the font color and background color for text cards, see Mobile UI styles.</td>
</tr>
</tbody>
</table>
4. Click Update.

After creating item views for each card style, you need to build and link components to display your mobile campaign. For more information, see Configure mobile campaign components.

Configure the details for the appearance of cards for a mobile campaign

Configure item views to provide the details that define the appearance of cards within a mobile campaign to enhance display options and make your campaign more effective.

When creating item views for cards for a mobile campaign, you need to construct required JSON code to define the appearance of the cards. You then provide this code in the Item view JSON field of the Item view form, as described in Create appearance templates for cards for a mobile campaign.

**Note:** If any of the configurable values are empty or the line of code is removed, the default value is used.

**Item view JSON for text cards**

For a text card, copy this sample JSON code and paste it in the Item view JSON field within an item view form.

```json
{
   "Type": "ViewGroup",
   "Distribution": "Equal",
   "Orientation": "Vertical",
   "Alignment": "Center",
   "Margin": {
      "Left": 0,
      "Right": 0,
      "Top": 0,
      "Bottom": 0
   },
   "Children": [
      {
         "Type": "Text",
         "CellId": "title",
         "TextAlignment": "center",
         "Margin": {
            "Left": 16,
            "Right": 16
         },
         "MaxLines": 3,
         "Font": {
            "Weight": "bold",
            "Size": 36,
            "MinSize": 24
         }
      }
   ]
}
```

The parameters in the sample JSON code are as follows:

- The **Type** parameter value is not configurable.
- The **CellId** parameter value in the sample code is based on the value in the default base system version of the HR Service Content Delivery table [sn_cd_content_mobile]. If the value in that table has changed, add the changed value in the sample JSON code.
Note: To access the table to check the values, enter sn_cd_content_mobile.list in the navigation filter of your application navigator.

- The other parameter values in this sample code for text cards are standard setup values that provide a balanced appearance for the cards. You can customize these values.

Item view JSON for video cards

For a video card, copy this sample JSON code and paste it in the Item view JSON field within the item view form.

```json
{
    "Type": "Media-Video",
    "CellId": "video_link.url",
    "TitleCellId": "title",
    "SubtitleCellId": "text"
}
```

The parameters in the sample JSON code are as follows:

- The **Type** parameter value is not configurable.
- The **CellId**, **TitleCellId**, and **SubtitleCellId** parameter values in the sample code are based on the values from the default base system version of the HR Service Content Delivery table [sn_cd_content_mobile]. If any of the values in that table have changed, add the changed value in the sample JSON code.

Note: To access the table to check the values, enter sn_cd_content_mobile.list in the navigation filter of your application navigator.

Item view JSON code for image cards

For an image card, copy this sample JSON code and set any values that differ from the default values. If you want to use the default values, you can delete that line of code. Once you have set your desired values, paste the JSON code in the Item view JSON field within the item view form.

```json
{
    "Type": "Media-Image",
    "CellId": "image",
    "TitleCellId": "title",
    "SubtitleCellId": "text",
    "Parallax": <true|false>,
    "DisplayType": "<over|under>",
    "DisplayTheme": "<light|dark>",
    "FocusOnFaces": <true|false>
}
```

The standard parameters in the sample JSON code are as follows:

- The **Type** parameter value is not configurable.
- The **CellId**, **TitleCellId**, and **SubtitleCellId** parameter values in the sample code are based on the values from the default base system version of the HR Service Content Delivery table [sn_cd_content_mobile]. If any of the values in that table have changed, add the changed value in the sample JSON code.
The custom parameters for the item view for image cards are as follows:

**Parallax**

Determines whether to create an illusion of depth and perspective on the image.

- **true** (the default): The text at the bottom of the card moves a slower speed than the background image, making the two objects appear as though they are on a different three-dimensional plane.
- **false**: The text at the bottom of the card and the image move at the same speed.

**DisplayType**

Determines whether text is displayed under the image (under) or as an overlay on the image (over). The text includes the title and the subtitle. For a smooth visual experience, use the same format for all the card types. The default value is **over**.

<table>
<thead>
<tr>
<th>Text displayed under image</th>
<th>Text displayed as overlay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Get The App!</td>
<td></td>
</tr>
<tr>
<td>With Now Mobile, employees can get stuff done</td>
<td></td>
</tr>
</tbody>
</table>

**DisplayTheme**

Determines the color of the text overlay. This option is available only if the display type is **over**. Use **light** for text that is to be displayed over a dark background. The default value is **light**.
FocusOnFaces

When set to `true` (the default), images on an image card are cropped to where the faces are captured in the frame.

Configure mobile campaign components

Configure components to display the image, video, and text cards in a campaign for your users to view and interact with on mobile devices.

To configure components to display a campaign on mobile devices, ensure that you have established the following items:

- Developed content for your campaign. For more information, see *Campaigns for HR Service Delivery* and *Configure mobile content*.
- Defined the appearance criteria for at least one card type. For more information, see *Create appearance templates for cards for a mobile campaign* and *Configure the details for the appearance of cards for a mobile campaign*.
- Have the HR Service Content Delivery plugin [com.sn_content_delivery] installed. For more information, see *Activate content delivery*.

Role required: admin

1. Create a data item to collect the data for the campaign content.
   a) In the web-based UI, enter `sys_sg_data_item.list` in the filter navigator to open a list of data items.
   b) Click New to create a data item.
   c) On the form, fill in the fields.

   **Data item form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A title for the data item. You can have multiple data items with the same name.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional additional details.</td>
</tr>
</tbody>
</table>
### Field Description

**Condition Type**
Select the method to use when building conditions to retrieve data from your data item. The available options are:

- **Declarative**
  Create conditions for the data item using the condition builder.

- **Scripted**
  Use a script to determine the conditions of your data item. A text window to enter a script appears below this field when you select this option.

**Table**
- sn_cd_content_mobile the HR Service Content Delivery table, which contains the content of the cards.

**Query Condition**
Condition that filters content data retrieved from the HR Service Content Delivery table. For example, if a field called month exists in the table, you could set the condition `[month][is] [September]` to indicate you want to pull content only when the month equals September.

**Group by**
Select the field used to group the records in your list.

---

d) Click **Submit**.

2. Create a master item for each item view you use.
   a) In the web-based UI, enter `sys_sg_master_item.list` in the filter navigator.
   b) Click **New** to create a master item.
   c) On the form, fill in the fields.

### Master item form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A title for the master item.</td>
</tr>
<tr>
<td><strong>Note:</strong></td>
<td>Include the card type in the name.</td>
</tr>
<tr>
<td>Item view</td>
<td>The item view to use for the card type.</td>
</tr>
<tr>
<td>Table</td>
<td>sn_cd_content_mobile the HR Service Content Delivery table, which contains the content of the cards.</td>
</tr>
</tbody>
</table>
3. Create an item stream, which connects the data item with the master item.
   a) In the web-based UI, enter \texttt{sys_sg\_item\_stream\_list} in the filter navigator.
   b) Click \textbf{New} to create an item stream.
   c) On the form, fill in the fields.

   \textbf{Item stream form}

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>A title for the item stream.</td>
</tr>
<tr>
<td>Data item</td>
<td>The data item you created.</td>
</tr>
</tbody>
</table>

   d) In the Item Stream M2M Master Item section, click \textbf{Insert a new row}.
   e) Insert the master item you created.
   f) Click \textbf{Save}.
   g) Click \textbf{Submit}.

4. Create an item stream container.
   a) In the web-based UI, enter \texttt{sys_sg\_item\_stream\_container\_list} in the filter navigator.
   b) Click \textbf{New}.
   c) In the \textbf{Name} field, enter a title for the item stream container.
   d) Right-click on the form header

   \includegraphics[width=0.5\textwidth]{form.png}

   and select \textbf{Save}.
   e) In the Item Stream Container M2M Item Streams section, click \textbf{New}.
   f) On the form, fill in the fields.

   \textbf{Item Stream Container M2M Item Stream form}

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item stream</td>
<td>The item stream that you want to add to the item stream container.</td>
</tr>
</tbody>
</table>
Field | Description
---|---
Item stream container | The item stream container that contains the item stream.
Order | A number that indicates where to place the item stream container. Lower-numbered items display before higher-numbered items.

5. Create an applet launcher content UI section to contain the campaign.
   a) In the web-based UI, enter `sys_sg_content_section.list` in the filter navigator.
   b) Click New.
   c) On the form, fill in the fields.

**Content section form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Title for the content UI section. This title is not displayed in the mobile UI.</td>
</tr>
<tr>
<td>Active</td>
<td>Option for enabling the display of the campaign in your mobile instance.</td>
</tr>
</tbody>
</table>

d) Click Submit.

Add the content UI section to an applet launcher. For more information, see *Configure content UI sections*.

**Additional mobile app configuration options**

Several system properties are available for you to further configure the mobile app. For example, use system properties to require a PIN, hide the image on the app homepage, configure the blur in background option, or disable sharing attachments from the mobile app.

**Disable the ServiceNow Classic mobile app**

Disable ServiceNow Classic on your instance to prevent users from using the legacy app and guide users toward the new ServiceNow mobile experience.
Disable ServiceNow Classic on your instance

Access to your instance using the ServiceNow Classic mobile app is controlled by the `glide.ui.m.allow_classic_mobile_app` system property.

This system property defaults to a value of `false` (blocked) on new instances, beginning with the Paris release.

For upgraded instances with existing ServiceNow Classic users, the property defaults to `true` (allowed). Your instance makes this determination based on the presence of records on the Mobile Devices [sys_mobile_devices] table. If there are records on this table, the instance sets the property value to `true` (allowed). With the property set to true, there will be no impact to your ServiceNow Classic users.

The `glide.ui.m.allow_classic_mobile_app` system property cannot be manually configured by an administrator. Contact ServiceNow Technical Support support to change this property from its default.
Configure the error message text for disabled ServiceNow Classic apps

Customize the message that appears on the ServiceNow Classic app when the Classic UI is disabled on your instance.

Role required: admin

Users accessing your instance with the ServiceNow Classic app will see a message if the classic UI is disabled. This message includes a title, a text area for a brief message, and a link.

Customize this information using records on the Messages [sys_ui_message] and System Properties [sys_properties] tables.

1. Navigate to System UI > Messages.
2. In the Messages list, the message record for the portion of the error message you want to change.

<table>
<thead>
<tr>
<th>Key</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Update required</td>
<td>The title that appears at the top of the message</td>
</tr>
<tr>
<td>A new version of the ServiceNow app is now available, visit the App Store to update.</td>
<td>The text that appears below the message title for iOS users.</td>
</tr>
<tr>
<td>A new version of the ServiceNow app is now available, visit the Play Store to update.</td>
<td>The text that appears below the message title for Android users.</td>
</tr>
<tr>
<td>Go to App Store</td>
<td>The text of the link that appears at the bottom of the message for iOS users.</td>
</tr>
<tr>
<td>Go to Play Store</td>
<td>The text of the link that appears at the bottom of the message for Android users</td>
</tr>
</tbody>
</table>

3. Open the record.
4. Update the text in the Message field. The value of this field is the text that will be displayed in the error message in place of the text that appears in the Key field.
5. Click Save.
6. Repeat steps 2 through 5 as needed, to modify your message.

Configure the error message link for disabled ServiceNow Classic apps

Customize the link that appears at the bottom of the ServiceNow Classic app, when the classic UI is disabled on your instance.
Role required: admin

By default, the link at the bottom of the message directs your users to the Mobile Agent app on the Apple App store or Google Play store, depending on their devices operating system. You can change the value of these system properties to direct users to an alternate URL.

1. Type `sys_properties.list` in the Application Navigator.
2. In the System Properties list, find and open the `glide.sg.agent_app_apple_store_url` system property.
3. In the Value field, replace the default value with the URL of your choice. This URL is used for users using iOS mobile devices.
4. Click Save.
5. In the System Properties list, find and open the `glide.sg.agent_app_play_store_url` system property.
6. In the Value field, replace the default value with the URL of your choice. This URL is used for users using Android mobile devices.
7. Click Save.

Set a default image for mobile users

To provide a default image to users whose records do not already have an image uploaded, configure the system property `glide.sg.image.default.sys_user`.

Role required: admin

If you do not upload images to user records, users in the mobile app receive an avatar by default. The file name for the avatar is `no_picture.jpg`.  

© 2020 ServiceNow, Inc. All rights reserved.
ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
To set a different default user image, upload an image and then enter its file name as the value for system property `glide.sg.image.default.sys_user`.

1. If you have not already uploaded the image that you intend to set as the default user image, navigate to **System UI > Images** and then upload your file to the Images (db_image) table.
2. Type `sys_properties.list` in the Application Navigator.
3. Open the record for `glide.sg.image.default.sys_user`.
4. In the form, match the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td><code>glide.sg.image.default.sys_user</code></td>
</tr>
</tbody>
</table>
Disable attachments in mobile apps

Disable attachments for mobile apps by using access control rules.

Role required: admin

Note: You need to elevate to the security_admin role to perform these steps. For details on this role, see security_admin role.

1. Navigate to System Security > Access Control (ACL).
2. Filter the list for <Name> <contains> <sys_attachemt> and <Operation> <is> <read>.
3. Find and open the record with the description: Allow read for records in sys_attachment, if the ACL script returns true.
4. Clear the Admin overrides check box.
5. In the Script field, add the following code to the bottom of the script:

```java
if( gs.isMobile() ){
    answer = false;
}
```
6. Click Update.
   The added code prevents attachments from appearing when the instance is accessed from a mobile device. If you want to prevent your users from uploading attachments, continue on to the next steps.
8. Filter the list for <Name> <contains> <sys_attachemt> and <Operation> <is> <create>.
9. Find and open the record with the description: Allow create for records in sys_attachment, if the ACL script returns true.
10. Clear the Admin overrides check box.
11. In the Script field, add the following code to the bottom of the script:

```java
if( gs.isMobile() ){
    answer = false;
}
```
12. Click Update.
   The added code prevents attachments uploading when the instance is accessed from a mobile device.

Note: The ability to rename or delete existing attachments on records in mobile is determined by the access control list (ACL) rules on your instance. For more information on these rule and how the effect user permissions on your instance, see Access control list rules.
Require an app PIN for the mobile app

Require uses to enter a PIN when the application has been inactive for five minutes. To require the mobile user to set and enter a local application PIN, add the system property `glide.sg.require_mobile_application_pin`.

Role required: admin

Users generate a six-digit code for the app PIN. The PIN must be entered when they log in to an instance from their mobile device, or after the application has been inactive for more than five minutes. If your users have faceID, touchID, or similar biometric security configured on their phone, they can use biometric authentication in place of the PIN.

1. Type `sys_properties.list` in the Application Navigator.
2. Open the record for `glide.sg.require_mobile_application_pin`.
3. In the form, match the following values:
### Configure the blur app option to improve security

As a security feature, administrators can configure the mobile app to blur when not in focus on a mobile device. When you double-click the home button on your mobile device to close apps or navigate back to where you left off, the ServiceNow app appears blurred.

1. Type `sys_properties.list` in the Application Navigator.
2. Open the record for `glide.sg.blur_ui_when_backgrounded`.
3. In the form, match the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>glide.sg.blur_ui_when_backgrounded</td>
</tr>
<tr>
<td>Type</td>
<td>true</td>
</tr>
<tr>
<td>Value</td>
<td>true</td>
</tr>
</tbody>
</table>

**Note:** The system property `glide.sg.blur_ui_when_backgrounded` is supported also in the ServiceNow Classic mobile app.

With the property in place, the app is blurred when not in focus.

---

**Note:** The mobile app is automatically locked after five minutes of inactivity. Users accessing the mobile app after a period of inactivity must enter their PIN code.
Configure the mobile app to clear the copy/paste clipboard and block ability to share content

To have the mobile app clear the pasteboard when the app enters the background, as well as block the ability to share content, add the system property `glide.sg.clear_pasteboard_when_backgrounded`.

Role required: admin
By default, content that you copy in the mobile app remains on your device's clipboard even after you close the mobile app. In addition, you can share copied content to an external app and to a non-native virtual agent. Use the system property `glide.sg.clear_pasteboard_when_backgrounded` to change this default behavior.

**Note:**
- Text edit menus on Android devices may display a share option, even though its functionality is disabled.
- Some third-party keyboards on Android devices are not disabled from the ability to paste and share content to the keyboard’s clipboard.
- The ability to copy/paste and share from some web views in the app may result in unexpected behavior.

1. Type `sys_properties.list` in the Application Navigator.
2. Open the record for `glide.sg.clear_pasteboard_when_backgrounded`.
3. In the form, match the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>glide.sg.clear_pasteboard_when_backgrounded</td>
</tr>
<tr>
<td>Type</td>
<td>true</td>
</tr>
<tr>
<td>Value</td>
<td>true</td>
</tr>
</tbody>
</table>

**Configure the maximum number of records returned for data items**

To set the maximum number of rows retrieved for the table defined in the data item, add the system property `glide.sg.data_item.row_count`.

Role required: admin

The system property `glide.sg.data_item.row_count` sets the maximum number of rows retrieved for the table defined in the data item. By default, the value is 1000. The system accepts no value greater than 1000.

1. Type `sys_properties.list` in the Application Navigator.
2. Click New, and then enter the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>glide.sg.data_item.row_count</td>
</tr>
<tr>
<td>Type</td>
<td>integer</td>
</tr>
<tr>
<td>Value</td>
<td>&lt;maximum-number-of-records-retrieved&gt;</td>
</tr>
</tbody>
</table>

**Configure pagination size for search lists**

To configure the amount of search list results that load to the screen as the user scrolls down, add the system property `glide.sg.list.pagination_size`.

