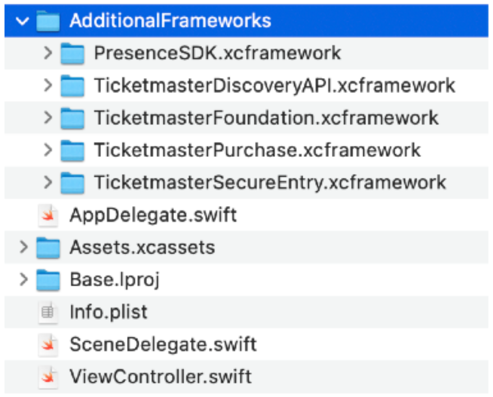
Ticketmaster SDK Quick Start Guide

1. Add Frameworks to your Project folder

1. Create a new sub folder somewhere in your project folder to hold these additional frameworks.
    1. We've named the folder "AdditionalFrameworks" here, but the name can be anything that makes sense to you.

2. Copy the required Ticketmaster frameworks into this new folder
    1. Required for both Presence SDK and Purchase SDK:
        1. TicketmasterFoundation
        2. TicketmasterSecureEntry
        3. PresenceSDK
    2. If using Purchase SDK, also add:
        1. TicketmasterDiscoveryAPI
        2. TicketmasterPurchase

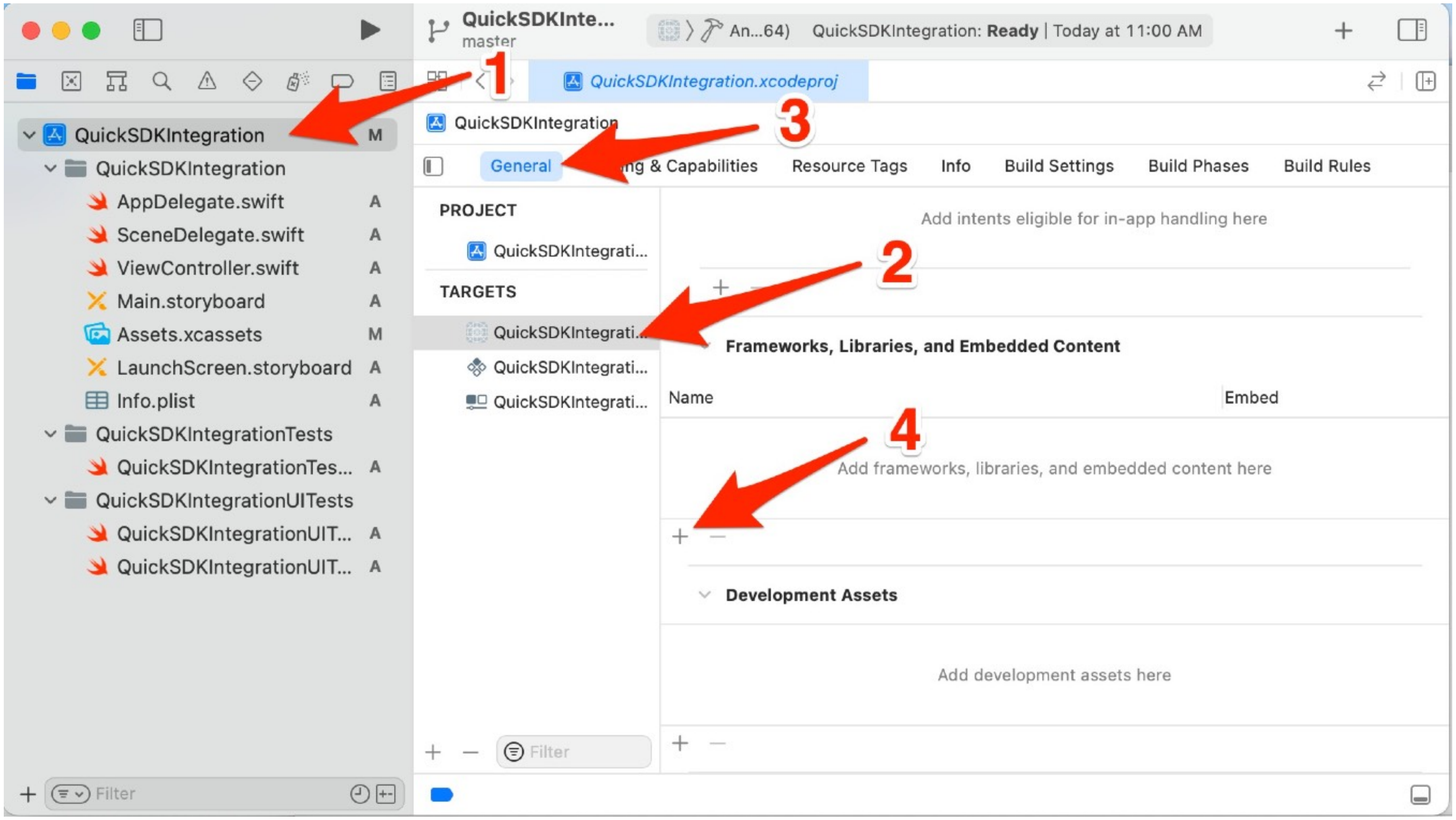
3. Example:



