

[GameKit](#) / Initializing and configuring Game Center

Article

Initializing and configuring Game Center

Enable Game Center, configure features, and test them locally in your Xcode project.

Overview

Game Center is a social gaming network that helps players discover your games across Apple devices. Players can track their scores on leaderboards, view achievement progress, participate in challenges, and more. Before you can use the GameKit framework and access Game Center data, you need to enable Game Center. When you enable Game Center in your project, Xcode adds the Game Center entitlement to your App ID.

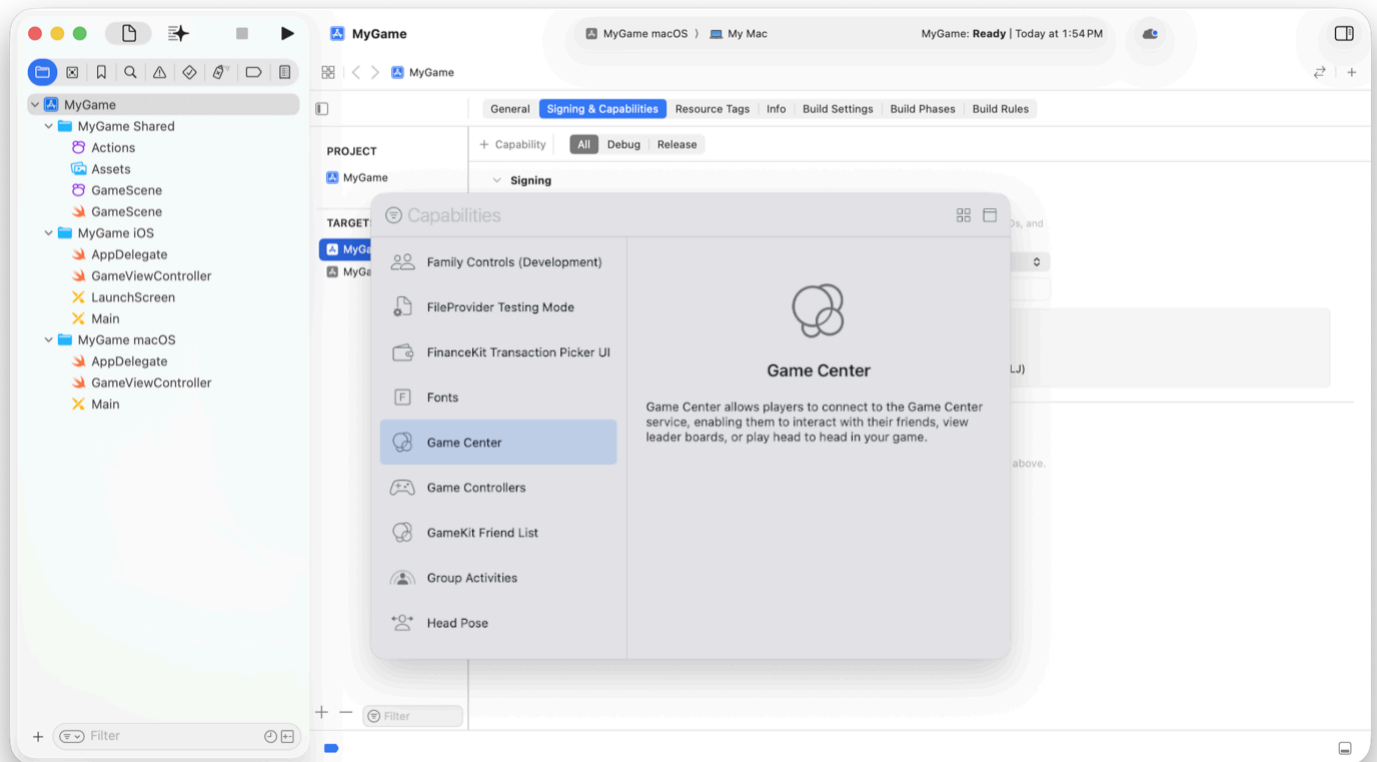
After you enable Game Center, you can configure the features you want to use in your game — like leaderboards, achievements, activities, and challenges — right in Xcode or in App Store Connect.

