

[Accessibility](#) / Performing accessibility testing for your app

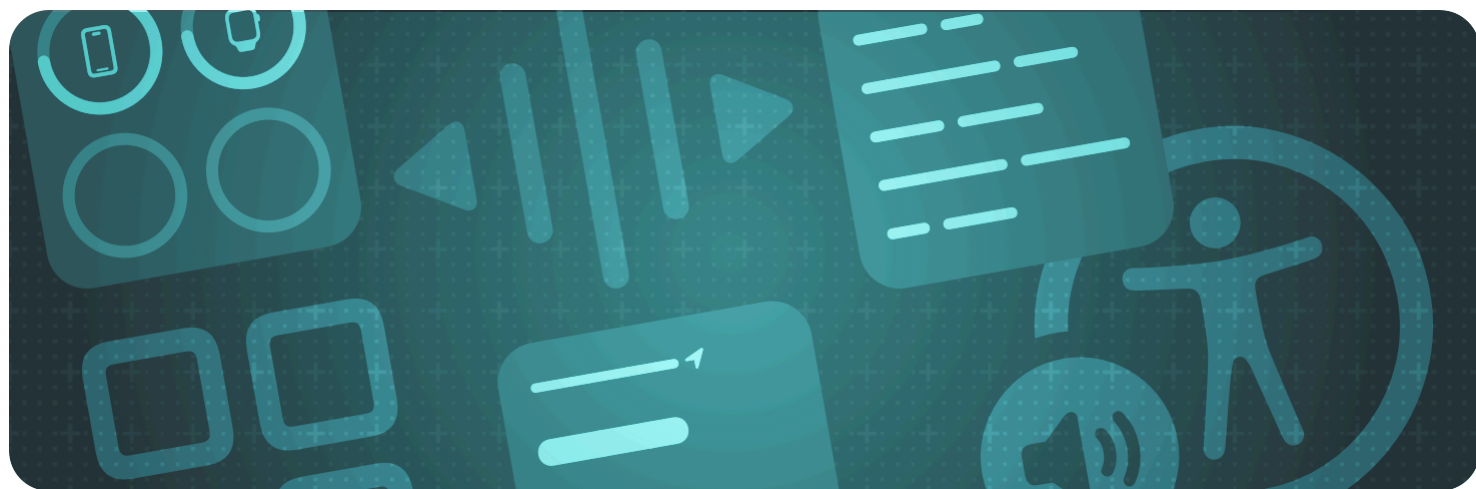
Article

Performing accessibility testing for your app

Test your app with accessibility settings and assistive technologies to discover and address accessibility issues.

Overview

It's always a good idea to experience your app from the perspective of the people using it. When you test your app from the user's perspective, make sure you also test with different accessibility settings and assistive technologies like VoiceOver, Voice Control, and Switch Control so you can experience the app in the same way as people who rely on these features. Testing your app with accessibility settings reveals which tasks are possible to complete, which elements are accessible and which aren't, and whether your navigation is clear and logical for every use case.



Identify the main tasks in your app

Before you begin accessibility testing, compile a list of the main tasks that a person can perform on each screen in your app. For example, a to-do list app might contain several screens with one

or more possible tasks as described in the following list.

First launch experience

The user may need to complete or opt out of an onboarding flow.

Login screen

The user may need to choose a login service, enter a username and password, interact with a login button, request a password reset, or enter information to create a new account.

Empty to-do list

The user may need to create a new task, or open app settings.

To-do list with tasks

The user may need to mark a task as complete, undo marking a task as complete, or create a new task.

New task screen

The user may need to enter information about the task, save the task to the list, or exit the screen without saving the task.

Settings screen

The user may need to adjust app settings, or exit the screen without changing anything.

Create an accessibility testing matrix

After you identify your app's main tasks, choose which devices, accessibility settings, and assistive technologies to test those tasks with.

It's recommended to test your app on each type of device your app supports (for example, iPhone, iPad, and Mac). Testing on each of those devices lets you experience the app in the same way as a person who downloads and installs it. Testing on device can also highlight subtle differences in your app's user experience across platforms.

After you prepare your devices for testing, identify which accessibility features to test. As a starting point, it's recommended that you test your app for the following accessibility categories, although you're encouraged to test with additional settings and technologies if you can:

- Visual accessibility for accessibility settings related to color, text, motion, and more
- Media accessibility for accessibility settings related to captions, audio descriptions, audio transcripts, and more
- Assistive technologies such as VoiceOver, Voice Control, Switch Control, and more

The following sections describe how to test your app for visual accessibility, media accessibility, and several key assistive technologies.