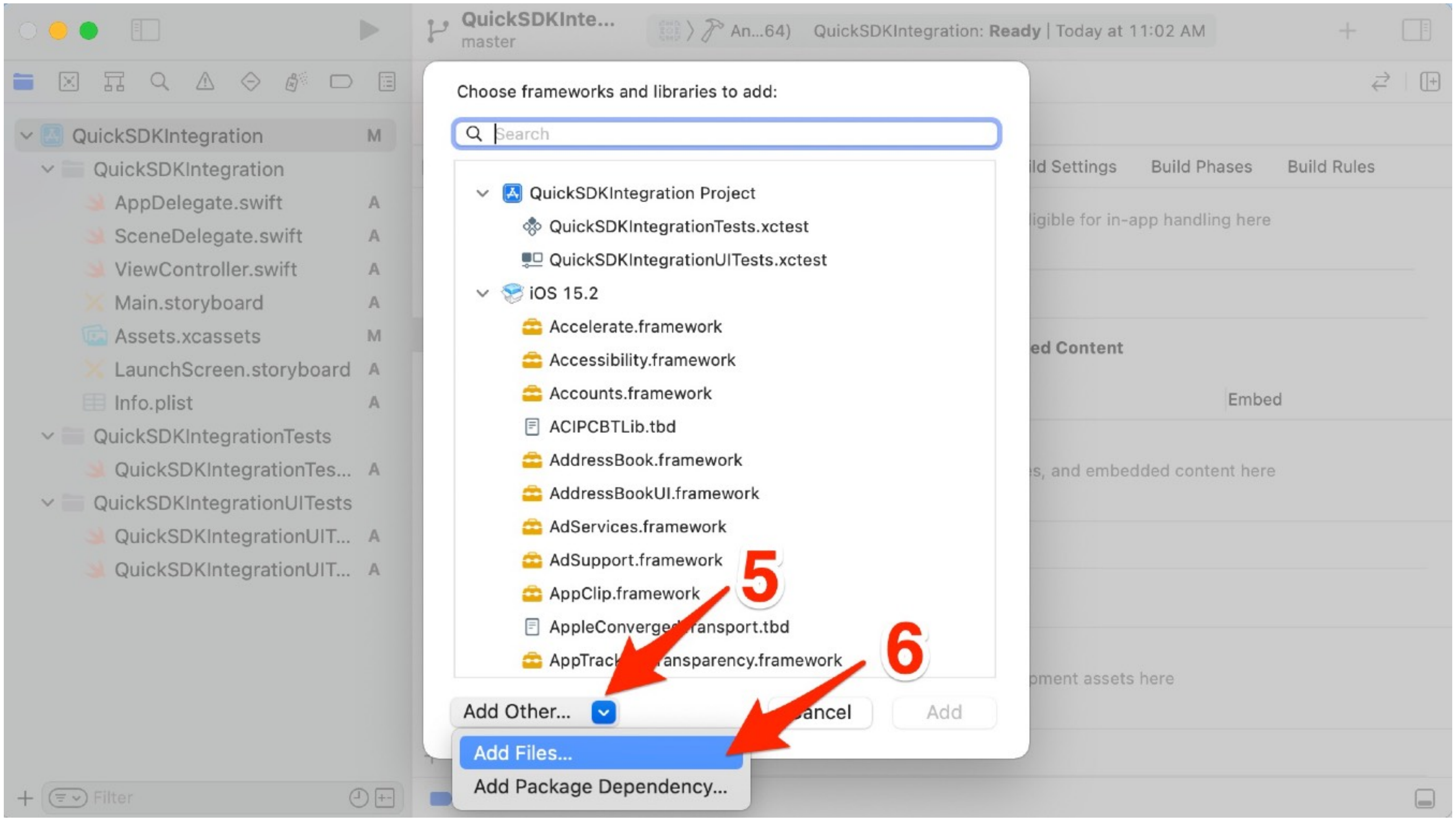
```
AdditionalFrameworks
    PresenceSDK.xcframework
    TicketmasterDiscoveryAPI.xcframework
    TicketmasterFoundation.xcframework
    TicketmasterPurchase.xcframework
    TicketmasterSecureEntry.xcframework
  AppDelegate.swift
  Assets.xcassets
  Base.lproj
  Info.plist
  SceneDelegate.swift
  ViewController.swift
```