For more information about the Game Center service, see [Overview of Game Center](#).

Enable Game Center

Add the Game Center capability to your project, so you can configure gameplay features for your game:

1. In Xcode, select the project in the main window.
2. Select the target in the project editor that appears on the right.
3. Click the Signing & Capabilities tab.
4. Click the add capability button (+ Capability) in the toolbar.
5. Add the Game Center capability by double-clicking on it.



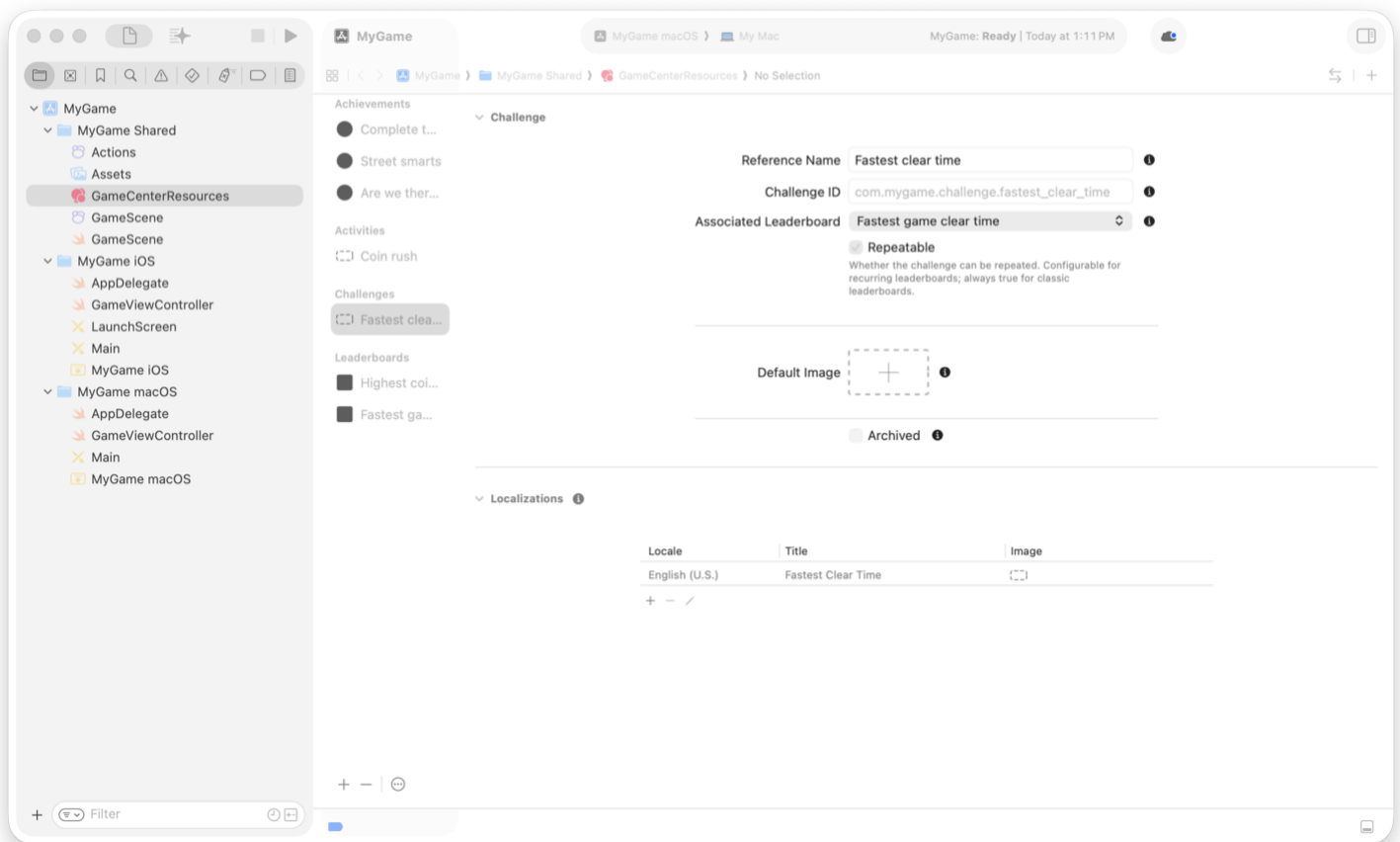
For Mac targets, ensure that you enable both Incoming Connections and Outgoing Connections in the App Sandbox capability under Network.

