

[Xcode](#) / [Capabilities](#) / Adding capabilities to your app

Article

Adding capabilities to your app

Configure your target to include and customize capabilities that provide access to Apple's app services.

Overview

A *capability* grants your app access to an *app service* that Apple provides, such as CloudKit, Game Center, or In-App Purchase. To use some app services, you must provision your app, adding a capability with Xcode's project editor that configures the app service correctly for you. Xcode edits the Entitlements and Information Property List files, adds related frameworks, and configures your signing assets.

However, some app services — such as Game Center and In-App Purchase — require additional configuration in App Store Connect and your developer account. For example, you need to upload a geographic coverage file using App Store Connect for an app that uses Maps to provide directions for other apps.

The platform, and whether you're a member of the Apple Developer Program, may limit the capabilities available to your app. For the supported capabilities, go to the Reference section of Developer Account Help — for example, go to Supported capabilities (iOS) for the capabilities available to iOS apps.

Before you begin, add your Apple Account and assign the project to a team so that Xcode can provision your app. For iOS, iPadOS, tvOS, visionOS, and watchOS apps, run your app on a device to register the device and create a development provisioning profile.

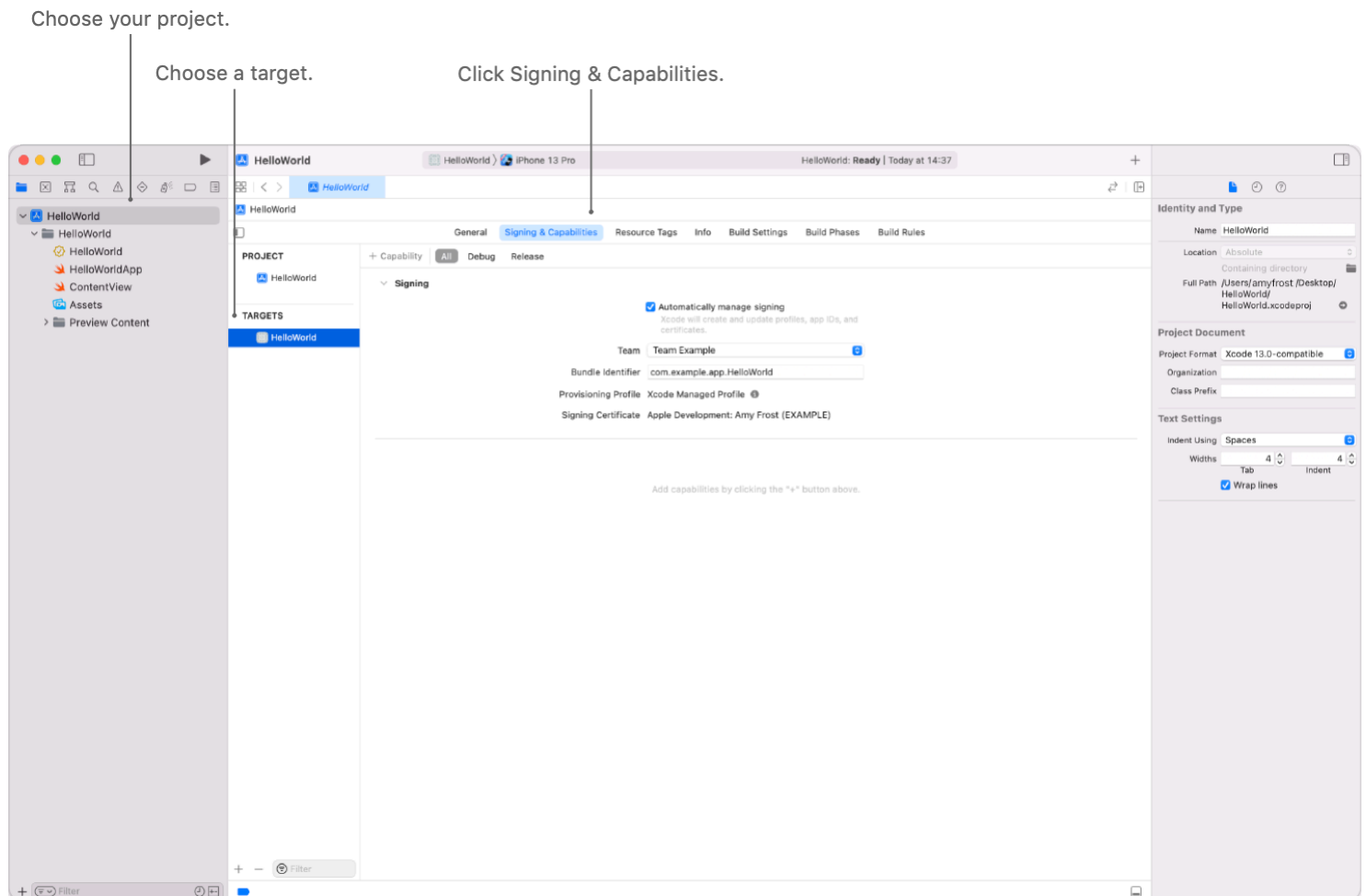
Important

Use the default automatic signing when you create a project from a template. If you manually sign your app, you need to perform the capability configuration steps yourself.

Add a capability

You add capabilities to your app using the Signing & Capabilities pane of the project editor.

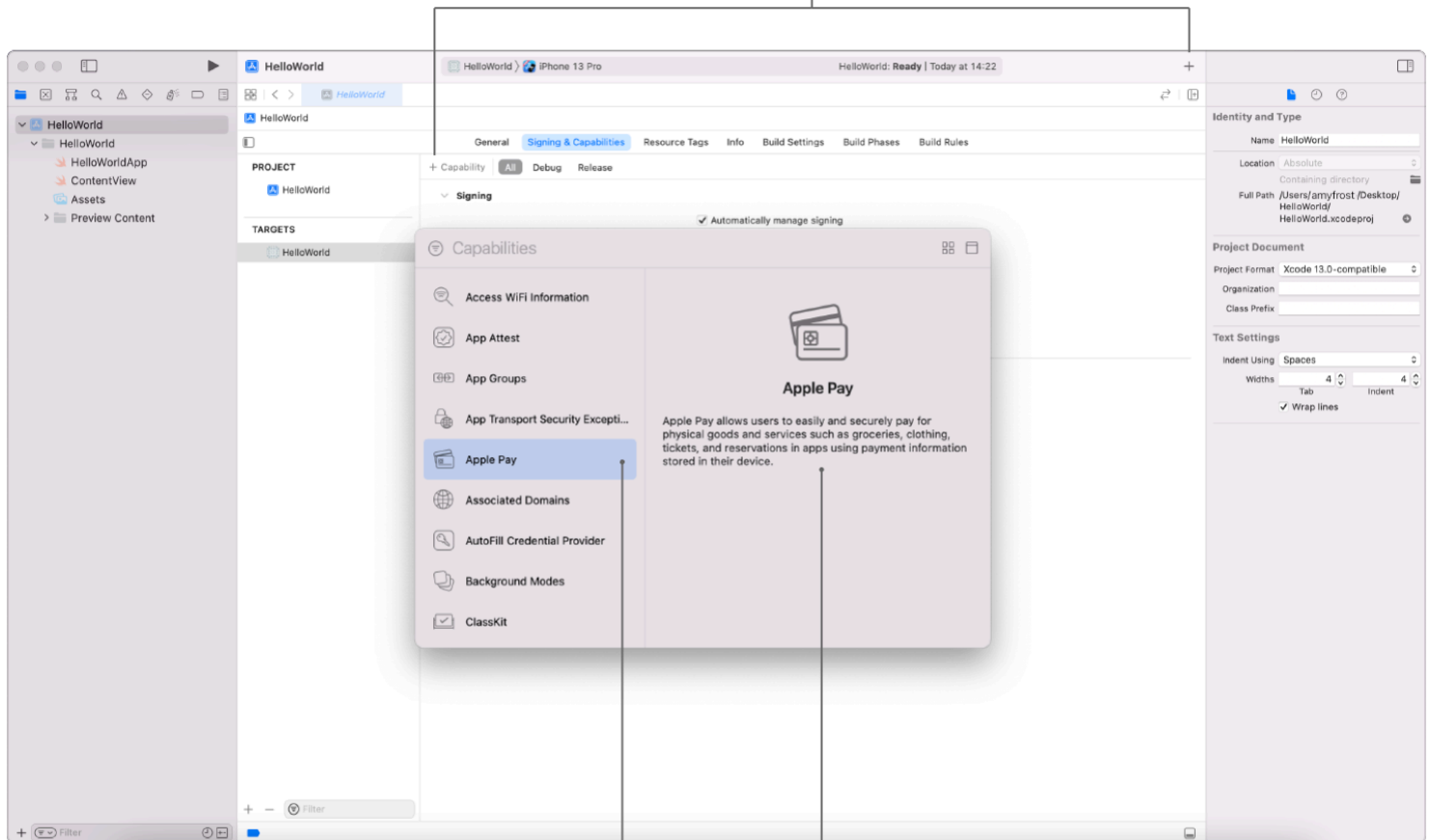
In the Project navigator of the main window, select the project — the root group with the same name as your app — and in the project editor that appears, select the appropriate target and then click the Signing & Capabilities tab.