Test with accessibility settings

To test with accessibility settings, turn on each of the following accessibility features in Settings > Accessibility, one at a time. Work through your testing matrix by completing the main tasks in your app while each accessibility setting is on.

Visual accessibility

Media accessibility

- Display & Text Size > Bold Text
- Display & Text Size > Larger Text > Larger Accessibility Sizes
- Display & Text Size > Button Shapes
- Display & Text Size > On/Off Labels
- Display & Text Size > Reduce Transparency
- Display & Text Size > Increase Contrast
- Display & Text Size > Differentiate Without Color
- Display & Text Size > Color Filters > Red/Green Filter, Green/Red Filter, Blue/Yellow Filter
- Motion > Reduce Motion
- Motion > Dim Flashing Lights

Test with assistive technologies

To test with assistive technologies, set up and turn on each of the following assistive technologies, one at a time. Work through your testing matrix by completing the main tasks in your app using only that assistive technology.

VoiceOver

Voice Control

Switch Control

Assistive Access

- Install your app on a physical device, since VoiceOver isn't available on Simulator.
- Learn how to turn VoiceOver off so you can exit it after testing. For example, in iOS, you can set up [Accessibility Shortcut](#) to turn VoiceOver on and off. In macOS, you can press Command-F5.
- Adjust your device volume to an appropriate level.
- Navigate to Settings > Accessibility > VoiceOver, and turn on VoiceOver.
- Adjust the Speaking Rate slider to an appropriate speaking pace.
- If you provide hints for any accessible elements in your app, make sure the Verbosity > Speak Hints setting is on.

With VoiceOver on, use VoiceOver gestures to navigate your app. Although you don't need to become an expert VoiceOver user to test your app with it, you do need to know several key gestures, which have slight variations across platforms. For example, the following gestures apply to devices with touchscreen and trackpad input.

Gesture	Action
Tap	Select and speak an item.
Swipe right or left	Select the next or previous item.
Double tap	Select and speak an item.
Two-finger tap	Pause or continue speaking.
Two-finger swipe up	Speak the entire screen from the top.
Three-finger triple tap	Turn the screen curtain on or off.

To begin testing, turn on the *screen curtain* to replicate the experience of someone who is solely relying on VoiceOver. When the screen curtain is on, the screen contents are active even though the display is black and appears to be turned off.

Tip

A great place to practice VoiceOver gestures is in the VoiceOver Practice area in Settings > Accessibility > VoiceOver. VoiceOver needs to be on for the VoiceOver Practice button to appear.

To learn more about VoiceOver gestures, read the platform-specific user guides listed in [VoiceOver](#).

Diagnose and address accessibility issues

As you test your app's workflows, make sure your app continues to provide a good user experience while accessibility settings and assistive technologies are on. Check that you can access every element and that the ordering of those elements is what you intend. Make note when you find it difficult to perform a task. Confirm that your UI adjusts appropriately when the system font size, color filter, or other visual settings change to something other than the default. You can use the following list as a starting point for ensuring that your app provides a good experience in various accessibility categories.

- A user with low vision can perceive all UI elements, images, and text in your app.
- Your app doesn't use color alone to convey information.
- Your app uses sufficient contrast for important UI elements.
- Your app responds to Dynamic Type accessibility text sizes by reflowing its layout, avoiding text truncation, and making sure UI elements don't overlap for larger sizes.
- Your app responds to the Increase Contrast accessibility setting by increasing color contrast between background and foreground elements.
- Your app responds to the Button Shapes accessibility setting to indicate where button controls exist.
- Your app responds to the Reduce Motion accessibility setting by reducing excessive motion where appropriate.
- Your app responds to the Dim Flashing Lights accessibility setting by reducing flickering in the UI or videos.

Take notes about any issues as you discover them so you can address them in your app's design and implementation. If you encounter accessibility issues that you aren't able to troubleshoot while testing on the device, try using [Accessibility Inspector](#) to help diagnose and resolve issues. You can also automate accessibility testing by adding accessibility audits in your UI tests, as described in [Perform accessibility audits for your app](#).

Address accessibility issues that you discover during testing so that you're able to successfully complete all the tasks in your app when accessibility settings and assistive technologies are on.

See Also

Essentials



Accessibility updates

Learn about important changes to Accessibility.



Accessibility

Accessible user interfaces empower everyone to have a great experience with your app or game.

