

[ActivityKit](#) / Launching your app from a Live Activity

Article

Launching your app from a Live Activity

Use deep links to enable people to open your app's scene that matches the data of your Live Activity.

Overview

People can tap a Live Activity to launch your app. To let them view more content or make changes to an ongoing task, open your app to the scene that matches the content of the Live Activity.

Respond to the system's default launch behavior

If you don't explicitly provide a deep link into your app, the system opens your app and passes an [`NSUserActivity`](#) to [`onContinueUserActivity\(_:perform:\)`](#), [`application\(_:continue:restorationHandler:\)`](#), or [`application\(_:continue:restorationHandler:\)`](#). Implement the appropriate callback and check whether the [`NSUserActivity`](#) object's [`activityType`](#) is [`NSUserActivityTypeLiveActivity`](#). Then, add code to open a screen in your app that fits the context of the active Live Activity.

Important

In CarPlay, tapping your Live Activity only launches your app if it supports CarPlay. For more information about adding CarPlay support to your app, refer to [CarPlay](#).

Add deep links to your app

With deep links, you specify an URL that directly launches a specific scene in your app, and choose different scenes for each Live Activity presentation:

- To create a deep link into your app from the Lock Screen, compact leading, compact trailing, and minimal presentations, use `widgetURL(_:_)`. When the compact leading and trailing presentations are visible, make sure both link to the same screen in your app.
- To create a deep link into your app from the extended presentation, use `widgetURL(_:_)` or SwiftUI's [Link](#).

Configure how your Live Activity launches your app in watchOS

Live Activities automatically appear in the Smart Stack on Apple Watch, and the system shows the Live Activity more prominently on Apple Watch when it receives updates with an alert. When a Live Activity receives an update:

- If a person doesn't actively use an app on Apple Watch, the system displays an alert that uses the compact presentation's leading and trailing views, and then launches the Smart Stack, displaying the Live Activity at the top of the stack.
- If a person actively uses any app on Apple Watch, the system shows an alert to let them know about the updated Live Activity.

Tapping the alert displays the Live Activity on Apple Watch and the option to launch the app on iPhone. You can opt-in to give people the ability to open your Watch app from your Live Activity. In the Build Settings for your watchOS app target, add the `Supports Launch for Live Activity Attribute Types` key in the `Info.plist` section. To launch your Watch app for all your Live Activities, leave its value empty. To launch for specific Live Activities, add an item for each `ActivityAttributes` conforming type that launches your watchOS app.

See Also

User interface

- 📄 [Creating custom views for Live Activities](#)
Create reusable custom views and layouts that support each Live Activity presentation.
- 📄 [Adding accessible descriptions to widgets and Live Activities](#)
Describe the interface elements of your widgets and Live Activities to help people understand what they represent.