2. Add Frameworks to your Project file

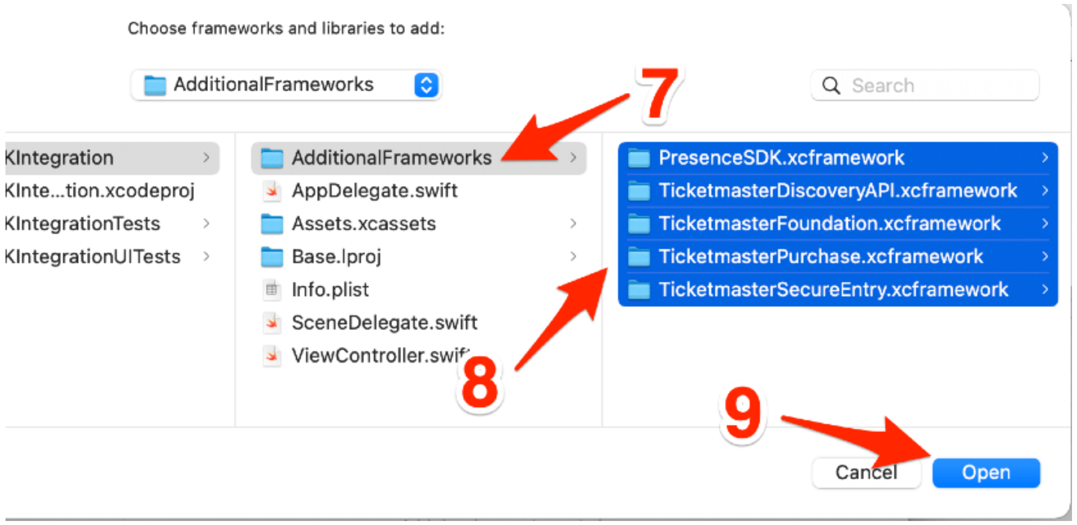
1. In Xcode, select your Project
2. Select your deployment Target
3. Select the General Tab
4. Click the + button under Frameworks, Libraries, and Embedded Content



