playAWARDS iOS SDK Installation Guide

This document is a comprehensive guide for installing the playAWARDS iOS SDK on a Mac computer running macOS. This document outlines multiple integration methods for incorporating the playAWARDS iOS SDK into your project, tailored to accommodate a native iOS development environment.

Development Environment Prerequisites

You must have the latest version of the playAWARDS iOS SDK xcframework, for example, **RewardsStoreSDK.xcframework**. You can download the SDK xcframework from the playAWARDS Game Console SDK Distribution page or our JFrog repositories.

Prerequisites for Installing the playAWARDS iOS SDK

- Understand the iOS operating system and Xcode.
- Have knowledge of the iOS development environment and best practices.
- Be familiar with the JFrog platform, especially downloading artifacts.
- Know how to manage dependencies using CocoaPods and CocoaPods-Art.
- Have proficiency in the Swift programming language and development environment for Swift projects.

Do the following to install the playAWARDS iOS SDK:

1. Either download your SDK directly or automate the installation process:
    - Directly download the playAWARDS iOS SDK through the playAWARDS Game Console.
    - Configure **Automated Installation** through JFrog.
2. Install the playAWARDS iOS SDK into Xcode.

(*Recommended*) Directly Download the playAWARDS iOS SDK through the playAWARDS Game Console

Direct download is the recommended and most straightforward method for installing the playAWARDS iOS SDK. You need access to the playAWARDS Game Console SDK Distribution page to download the latest SDK version and integrate the SDK into your project directory.

Configure Automated Installation through JFrog

Another method for gaining access to the playAWARDS iOS SDK is through JFrog. The value in using this method is automation. You can automate connecting to the playAWARDS iOS SDK and integrating the SDK into your code.

You must configure your JFrog credentials to use methods that require access to our JFrog artifactory (SDK) at build time. Upon receiving your credentials, you can log in to our JFrog account to either browse or directly download our SDKs. Alternatively, you can configure the API key from your JFrog account to use in-build scripts, facilitating seamless integration.

Note: Reach out to PLAYSTUDIOS IT to create your JFrog account and gain access to our repository.

Do the following to configure your JFrog **API Key**:

1. Click **Edit Profile** in the JFrog User Menu.
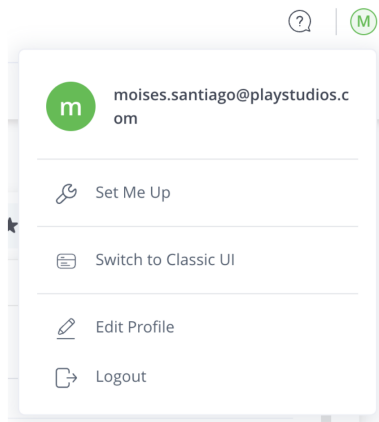
2. Go to **Authentication Settings** and click **Generate an Identity Token**.
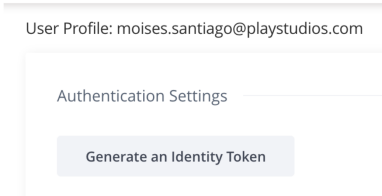
3. Enter the description of the token and click **Next**.
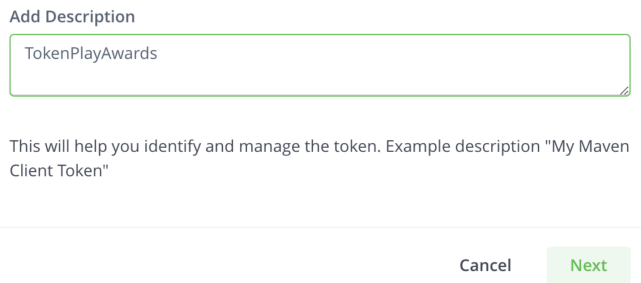
4. A new dialog window with the token is displayed. Copy and save the generated **Reference Token**.
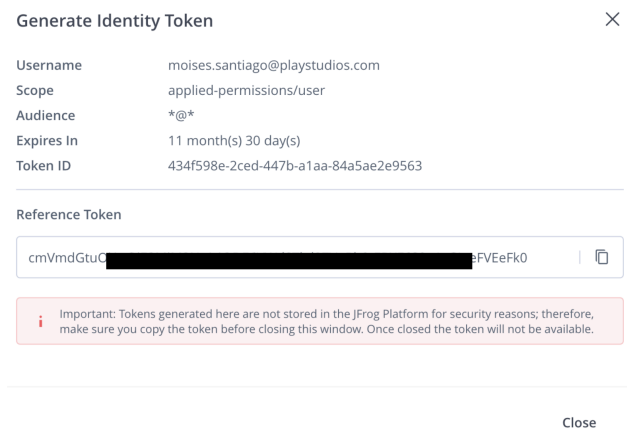
5. Copy the **API Key**.

	Description	Token ID	Issued At	Expiry Date
☐	TokenPlayAwards	f58b318d-93d1-4fc8-9f4b-4f958b0ac509	16-08-23 13:58:19 -0...	15-08-24 13:58:19 -0...

Showing 1 - 1 from 1 items

API Key   ⊙

░░░░░░░░░░░░░░░░░░░░░░░░ ⊙   ⧉ ⟳   ✕ Revoke API Key

After configuring your JFrog credentials, you must continue your installation using either of the following:

- Gradle - Gradle is a build system. The JFrog credentials allow Gradle to pull updated SDK packages for its build.
- Cocoapods - Cocoapods is a dependency manager. The JFrog credentials allow Cocoapods to place the SDK in a development environment. Updated versions of the SDK are added to the environment.