Optionally, select a build configuration (All, Debug, or Release). For example, if you want to add the capability to the Debug configuration only, select Debug; otherwise, select All.

In the toolbar, click the Library button (+) to open the Capabilities library. Alternatively, click + Capability to the left of the build configurations, or choose Editor > Add Capability. The Capabilities library displays only the capabilities available to the target platform and your program membership. Select a capability in the list to view its description on the right.

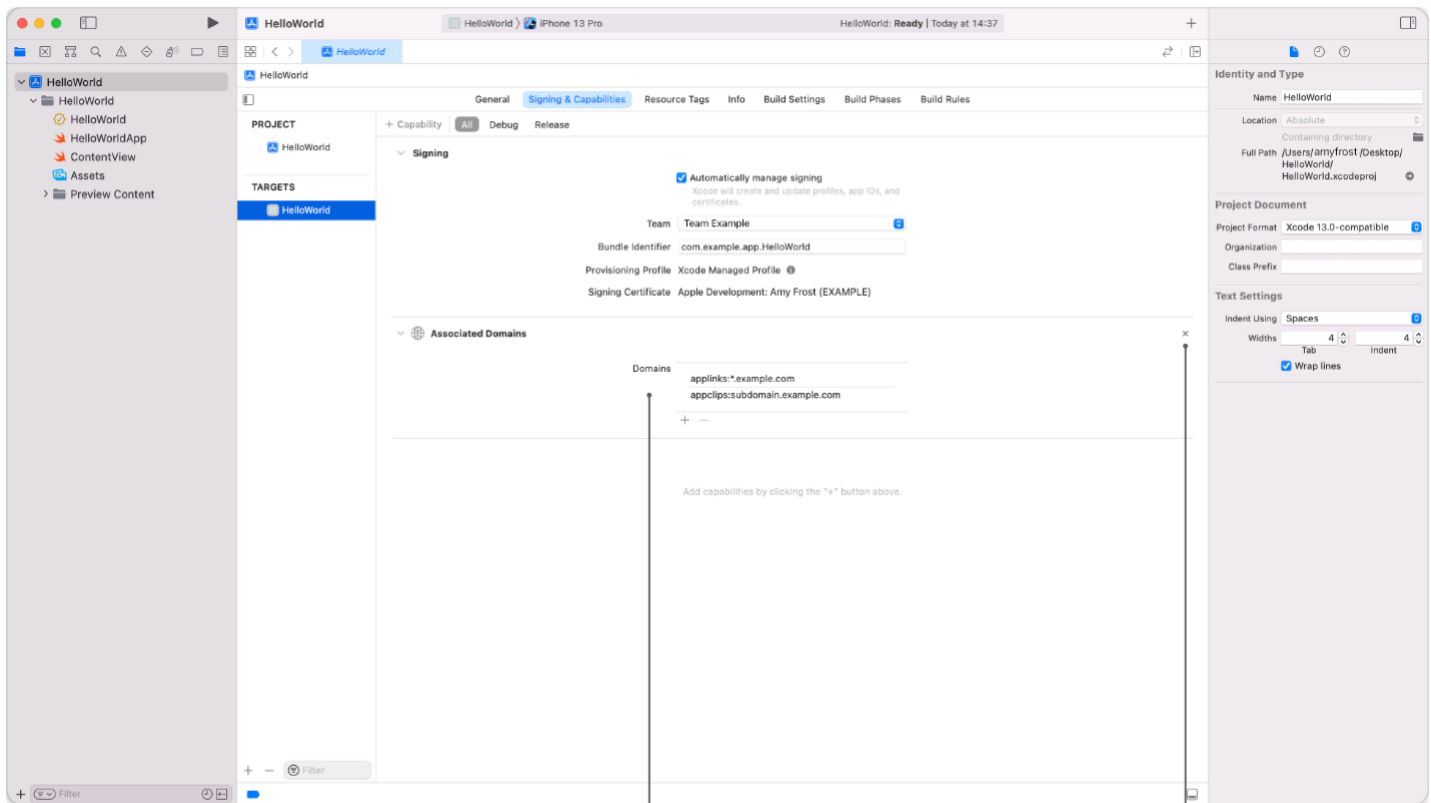
Open the Capabilities library.