Role required: admin

By default, the mobile app returns 50 results at a time in a search list. Increasing the pagination size may cause the search request to take longer.

1. Type `sys_properties.list` in the Application Navigator.
2. Open the record for `glide.sg.list.pagination_size`.
3. In the form, match the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td><code>glide.sg.list.pagination_size</code></td>
</tr>
<tr>
<td>Type</td>
<td>integer</td>
</tr>
<tr>
<td>Value</td>
<td><code>&lt;pagination-size&gt;</code></td>
</tr>
</tbody>
</table>

**Configure the maximum number of records returned for list UI parameters**

To configure the maximum number of records returned for a list of parameters, add the system property `glide.sg.list.max_items_number`.

Role required: admin

Default is 1000.

**Note:** The maximum number of rows returned for a parameters list is limited by the maximum number of rows returned for data items. In other words, the value for system property `glide.sg.list.max_items_number` cannot be greater than the value for system property `glide.sg.data_item.row_count`. For example, if you set the value of `glide.sg.list.max_items_number` to 50 but the value of `glide.sg.list.max_items_number` is 20, then you may only receive 20 records in your parameters list.

1. Type `sys_properties.list` in the Application Navigator.
2. Open the record for `glide.sg.list.max_items_number`.
3. In the form, match the following values:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td><code>glide.sg.list.max_items_number</code></td>
</tr>
<tr>
<td>Type</td>
<td>string</td>
</tr>
<tr>
<td>Value</td>
<td><code>&lt;maximum-number-of-parameters-items&gt;</code></td>
</tr>
</tbody>
</table>

**Force local login in mobile apps**

Configure the force local login option to provide local login experience on mobile apps even when the instance is configured with Single Sign On (SSO) configuration. You can configure this feature independently on any available ServiceNow® app.

Role required: admin

1. Navigate to **System Mobile > Mobile Publishing > Native Clients**.
2. Open the record for the mobile app where you want to force local logins.
3. Enable the **Force Local Login** field.
4. Click **Update**.

The selected mobile app routes your users to the local login authentication page. The app will default to the authentication method defined on your instance if this field is disabled.
Enable Virtual Agent for mobile applications

Give your users the ability to chat with a virtual agent through a ServiceNow mobile application.

Role required: admin

To include virtual agent functionality in your mobile applications, you first must activate and configure virtual agent. For details on this process see Implementing Virtual Agent

1. Navigate to System Applications > Studio.
2. In the application explorer on the left edge of the screen, select Functions > Actions, and click the pop-out icon that appears to the right of Actions.
3. Click the Create New button to create a new function.
4. In the Type field, select Chat Launcher.
5. In the Context field, select Global.
6. Click Submit.
7. On your instance, outside of Studio, navigate to System Mobile > Applet Launchers.
8. Open the record for the applet launcher where you want to add your Virtual Agent quick action.
9. In the Body section, click Insert a new row under the Quick Actions Menu Maps list.
10. Select the function created in the previous steps.
11. Click Update.

Configure a placeholder image for missing images in mobile apps

You can specify an image on your instance as a placeholder for missing images. This image appears in your mobile apps when a record has an image field with an empty value, such as a user avatar or catalog item. You can select a different image to use for each table on your instance.

Role required: admin

You configure placeholder images by creating properties on the System Properties [sys_properties] table. The property is table-specific. If you want to define placeholder images for many tables, you must create multiple properties. Multiple tables can use the same image as a placeholder.

1. Upload an image to your instance to use as a placeholder. For details on uploading images, see Storing images in the database
2. To open the system properties list, type sys_properties.list in the Application Navigator.
   Some tables already have a placeholder image defined. You can search the system properties table for properties that start with glide.sg.image.default to see any existing properties. Creating multiple properties for the same table can cause inconsistent results.
3. Click New.
4. Use the following information to complete the fields on the system property form.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter glide.sg.image.default.[tablename]. Replace [tablename] with the name of the table you want this property to apply to. For example to use the Catalog Item table, enter glide.sg.image.default.sc_cat_item.</td>
</tr>
<tr>
<td>Type</td>
<td>Select String</td>
</tr>
<tr>
<td>Value</td>
<td>Enter the name of your image. This value is the same as the Name field on the image [db_image] record.</td>
</tr>
</tbody>
</table>
5. Click Update.

Mobile authentication

Users are required to log in to an instance on their mobile device. Depending on how you configure authentication for mobile devices, users may be required to enter additional information.

For more information on configuring authentication for mobile devices, see Set up OAuth.

Follow the instructions for using a third-party OAuth provider.

Create a QR code for mobile login

Create and use a QR code containing JSON to provide a method for your users to log in with pre-defined parameters.

Role required: none

1. Use a text editor to create JSON using the following parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>InstanceUrl</td>
<td>The URL for your instance. For example, <a href="https://example.servicenow.com">https://example.servicenow.com</a></td>
</tr>
<tr>
<td>Name</td>
<td>The name of your instance. For example, Example</td>
</tr>
</tbody>
</table>

The JSON consists of parameters and values, separated by commas, and enclosed in curly braces. Each parameter is separated from its value by a colon, and each parameter and value is enclosed in double quotes, as shown in these examples. The image shows how the instance appears in instance selection screen for the app.

```json
{
  "InstanceUrl":"https://example.servicenow.com",
  "Name":"Example"
}
```

This second example includes only the InstanceUrl value, which is also valid.

```json
{
  "InstanceUrl":"https://example.servicenow.com"
}
```
Note: ServiceNow instances do not provide a method to create JSON files. You can create these files using your text editor of choice.

Note: Parameters are not supported for the Android OS. To use a QR code for Android, create a QR code containing only the URL for the instance as text rather than JSON. For example, https://example.servicenow.com.

2. Use a QR code generator of your choice to encode this JSON or text into a QR code.

Note: ServiceNow instances do not provide a method to create QR codes, however there are many online resources you can use to create QR codes using the JSON created in the previous steps.

3. Use your QR code to access the mobile app. For steps to use a QR code with the app, see Log in to an instance with a mobile app.

Configure mobile app token lifespan

Configure the length of time it takes for the app to time out.

Role required: admin

Mobile apps timeout when their associated OAuth token expires. Token lifespans are considered active if the app is in the foreground or if the app is processing a long running task in the background.

Configure the OAuth entity for a mobile app.

a) Navigate to System OAuth > Application Registry and open the OAuth entity for the application you want to modify. For example, open the ServiceNow Agent record to modify the timeout for your ServiceNow Agent app.

b) In the Refresh Token Lifespan field, change the number in seconds to the amount of time the refresh token is valid.

c) In the Access Token Lifespan field, change the number in seconds to the amount of time the access token is valid.

For example, if you want the mobile app's token to expire after 30 minutes, use the following configuration.

<table>
<thead>
<tr>
<th>Configuration point</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Refresh Token Lifespan for the ServiceNow Agent mobile app</td>
<td>1800 seconds</td>
</tr>
<tr>
<td>Access Token Lifespan for the ServiceNow Agent mobile app</td>
<td>1800 seconds</td>
</tr>
</tbody>
</table>

Sign out and sign back in to the mobile app. Otherwise the app uses the previously granted tokens.

Configurations for Now Mobile

Configure options for the Now Mobile app. For example, you can link the app with a service catalog and knowledge base, personalize the greeting for the home page, and specify which records appear under My Requests in the For Me tab.
Note: Offline mode is not supported in the Now Mobile app.

For Me

This page includes these configuration options:

Greeting
Configure the greeting that your users see when they log in to the Now Mobile app. For example, you can add a hello message that includes the user's first and last name.

My To-Dos
Users can view items that are assigned to them and complete their tasks. By default, My To-Dos show the user things that they need to approve from the Requests [sc_request] and Requested Items [sc_req_item] tables. Other applications, such as HR Service Delivery, might include other types of tasks. For more information about HR Service Delivery, see Now Mobile app for HR Service Delivery.
Configure default welcome message

2:59

Hello Beth Anglin

Search

Popular Services

Apple iPad 3
Apple iPad 3

Welcome Beth Anglin
How's your day going?

Popular Articles

How can I find the MAC address of my Ethernet or...

33 Views

Hi Beth Anglin
How can I help you?

Hello Beth Anglin
Ready to assist you
My Requests

Specify which records that you want your users to see under My Requests so that they can track their work assignments. For example, you can add a filter to display records that are opened by the user from the Problem table. By default, the app displays records that are opened by the user from the Incident and Requested Item tables.
Search

Search includes these configuration options:

People search

Configure whether users can search for other users in the system. By default, people search is enabled. For people search to display results, your users must have read-only access to the User [sys_user] table. You can test people search by logging in as a user without any roles and searching for another user. If search results do not include meaningful data, for example, locations and phone numbers, update the access control lists (ACLs) on the User table to allow read access. For more information, see *Access control list rules*.

Analytics and suggestions

The Now Mobile app collects search data and analytics that generate search suggestions. If you are upgrading from a previous release, the search analytics do not contain any data yet. To immediately provide suggestions to your users, you can populate the search suggestions using knowledge, catalog, and user search records from the Text Searches [text_search] table.

Search analytics and suggestions is a Now Platform feature. For more information, see *Search analytics and suggestions*.

Items and services

Service Catalog items and services include these configuration options.

Catalog

Enable your users to view and request their associated items in the Now Mobile app. If no catalogs are selected, users can view and request items from all catalogs in the system. By default, the app uses Service Catalog.

For more information, see *Now Mobile for Service Catalog*.

Quick actions

Select catalog items to display as additional user menu actions. For example, add the **Report an Incident** catalog item to enable users to quickly navigate to the form. Users can only see items if they have the required user criteria permissions.
Configure service catalogs

Facilities catalog
  - Maintenance and Repair
  - Office

IT catalog
  - Departmental Services
  - Hardware

Configure quick actions

Create links to critical items accessible throughout the app

- Report an Outage
- Report an Issue
- Chat
Knowledge

Enable users to view knowledge articles from the mobile app. If no knowledge bases are selected, users can view articles from all knowledge bases in the system. By default, the app uses the IT knowledge base.

For more information, see *Now Mobile for Knowledge Management*. 
Configure knowledge bases

App uses IT knowledge base by default

IT knowledge
- Applications
- Devices
- Email

Configure which knowledge bases to display

Facilities knowledge
- Maintenance and Repair

HR knowledge
- Benefits
- Hiring
**Siri shortcuts**

In the base system, iOS users can use Siri shortcuts to open these pages in the app:

- Open a chat window.
- Browse items and services.
- Open my tasks.
- Open my requests.

**Configure a personalized greeting**

Configure the greeting that your users see when they log in to the Now Mobile app. For example, you can add a hello message that includes the user’s first and last name.

Role required: admin

1. Navigate to **Now Mobile App > Applet Launchers**.
2. In the Applet Launchers [sys_sg_applet_launcher] table, open the Homepage record.
3. Add the **Header Titles** field to the form.
4. In the **Header Titles** field, open the record.
5. Update the **Text** field as desired.
   - This text displays in the Now Mobile app heading.
   - This field uses the `{{sys_id}}` syntax to reference a variable for the first and last name of the currently logged-in user. Changing the variable is not supported.
6. Click **Submit**.

When a user logs in to the Now Mobile app, they see your personalized greeting.

**Configure My Requests to track open records**

Specify which records that you want your users to see under **My Requests** so that they can track their work assignments. For example, you can add a filter to display records that are opened by the user from the Problem table. By default, the app displays records that are opened by the user from the Incident and Requested Item tables.

Role required: admin

1. Navigate to **Now Mobile App > My Request Filters**.
2. In the My Request Filters [request_filter] table, create a filter to display records from a specific table:
   a) Click **New**.
   b) On the form, fill in the fields.

**Request filter form**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Name for the filter.</td>
</tr>
<tr>
<td>Table</td>
<td>Table that extends the Task table. The app displays records from this table under <strong>My Requests</strong>.</td>
</tr>
<tr>
<td>Filter</td>
<td>Filter for the table. For example, you can set the filter to <code>[Opened by] [is (dynamic)] [Me]</code> to enable users to view their own requests.</td>
</tr>
<tr>
<td>Active</td>
<td>Option that you can select to make the filter active.</td>
</tr>
</tbody>
</table>
ServiceNow	Paris	ServiceNow Mobile

Field | Description
--- | ---
Applies to | Option that you can select to apply to Mobile apps.

- c) Click Submit.

Records from the defined filter are displayed in the app under My Requests in the For Me tab.

**Disable people search**

Configure whether users can search for other users in the system. By default, people search is enabled.

Role required: admin

1. Navigate to Now Mobile App > Applet Launchers.
2. Open the page where you want to disable people search.
3. In the Search Configuration field, select Homepage Search - Catalog and Knowledge.
4. Click Update.

Your users can no longer see search results for other users in the system.

**Configure catalogs**

Enable your users to view and request their associated items in the Now Mobile app. If no catalogs are selected, users can view and request items from all catalogs in the system. By default, the app uses Service Catalog.

Role required: admin

1. Navigate to Now Mobile App > Catalogs.
2. Optional: In the portal catalogs [m2m_sp_portal_catalog] table, add a catalog in addition to the base system catalog, or change the catalog that is associated with the base system record:
   - a) Click New or open the base system record.
   - b) On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Portal</td>
<td>Portal that you select. Select Mobile Employee Service Portal.</td>
</tr>
<tr>
<td>Catalog</td>
<td>Catalog that you would like to associate with the app.</td>
</tr>
<tr>
<td>Order</td>
<td>Number that indicates the order that the configurations should run. If there are multiple configurations on a portal, the system runs the configurations from the lowest to the highest order that you selected.</td>
</tr>
</tbody>
</table>

Your users can view and request items from all added catalogs.

**Configure knowledge bases**

Enable users to view knowledge articles from the mobile app. If no knowledge bases are selected, users can view articles from all knowledge bases in the system. By default, the app uses the IT knowledge base.
Role required: admin

1. Navigate to **Now Mobile App > Knowledge Bases**.
2. Optional: In the portal knowledge bases [m2m_sp_portal_knowledge_base] table, add a knowledge base in addition to the IT knowledge base.
   a) Click **New**.
   b) On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Active</td>
<td>Option that you can select to make the configuration active.</td>
</tr>
<tr>
<td>Portal</td>
<td>Portal that you select. Select <strong>Mobile Employee Service Portal</strong>.</td>
</tr>
<tr>
<td>Knowledge Base</td>
<td>Knowledge base that you would like to associate with the app.</td>
</tr>
<tr>
<td>Order</td>
<td>Number that indicates the order that the configurations should run. If there are multiple configurations on a portal, the system runs the configurations from the lowest to the highest order that you selected.</td>
</tr>
</tbody>
</table>

3. Optional: Change the default knowledge base that is associated with the Now Mobile app.
   a) In the **Portal** field, open the record with **Mobile Employee Service Portal**.
   b) In the **Knowledge Base** field, select the knowledge base that you would like to associate with the app.
   c) Click **Update**.

Users can select the icon on the applet launcher page to open the quick action.

**Configure Siri shortcuts for Now Mobile**

Configure Siri shortcuts for quick access to common features on your iOS mobile device.

Role required: admin

1. Log into a ServiceNow mobile app.
2. On the navigation bar, tap the **Settings**.
3. On the settings page, tap **Siri shortcuts**.
4. In the **Siri shortcuts** section tap the plus button

   ( )

   to configure a Siri shortcut.

Note: The currently available shortcuts are limited to **Open My Tasks**, **Browse Services**, and **Open My Requests**.
5. In the **When I say** field, add the phrase you want to say to Siri to open your app and use the selected shortcut.
Cancel

Add to Siri

Add a custom phrase Siri can use to tell Mobile to run this shortcut.

When I say:

Shortcut Name

Do:

now

Open My Requests
6. Tap Add to Siri.

Note: You can repeat this process to configure the remaining shortcuts, or change the phrase used to activate previously configured shortcuts.

Add a quick action in your mobile applications

Provide your users with a shortcut, which is also known as a quick action, to an item or action in your mobile apps. Quick actions appear on the applet launcher page.

Role required: admin

1. Navigate to Now Mobile App > Applet Launchers.
2. In the applet launchers [sys_sg_applet_launcher] table, open the applet launcher record that you would like to add the catalog item to.

For example, open the Homepage record to add the catalog item under the icon on the home page.
3. Select the Body tab.
4. Add a record under the Quick Actions Menu Maps related list.
   If you're not able to insert a row, make sure that you're in the ServiceNow Now Mobile App Screens and Applet Launcher application scope.
5. Add one of the quick actions from the list.

<table>
<thead>
<tr>
<th>Quick action</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Report an Issue</td>
<td>Opens the Create Incident catalog item.</td>
</tr>
<tr>
<td>Chat</td>
<td>Opens Agent Chat.</td>
</tr>
</tbody>
</table>

Your users can open the quick action by selecting the icon on the applet launcher page.

Create a quick action in your mobile applications

Provide easy access to an important item by creating a quick action. For example, you can create a quick action that opens a Service Catalog item.

Role required: admin

1. Create a function for the quick action.
   a) Navigate to System Mobile > Functions.
      The Function [sys_sg_button] table opens.
   b) Click New.
   c) On the form, fill in the fields.