5. On the Choose Frameworks popup that appears, click the Add Other… drop-down
6. Choose Add Files… option

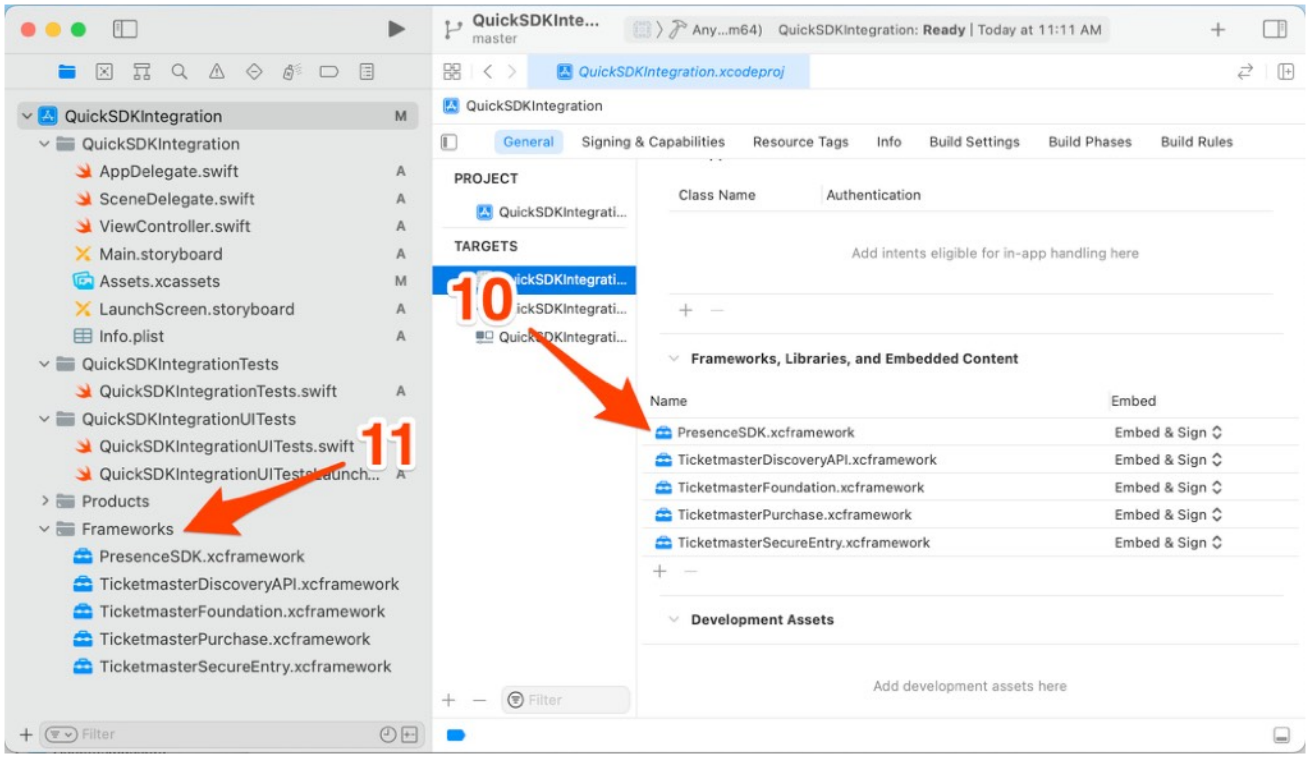


7. Find and select the sub folder where you copied the frameworks earlier
8. Select all the .xcframework folders inside
9. Click Open



At this point, all of the required xcframework files should be added to your project.