Read a description of the selected capability.

Double-click the capability to add it to your project.

To add a capability to the app target, double-click the capability in the library or drag the capability from the library to the Signing & Capabilities pane. The capability appears below the Signing section. If there are more configuration steps, the capability expands to show additional controls (see [Perform additional configuration steps](#) below). To remove a capability, click the Delete (x) button in the upper-right corner of the capability in the Signing & Capabilities pane.



Configure the capability.

Remove the capability.

If errors appear in the Signing section, read the message, correct the problem, then click Try Again. For example, the bundle ID (`CFBundleIdentifier`) that appears in the Bundle Identifier field under Signing needs to be unique. The default value for the bundle ID is the organization identifier concatenated with the app name that you enter when creating a project.

Perform additional configuration steps

For some capabilities, you may need to perform additional configuration steps in Xcode, your developer account, or App Store Connect. For other capabilities, you may need to write some code.

For more guidance on specific capabilities, see the table below:

| Capability | Additional information |
|--------------------|--|
| App Groups | Configuring app groups |
| App Sandbox | Configuring the macOS App Sandbox |
| Apple Pay | Configuring Apple Pay support |
| Associated Domains | Configuring an associated domain |
| Background Modes | Configuring background execution modes |

| Capability | Additional information |
|---------------------------|---|
| ClassKit | <u>Enabling ClassKit in your app</u> |
| Fonts | <u>Configuring custom fonts</u> |
| Game Controllers | <u>Configuring game controllers</u> |
| Group Activities | <u>Configuring Group Activities</u> |
| Hardened Runtime | <u>Configuring the hardened runtime</u> |
| HealthKit | <u>Configuring HealthKit access</u> |
| HomeKit | <u>Configuring HomeKit access</u> |
| iCloud | <u>Configuring iCloud services</u> |
| In-App Purchase | <u>Configuring in-app purchases</u> |
| Keychain Sharing | <u>Configuring keychain sharing</u> |
| Maps | <u>Configuring Maps support</u> |
| Media Device Discovery | <u>Configuring media device discovery</u> |
| Network Extensions | <u>Configuring network extensions</u> |
| On-Demand Install Capable | <u>Creating an App Clip with Xcode</u> |
| Push Notifications | <u>Registering your app with APNs</u> |
| Sign in with Apple | <u>Configuring Sign in with Apple support</u> |
| Siri | <u>Configuring Siri support</u> |
| Wallet | <u>Configuring Wallet support</u> |