Button fields form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the function record.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>Description</td>
<td>Description to enable other users to easily understand the purpose of the function.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of quick action that you want to create. To create a quick action that opens a Service Catalog item, select URL. For more information about function types, see Configure a smart button.</td>
</tr>
<tr>
<td>Context</td>
<td>Whether the function applies at the record or table level. • Record: Action only applies at the record level. For example, updating a field. • Global: Action that only applies at the table level. For example, creating or deleting a record. For this example, select Global.</td>
</tr>
<tr>
<td>Link Label</td>
<td>Label for the link. This value does not display in the user interface.</td>
</tr>
<tr>
<td>Link URL</td>
<td>Relative URL for the item that you want to open when the user clicks the function. For example, to open a Service Catalog item in the Now Mobile portal, enter /mesp?id=sc_cat_item&amp;sys_id=060f3afa3731300054b6a3549d6e5d3e. This field only displays if the Type field is URL.</td>
</tr>
<tr>
<td>Relative URL</td>
<td>Option that is selected to determine whether the URL is relative. For this example, choose Selected. This field only displays if the Type field is URL.</td>
</tr>
<tr>
<td>Condition</td>
<td>Table on which the condition runs.</td>
</tr>
<tr>
<td>Table</td>
<td>Table on which the condition runs.</td>
</tr>
<tr>
<td>Condition</td>
<td>Condition that should be met for the action to be successful. For a catalog item, enter var item = new sn_sc.CatItem('3f1dd0320a0b99000a53f7604a2ef9'); answer = item.canView() &amp; item.isVisibleServicePortal(); to display the quick action only when the end user has permissions to view the item.</td>
</tr>
<tr>
<td>Roles</td>
<td>Roles that you want to user to have to view the quick action.</td>
</tr>
</tbody>
</table>

2. Add an instance of the function that you created to the page.  
   a) In the native UI, navigate to **Now Mobile App > Applet Launchers**.  
   b) Open the applet launcher record that you would like to add the quick action to.  
   c) Select the **Body** tab.  
   d) Insert a new row in the Quick Actions Menu Maps related list. If you're not able to insert a row, make sure that you're in the **ServiceNow Now Mobile App Screens and Applet Launcher** application scope.  
   e) Click the magnifying glass to look up an item  
      The Function Instances [sys_sg_button_instance] table opens.  
   f) Click **New**.  
   g) On the form, fill in the fields.
Function instance form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the function instance.</td>
</tr>
<tr>
<td>Description</td>
<td>Description to enable other users to easily understand the purpose of the function instance.</td>
</tr>
<tr>
<td>Parent</td>
<td>Page on which you are adding the quick action. For example, select Applet Launcher: Homepage to add the quick action to the home page. Select a value in the Parent table field first.</td>
</tr>
<tr>
<td>Application</td>
<td>Application scope.</td>
</tr>
<tr>
<td>Parent table</td>
<td>Table that contains the record type where the button appears. For this example, select applet Launcher [sys_sg_applet_launcher] to add the quick action to an applet launcher page.</td>
</tr>
<tr>
<td>Function</td>
<td>Function record that you created earlier.</td>
</tr>
<tr>
<td>Label</td>
<td>Label to help the user understand what the quick action opens. For example, Report an outage.</td>
</tr>
<tr>
<td>Location</td>
<td>Location where the button appears in the UI. For this example, select Quick Action.</td>
</tr>
<tr>
<td>Icon</td>
<td>Icon to display next to the label. For more details on mobile icons, see Mobile icons.</td>
</tr>
</tbody>
</table>

**Note:** Not all listed icons work with quick actions. To see a list of compatible icons, filter your list where the Icons field contains now-mobile-icons-buttons.

**Note:** The Icons field is not on the icon list by default. To add the Icon field to your list, right-click the list header and select Configure > List Layout. Then, add the Icon field to the selected list. You should see the icon field on your reference field list.

| Order         | Number that indicates the order that the functions are listed. If there are multiple instances in the same location, the app displays them from the lowest to the highest. |

h) Click Submit.

i) Save the applet launcher record.

Your users can open the quick action by selecting the icon on the applet launcher page.

**Enable Live Agent in the Now Mobile app**

Activate the Chat quick action to allow your users to ask for help from a virtual or live agent from the Now Mobile home page.

Role required: admin

Ensure that the Agent Chat plugin is active. For more information, see Agent Chat.
Chat is enabled from the Requests and Knowledge Articles pages by default.

**Note:** Connect Chat and pre-chat conversation routing is not supported in the Now Mobile app.

1. Navigate to **Now Mobile App > Applet Launchers**.
2. In the Applet Launchers [sys_sg_applet_launcher] table, open the Homepage record and select the **Body** tab.
3. Under the Quick Actions Menu Maps related list, open the Chat record.
4. Select **Active** to activate the record.
5. Click **Update**.

Users can select Chat in the Homepage quick actions menu to ask for help from a live or virtual agent.

### Enable voice search

Enable your users to search for items, articles, and people using native speech recognition from an app on their mobile device.

**Role required:** admin

**Caution:** Voice search uses native speech recognition and relies on your operating system's cloud server to transcribe voice into text search. If you have data privacy concerns about search queries moving to the operating system cloud server, do not turn on voice search.

1. In the application navigator, enter **sys_properties.list**.
2. Click **New** to add a new system property.
3. On the form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td><code>glide.sg.voice_search.enabled</code></td>
</tr>
<tr>
<td>Type</td>
<td>Select **true</td>
</tr>
<tr>
<td>Value</td>
<td>Enter <strong>True</strong></td>
</tr>
</tbody>
</table>

4. Click **Submit**.

The global search bar in the Now Mobile app includes a microphone icon. Users must select this icon and allow the app to access speech recognition on their mobile device to use voice search.

### Populate search suggestions in the mobile apps

If you are upgrading from a previous release, run a script to populate search suggestions with data from a platform search table to provide search suggestions to your users. Alternatively, you can wait until users search for keywords instead of running this script.

**Role required:** admin

In new instances, the search suggestions are enabled by default. In upgraded instances, you must enable the search suggestions. For more information, see [Enable search suggestions](#).

The Now Platform collects search data and analytics that generate search suggestions. If you are upgrading from a previous release, the search analytics do not contain any data yet. To immediately provide suggestions to your users, you can populate the search suggestions using knowledge, catalog, and user search records from the Text Searches [text_search] table.
Search suggestions improve over time as more people use the app. Search analytics and suggestions is a Now Platform feature. For more information, see Search analytics and suggestions.

**Caution:** Populating search suggestions can be a resource-intensive task that may take a while to complete. Do not run this script during peak hours. Populating search suggestions is not supported on domain-separated instances.

1. Navigate to **System Definition > Scheduled Jobs**.
2. Open the **Populate Suggestions to avoid Cold Start - NowMobile App** scheduled job.
3. Activate the record and select **Execute Now**.
   
   Running this scheduled job populates the Search Events \([sys_search_event]\), Search Source Events \([sys_search_source_event]\), and Search Suggestions \([sys_search_suggestion]\) tables with records from the Text Searches \([text_search]\) table.

Users see suggestions when they start typing in a search field in the Now Mobile app.

**User Experience Analytics**

User Experience Analytics helps you understand usage and adoption of your mobile applications. User Experience Analytics was formally called Mobile Analytics.

The ServiceNow® User Experience Analytics application provides dashboard views for monitoring key performance indicators of users of your mobile applications built with the ServiceNow mobile platform. Visualize metrics and interactions to better understand the user experience, and create more intuitive journeys for your users.
Mobile location tracking configuration

As an administrator, you can activate geolocation tracking for your ServiceNow® ServiceNow Agent app users by installing the Geolocation plugin (com.snc.geolocation). Complete other geolocation tracking tasks to finalize the configuration.

Activate the Geolocation plugin

Enable geolocation features on your instance by activating the Geolocation plugin (com.snc.geolocation). For details on plugin activation, see Activate a plugin.

Enable location tracking on your user records

After activating the geolocation plugin, you must enable location tracking for users who need to use the geolocation feature. Enable geolocation records for a user by opening a User [sys_user] record, and enabling the Geolocation Tracked field.

![User record screenshot](image)
Users must enable location tracking on their mobile devices

After enabling geolocation tracking on user records, your user will see geolocation tracking settings in the Settings tab of their mobile devices. Your users can enable or disable location tracking on their device, and set a duration to enable location tracking. For details on these settings, see Location tracking for mobile.

Offline mode

Offline mode allows users who have no internet connection to continue working from a mobile device. Configure specific application, applets, or functions for users to use offline in the ServiceNow Agent app. Watch this three-minute video to learn how offline mode works, how to download data, enable and disable offline mode, synchronize your outbox, and resolve synchronization errors.

Request Offline Mode

To activate this feature, request activation of the SG Offline support plugin (com.glide.sg.offline).

Note: Offline mode is supported on on-premise instances.

Role required: admin

1. Navigate to System Applications > All Available Applications > All.  
2. On the All Applications page, click Request Plugin to open the request form on HI.
3. On HI, select to be redirected to the HI Service Portal Service Catalog.
4. On the Activate Plugin request form, fill in the fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target Instance</td>
<td>Instance on which to activate the plugin.</td>
</tr>
<tr>
<td>Plugin Name</td>
<td>Name of the plugin to activate.</td>
</tr>
<tr>
<td>Specify the date and time you would like this plugin to be enabled</td>
<td>The date and time must be at least two business days from the current time.</td>
</tr>
</tbody>
</table>

**Note:** Plugins are activated in two batches each business day in the Pacific time zone, once in the morning and once in the evening. If the plugin must be activated at a specific time, enter the request in the Reason/Comments field.

| Reason/Comments | Information that would be helpful for the ServiceNow personnel who are activating the plugin. For example, if you need the plugin activated at a specific time instead of during one of the default activation windows, specify it in the comments. |

5. Click Submit.

**Configure Offline Mode behavior**

Configure system properties to customize the Offline Mode behavior of the mobile application.

Install or request installation of the SG Offline support plugin (com.glide.sg.offline).

Role required: admin

1. In the application navigator, enter: sys_properties.list.
2. In the Name search box, enter the name of a property from the table, and press Enter to display the property record.
3. Per the descriptions in this table, set values to the property record according to how you would like offline mode configured.
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.sg.offline.attachment.max_size</td>
<td>The maximum size, in bytes, for attachments that are cached while in offline mode. If no download, a placeholder displays. The default size is 50 MB.</td>
</tr>
<tr>
<td>glide.sg.offline.expiration</td>
<td>The length of time before cached data is expired. The default length is 48 hours. After 48 hours, the system deletes the data due to security protocol.</td>
</tr>
<tr>
<td>glide.sg.offline.attachment.allowed_content_types</td>
<td>A comma-separated list of allowed file types for attachments in offline mode. The default list includes these types: image/png, image/jpg, image/gif, video/quicktime. The default is empty. If empty, offline mode does not download any files.</td>
</tr>
<tr>
<td>glide.sg.offline.roles</td>
<td>A comma-separated list of role names that are allowed to work in offline mode. If empty, all users may use offline mode.</td>
</tr>
<tr>
<td>glide.sg.offline.enabled</td>
<td>Enable offline capabilities on your instance. Offline mode is enabled by default. To disable offline mode, create a system property with this name and set the value to false.</td>
</tr>
<tr>
<td>glide.sg.offline.scheduled_download.enabled</td>
<td>Enable scheduled downloading for all Agent app users</td>
</tr>
<tr>
<td>glide.sg.offline.scheduled_download.reminder_offset</td>
<td>Number of minutes before a user's schedule that they receive a manual download reminder. If no there is no cache or the cache is expired, offline mode sends a reminder for the user to manually download a cache. Offline mode sends this reminder once a day before the start of the users schedule.</td>
</tr>
</tbody>
</table>
| glide.sg.offline.scheduled_download.cachesync_offset | Number of minute before a user's schedule to begin payload generation. The property value is in minutes.  

**Note:** Be sure not to set the value of this property to a value that was less than the interval that the Mobile Offline Scheduling job is running. Doing so will prevent offline payloads from being generated.  

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| glide.sg.offline_payload.refresh_frequency   | The number of minutes before a cached record is flagged for refresh. The default is 240 minutes. The property value is in minutes.  

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.sg.offline.attachment.max_total_bytes</td>
<td>Maximum space allocated for all attachments in cache. Default value is 400 MB, max is 2 GB. The property value is in MB.</td>
</tr>
</tbody>
</table>

4. Repeat steps 2 and 3 for the remaining properties in the table.

For a view of the offline mode end-user experience, see: *Working offline in mobile.*

**Configure Offline Mode for applications or applets**

Configure applications and applets to be available for users that are not connected to the internet.

Because applications contain applets, if you set an applet to work offline, you must also set the application to work offline.
Role required: admin

1. In Studio, navigate to the application you want to configure Offline Mode for, then click the pop-out icon that appears when you point to the application name.
2. Optional: Open an applet you want to allow users to use offline.
3. In the header of the application or the applet, click Properties.

4. Use the Available offline toggle switch to determine if the application or applet is available offline.

Determine which functions are available to users in Offline Mode.
Configure Offline properties for action functions

Determine which fields and functions are available to users accessing the app in Offline Mode

Role required: admin

1. In Studio, navigate to Mobile Studio > Functions > Actions.
2. Click the pop out icon to open the Actions list in a tab.
3. Open an action you want to configure for Offline Mode.
4. Select the Offline toggle.

5. Use the Hide field and Show field lists to determine which fields are available after the user performs the action in offline mode. For example, after a user assigns a task to themselves, you could hide the Assigned to field or show the Work notes field.

6. Use the Hide functions and Show functions lists to determine which fields are available after the user performs the action in offline mode. For example, when a user taps the Start Work function in offline mode, that action function is hidden and the Close Complete and Close Incomplete functions display instead.

7. From the Mark as zombie on screens field, search for screens to gray out when a user performs an action. In Offline Mode, an indicator appears next to a record after a user makes a change. Setting a screen to be marked as a zombie allows the list to mimic the online behavior where a record is grayed out in the list after a user makes a change.
INC0000039
5 - Planning
Trouble getting to Oregon mail server
Assigned to
State New

INC0000057
5 - Planning
Performance problems with wifi
Assigned to Beth Anglin
State In Progress

INC0010003
1 - Critical
Need access to sales DB for the West
Assigned to
State New

INC0000059
3 - Moderate
Unable to access team file share
Assigned to
State New

Online behavior for "zombie mode"
8. Click **Update** to save the offline properties configuration.

You can also configure Offline Mode for *navigation* and *smart button* functions.

**Scheduled offline caching**

Configure offline caching so your field technicians can receive scheduled updates to their offline data cache. Scheduled downloads are based on the user's work schedule.

**Activate the Agent Schedule plugin**

To enable and configure scheduled offline caching, the Agent Schedule plugin [com.snc.agent_schedule] must be activated. For details on plugin activation, see *Activate a plugin*.

**Create work schedules for offline caching**

After activating the plugin, you must create work schedules for the agents or technicians to enable users to automatically receive scheduled offline caches. This can be done directly through the Agent Work Schedules [agent_work_schedule] table. For more information on how to create schedules through this table, see *Create a work schedule for an agent or technician*.

Based on the records from the Agent Work Schedules table, your instance runs background scheduled jobs that create schedules in the Agent Daily Schedules [agent_daily_schedule] table. Within these schedules, agents will receive a silent push sometime throughout their scheduled day.

The offline payloads that your instance generates are based on the times recorded in the Agent Daily Schedules. These can be found in the Events [sysevent] table. Use the records in the Event table to track these payloads, and info about when they are created and when they are sent to the agents.

**Scheduled Jobs associated with offline caching**

These scheduled jobs are automatically scheduled for only users who enable background downloading on their app. For information on how users can enable this feature, see *Offline mode for mobile*.

**Populate Agents Daily Schedule Table**

This job runs once daily for all users with background downloading set to **true**.

**Scheduled Download of Offline Payload**

This job creates an event for the first payload of the day in the [mobile_offline_payload_gen_queue] table.

**Offline scheduling system properties**

Use the following properties on the System Properties [sys_properties] table to configure scheduled offline caching.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>glide.sg.offline.scheduled_download.enabled</td>
<td>Enables scheduled downloading for all Mobile Agent app users.</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>glide.sg.offline.scheduled_download.cachesync_offset</td>
<td>Determines when your instance starts payload generation prior to schedule. Value determines how many minutes before the user schedule payload generation begins.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> Be sure not to set the value of this property earlier than your Mobile Offline Scheduling job is runs. Doing so will prevent offline payloads from being generated.</td>
</tr>
<tr>
<td>glide.sg.offline.scheduled_download.reminder_offset</td>
<td>Determines when your instance sends your users a daily reminder to manually download a cache. The instance sends this reminder only when there is no cache or the cache is expired. Value determines how many minutes before a user schedule your instance sends this reminder.</td>
</tr>
<tr>
<td>glide.sg.offline_payload.refresh_frequency</td>
<td>Determines <code>refreshTimestamp</code> frequency of payloads. Value represents this frequency in minutes. The default value is 480.</td>
</tr>
<tr>
<td>glide.sg.offline.attachment.max_total_bytes</td>
<td>Determines the maximum size allocated for all attachments in a cache. Value is represented in Megabytes (MB). The maximum value is 2048 MB.</td>
</tr>
</tbody>
</table>

For information on more offline related system properties, see, *Configure Offline Mode behavior*.

**Mobile bar-code scanning**

Use the camera on your mobile device to scan bar-codes into your instance using mobile applications.
Multi-scan bar-code scanning

Use multi-scan to scan multiple bar-codes sequentially without leaving the scanning interface. Input information for your assets faster than manual entry or individual scans. After scanning in multiple bar-codes, users can review a list of their scanned items.
Grouped bar-code scanning

Use grouped bar-code scanning to scan multiple bar-codes and group them into a set. For example, a single item that has an asset number, serial number, and model number. You can configure a button to use grouped inputs to accept multiple bar-code scans.
Supported barcode types

The ServiceNow mobile apps for iOS and Android support scanning the following barcode types:

- 2D barcodes: QR Code, Data Matrix, PDF-417, AZTEC

Zebra scanning devices

The Android version of ServiceNow mobile apps can be used with most Zebra scanning devices. The scanner works only when the Zebra device camera is off. In the app, when the cursor is in a QR/Barcode input field, the scanner outputs the results. Users can modify the barcode output to include a return/enter string at the end of each scan with other apps, such as DataWedge.