Configure Game Center features

Use a GameKit bundle file to configure achievements, activities, challenges, leaderboards, and leaderboard sets. You need to configure these features in Xcode before you can access them in your code and begin testing them. When you configure resources, you can organize the list in the way you want Game Center to present them.

Begin configuring the feature you want by creating a GameKit bundle file:

1. In Xcode 16.3 and later, choose File > New > File from Template.
2. Choose either iOS or macOS as the platform.
3. Scroll down to Other, and select the GameKit Bundle template.
4. Click Next.
5. In the sheet that appears, enter a name for the configuration and select the appropriate targets.
6. Click Create.



When you're ready to deploy your configuration updates, sync them with App Store Connect. If you already configured features in App Store Connect, you can sync your existing configuration with the GameKit bundle file:

1. At the bottom of the left column, click the More (...) button.
2. Choose Pull from App Store Connect.
3. Select your development team.
4. Select the bundle ID or Game Center group.
5. Click Pull.

The details for the resource then appear in the editor at right. If you already pushed your configuration changes to App Store Connect, removing a resource from the local configuration file doesn't remove it from App Store Connect.

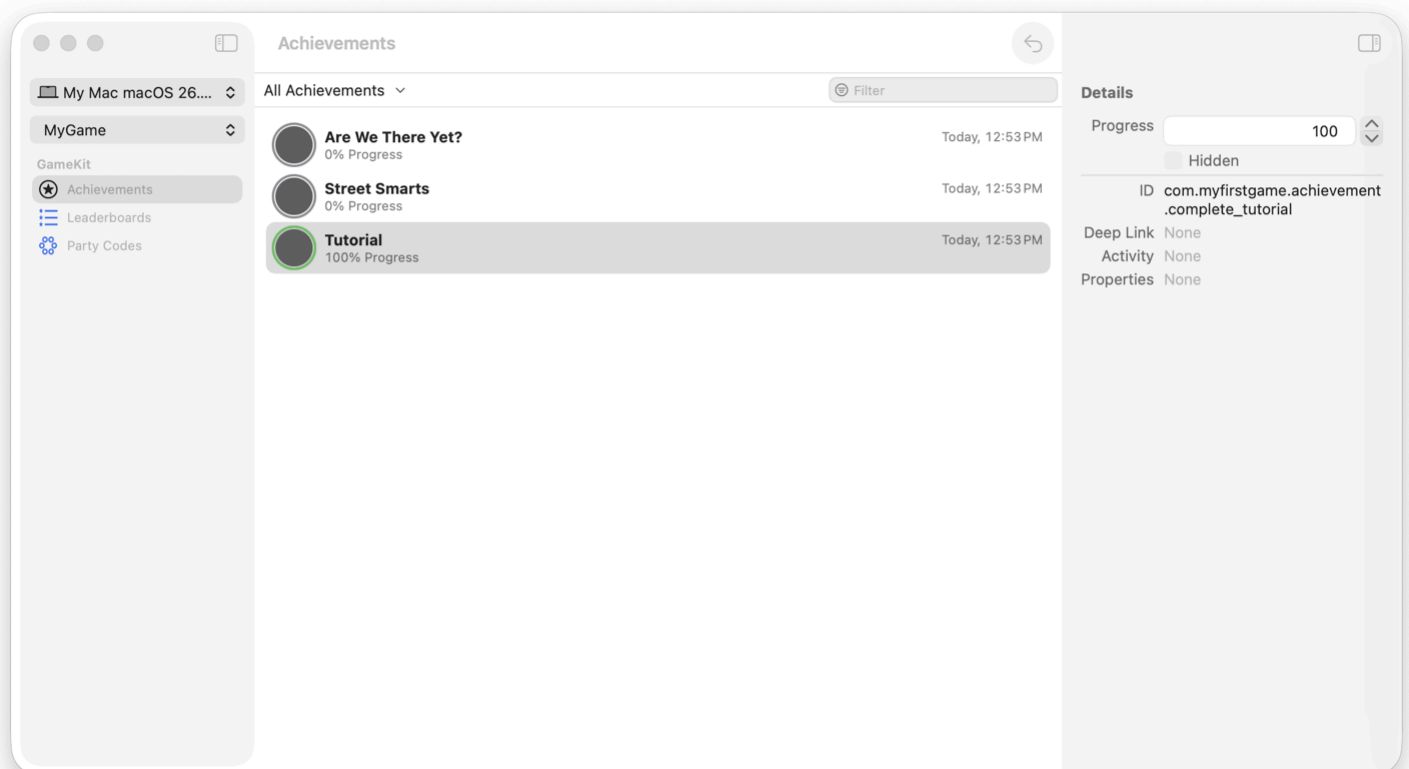
If you don't have an app record in App Store Connect, create one so you can sync your configuration while you develop your game. To create an app record in App Store Connect that matches your App ID in your developer account, see [Add a new app](#) in App Store Connect Help. To enable Game Center for versions of your app, see the "Enable app version for Game Center" section of [Manage an app version for Game Center](#).