Continue Configuring your Installation through Gradle

Do the following to continue your installation through Gradle:

1. Store your username and API key in the global Gradle properties file. Navigate to **${HOME}/.gradle** and create a **gradle.properties** file if one does not already exist.
2. Open the **gradle.properties** file and enter your JFrog credentials as follows (replace the placeholders with your actual credentials):
   - `jfrogUser=example@playstudios.com` (replace with your username)
   - `jfrogPass=APIKEY` (use your JFrog account's API key)

This credentials file ensures that your build system can securely fetch the necessary SDK components during the build process without manual intervention, streamlining the development workflow.

Continue Configuring your Installation through Cocoapods

Do the following to continue your installation through Cocoapods:

1. Install `cocoapods-art` on your local machine:

```
$ sudo gem install cocoapods-art
```

If you receive the following error message when trying to install `cocoapods-art` on your local machine:

```
Error installing cocoapods:
The last version of activesupport (>= 5.0, < 8) to support your Ruby &
RubyGems was 6.1.7.4.
Try installing it with `gem install activesupport -v 6.1.7.4` and then
running the current command again. activesupport requires Ruby version >=
2.7.0. The current ruby version is 2.6.10.210.
```

try installing `cocoapods-art` using the following expanded code:

```
$ sudo gem install activesupport -v 6.1.7.4
Fetching zeitwerk-2.6.11.gem
Fetching activesupport-6.1.7.4.gem
Successfully installed zeitwerk-2.6.11
Successfully installed activesupport-6.1.7.4
Parsing documentation for zeitwerk-2.6.11
Installing ri documentation for zeitwerk-2.6.11
Parsing documentation for activesupport-6.1.7.4
Installing ri documentation for activesupport-6.1.7.4
Done installing documentation for zeitwerk, activesupport after 1 seconds 2
gems installed

$ sudo gem install cocoapods
Successfully installed cocoapods-core-1.12.1
Successfully installed cocoapods-1.12.1
Parsing documentation for cocoapods-core-1.12.1
Installing ri documentation for cocoapods-core-1.12.1
Parsing documentation for cocoapods-1.12.1
Installing ri documentation for cocoapods-1.12.1
Done installing documentation for cocoapods-core, cocoapods after 2 seconds
2 gems installed

$ pod setup
Setup completed
```

2. Create a **.netrc** file in your **/home** directory with the following contents:

```
machine playstudios.jfrog.io
login ${USER_EMAIL}
password ${API_KEY}
```

3. Add the **Artifactory Specs** repo:

```
$ pod repo-art add playawards-iossdk-cocoapods-release-local
"https://playstudios.jfrog.io/artifactory/api/pods/playawards-iossdk-
cocoapods-release-local"

$ pod repo update
```

4. Add the `rewardsstoresdk` podspec to your project's Podfile. Create your Podfile with the `pod init` command:
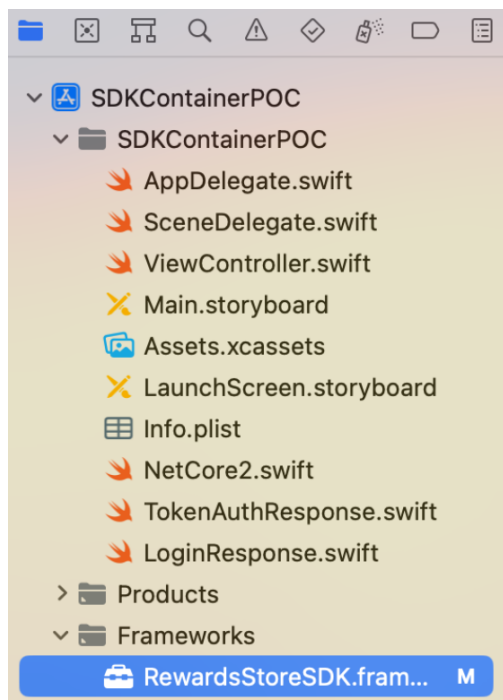
```
$ pod init
```

5. Run `pod install` to download the **xcframework**:

```
$ pod install
```

Install the playAWARDS iOS SDK into Xcode

Do the following to install the playAWARDS iOS SDK into Xcode:

1. Create a new project in Xcode. Alternatively, you can integrate the SDK into an existing project.

2. Drag the **.xcframework** folder to the **Frameworks** group of the iOS project in Xcode.

3. With the **Target** selected, go to **General** > **Frameworks, Libraries, and Embedded Content**. There are two **Embed** options:

   ○ Select **Embed Without Signing** next to the **RewardsStoreSDK.framework** if you are going to run the project in the **Development** mode.

   ○ Select **Embed & Sign** if you are going to run the project in the **Production** mode.

General   Signing & Capabilities   Resource Tags   Info   Build Settings   Build Phases   Build Rules

PROJECT

SDKContainerPOC

TARGETS

SDK_Debug

☑ Landscape Left
☑ Landscape Right

Status Bar Style   Default

☐ Requires full screen

Supports multiple windows

App Icons and Launch Screen

App Icon   AppIcon   +

App Icons Source   ☐ Include all app icon assets

Launch Screen File

Supported Intents

Class Name	Authentication
	Add intents eligible for in-app handling here

+ —

Frameworks, Libraries, and Embedded Content

Name	Embed
📁 RewardsStoreSDK.framework	Embed Without Signing ⇕