Multi-scan for mobile applications

Use multi-scan to quickly scan multiple barcodes or groups of codes. With multiscan, you can input information for your assets faster than manual entry or individual scans.

Use the scan screen to scan items into your instance by using your mobile device.
Scan screen

Use the scan screen to scan in your barcode items. You see this screen when you access a function that your administrator configured for barcode scanning.

Mobile scan screen features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Close (1)</td>
<td>Tap Close to leave the scan screen.</td>
</tr>
<tr>
<td>Disable camera (2)</td>
<td>Tap Disable camera to disable the camera on your mobile device. This button is available only on Android. Disable your camera to scan using a non-phone scanner.</td>
</tr>
<tr>
<td>Enable flashlight (3)</td>
<td>Tap Enable flashlight to enable the flashlight feature on your mobile device.</td>
</tr>
<tr>
<td>Scan crosshair (4)</td>
<td>Center a barcode to be scanned using the crosshair icon in the center of the screen.</td>
</tr>
<tr>
<td>Last Scan (5)</td>
<td>See the last three entries that you scanned. Press the Rescan button to discard the scanned values.</td>
</tr>
<tr>
<td>Asset tag (6)</td>
<td>See the UI parameters that are used in the scanned input. Scanning a barcode adds a value to the top parameter. Further scans add values to the next parameters in the display order. You can tap any entry to manually enter a value.</td>
</tr>
<tr>
<td>Scan next item (7)</td>
<td>Tap Scan next item to accept the current scanned values and continue to a new item. If all fields in the Inputs (6) section have values, the scan screen automatically continues to the next item. If there are any mandatory fields...</td>
</tr>
</tbody>
</table>
Review screen

After scanning your items, use the review screen to review, delete, and submit scanned items.

Mobile review screen features

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Submit (1)</td>
<td>Tap Submit to save your scanned items.</td>
</tr>
<tr>
<td>Scanned items (2)</td>
<td>See your scanned items in this section. Each item displays the UI parameter name and its scanned value.</td>
</tr>
<tr>
<td>Delete item (3)</td>
<td>Tap the delete icon next to a scanned item to remove that item.</td>
</tr>
</tbody>
</table>

Add Asset

Count: 2

Asset tag 0000002
Serial number [empty]

Asset tag 0000002
Serial number 0000002
Configure a grouped input for multiple scans

Enable your users to scan several barcodes sequentially. Users can scan barcodes without leaving the scanning interface when you configure a grouped input. You can configure grouped inputs on buttons to accept multiple barcode scans.

Role required: admin

1. In the application navigator, enter `sys_sg_button` to open a list of mobile functions.
2. Open the record for the button where you want to include multi-scan functionality.
3. Click the Grouped Inputs tab on the form to display the grouped inputs list.
4. Click New.
5. On the form, fill in the fields.

Grouped input form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of your grouped input. Because this name is used in scripts, do not include spaces in names.</td>
</tr>
<tr>
<td>Label</td>
<td>Label to describe your grouped input. You may use spaces in your label.</td>
</tr>
<tr>
<td>Source</td>
<td>Source for the grouped input. Select Button.</td>
</tr>
<tr>
<td>Button</td>
<td>Button used for the grouped input. The Button field appears after you select Button in the Source field.</td>
</tr>
<tr>
<td>Order</td>
<td>Order value for your grouped input.</td>
</tr>
<tr>
<td>Mandatory</td>
<td>Field that you enable to require that at least one value is entered for this input.</td>
</tr>
<tr>
<td>Multiple entries</td>
<td>Ability to scan multiple entries.</td>
</tr>
<tr>
<td>Max entries</td>
<td>Maximum number of entries for this input. Enter 0 for an unlimited number of entries. This field is visible only when the Multiple entries field is enabled.</td>
</tr>
</tbody>
</table>

Create UI parameters for your grouped input

Create UI parameters to contain the values that your users scan in with your grouped input.

Role required: admin

1. In the UI Parameters field, click the lookup icon

   ![Lookup icon](image)

   to display the UI parameters list.

2. Click New.
3. On the form, fill in the fields.

UI parameter form

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Descriptive name of your UI parameter</td>
</tr>
</tbody>
</table>
ServiceNow
Paris
ServiceNow Mobile

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter type</td>
<td>Type of parameter. Select None</td>
</tr>
<tr>
<td>Input type</td>
<td>Type of entry for the parameter value. Select QR/Barcode</td>
</tr>
</tbody>
</table>

4. Click Submit.
5. Optional: Repeat steps 1 through 4 to add additional parameters. For example, if you are creating a record to track hardware, you may include UI parameters for a serial number and a model number.
6. Click Submit.

**Configure your action item to use your grouped input UI parameters**

Configure the action item on your button so that you can use the values that your users scan in.

Role required: admin

1. From your grouped input record, click the reference icon next to the **Button** field, and then click **Open record** to open your function record.

2. From your function record, click the reference icon next to the **Action item** field, and then click **Open record** to open your action item record.

3. Modify the **Execution Script** field to use your grouped input UI parameters.

These parameters are contained within the **input** object. You can access the parameters by using this format: `input.grouped_input_name[input index]"UI Parameter Name"`. For example, if you have a grouped input called `add_asset` that has a UI parameter named `Serial number`, you would use `input.add_asset[0]"Serial number"` to access the first serial number asset.

In this example, the script checks the length of `input.add_asset` to see how many items were scanned. It then adds an asset for each scanned item using the scanned-in serial number and asset tag.

```
/* Execution Script */

(function WriteBackAction(input) {
  for (i=0; i< input.add_asset.length; i++) {
    var assetString = input.add_asset[i]"Asset tag";
    var serialNumber = input.add_asset[i]"Serial number";
    var SMAAssetUsage = new global.SMAAssetUsage;
    SMAAssetUsage.addAssetToStockroom(assetString, serialNumber);
  }
})(input);
```
Note: The action item type must be set to Script to display this Execution Script field.

Your button is configured for multi-scan, and your users can begin creating records. For information on using the mobile scanning interface, see Multi-scan for mobile applications.

Localization on mobile devices

ServiceNow mobile apps are localized in 20 languages.

The native mobile applications are localized in two different ways, which means that there is a blending of localization visible inside native applications.
Native localization

Native (on device) localization: Controlled by the device's language preference, which means that many components are localized with the language preference for the user's device. These components can include local screen titles (such as Settings) and local button titles (such as the Clear All button on the filters screen).

The on-device localization supports the following languages:

- I18N: Brazilian Portuguese Translations
- I18N: Czech Translations
- I18N: Dutch Translations
- I18N: Finnish Translations
- I18N: French Canada Translations
- I18N: French Translations
- I18N: German Translations
- I18N: Hungarian Translations
- I18N: Italian Translations
- I18N: Japanese Translations
- I18N: Korean Translations
- I18N: Norwegian Translations
- I18N: Polish Translations
- I18N: Portuguese Translations
- I18N: Russian Translations
- I18N: Simplified Chinese Translations
- I18N: Spanish Translations
- I18N: Swedish Translations
- I18N: Thai Translations
- I18N: Traditional Chinese Translations
- I18N: Turkish Translations

These translations can't be customized since they ship with the application binary and translated natively on the device according to the language settings for the device.

Server-side localization

Server-side localization: Controlled the same way as desktop web localization (server system language / user preference on server). Localized components on the server include things like field labels, web content, and other data stored on the server in a translated field.

You can customize translations that use server-side localization the same way you do on the desktop. Translated these elements using the translated name/field table on your instance. For more details on this table, see Translated Name / Field table

Note: Right-to-left languages are not supported in the native mobile app.

This series of screenshots shows a mobile app on a mobile device with Spanish set as its native language. The device is connected to a ServiceNow instance where the instance language is set to English. The user's mobile device translates the Spanish portions the text according to the device's user preferences. These elements cannot be changed with your instance localization settings.

The English portions are not translated by the mobile device. The English text can be translated using your instances localization settings. As a admin you have control over how these elements are translated.
Native Device Language: Spanish
Server Language: English
Domain separation in mobile

This is an overview of domain separation in the mobile application. Domain separation enables you to separate data, processes, and administrative tasks into logical groupings called domains. You can then control several aspects of this separation, including which users can see and access data.

Support level: Basic

- Business logic: Ensure that data goes into the proper domain for the application’s service provider use cases.
- The application supports domain separation at run time. This includes domain separation from the user interface, cache keys, reporting, rollups, and aggregations.
- The owner of the instance must set up the application to function across multiple tenants.

Use case: When a service provider (SP) uses chat to respond to a tenant-customer’s message, the client must be able to see the SP's response.

Overview

The mobile platform supports domain separation for all native clients. The mobile UI design clearly indicates the domain that a record is associated with.

Before extending the domain separation functionality to mobile, the feature must be enabled on the platform web-based interface. For further information on configuration, see Domain separation setup and administration.

How domain separation works in mobile

You can use the company or account fields to display appropriate records by domain. These fields are available in tables when the domain separation plugin is enabled. Because each company or account is linked to a single domain, when you create a record, you can use these fields to specify the domain in which you want to create the record.

After the feature has been set up through the platform web-based interface, a two-part process is required to further extend the domain separation functionality on mobile. For additional information about configuring domain separation on mobile devices, contact ServiceNow Technical Support.

Note: ITSM Mobile Agent contains an automatically configured domain separation feature. For more information, refer to the ITSM mobile agent documentation.

ServiceNow Classic mobile app

Use the ServiceNow Classic mobile app to access records, update information, and collaborate with other users.

Any differences between the two platforms are designed to make the app accessible to a user on either device. For example, the location of the Navigation Menu varies between the two platforms to stay consistent with the platform-specific UI.

Note: If your company uses multi factor authentication (MFA), when you sign in on the classic mobile app, append the MFA code to your password.
Not supported

- Custom app configuration parameters
- Custom buttons
- Deep linking to the mobile app
- Formatters
- Form Templates
- Internal distribution
- Knowledge v2
- Connect Support
- UI Scripts
- UI macro variables in the mobile Service Catalog

Limited support

- List filtering: Several of the following fields have limited support on mobile devices. You can still create a complex filter in a desktop instance and open it in the mobile app. However, you will not be able to edit any of the limited fields.
  - between
  - dates
  - tags
  - related fields
- Visualizations are specific to a device. If you set up a visualization in the app on your iPhone, then view the app on an iPad, the visualization may not be the same.

Migrate to ServiceNow mobile

The ServiceNow Classic app no longer receives enhancements or non-priority bug fixes. Migrate to ServiceNow® mobile to take advantage of features such as rapid development, offline capability, and integration with native mobile device features. For more detail on the migration process, see Migrate from Classic mobile to ServiceNow mobile

Device security for ServiceNow Classic

This document applies to the ServiceNow Classic app for iOS and Android. This document may be subject to change for future mobile releases and re-platforming efforts.

Components and architecture

The ServiceNow solution consists of the ServiceNow server instance and the iOS and Android hybrid apps. A hybrid app includes both native and web components. The mobile client applications communicate over a wireless connection with the server and pull live data for the end user.

Component explanations

App for iOS

The ServiceNow app for iOS is a hybrid application that can be used on iPhone, iPad, and Apple Watch. Most components are native, however, there are web components, such as forms. It can be downloaded from the app
store directly by a user, or can be pulled, dynamically configured, and distributed using MDM (more information available in EMM section). ServiceNow does not currently distribute the ipa file to customers.

**App for Android**

The ServiceNow App for Android is a hybrid application that can be used on Android devices. Most components are native, however, there are web components, such as forms. It can be downloaded from the app store directly by a user, or can be pulled, dynamically configured, and distributed using MDM (more information available in EMM section). ServiceNow does not currently distribute the apk file to customers.

**Identity and access management**

Control user access with user authentication, session timeout, and termination.

**User authentication for ServiceNow Classic**

The mobile app supports platform authentication using OAuth 2.0. Authentication mechanisms include multi provider SSO, MFA, LDAP, Local DB, and Digest.

**Multi Provider SSO**

The mobile app uses federated login when using the multi provider SSO plugin [com.snc.integration.sso.multi.installer]. For more information on configuring multi provider SSO, see *Single sign on for the ServiceNow Classic mobile app*.

**Multifactor authentication**

Users can access the instance via Multifactor Authentication using the MFA plugin [com.snc.integration.multifactor.authentication]. For more information on MFA configuration, see *mobile multifactor authentication*.

**LDAP**

Use LDAP authentication to access using LDAP credentials. For more information on LDAP configuration, see *LDAP integration and authentication*.

**Local DB**

The user name and password in the user record in the instance database.

**Digest**

The digest token authentication passes user credentials and a digest token within an unencrypted HTTP header. For more information on digest configuration, see *Digest token authentication*.

**Not supported**

- SAML 2.0 plugin, however SAML with the Multi-provider SSO plugin is supported.
- Kerberos
- Certificate-based authentication
Storage/Keychain
When you sign in to the app on your mobile device, the app uses your credentials to negotiate an OAuth Token with the instance. The iOS Keychain stores the token and Android uses KeyStore. The keychain encryption is AES 256 in Galois/Counter Mode (GCM).

The mobile app never stores the user password.

Session length and timeout
The session length and timeout is configurable by the organization. For more information on configuring session timeout, see mobile session app timeout.

User termination
When an administrator deletes or removes a user from the system, they are logged out of the mobile client.

Mobile data flow for ServiceNow Classic
Data can be retrieved, downloaded from, and written back to a mobile device.

Retrieval
The following describes how data is retrieved from the ServiceNow mobile app.

Read data
When a user requests to view information on the mobile app, the following steps occur.
1. The mobile app sends a request to access data from the instance.
   The request includes the token and any relevant data field needed for the request.
2. The instance receives the request and checks if the Token is valid.
3. If the token is valid, the request is directed to the relevant API to fetch the information.
4. The information is returned to the mobile app.

Downloading documents
When a user requests to download documents from the app, the following steps occur.
1. The mobile app sends a request to access the document.
   The request includes the Token.
2. The instance receives the request and checks if the Token is valid.
3. If valid, the document becomes available to view or take further actions on the device.

Write-backs
The following describes how data is written back from the ServiceNow mobile app.

Updating fields
When a user updates a field in the mobile app, the following steps occur.
1. The mobile app sends the Token and the action metadata, for example the ID, or the field to be updated, to the instance.
2. The instance directs the action based on the relevant API.
3. The instance completes the action and sends a response to the mobile app.
4. Based on the response, the mobile app reflects the field changes and action availability in the UI.

### Attaching documents

When attaching documents, the following steps occur.

1. The mobile app asks the user to attach a document, for example, an image.
2. The mobile app sends the document and Token to the instance.
3. The instance places the document based on the relevant API.
4. The instance sends a response back to the mobile app.

### Internal mobile app distribution

Internal distribution of the ServiceNow Classic app is supported through all major EMM vendors.

Customers are able to pull the app for iOS or Android from the Apple App store and Google Play respectively, dynamically configure the apps to point to the correct ServiceNow instance, and distribute using the EMM hub. This way, the MDM can fully manage the app as part of a customer portfolio.

**Note:** ServiceNow does not currently distribute the ipa/apk files, or any other unpublished app to customers as it breaches the Apple Enterprise Developer License Agreement.

Mobile app distribution providers:

- *AirWatch Mobile Device Management (MDM)*
- *Blackberry Unified Endpoint Management (UEM)*
- *Citrix XenMobile*
- *Intunie mobile device management (MDM)*
- *Jamf Pro mobile device management (MDM)*
- *IBM Maas360 mobile device management (MDM)*
- *MobileIron Mobile Device Management (MDM)*

### Data security for ServiceNow Classic

The ServiceNow Classic app uses SSL/TLS for Over-the-Air communication encryption. The OAuth authorization endpoints are HTTPS.

#### Data at rest

Application preference data such as favorites, home screen, and the mobile navigator items are stored and cached locally on the device. The mobile app does not store record data such as incidents, problems, etc. on the device unless the organization has specifically enabled offline syncing for Field Services. The record data is encrypted with AES 128.

#### Data in motion

Data in motion is over a secure SSL/TLS channel and encrypted with HTTPS.
Offline access and data cache configuration

Some field service tables are available to cache locally on the device at the customer’s discretion.

Push notifications

Administrators create push notifications and users are able to receive them.

Cloud

For more information on the push notification system including process, configuration, and architecture, see Push notification system. Administrators can configure push notification delays using scheduled jobs. To view an example included with the base system, navigate to System Scheduler > Scheduled jobs, then search for a job with the name Push. 5 seconds is the minimum time allowed for the push delay.

Mobile security practices

Mobile security practices include mobile-specific system properties, attachment control, password reinforcement, security patching, and controlling shared data.

Security controls for ServiceNow Classic

Configure security controls to restrict copy/paste, enable biometric controls, enforce passwords, or control attachment functionality.

Restrict copy/paste

Copy/paste restrictions are defined in the system properties [sys_properties] table. There are two applicable security properties.

- glide.ui.m.clear_pasteboard_when_backgrounded: Clears the copy/paste clipboard when the ServiceNow app enters the background
- glide.ui.m.blur_ui_when_backgrounded: Forces the app to blur the screen when the app enters the background on iOS. This property prevents users from being able to take screenshots and also blurs the screen when in app switcher on Android.