Test Game Center features by using the Game Progress Manager

Use Game Progress Manager to test your game features locally during development by modifying properties, reporting updates, and resetting resources. You can also use it to test features and deep links to your game.

Before testing your GameKit configuration, turn on Debug Mode in Xcode:

1. Choose Product > Scheme > Edit Scheme.
2. Select the Run configuration in the left column.
3. Select Options.
4. Scroll down to GameKit Configuration, and click the Enable Debug Mode checkbox.
5. Click Close.



To test your configuration, open Game Progress Manager in Xcode:

1. Choose Debug > GameKit > Manage Game Progress.
2. In the sidebar, click the "Select a device" pop-up button, and select the physical device you use for testing.
3. Click the pop-up button beneath that and select the project you want to debug.

The test data stays local to your machine and doesn't rely on App Store Connect to test.

Important

Testing your app with the Game Progress Manager requires a physical device and is available for iOS 18.4 and later, macOS 15.4 and later, tvOS 18.4 and later, and visionOS 2.4 and later.

After selecting the device and app, choose the Game Center feature you want to test. In the inspector panel for a resource, you can simulate a variety of updates, like achievement progress, leaderboard scores, and deep links to your game activities.

Tip

When testing score submissions to a leaderboard, the system can notify you when they happen instead of needing to manually check. In iOS, open Settings > Developer and turn on Notify About Score Submissions.

To reset the progress for your Game Center resource, click the reset button at the top of the Game Progress Manager. You can't access previous occurrences of a resource after resetting the Game Progress Manager.

Initialize the local player

In your game, you need to initialize the local player before you can use any GameKit APIs and Game Center services. Game Center isn't a single sign-on (SSO) authentication service, so don't use it for authentication. When you initialize the player, you connect the player with the Game Center services.

The best time to initialize Game Center is when the player launches your game, so don't hide initialization behind game menus or the settings screen. When you initialize the local player, Game Center:

- Checks whether you configured your game for Game Center
- Verifies the credentials of the player and ensures their account is ready for use

To initialize the player, set [authenticateHandler](#) on the shared instance of [GKLocalPlayer](#) that represents the player of your game:

```
GKLocalPlayer.local.authenticateHandler = { viewController, error in
    // Handle the initialization callbacks.
}
```

For more information about initializing a local player, see [Authenticating a player](#).

You use the handler to initialize Game Center. If you need to authenticate a person with your own server, continue the flow with [fetchItems\(forIdentityVerificationSignature:\)](#).

See Also

Essentials



Authenticating a player

Confirm player credentials and device capabilities and check for account restrictions.



Improving the player experience for games with large downloads

Provide ample content in your base installation and then use on-demand resources and the Background Assets API to handle additional content.

Game Center Entitlement

A Boolean value that indicates whether users of the app may see and compare achievements on a leaderboard, invite friends, and start multiplayer games.