PIN/Password reinforcement

Standard platform password requirements are enforced. Any additional device hardening is the responsibility of the customer.

Attachment control

Use an ACL to block specific access on mobile. Use the isMobile method to check if a request comes from a mobile device. For example, you could add an ACL for the attachment [sys_attachment] table where the read and write scripted ACLs includes the following check. You can also add this code to any existing ACLs you have for the attachment table. If have multiple attachment ACLs, all of the need to have Admin override option unchecked.

```java
if( gs.isMobile() ){
    answer = false;
}
```
Note: You need elevated privileges to create ACLs.

Security patching
In the event a security patch is needed, the mobile development team aligns with standard SDLC properties in order to patch.

User data collection
The mobile app does not specifically collect any user data.
Any user transactions or usage within the app is tracked on the ServiceNow instance just as it is on the web. For user credentials, after a user logs in, the mobile app negotiates an OAuth Token that is stored in the Apple Keychain or the Android Keystore. User credentials are never saved. If the user opts in, the following information is collected:

- Location
- Access to camera
- Notifications

Shared data
The mobile app communicates with a third party Google program called Fabric for app crash reporting. No customer information is shared.

Incident reporting
Mobile app issues should be reported through the standard support channels. You can report incidents by contacting ServiceNow Technical Support.

Mobile configuration for ServiceNow Classic
As an administrator, configure the ServiceNow Classic mobile experience for your users to access an instance on a tablet or smartphone.

Requirements
Configuration for the mobile experience takes place in a desktop browser. Most of the configuration options are available for both the ServiceNow Classic mobile app and the mobile web experience.
Use ServiceNow Classic or the mobile web to test your configuration. Depending on your device, go to the Apple App Store or the Google Play store and search for ServiceNow to download the ServiceNow Classic mobile app.

Role required
admin

Before you begin
Consider the following questions with your stakeholders before configuring the mobile experience:

- What applications and modules do you want to be accessible from a mobile device?
- How do you want mobile lists to appear on a mobile device?
- Which users or roles should have access to Connect Chat?
What to do

**Configure the application menu**
Determine which applications and modules you want to appear in the mobile application navigator.

**Customize the mobile home screen**
Customize the appearance of the mobile app or on a mobile web browser by navigating to System Mobile UI > Home Screen. Use the home screen configuration options to customize the mobile experience for each user by role.

**Create a mobile theme**
Add a consistent look and feel to the mobile experience by creating a mobile theme.

**Set up mobile lists**
Mobile devices have less screen real estate so you may need to change the length of titles or the default fields that appear in a mobile list. Customize the mobile list view to determine the length of a title in a mobile list, as well as the default fields that appear in a table.
Customize sortable columns in a mobile list.
Configure available search fields.

**Configure Connect Chat for mobile**
Configure the users or roles that can access Connect Chat on a mobile device.

Next steps

Most configuration items for the mobile UI are optional. Click any of the topics below for additional configuration.

**Mobile authentication for ServiceNow Classic**
Administrators can configure mobile devices to use different levels of authentication.
Customizing the mobile login page is not supported.
To use multi-provider SSO, the multi-provider SSO plugin must be enabled.
The following authentication options are not supported or have limited support for mobile:
- SAM L 2.0 plugin: Not supported for mobile.
- Kerberos: Native mobile apps do not support Kerberos authentication.
- Okta: Using the Okta app to launch the ServiceNow app is not supported. You can, however, launch the native mobile apps using the web version of Okta as long at the multi-provider SSO plugin is enabled.

**ServiceNow Classic multifactor authentication**
Users can access an instance on a mobile device - with either the ServiceNow Classic mobile app or a mobile browser - using multifactor authentication (MFA). The Integration - Multifactor Integration plugin [com.snc.integration.multifactor.authentication] must be active for MFA to apply to mobile authentication.
For more information on configuring multifactor authentication for users, see Configure multifactor authentication (MFA).

The mobile web experience has a secondary authentication screen that requests the MFA code. The native mobile apps do not currently prompt for the MFA code on a second screen. The MFA code needs to be appended to the password on the login screen when using the ServiceNow Classic mobile app. For example, P@ssw0rd135642, where 135642 is the MFA code.
Single sign on for the ServiceNow Classic mobile app

The classic mobile app leverages federated login when using the multi-provider SSO plugin.

Make sure the Integration - Multiple Provider Single Sign-On Installer [com.snc.integration.sso.multi.installer] plugin is activated. The SAML 2.0 plugin is not supported on mobile. For more information on how to configure multi-provider SSO, see Set up Multi-Provider SSO.

Enable multi-provider SSO

For multi-SSO to work on mobile devices, you need to enable the glide.authenticate.multisso.enabled property.

1. Navigate to Multi-Provider SSO > Administration > Properties.
2. Select the Enable multiple provider SSO check box.
3. Click Save.

SSO and the OAuth token

Once a successful session is established with the federated identity provider and the instance, the ServiceNow Classic mobile application negotiates an OAuth token with the instance. This allows the applications to re-establish connections without storing any user credentials on the device.

The glide.authenticate.sso.redirect.idp system property is not required to use SSO with mobile applications. Mobile applications bypass the local login page and go directly to the SSO/IDP if this property is set.

To enable both local and external login for mobile applications, disable the glide.authenticate.sso.redirect.idp property.

Enable e-signature for approvals for the ServiceNow Classic mobile app

Administrators can configure e-signature to enforce approvals with passwords or biometrics in mobile browsers and the ServiceNow Classic mobile app.

Role required: admin

Note: This feature is only available in mobile browsers and the ServiceNow Classic mobile app.

E-signature uses the following re-authentication behavior:

- If a user logs in with SSO, e-signature attempts to re-authenticate using the normal SSO authentication flow.
- If a user logs in with a local username and password, e-signature attempts to re-authenticate the user with the local password only, not the username.
- If a user has biometrics, or TouchID, enabled on their device, e-signature attempts to re-authenticate with the user's biometrics.

   For more information on activating a plugin, see Activate a plugin.
2. To enable a specific table for e-signature approval, navigate to System Definition > e-Signature Registry, then click New and complete the form.

Authentication using biometrics is enabled by default. To disable biometric authentication, enter sys_properties.list in the navigation filter, open the glide.ui.m.auth.allow_biometrics property, and set it to false.
Enable an application menu for the ServiceNow Classic mobile app

Define which application menus and modules are available on the mobile application navigator. Enable applications to show new applications in the ServiceNow Classic app or restore applications that were previously removed.

You cannot hide the application navigator from users, however you can control the contents based on role. Use the Roles field for both the application menu and the modules as listed in the following tables to configure the application navigator specifically by role.

1. On a desktop browser, navigate to **System Mobile UI > Navigator Apps**.
2. Click **New**.
3. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name for the application menu.</td>
</tr>
<tr>
<td>Order</td>
<td>Enter a number to specify the order of the menu. For example, an entry of 100 would place this application menu before one with an Order entry of 200.</td>
</tr>
<tr>
<td>Roles</td>
<td>Click the lock icon to select the roles for this application menu. Only users with the designated roles can access the modules under this application menu.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to activate this application menu.</td>
</tr>
</tbody>
</table>

4. Save the application menu.

**Note:** Until you add at least one module, the new application menu does not appear on the mobile application navigator.

5. Click **New**.
6. Fill in the fields, as appropriate.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Enter a name for the module.</td>
</tr>
<tr>
<td>Order</td>
<td>Enter a number to specify the order of the module within the application menu. For example, an entry of 100 would place this module before one with an Order entry of 200.</td>
</tr>
<tr>
<td>Application menu</td>
<td>Displays the application menu from which you accessed this screen. Select a different application menu, if appropriate.</td>
</tr>
<tr>
<td>Table</td>
<td>Select the table for this module.</td>
</tr>
<tr>
<td>Updated</td>
<td>Displays the date and time when the module record is updated.</td>
</tr>
<tr>
<td>Roles</td>
<td>Click the lock icon and select the roles for this module. Only users with the designated roles can access this module.</td>
</tr>
<tr>
<td>Active</td>
<td>Select the check box to activate this module. Only active modules appear in the application menu.</td>
</tr>
<tr>
<td>Filter</td>
<td>Create a filter for identifying which fields this module uses from the selected Table.</td>
</tr>
</tbody>
</table>
### Field Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| Path                                           | Enter a custom URL for the module in this format: `<type_of_link>/table/parameters`.  
  - `type_of_link`: determines what kind of page opens, for example, form, list, or view.  
  - `table`: Refers to the table being referenced. For example, incident.  
  - `parameters`: Any additional information you want to use to direct the user a specific URL. For example, you can include a query as part of the URL to direct users to a specific filtered list.  
  You can also use the URL format, `$___.do`.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| Path Relative To Root                          | Allows mobile navigator modules that link to desktop pages. Select this check box to use mobile URL in the Path field, such as `$sp.do`, `$chat.do`, or `$vtb.do`.  
  When the Path Relative To Root is cleared, the path is considered a mobile specific path and is relative to `$m.do`.  
  For example, a path like `/form/incident/-1` would be a mobile specific path and would be considered relative to `$m.do`. This would ultimately result in navigating to a path of `$m.do#/form/incident/-1`.  
  Use mobile specific paths when possible and only use Path Relative To Root when there is not a mobile specific path for the desired resource.                                                                                                                                                                                                                                                                                                                                                           |

**Note:** If you are creating a module for a map page, see *Create a smartphone map page module.*

7. Click Submit.

### Customize lists and forms for the ServiceNow Classic mobile app

In order to optimize views on smaller screens, lists and forms display differently on a mobile device.

Use any of the following options to customize different aspects of mobile lists.

**Customize a mobile list**

Customize which fields display in a list on a mobile device using table titles.

**Configure field styles**

Field styles allow you to set individual styles for each item in a list. You can configure field styles for a mobile list that are separate from the desktop view.

**Configure the navigation behavior when a user taps an item in a list**

Determine whether tapping an item in a list opens the record or the activity stream for an item.

**Configure how a form displays on a mobile device**

Limit the number of fields on a form for a better mobile experience.

**Hide mobile filters**
Hide the filter option or the activity stream from users to prevent them from filtering out search results.

**ServiceNow Classic mobile list view**

The information that appears in a mobile list is different than a list on a desktop. Smaller screen real estate limits the information that is easily viewable. You can control mobile list text using the table titles module.

Use Table Titles to control the information that appears in a mobile list. The information that appears in a list item is made up of the following components.

- **Title:** Appears prominently at the top of the list item. You can configure the list item title by selecting the fields that appear as the title. For a more advanced configuration, you can use a script to determine how and what displays as the list item title.
- **Display title:** Appears as subtext beneath the title for a list item. You can customize the display title by using the configure dictionary option for a field on a form.
- **Extras:** Any extra information you want to appear in the list. Extras are only configurable using a script.
Customize the list title for the ServiceNow Classic mobile app

The information that appears in a mobile list is different than a list on a desktop. Smaller screen real estate limits the information that is easily viewable. You can control mobile list text using the table titles module.

1. In the desktop interface, navigate to System Mobile UI > Table Titles.
2. Click New, or select an existing table to edit.
3. If you are creating a new list, select the table from the list.
The list shows only tables and database views that are in the same scope as the title.

4. Use the Fields slushbucket to add items that appear in the list item.

The order of the items in the selected column determines the order they appear in on a mobile device.
Table titles define the title appearance for a list on a mobile device as well as in the split pane view on the desktop interface. You can change the title for any table or add a title record for a different table. However, the change affects both mobile lists and the list card in the split pane view.

For example, the default title for the Incidents [incidents] table is the short description. If you change the table title to display the priority field instead, the priority field appears as the table title for incidents on a mobile device as well as in the list card in the split pane view.

5. Click *Submit*.
This example shows a list of incident records with the Short Description, Assigned to, and Updated fields included.

**Script items in a list for the ServiceNow Classic mobile app**

For a more advanced option for configuring the information that appears in a mobile list, you can create customized scripts.

Role required: admin

Use scripts to control the following components in a mobile list.
• Title: Use the `setTitle()` method to script a title. The list title is connected to a specific table in the system, for example Problem. Each table contains a list of field options, defined by `field_list` that you can use to determine the title.
• Extras: Use the `setExtras()` method with an array of strings to configure the information that displays after the list title. Scripting extras adds new rows to the list items.

1. Navigate to **System Mobile UI > Table Titles**.
2. In the **Script** field, create a script to define which fields appear in the mobile list. For example, the following code block uses `setTitle` and `setExtras` to configure a mobile list for the problem table. The title shows the short description for a problem. The extras include additional information for the problem priority.

   ```javascript
   var title = current.short_description;
   var extras = [];
   if (current.priority === 1){
     extras.push("Critical!");
   }
   else if(current.priority === 2) {
     extras.push("High Priority!");
   }
   else {
     extras.push("Carry on");
   }
   titleValues.setTitle(title);
   titleValues.setExtras(extras);
   ```
3. Click **Submit**.

   *Add a display title for the ServiceNow Classic mobile app*

   Use the display title option to add a subtitle to list items on a mobile device.

   Role required: admin

1. Navigate to the table you want to configure the display title for. For example, Problem.
2. Right-click the field you want to display and click **Configure Dictionary**. For example, Urgency.
3. Select the **Display** check box.

   You can only have one display title for a table. The system automatically clears the selection for any previously selected display titles when you select a new one.

   ![Performance issue in the custom extension written for Workday Payroll](image)

   Display title

---

**Configure field status indicators for the ServiceNow Classic mobile app**

Differentiate items in a list by configuring the field status indicators.
Role required: admin

1. Navigate to System Mobile UI > Table Titles.
2. Select a table to add field status indicators to.
3. From the Style Field list, select the styled field you want to appear with the list on a mobile device.

   **Note:** If the Style Field does not appear on the form, you may need to configure the form.

**Configure list item navigation for the ServiceNow Classic mobile app**

Determine whether a tapping an item in a list opens the record or the activity stream using a system property.

Role required: admin

1. Navigate to System Properties > Mobile UI properties.
2. From the Destination when navigating to a record from a list property [glide.ui.m.default_record_navigation], select Activity Stream or Form.
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Activity Stream</td>
<td>Opens the list item in the activity stream</td>
</tr>
</tbody>
</table>

### How do I create a sub-folder

**Number:** INC0000017  
**Category:** Inquiry / Help  
**Priority:** 1 - Critical

**Related Lists** (1)

**Participants**

**Aug 25, 2015**

- **Don Goodcliffe**  
  I got this

- **Don Goodcliffe**  
  Task reassigned due to employee termination
<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Form</td>
<td>Opens the list item in the form</td>
</tr>
</tbody>
</table>

Configure list search fields for the ServiceNow Classic mobile app

Configure search fields to determine what fields are searched for in the table. If search fields are not configured, only the columns in the mobile list layout are searched.

1. Navigate to **System Mobile UI > Table Titles**.
2. Open the table you want to configure search fields for, or click **New**.
3. Move fields from the **Available** list to the **Selected** list to add fields to the mobile list.
**Note:** The **Search fields** field might not appear on the Table Title form. Add **Search Fields** to the form by editing the form layout.
Configure the list layout for the ServiceNow Classic mobile app

Configure and order the fields displayed in a list view in the ServiceNow Classic mobile UI. Ensure that your users have the most relevant information when browsing on their mobile devices.

1. From a desktop browser, navigate to any list you want to configure, for example, Incidents.
2. Click the list header menu and navigate to Configure > List Layout.
3. From the View name list, select Mobile.
4. Move fields from the Available list to the Selected list to add fields to the list.
Any fields you add to the list appear as a sortable column and as a field in the card of a specific record in the mobile UI.

**List layout results**

<table>
<thead>
<tr>
<th>Record field</th>
<th>Sortable column</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Title</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Number</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Category</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Priority</strong></td>
<td></td>
</tr>
</tbody>
</table>

**Caller First name: Margaret**

**Oct 20, 2015**

**System Administrator** 8 mos
Hide filters for the ServiceNow Classic mobile app

Prevent users from filtering content in a list by hiding the filter.

Role required: admin

1. Navigate to the table you want to hide filters for.
2. Open the list configuration page by performing the appropriate action for the list version. For more information on configuring list controls, see Configure list controls.

<table>
<thead>
<tr>
<th>Version</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>List v2</td>
<td>Right-click any column heading and select Configure &gt; List Control.</td>
</tr>
<tr>
<td>List v3</td>
<td>Open the list title menu and select List Control.</td>
</tr>
</tbody>
</table>

3. Select the **Omit filters** check box.
   
   This field is only available for standard lists.

Customize default mobile home pages for the ServiceNow Classic mobile app

Administrators can create customized home pages for their users on the ServiceNow Classic mobile app. Administrators can customize mobile home pages by section and module. Depending on the level of customization required, administrators can limit user access by role to each component of the mobile home page. Mobile home pages apply to both the ServiceNow Classic mobile app and the mobile browser.

Role required: admin

Use the following components to create a customized home page:

- Home Page Collections
- Home Pages
- Home Page Section
- Home Page Modules

Each component can be configured separately or using the following steps.

1. Navigate to **System Mobile UI > Home Screen > Home Page Collections**.
2. Click **New** or open an existing Home Page Collection. For more information on completing the collection fields, see [Home Page Collections](#).

   If you want to configure an entire homepage to be role specific with no default modules, you can create a new Home Page Collection. Otherwise, simply open an existing collection to configure the page.
3. From the related list on the Home Page Collection form, click **Edit...** to add an existing Home Page or click **New**.

   For more information on completing the Home Page form, see [Mobile Home Pages form](#).
4. From the related list on the Home Page form, click **Edit...** to add an existing Home Page Section to the Home Page, or click **New**.

   For more information on completing the Home Page Section form, see [Mobile Home Page Sections](#).
5. To add modules to a homepage, from the related list on the Page Section form, click **Edit...** to add an existing module, or click **New**.

   For more information on completing the Home Page Modules form, see [Mobile Home Page Modules](#).

Home Page Collections

Use the Home Page Collections as a starting point for your home page customization.
Administrators can configure Home Page Collections by role so only specific users can see a particular home page. For example, you can configure an ITIL user homepage to only have modules that are relevant to the ITIL user. However, if you have default apps that different roles use, you can also create a basic homepage, with the option of customize specific pages by roles.

**Home Page Collections fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the collection. This field won't appear in the app or on the mobile web.</td>
</tr>
<tr>
<td>Priority</td>
<td>The numbered priority of the collect. Set a higher priority than the default collection if you want the homepage collection to show up for specific roles. For example, if the default is set to 500, set the role specific collection to 200.</td>
</tr>
<tr>
<td>Hide Favorites Page</td>
<td>Users can configure their own homepage by favoriting items in the system. The favorites page is accessible with a right swipe. Administrators can disable the favorites page by selecting this option.</td>
</tr>
<tr>
<td>Roles</td>
<td>Use this option to configure home page collections for specific roles.</td>
</tr>
</tbody>
</table>

**Mobile Home Pages form**

Configure the look and feel of mobile home pages.

Administrators can configure more than one home page for their users. Each additional home page is accessible to users by swiping to the left from the main home page. The sort order, accessible from the Home Page Collections related lists, defines the Home page order. Administrators can also configure home pages by role.

**Mobile home page fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Title of the home page. The title appears on the home page.</td>
</tr>
<tr>
<td>Title position</td>
<td>Determines the location of the title in the app or mobile web. Selecting Navigation Bar overrides the theme configuration and adds the title in place of the main header bar. Use Hidden to hide the title.</td>
</tr>
<tr>
<td>Subtitle</td>
<td>Create an additional title to describe the page. Subtitle appears below the title or header image.</td>
</tr>
</tbody>
</table>
Field | Description
--- | ---
Header image | Optional. Select an image to appear at the top of the mobile home page. A header image lets you visually communicate the purpose of the home page. Note the following guidelines:

- Use a high-resolution image that looks good at different sizes and aspect ratios. The display size and aspect ratio of a header image vary by the device screen size. Since it is not possible to specify a fixed display size for a header image, use a high-quality image that works for different screen sizes.
- Avoid text on a header image. Text can be cropped or appear blurry when the image is scaled for different screen sizes. Also, text is not localized because the header image is a static file. Use the home page title and subtitle properties to include any text on a home page.
- Avoid using a company logo as a header image. Ideally a header image artistically represents the purpose of the home page. If a logo is needed, create a custom mobile theme (sys_ui_mobile_theme) with a title image that includes the company logo in the navigation bar of the home screen.
- For domain separation, header images uploaded in the parent domain do not appear for users in the child domain. Either use a global domain or the child domain that the users are in.

Roles | Configure a home page by role.

Mobile Home Page Sections

Administrators can configure Home Page Sections to determine how apps appear in a mobile home page.

Mobile home page section fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Title of the home page section. This appears in the app or mobile home page above the list of modules.</td>
</tr>
<tr>
<td>Hide Title</td>
<td>Hide the home page section title so it doesn't appear on the page.</td>
</tr>
<tr>
<td>Module Style</td>
<td>How the modules appear in the section. <strong>Regular</strong> shows the modules as rectangles that stretch the length of the app or mobile web. <strong>Compact</strong> displays the modules as smaller &quot;app-like&quot; squares.</td>
</tr>
<tr>
<td>Roles</td>
<td>Configure Home Page Sections by role.</td>
</tr>
</tbody>
</table>

Mobile Home Page Modules

Home Page Modules define what information appears on a mobile home page.
Modules must be enabled for mobile to appear on a mobile home page. Administrators can also configure Home Page Modules by role.

**Mobile Home Page Module fields**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Name of the module that appears in the app or mobile web.</td>
</tr>
<tr>
<td>Subtitle</td>
<td>Optional module description that appears beneath the title.</td>
</tr>
<tr>
<td>Module</td>
<td>Select a module from the list of available mobile modules</td>
</tr>
<tr>
<td>Active</td>
<td>Whether the module is available or not. Clear this check box to disable mobile access to a mobile module.</td>
</tr>
<tr>
<td>Roles</td>
<td>Configure module access by role. Users without the specific role won’t see the module.</td>
</tr>
<tr>
<td>Icon</td>
<td>Select an icon from the list of available icons.</td>
</tr>
<tr>
<td>Image...</td>
<td>Upload your own module image</td>
</tr>
<tr>
<td>Background color</td>
<td>Select a background color for the module. Use color name or Hex color. For example, #81878e or light gray.</td>
</tr>
<tr>
<td>Content Style</td>
<td>Select a style from available themes.</td>
</tr>
</tbody>
</table>

**Configure a theme for the ServiceNow Classic mobile app**

Configure a theme for the ServiceNow Classic mobile UI to make the experience more consistent for your users. Theme changes apply to both the ServiceNow Classic mobile app and the mobile web experience.

Role required: admin

1. Navigate to **System Classic Mobile UI > Themes**.
2. Click an existing theme to edit, or click **New**.
3. Complete the mobile theme form fields.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Title for the theme. This does not appear anywhere for end users.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Base theme</td>
<td>The underlying theme to build your styles on. Use the base theme to determine if the main background is darker or lighter.</td>
</tr>
<tr>
<td>Navigation bar title</td>
<td>A title that appears in the navigation bar of the app or mobile web page. For example, your company name or the group you are creating the app for. The navigation bar title also appears at the title when you save a mobile web shortcut on your phone.</td>
</tr>
<tr>
<td>Title image</td>
<td>An image that appears below the navigation bar. The title image overrides the navigation bar title.</td>
</tr>
<tr>
<td>Theme Colors</td>
<td>Navigation bar and toolbar colors for the native app or mobile web. Use color names or Hex codes. For example, red or #ff0000.</td>
</tr>
</tbody>
</table>

4. Click **Activate Theme**. The sysID for the new theme appears in the Set the active theme for the mobile experience field on the Classic Mobile UI properties page. You can change the theme the ServiceNow Classic mobile app uses by changing the sysID in this field.
Configure Connect Chat for the ServiceNow Classic mobile app

Configure Connect Chat to show or hide on the ServiceNow Classic mobile app homepage. Connect Chat appears on the ServiceNow Classic mobile app homepage by default if it is active on the instance and the user has the correct roles. Hide Connect Chat in the mobile app by disabling the Connect module in the mobile homepages.

Connect Chat has only been optimized for the ServiceNow Classic mobile app, not the mobile web.

1. Navigate to System Mobile UI > Home Screen > Home Page Modules.
2. Open the Connect home page module.
3. Clear the Active check box.
   You can also use the Roles field to configure access to Connect Chat by role.

Configure the appearance of the Connect Chat homepage module. For more information on configuring home page modules, see Mobile home page module fields.
Mobile UI actions for the ServiceNow Classic mobile app

UI actions function the same on the mobile interface as on desktop interface, but are configured on a different table: UI Action - Mobile [sys_ui_ng_action].

You can use the following types of UI actions in the mobile interface:

- List buttons
- Form buttons
- Form more items (items that display when the user taps the More button)

![Table containing UI actions](image)

UI action

The lowest number in the Order column identifies the primary button at the top of the form.

**Note:** Mobile UI actions automatically reload the form when an action is submitted. You do not need to configure a UI action to make this happen.

Back navigation in the ServiceNow Classic mobile app and mobile web interface

Mobile UI actions support back navigation when a UI action finishes. You can navigate back one logical navigation item in the current navigation stack.

When the `navigate_back` flag is set to true for a mobile UI action in the UI Action - Mobile [sys_ui_ng_action] table, you can move back to the previous screen.

Sometimes one or more discrete screens, or a nested or partial view of the screen, can represent a single navigation item. For this reason, back navigation can result in different UI behaviors depending on the current navigation content or device.

For example, the activity stream and form for a record are considered as one logical navigation item (the current record). However, some devices use multiple discrete screens for this navigation. When back navigation is triggered from a UI action on a form, multiple screens may be dismissed to return to a previous item, such as a list.
Enable client scripts for the mobile browser for the ServiceNow Classic mobile app

Control whether client scripts for forms run on the mobile interface, the desktop UI, or both. Use this option to create scripts specific to one interface, or prevent long running from executing on the more limited hardware of a mobile device.

1. On a desktop, navigate to System Definition > Client Scripts.
2. Create a client script that is compatible with the mobile interface.
   To ensure compatibility, see Mobile client GlideForm (g form) scripting and migration.
3. In the UI Type field, select Mobile/Service Portal or All to have the script run on the mobile interface.
   If the UI type field is not available on the form, configure the form to add the UI Type field.
4. Fill in the fields, as appropriate.
5. Click Submit.
Mobile client GlideForm (g form) scripting and migration

Client scripting for mobile is identical to scripting for the web, with some exceptions. All new scripts must conform to certain guidelines. The following items are affected on the mobile platform: client scripts, UI policies, navigator modules, and UI actions.

Client scripts

For new or existing scripts to be valid for mobile, they must conform to the following requirements:

- Use the new mobile methods in place of `g_form.getControl()`.
- Do not use deprecated methods.
- Do not reference unsupported browser objects.
- Do not make synchronous JavaScript, GlideAjax, and GlideRecord calls.
- Do not call methods that are not available for mobile.
• Enable scripts to run on the mobile UI.

### Requirements

#### Use the new mobile methods

Several new methods are available for modifying form fields instead of directly manipulating the HTML. These methods replace previous usages of `g_form.getControl()`, which is deprecated for the mobile platform. In your existing scripts, ensure that the new methods are used in place of methods that are not valid on the mobile platform. For information on these new methods, refer to .

#### Do not use deprecated methods

The following methods have been deprecated for the mobile platform because direct access to HTML elements is not allowed:

- `g_form.getControl()`
- `g_form.getFormElement()`
- `g_form.getElement()`

To ensure that existing scripts are compatible, remove all calls to deprecated methods from your code. For new scripts, do not use deprecated methods if you want the script to be valid for mobile.

For `g_form.getControl()`, some of the functionality previously included with this method has been extracted to individual methods. Instead of `g_form.getControl()`, use the new methods described on the developer site.

#### Do not reference unsupported browser objects

The following browser objects are not supported in mobile scripts:

- Window
- jQuery or Prototype ($, $j, or $$)
- Document

Make sure that new scripts do not use these objects, and remove any usage of these objects from your existing scripts. Use GlideForm (`g_form`) instead, which provides methods such as `setLabel()`, `addDecoration()`, and `hasField()` for accomplishing the same tasks.
| Do not make synchronous JavaScript calls | The mobile platform does not allow synchronous JavaScript calls. The `g_form.getReference()` method must now have the callback parameter defined. For example:

```javascript
    g_form.getReference(fieldName, callback)
```

Be sure that all `g_form.getReference()` calls include the callback parameter. For example, the following script does not include a callback and is incompatible with the mobile platform:

```javascript
    var userName = g_form.getReference('assigned_to').user_name;
    g_form.setValue('u_assigned_user_name', userName);
```

The following script has been updated to include the callback and is compatible with the mobile platform:

```javascript
    g_form.getReference('assigned_to', function(gr) {
        g_form.setValue('u_assigned_user_name', gr.user_name);
    });
```

| Do not make synchronous Ajax calls | The mobile platform does not allow synchronous GlideAjax calls. Any use of `getXMLWait()` in a GlideAjax call will not work on the mobile platform. Be sure that all GlideAjax calls are asynchronous. For more on synchronous versus asynchronous GlideAjax calls and `getXMLWait()`, see AJAX. For information on the available GlideAjax methods, refer to the |
| Do not make synchronous GlideRecord calls | The mobile platform does not allow synchronous calls. Make sure that any existing GlideRecord calls include a callback. For example, the following script does not include a callback and is incompatible with the mobile platform:  

\[
\begin{align*}
\text{var gr = new GlideRecord('incident');} \\
\text{gr.addQuery('number', g_form.getValue('related_incident'));} \\
\text{gr.query();} \\
\text{gr.next();} \\
\text{g_form.setValue('u_related_incident_description', gr.short_description);}
\end{align*}
\]  

The following script has been updated to include the callback, and is compatible with the mobile platform:  

\[
\begin{align*}
\text{var gr = new GlideRecord('incident');} \\
\text{gr.addQuery('number', g_form.getValue('related_incident'));} \\
\text{gr.query(function(gr) {} } \\
\text{\quad gr.next();} \\
\text{\quad g_form.setValue('u_related_incident_description', gr.short_description);} \\
\text{});}
\end{align*}
\] |
| Do not use methods unavailable on the mobile platform | Due to the limitations and reduced functionality that is imposed by the mobile platform, the following methods are not deprecated but are not available on the mobile platform. If these run on the mobile platform, no action occurs:  

- `showRelatedList()`  
- `hideRelatedList()`  
- `showRelatedLists()`  
- `hideRelatedLists()`  
- `flash()`  
- `getSections()`  
- `enableAttachments()`  
- `disableAttachments()`  
- `setReadOnly()` (Note that `setReadonly()` is available)  
- `getParameter()` |
| Enable scripts for mobile | Scripts must be enabled for the mobile platform. See Enable client scripts for the mobile browser for the ServiceNow Classic mobile app. |

**Note:** Focusing an element on a mobile form is not supported.
UI policies

Use the **Run scripts in UI type** field to determine whether scripts run on the mobile platform, the desktop, or both. Update existing policies so that they apply to either the mobile platform or both. For new scripts, also ensure that the mobile option or both is selected. For more on UI policies for mobile, see *Enable UI policies for the mobile browser*.

Navigator modules

For existing code, modules must be transferred to either the sys_ui_application or sys_ui_module tables to be available on the mobile platform. When developing new code, be sure that all modules are created in the sys_ui_application or sys_ui_module tables. For more information, see *Enable an application menu for the ServiceNow Classic mobile app*.

UI actions

UI actions must be transferred to the sys_ui_ng_action table to appear on the mobile platform. UI action scripts that do not use deprecated methods do not require changes to the script itself. For new UI actions, be sure that they are created in the sys_ui_ng_action table. For more information, see *Mobile UI actions for the ServiceNow Classic mobile app*.

Enable UI policies for the mobile browser

Define UI policies to run on forms on the mobile interface, the desktop UI, or both.

1. In the desktop interface, navigate to **System UI > UI Policies**.
2. **Create a UI policy** that is compatible with the mobile interface.
   
   To ensure the UI policy is compatible on a mobile device, see *mobile UI policies*.
3. Configure the UI Policies form to add the **Run scripts** and the **Run scripts in UI type** fields.
4. Select the **Run scripts** check box.

   **Note:** The **Run scripts in UI type** field does not appear unless the **Run scripts** check box is selected.

5. In the **Run scripts in UI type** field, select **Mobile/Service Portal** or **All** to have the policy run on the mobile interface.
Configure 3D Touch options for iOS for ServiceNow Classic

Favorites determine the options that appear on the 3D Touch option for the ServiceNow app on iOS. You can configure your own options for your users using the Shortcut Items module.

Role required: admin

3D Touch allows users to access shortcuts to items in an app with a hard press on the app. 3D Touch is only available on the iPhone 6s and newer.

1. Navigate to Mobile App > Shortcut Items.
2. Click New.

6. Complete the fields, as appropriate.
7. Click Submit.
3. Complete the fields on the form.

Shortcut item fields

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Name for the shortcut that appears in the 3D Touch menu. For example, <em>Create new incident</em>.</td>
</tr>
<tr>
<td>Subtitle</td>
<td>Optional secondary title for the shortcut that appears in the 3D Touch menu. For example, <em>Create a new incident if there's a problem</em>.</td>
</tr>
<tr>
<td>Icon</td>
<td>A visual indicator for the shortcut.</td>
</tr>
<tr>
<td>Module</td>
<td>Module that opens when a user taps the shortcut. The module must be <em>configured to appear in a mobile device</em>. For example, the create new incident module.</td>
</tr>
<tr>
<td>Order</td>
<td>The order in which shortcuts appear. Lower numbers appear in the list first. Configured shortcuts also appear before user favorites in the 3D Touch menu.</td>
</tr>
<tr>
<td>Roles</td>
<td>Limit the users who can access the 3D Touch menu item by role.</td>
</tr>
</tbody>
</table>
Enable mobile location and barcode scanning for the ServiceNow Classic mobile app

Take advantage of mobile devices by allowing location and barcode scanning in the ServiceNow Classic mobile app.

Role required: admin

1. On a desktop, navigate to the form where you want to add the location or barcode scanner.
2. Use the form context menu on the left side of the form header to switch the form view to **Mobile**.

3. From the form context menu, navigate to **Configure > Form Layout**.

4. From the **View name** list, select **Mobile**.

5. In the **Create new field** section, set the field type to **String**.

6. From the form, right-click the new field and click **Configure Dictionary**.

7. In the **Attributes** related list, click **New**.

8. Use the **Attribute** field on the Dictionary Attribute form to search for **barcode** or **location**.

The added fields appear on the form in the native mobile app. This feature is not available on the mobile web. Users must allow the app to access their location and camera on their devices. These fields only appear when you create a record, they do not appear in one that is already existing.

The native apps for iOS and Android support scanning the following barcode types:

- 2D barcodes: QR Code, Data Matrix, PDF-417, AZTEC

**Disable mobile access for the ServiceNow Classic mobile app**

Administrators can disable or enable different parts of the mobile UI.

Navigate to **System Properties > Mobile UI Properties** to enable or disable any of the properties below.

<table>
<thead>
<tr>
<th>Name</th>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable mobile web UI</td>
<td>glide.ui.m_enabled</td>
<td>Turns on/off the mobile web UI. Even if this property is disabled, users still have access to the mobile web</td>
</tr>
<tr>
<td>Enable native mobile applications (Android, iOS)</td>
<td>glide.ui.m.native_apps_enabled</td>
<td>Enables/disables user access to the ServiceNow Classic mobile app. This property only applies to the classic mobile app.</td>
</tr>
</tbody>
</table>
## Get started with the ServiceNow Classic app

Use your smartphone or tablet to access an instance in a web browser or the ServiceNow Classic mobile app. The mobile UI varies depending on your device, the way your administrator has configured the mobile UI for your company, and whether you access your instance from the ServiceNow Classic mobile app or mobile browser.

### Before you begin

You can access your instance from a web browser on your mobile device or from the ServiceNow Classic mobile app. Depending on your device, go to the Apple App Store or the Google Play store and search for ServiceNow to download the native mobile app. If you do not have access to the mobile platform, contact your administrator.

### What to do

#### Access an instance on your mobile device

To open an instance on a mobile device:

- open an instance in your mobile app
- open an instance in your mobile browser

For mobile web access, add a shortcut to an instance browser page on your mobile device.

#### Set up homepage favorites

Use the application navigator to find modules and applications. Create and edit favorite modules or applications to appear as icons on your homescreen. In the ServiceNow Classic mobile app, add a visualization to the favorites on your homepage.

#### Use lists on a mobile device

View a list on your mobile device by tapping a homescreen favorite or by opening a module from the application navigator. Search for specific records in a mobile list. Use Mobile lists for ServiceNow Classic on a tablet.

#### Use Service Catalog, Connect Chat, location, and barcode scanning

Use mobile Service Catalog to order materials. Collaborate and stay connected with other users using Connect Chat. Allow the app to access your location and camera to check in or scan barcodes.

#### Manage your notification settings

Use Notification Settings to enable or disable the notifications that you receive and the channels for receiving them.
Log in on a mobile device

You can use the URL for your instance to log in using a mobile device. Access your instance from a mobile web browser or using the ServiceNow Classic mobile app.

Depending on your device, go to the Apple App Store or the Google Play store and search for ServiceNow to download the native mobile app.

Access an instance from the ServiceNow Classic mobile app

Access an instance using the ServiceNow Classic mobile app for supported devices.

1. After you download the app from the Apple App Store or the Google Play store, tap the icon on the homepage. Even with the newest version of the app, you can access any instance version as far back as Geneva patch 8. Some functionality is version-specific so if you do access an older version you might have more limited access. For example, only instances starting with Istanbul have access to customizable mobile home pages.

2. Use your regular login to sign in to the app.

After you log in, the app remembers your information. As long as you do not log out, the app opens the last screen you used when you open it again.

If you do log out, the app remembers your instance. Select the instance from the History list to log back in.

Note: The ServiceNow Classic mobile app does not have a screen for multi factor authentication (MFA). If your organization uses multi factor authentication, append the MFA code to your password on the login screen. For example, P@ssw0rd642135, where 642135 is the MFA code.

After logging in, use the mobile app to perform tasks on your instance. For more information on using ServiceNow Classic mobile app, see Using the ServiceNow Classic mobile app.

Access an instance from a mobile browser

Access an instance using a supported mobile browser.

On a supporte device, navigate to the URL of an instance in a supported mobile browser. Devices are automatically detected and the interface appends $m.do to the end of the URL. Accessing the mobile interface does not prevent the user from also accessing the tablet or desktop interfaces.
1. Enter the base URL in any supported mobile web browser.
2. Enter your user name and password.
3. Optional: Select the **Remember Me** check box to remain logged in until you manually log out.
   
   This option can be enabled or disabled by the administrator. For more information, see *Change settings for the Remember me check box and cookie*. 

Mobile web log in page
4. Optional: If you forget your password, tap **Forgot password?**. The system walks you through a verification process to confirm your information.

5. Tap **Continue** to complete your login.

   If your organization uses multi factor authentication (MFA), the web login includes an additional screen to input the MFA code.

**Add a shortcut to the home screen**

Add an Apple home screen shortcut that opens a mobile browser page.

1. In Safari on a mobile device, navigate to a page that you want to access as a shortcut, such as a list of open incidents.

2. Tap the sharing icon ( Sandwich ) at the bottom of the browser.

3. Tap the **Add to Home Screen** icon.

4. Type a descriptive name for the page and tap **Add**.

   The shortcut is saved to your device.

**Change the home screen icon**

Administrators can change the icon that appears for browser home screen shortcuts in the mobile interface.

To replace the default icon:

1. In the desktop interface, navigate to **Self-Service > My Profile**.

2. Switch to the **Default** view.

3. Click the reference icon by the **Company** field to open the company record.

4. Configure the form to add the **Apple icon** field.

5. Upload an image up to 57x57 pixels in the **Apple icon** field.

When users associated with the company **add a home screen shortcut**, the shortcut uses the new icon.
Configure a BlackBerry device
BlackBerry devices require some configuration to use the smartphone interface.

1. Open the browser application.
2. Open the menu and select Options > Browser Configuration.
3. Select the following options.
   - Support JavaScript
   - Use Background Images
   - Show Images: On WML & HTML Pages
   - Browser Identification: BlackBerry
4. Save your changes.

Depending on the size of the BlackBerry screen, you can change the default font size to a smaller size. Navigate to the General Properties browser submenu and set the default font size and minimum font size to a smaller size. Smaller text displays more content, but is difficult for some users to read.

Mobile application navigator in the ServiceNow Classic app
The application navigator provides access to all applications and the modules they contain, enabling users to quickly find information and services.

An application is a group of modules, or pages, that provide related information and functionality in an instance.

The application navigator also provides access to favorites and recently viewed items.

Administrators need to customize the application navigator so that the required modules appear on a mobile device. Most modules have not been enabled for mobile. For more information, see Enable an application menu for the ServiceNow Classic mobile app.

Use the application navigator in the ServiceNow Classic app
Use the application navigator to access different modules in the mobile interface.

To open the application navigator, tap the menu icon ( for iPhone and for Android).
Scroll the navigator up or down using one or two fingers. To expand an application, tap the application name. To hide the navigator, tap Close or the back arrow.
Use the tablet application navigator in the ServiceNow Classic app

Access different modules in the user interface using the application navigator just as you do in the mobile app.

1. To open the application navigator, tap the menu icon ( ).
2. Scroll the navigator up and down using your fingers. Tap a menu application to open.
To close the navigator, tap outside of the window.

**Mobile lists in the ServiceNow Classic app**

Lists appear as a single column of records on the mobile app UI display. Each row represents a separate record.

Live list updates are not available on mobile devices. Mobile lists only automatically reload when a record change is made by the currently logged in user. The list does not automatically update when other users make changes.
<table>
<thead>
<tr>
<th>Mobile UI lists for iPhone</th>
<th>Mobile UI lists for Android</th>
</tr>
</thead>
</table>
| 2
| 1
| 3
| 4
| 5
| 6
| 7

**Unable to get to network file shares**
INC0000002

**Wireless access is down in my area**
INC0000003

**Need access to sales DB for the West**
INC0000007

**I can't launch my VPN client since the last software update**
INC0000015

**Rain is leaking on main DNS Server**

**How do I create a sub-folder**
INC0000017

**Unable to get to network file shares**
INC0000002

**Wireless access is down in my area**
INC0000003

**Need access to sales DB for the West**
INC0000007

**I can't launch my VPN client since the last software update**
INC0000015

**Rain is leaking on main DNS Server**
INC0000016

**How do I create a sub-folder**
INC0000017
Mobile lists consist of the following elements.

<table>
<thead>
<tr>
<th>Number</th>
<th>UI element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>List name</td>
<td>Displays the name of the list favorite.</td>
</tr>
<tr>
<td>2</td>
<td>Back button</td>
<td>Navigates back to the home page.</td>
</tr>
<tr>
<td>3</td>
<td>UI actions</td>
<td>Displays buttons to display the list activity stream for iPhone and add a record.</td>
</tr>
<tr>
<td>4</td>
<td>Application search</td>
<td>Displays records that match the search string. On Android, search is located in the List options menu.</td>
</tr>
<tr>
<td>5</td>
<td>Current filter</td>
<td>Displays the conditions filtering the list.</td>
</tr>
<tr>
<td>6</td>
<td>Records</td>
<td>Displays one row for each record in the list.</td>
</tr>
<tr>
<td>7</td>
<td>List options</td>
<td>Displays list options to add to favorites, share, and sort the list.</td>
</tr>
</tbody>
</table>

Mobile activity streams in the ServiceNow Classic

List activity streams appear as a single column window with a separate row for each record update in the mobile UI. Access a list activity stream by pressing the activity stream icon. Activity streams open on a new page. Use the List options menu to open the Activity Stream on Android.

Rendering HTML text in activity streams is not supported for the mobile apps.
iPhone list activity stream
List activity stream for Android

Elements of the mobile UI: activity stream

© 2020 ServiceNow, Inc. All rights reserved.

ServiceNow, the ServiceNow logo, Now, and other ServiceNow marks are trademarks and/or registered trademarks of ServiceNow, Inc., in the United States and/or other countries. Other company names, product names, and logos may be trademarks of the respective companies with which they are associated.
Activity stream updates consist of the following elements.

**Mobile UI elements: activity stream**

<table>
<thead>
<tr>
<th>UI element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Back button</td>
<td>Return to the list.</td>
</tr>
<tr>
<td>User</td>
<td>Displays the user photo and name of the user who made the activity update.</td>
</tr>
<tr>
<td>Record details</td>
<td>Displays the record number and relative time since the update was made.</td>
</tr>
<tr>
<td>Activity details</td>
<td>Displays the most recent activity update.</td>
</tr>
</tbody>
</table>

Swipe up or down to see more activity stream updates.
Mobile filters in the ServiceNow Classic app

Use the condition builder to create complex filters in the native mobile app.

Not all filters are available on the mobile app. Saving a mobile filter is not supported. You can, however, add a filtered list to the favorites menu by selecting the star icon, which saves the list with the filter.

<table>
<thead>
<tr>
<th>iPhone UI filter</th>
<th>Android UI filter</th>
</tr>
</thead>
</table>

Mobile filters consist of the following elements.
## Mobile UI filters

<table>
<thead>
<tr>
<th>Number</th>
<th>UI element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Current set</td>
<td>Displays the current condition set and the number of records returned by the condition set.</td>
</tr>
<tr>
<td>2</td>
<td>Current rule</td>
<td>Displays the type of rule set that applies to the current filter.</td>
</tr>
<tr>
<td>3</td>
<td>Conditions</td>
<td>Displays the conditions that are part of the current condition set.</td>
</tr>
<tr>
<td>4</td>
<td>Add rule button</td>
<td>Displays a pop-up to add a rule.</td>
</tr>
<tr>
<td>5</td>
<td>New filter button</td>
<td>Creates a condition set.</td>
</tr>
<tr>
<td>6</td>
<td>Delete set</td>
<td>Deletes the current condition set.</td>
</tr>
</tbody>
</table>

### Mobile condition sets

Condition sets generate a list by combining the results of multiple sets of conditions. Use condition sets to create complex filters.

Each set specifies whether a record must match all conditions or whether a record can match any condition.

In general, condition sets requiring matching to all conditions return fewer records than sets that allow matching on any condition within a set.

For example, consider the following filter consisting of two condition sets.

### Sample list filter with two condition sets
In this example, set 1 requires matching all the conditions and returns 8 results. Set 2 allows matching to any condition and returns 29 results. Total results return 37 records for the combination of record results from the individual condition sets.

**View or modify a mobile list in the ServiceNow Classic app**

Modify a list on a mobile interface.

When you create a favorite list, it automatically appears on the mobile homepage. To see the details of a list or selection, tap its icon on the home page.

To modify a list:

1. Tap the list you want to modify.
2. Tap the filter.
3. To add an "And" rule, tap **Add an All Rule**. On an Android device, tap the add button and tap **'All' Rule**.
   a) In the Add All section, click **Add New Condition**.
   b) Select a condition from the list.
   c) From the Select Operator list, select a condition qualifier.
   d) Click **Save**.

4. To add an "Or" rule, tap **Add an Any Rule**. On an Android device, tap the add button and tap **'Any' Rule**.
   a) In the Add Any Rule section, tap **Add New Condition**.
   b) Select a condition from the list.
   c) From the Select Operator list, select a condition qualifier.
d) Click **Save**.

Both sets of conditions appear in the condition filter.

5. To delete a condition, tap the adjacent **×**.
6. Tap **Save**.

**Search a mobile list in the ServiceNow Classic app**

Search in a list on a mobile device.

1. From the mobile homepage or the navigation menu, tap the list you want to search within.
2. In the search field at the top of the screen, type your search terms. On an Android device, access the search field from the List options menu ( ).
<table>
<thead>
<tr>
<th>iPhone search field</th>
<th>Android search field</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Incidents" /></td>
<td><img src="image" alt="Active" /></td>
</tr>
<tr>
<td><img src="image" alt="Search" /></td>
<td><img src="image" alt="Activity Stream" /></td>
</tr>
<tr>
<td>Manager can't access SAP Controlling application</td>
<td><img src="image" alt="INCIDENTS" /></td>
</tr>
<tr>
<td>INC0000051</td>
<td><img src="image" alt="Active = true" /></td>
</tr>
<tr>
<td>Category: Software</td>
<td><img src="image" alt="Search" /></td>
</tr>
<tr>
<td>Can't access SFA software</td>
<td><img src="image" alt="INCIDENTS" /></td>
</tr>
<tr>
<td>INC0000046</td>
<td><img src="image" alt="Active = true" /></td>
</tr>
<tr>
<td>Category: Software</td>
<td><img src="image" alt="Search" /></td>
</tr>
<tr>
<td>Please remove the latest hotfix from my PC</td>
<td><img src="image" alt="INCIDENTS" /></td>
</tr>
<tr>
<td>INC0000027</td>
<td><img src="image" alt="Active = true" /></td>
</tr>
<tr>
<td>Category: Software</td>
<td><img src="image" alt="Search" /></td>
</tr>
<tr>
<td>Can't launch 64-bit Windows 7 virtual machine</td>
<td><img src="image" alt="INCIDENTS" /></td>
</tr>
<tr>
<td>INC0000019</td>
<td><img src="image" alt="Active = true" /></td>
</tr>
<tr>
<td>Category: Software</td>
<td><img src="image" alt="Search" /></td>
</tr>
<tr>
<td><img src="image" alt="Sort" /></td>
<td><img src="image" alt="Sort" /></td>
</tr>
<tr>
<td><img src="image" alt="Add" /></td>
<td><img src="image" alt="Add" /></td>
</tr>
<tr>
<td><img src="image" alt="Star" /></td>
<td><img src="image" alt="Star" /></td>
</tr>
</tbody>
</table>
3. To change the sort order, tap **Sort**.

**Mobile lists for ServiceNow Classic on a tablet**

Lists appear as a single column of records on the mobile tablet UI display. Each row represents a separate record.

Tap a record in the column on the left to see item details.
Mobile lists on a tablet consist of the following elements.
Elements of the mobile UI: Lists

<table>
<thead>
<tr>
<th>Number</th>
<th>UI element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>List name</td>
<td>Displays the name of the list favorite.</td>
</tr>
<tr>
<td>2</td>
<td>Back button</td>
<td>Navigates back to the home page.</td>
</tr>
<tr>
<td>3</td>
<td>UI actions</td>
<td>Displays buttons to display the list activity stream and add a record.</td>
</tr>
<tr>
<td>4</td>
<td>Application search</td>
<td>Displays records that match the search string.</td>
</tr>
<tr>
<td>5</td>
<td>Current filter</td>
<td>Displays the conditions filtering the list.</td>
</tr>
<tr>
<td>6</td>
<td>Records</td>
<td>Displays one row for each record in the list.</td>
</tr>
<tr>
<td>7</td>
<td>List options</td>
<td>Displays list options to add to favorites, share, and sort the list.</td>
</tr>
<tr>
<td>8</td>
<td>Record details</td>
<td>View record details in a split pane view.</td>
</tr>
<tr>
<td>9</td>
<td>Record UI actions</td>
<td>Make a particular record a favorite, share the record, add an attachment, work notes, or comments.</td>
</tr>
</tbody>
</table>

Mobile activity streams in the ServiceNow Classic app for a tablet
List activity streams appear as a single column window with a separate row for each record update in the mobile UI.

Access a list activity stream by pressing the activity stream icon. Activity streams open in the same panel as the list. Select an item to see more details in the right panel.

Elements of the tablet UI: activity stream
<table>
<thead>
<tr>
<th>Issue with email</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spoke to customer, appears to need a new VPN token.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Joe Employee</th>
<th>INC0000047 • 4 mos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Issue with email</td>
<td></td>
</tr>
<tr>
<td>Will deliver new token tomorrow when I'm in the San Diego office.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Joe Employee</th>
<th>INC0000047 • 4 mos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Issue with email</td>
<td></td>
</tr>
<tr>
<td>Can't connect to Email from home. Webmail appears to be down.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Joe Employee</th>
<th>INC0000047 • 4 mos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Issue with email</td>
<td></td>
</tr>
<tr>
<td>Be sure to record VPN ID number in the CMDB.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Issue with email</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spoke to customer, appears to need a new VPN token.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Joe Employee</th>
<th>4 mos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Issue with email</td>
<td></td>
</tr>
<tr>
<td>Will deliver new token tomorrow when I'm in the San Diego office.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Joe Employee</th>
<th>4 mos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Issue with email</td>
<td></td>
</tr>
<tr>
<td>Can't connect to Email from home. Webmail appears to be down.</td>
<td></td>
</tr>
</tbody>
</table>
Activity stream updates consist of the following elements.

**Tablet UI elements: activity stream**

<table>
<thead>
<tr>
<th>Number</th>
<th>UI element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Back button</td>
<td>Return to the list.</td>
</tr>
<tr>
<td>2</td>
<td>User</td>
<td>Displays the user photo and name of the user who made the activity update.</td>
</tr>
<tr>
<td>3</td>
<td>Record details</td>
<td>Displays the record number and relative time since the update was made.</td>
</tr>
<tr>
<td>4</td>
<td>Activity details</td>
<td>Displays the most recent activity update.</td>
</tr>
</tbody>
</table>

Swipe up or down to see more activity stream updates.

*Filters for ServiceNow Classic on a tablet*

Construct complex filters with the ServiceNow Classic mobile app UI.
Mobile filters consist of the following elements.

### Mobile UI filters

<table>
<thead>
<tr>
<th>Number</th>
<th>UI element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Current set</td>
<td>Displays the current condition set and the number of records returned by the condition set.</td>
</tr>
<tr>
<td>2</td>
<td>Current rule</td>
<td>Displays the type of rule set that applies to the current filter.</td>
</tr>
<tr>
<td>3</td>
<td>Conditions</td>
<td>Displays the conditions that are part of the current condition set.</td>
</tr>
<tr>
<td>4</td>
<td>Add rule button</td>
<td>Displays a pop-up to add a rule.</td>
</tr>
<tr>
<td>5</td>
<td>New filter button</td>
<td>Creates a condition set.</td>
</tr>
<tr>
<td>6</td>
<td>Delete set</td>
<td>Deletes the current condition set.</td>
</tr>
</tbody>
</table>

Not all filters are available on the ServiceNow Classic mobile app.

### Mobile favorites in the ServiceNow Classic app

Mobile favorites provide links to records in the system. Favorites display as icons on the homepage in the ServiceNow Classic mobile app.
Note: There are a few limitations for mobile favorites:
- You cannot disable the favorite icon on mobile lists or forms.
- Fixed Query modules are not supported and do not appear on a mobile device even when selected as a favorite.
- Favorites containing `home\.do` are not supported and do not appear on mobile devices.

Elements of the mobile UI home

<table>
<thead>
<tr>
<th>Number</th>
<th>UI element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>User profile button</td>
<td>Displays the current profile record.</td>
</tr>
<tr>
<td>2</td>
<td>Favorites</td>
<td>Links to records in the instance, sometimes with visualizations.</td>
</tr>
<tr>
<td>3</td>
<td>Application navigator button</td>
<td>Displays the list of menus and modules available to the current user.</td>
</tr>
</tbody>
</table>

Long press a favorite to edit the following properties.
- Label
- Text and icon color
- Icon image
- Enable visualizations for some favorites

Favorites automatically synchronize between the mobile UI and the desktop UI for your instance.

Add a mobile favorite

Make any list, record, or module a favorite to add it as an icon to the mobile favorites page.
1. Navigate to the screen that you want to add as a favorite.
2. Tap the star icon.
3. From the Customize Favorite page, give the favorite a name, tag color, and icon. Then tap Save.
Edit a mobile favorite

Make changes to the appearance of a favorite on a mobile favorites page.

1. On the app home page, long press a favorite icon.
2. From the icon customization screen, change the icon name, color, or icon.
3. To delete a favorite, long press the icon then on iOS tap Delete or on Android tap the delete icon.

Add a Visualization to a favorite

Add visualizations to mobile favorites. Visualizations are specific to your device. If you add a visualization to a list on one mobile device, it won't display the same way on another device.
Visualizations allow users on the mobile app UI to see relevant table information directly from the home page favorite icon.

1. In the mobile app homepage, long press the favorite you want to add a visualization to.
2. Tap the **Visualizations** tab.
3. Select a visualization from the available options. Visualization options vary depending on the type of favorite you select.
Visualizations tab
Visualization options
Use visualizations to customize the appearance of favorites on the mobile app homepage

Favorite visualization options

<table>
<thead>
<tr>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Favorite icons displays the number of records returned by the list filter.</td>
<td><img src="assigned-to-me.png" alt="Count" /></td>
</tr>
<tr>
<td>Favorite icons displays the number of records with a particular field value. Select the field values you want to display when you set up the visualization.</td>
<td><img src="assigned-to-me.png" alt="Bar chart" /></td>
</tr>
</tbody>
</table>

Using the ServiceNow Classic mobile app

Access an instance from the mobile interface using the ServiceNow Classic mobile app.

Download the ServiceNow Classic mobile app from the Apple App Store for devices running iOS 9 and above or from the Google Play store and for Android phones running version 4.4 (KitKat) and above.

Use the ServiceNow Classic mobile app to do the following:

- Access lists and forms.
- Save favorite lists and records to the app homescreen.
- Access the Service Catalog from your mobile device.
- Communicate with other users within the platform using Connect Chat for mobile.
- Share your location.

The mobile UI is configurable by your administrator.

**Manage notification settings in the ServiceNow Classic mobile app**

In the ServiceNow Classic mobile app, use Notification Settings to enable or disable your notifications and the channels for receiving them.

Role required: none

You can set additional notification preferences, such as conditions or filters that affect notification delivery, through the System Settings on a desktop instance or mobile web browser. For details, see Setting notification preferences in UI16.

1. On your profile screen, tap Notification Settings.
2. On the Notifications Settings screen, enable or disable your notifications.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Notifications</td>
<td>Global switch for enabling or disabling all your notifications.</td>
</tr>
</tbody>
</table>
|                         | ![Note: The system does not disable any notifications configured as mandatory by your administrator.](image)
| Notification channels   | Switches for enabling or disabling the notifications that you receive by device. |
| Notification categories | List of notification categories. Each category identifies and groups related notifications. To select a notification:
|                         | 1. Tap the notification category.
|                         | 2. Tap the notification to be updated.
|                         | 3. Enable or disable the channels for that notification.                     |

3. Tap the back button (arrow) to navigate back to your profile screen.

**Mobile Connect Chat in the ServiceNow Classic app**

Use Connect Chat to communicate with coworkers in the ServiceNow Classic mobile app.

Activate Connect before using chat in the ServiceNow Classic mobile app. Many, but not all the features supported in the desktop interface are available in the mobile app UI. Administrators can show or hide Connect Chat by configuring it by role.

Connect Chat is not supported on the mobile web although some limited functionality exists.
### Elements of the mobile UI: Connect Chat

<table>
<thead>
<tr>
<th>UI element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Back button</td>
<td>Returns to the previous screen.</td>
</tr>
<tr>
<td>Edit button</td>
<td>Turns on conversation editing mode, in which you can delete conversations or mark them as read. Deleting a conversation only removes it from the list. Chat history is preserved and you can add the conversation back to your list using the new conversation button.</td>
</tr>
<tr>
<td>New conversation</td>
<td>Opens a new conversation to which you add one or more members.</td>
</tr>
<tr>
<td>Filter</td>
<td>Enables you to filter conversations by member name.</td>
</tr>
<tr>
<td>Conversations</td>
<td>Lists your open conversations. You can view All your conversations or only conversations with Unread messages.</td>
</tr>
<tr>
<td>Application navigator button</td>
<td>Displays the list of menus and modules.</td>
</tr>
</tbody>
</table>

**Mobile Connect Chat conversations**

Have conversations with groups of people in Connect in the ServiceNow Classic mobile app.
Have you seen this?

**ITIL User**
System Administrator has been added to the group

**System Administrator**
that's not your car is it??

**ITIL User**
No! So glad I parked on the street today!

_incident.do?sys_id=23d487264f8c12002fa02f1e0210c746&sysparm_record_target=incident&sysparm_record_row=2&sysparm_record_rows=32&sysparm_record_list=active%3Dtrue%5EORDERBYDESCNumber

Parking garage flooded! INC0010004
Elements of the mobile UI: Connect Chat conversation

<table>
<thead>
<tr>
<th>UI element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conversation name</td>
<td>Displays the name of the conversation.</td>
</tr>
<tr>
<td>Conversation details</td>
<td>Opens the conversation details page, which displays the following information.</td>
</tr>
<tr>
<td></td>
<td>• Record details (record conversations only)</td>
</tr>
<tr>
<td></td>
<td>• Push notification preferences</td>
</tr>
<tr>
<td></td>
<td>• Conversation members</td>
</tr>
<tr>
<td></td>
<td>For group and record conversations, the conversation details page also provides capabilities to add or remove conversation members and to leave the conversation.</td>
</tr>
<tr>
<td>Avatar</td>
<td>Displays an image or initials to represent a user. Tap an avatar to view details about the user, including email address and online presence status.</td>
</tr>
<tr>
<td>Attachment button</td>
<td>Enables you to include attachments in a message, including photos and documents.</td>
</tr>
<tr>
<td>Message field</td>
<td>Enables you to enter and send messages.</td>
</tr>
</tbody>
</table>

Note: For record conversations, all messages are sent as comments. Sending a message as a worknote is not supported.

Mobile device location on the ServiceNow Classic app

Administrators can set up the ServiceNow Classic mobile app UI to request a current location for a mobile device and store GPS coordinates in a string field.

Take advantage of mobile device tracking features such as GPS location by configuring a string field to use the current_location dictionary attribute. This attribute causes the ServiceNow Classic mobile app UI to display a special icon to request the current location of the mobile device.
Elements of the mobile UI: current location field

GPS coordinates are stored in the string field. Administrators can assign any label they want to this field.  

Get current location allows the app to access the current location of your mobile device.  

Current location field asks for permissions to use your current location the first time you use that option.  

Change the app access to location from your device Settings.  

Mobile barcode scanning on the ServiceNow Classic app

Administrators can set up the mobile app UI to request access to a mobile device camera to scan and store barcodes in a string field.  

Take advantage of mobile device barcode scanning features by configuring a string field to use the barcode dictionary attribute. This attribute causes the mobile app UI to display a special icon to request access to the mobile device camera.
Elements of the mobile UI: barcode scanning field

String field displays and stores the numeric value for the barcode. Administrators can assign any label they want to this field.

Scan barcode opens the mobile device camera to scan a barcode.

When you open the barcode scanner for the first time, the app asks your mobile device for permissions to access the camera.

The native apps for iOS and Android support scanning the following barcode types:

- **2D barcodes**: QR Code, Data Matrix, PDF-417, AZTEC
Support for ServiceNow Classic

Specific devices and operating systems are supported by the ServiceNow Classic mobile app and by the mobile browser interface.

### Devices supported by the ServiceNow Classic mobile app interface

<table>
<thead>
<tr>
<th>Device</th>
<th>Supported versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apple® iPhone®</td>
<td>Version 6.0 and higher of the app for iOS requires iOS 11 and above for all models of iPhone. Previous versions of the app support all models of iPhone running iOS 9 and above.</td>
</tr>
<tr>
<td>Apple® iPod touch</td>
<td>Version 6.0 and higher of the app requires iOS 11 and above for all models of iPod. Previous versions of the app support all models of iPod running iOS 9 and above.</td>
</tr>
<tr>
<td>Apple® iPad®</td>
<td>Version 6.0 and higher of the app requires iOS 11 and above for all models of iPad. Previous versions of the app support all models of iPad running iOS 9 and above.</td>
</tr>
<tr>
<td>Apple® watch</td>
<td>All models of watch running iOS 2 and above</td>
</tr>
<tr>
<td>Android™ mobile devices</td>
<td>All models of Android phone running KitKat (4.4) and above.</td>
</tr>
</tbody>
</table>

### Limitations

The platform does not have a native BlackBerry or Windows phone app. Use the mobile web experience on these devices.

The mobile apps do not give access to dashboards as dashboards are not optimized for mobile screen sizes. You can access dashboards on a tablet using the standard web interface.

### Devices supported by the mobile browser interface

<table>
<thead>
<tr>
<th>Device</th>
<th>Supported versions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apple iPhone®</td>
<td>All models of iPhone running iOS 7 and above. Safari is supported.</td>
</tr>
<tr>
<td>Apple® iPod®</td>
<td>All models of iPod running iOS 7 and above. Safari is supported.</td>
</tr>
<tr>
<td>Apple® iPad®</td>
<td>All models of iPad running iOS 9 and above are supported. Safari® is supported.</td>
</tr>
<tr>
<td>Android™</td>
<td>Android version 4.0/Ice Cream Sandwich and above. Use the latest available version of Chrome to access the mobile browser interface on Android devices. Native browsers and older versions of Chrome support major interface functionality, but have some known issues.</td>
</tr>
<tr>
<td>Device</td>
<td>Supported versions</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Android® tablet</td>
<td>Any Android tablet running 4.4 (KitKat) and above with the latest available version of Chrome™. Native browsers and older versions of Chrome support major tablet interface functionality, but have some known issues. Tablet UI is optimized for the iPad form factor and resolution. Not all Android tablets have the same form factor and resolution. Gesture-based zoom functionality is not supported.</td>
</tr>
<tr>
<td>BlackBerry®</td>
<td>All BlackBerry devices running BlackBerry 10 and above. Some configuration is required.</td>
</tr>
</tbody>
</table>

In UI16, browsers on a tablet use the same interface as the desktop browser. Mobile browsers do not support the UI16 interface. Instead they use the same interface as the ServiceNow Classic mobile app.

**Note:** Do not use the mobile interface on a desktop browser except for testing purposes.

**Mobile browser limitations**

The iOS version of Firefox does not support Now Community or other Service Portal-based pages.
Index

A

action function 232
action item 232
AirWatch EMM 89
AirWatch MDM 89
AirWatch mobile device management 89
applet 232
offline mode 579
Offline Mode 577
parameterized action functions 483
parametrized navigation function 227
URL applet 433
mobile 3D touch 633
mobile advanced list customization 609
mobile app
configure Connect 625
supported devices 679
mobile app UI
blur 546
Connect Chat 672
mobile authentication 601
mobile browser interface
back navigation 626
shortcut 640, 640
UI actions 626, 626
UI policies 632
mobile button instances 505
mobile classic
default home page customization 619
field status indicators 610
navigator 603
Mobile Client GlideForm (g form) scripting 628
mobile configuration
list search fields 613
mobile customization 604
mobile data item
parameters 300
mobile device management
add app XenMobile 101
Citrix managed configuration 102
Citrix XenMobile 101
Jamf Pro 116
mobile device management (MDM) 83
mobile device management Intune 104
mobile device options 183
mobile experience
setup for administrators 202, 600
mobile force touch 633
mobile functions 505
mobile hide filters 619
mobile home page fields 620
mobile home page modules 621
mobile home page sections 621
mobile interface
access 638
client scripts 627
disable 636
home page
favorite 668
navigating applications 642
mobile list
display title 610
mobile list view 605
mobile lists 611
mobile localization 590

B

BlackBerry MDM 93
BlackBerry mobile device management 93
BlackBerry UEM MDM 93
BlackBerry Unified Endpoint Management mobile device management 93

C

components and architecture 72, 595
configure a mobile theme 622
configure mobile classic location and barcode scanning 635
customize mobile homepages 619

data item parameters 302
display title 610
discover search 656
discovery title 606

E

e-signature for mobile approvals 602
technology mobility management (EMM) 83

event data item
parameters 300

event data item
parameters 300

F

functions 232

I

IBM MaaS360 EMM 102
IBM MaaS360 enterprise mobility management 102
IBM MaaS360 iOS distribution 102
IBM MaaS360 MDM 102
IBM MaaS360 mobile device management 102
Intune mobile device management 104

M

mobile
action 208
action item parameters 478
action tutorial 208
button actions 473
configure employee directory applet 417
data item parameters 302
home page collections 619
list applet 331
list search 656
list title 606
map applet 420
navigation functions 490
mobile log in 638
mobile multifactor authentication 601
mobile navigation
parameters 219
mobile parameterized action 208
mobile parameters 232
mobile parametrizations 203
mobile push notifications 513
mobile security
architecture 72, 595
components 72, 595
components and architecture 72, 595
controls and practices 599
data flow retrieval 597
data security 598
identity and access management 72, 596
information storage 75, 597
internal mobile app distribution 598
security controls 78, 599
security patching 600
session length 597
session length and timeout 597
shared data 600
timeout 597
user authentication 72, 596
user data collection 600
user termination 597
mobile SSO 602
mobile studio
applet 319
create action item 477
smart button function 502
mobile Studio
action functions 480
mobile tested devices 121
mobile theme configuration 622
mobile tutorial
applet parameter configuration 204
data item with parameters 204
parameterized navigation 219
mobile UI
activity streams 648, 661
application navigator 642
barcode scanning 677
condition sets 653
Connect Chat
conversations 674
device location 676
favorites 665
filters 652, 663
get started for end users 637
home page
favorite 667
list 654
list configuration 616
lists 646, 659
mobile app 671, 672
visualizations 668, 671
mobile UI styles 449, 451
MobileIron MDM 117
MobileIron mobile device management 117

N
navigation function 232
notification settings 672

O
offline mode
mobile 579
Offline Mode
mobile 577

S
ServiceNow Agent 202, 461
set up
  mobile experience admin 202, 600
smart button function 232
smartphone interface
  blackberry 642

T
tablet app
  application navigator 644
tablet support 184