Please make sure that both:
10. Frameworks appear in the Framework list as “Embed & Sign”
11. Frameworks appear in the Project Navigator list



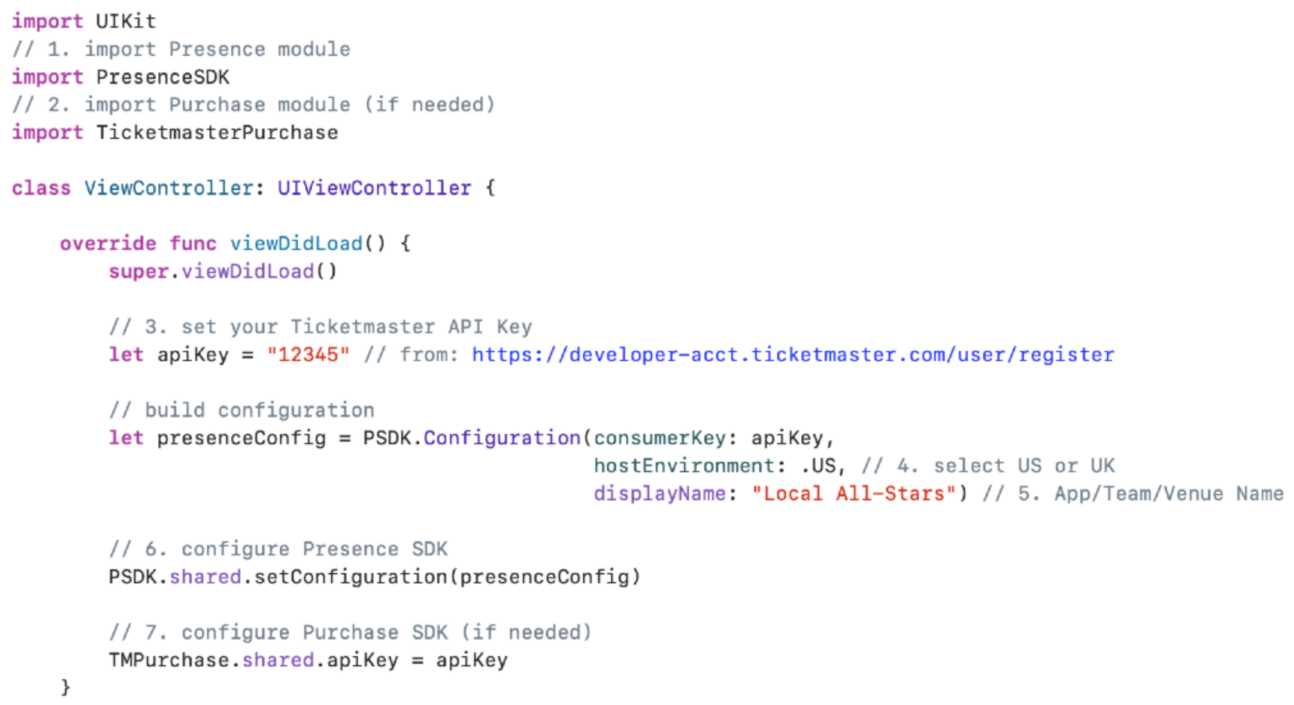
Name	Embed
PresenceSDK.xcframework	Embed & Sign
TicketmasterDiscoveryAPI.xcframework	Embed & Sign
TicketmasterFoundation.xcframework	Embed & Sign
TicketmasterPurchase.xcframework	Embed & Sign
TicketmasterSecureEntry.xcframework	Embed & Sign

If you have additional deployment Targets, you will need to add these frameworks to each of those Targets as well by repeating steps 1 through 4. The new frameworks will now appear on the list, so you won’t need to repeat steps 5 to 11.

3. Import and Configure SDKs

Now that you have added the frameworks to your Project folder and Project file, it is time to configure the SDKs. It can be done at the launch of the app (in the AppDelegate or SceneDelegate) or it can be done later after your App is setup, but before you actually present the SDK's UI.

1. Import module for PresenceSDK
2. Import module for TicketmasterPurchase (if you are using the Purchase SDK)
3. Enter your API Key
    1. provided by the Ticketmaster Client Support Team or created at https://developer-acct.ticketmaster.com/user/register
4. Determine .US or .UK Host:
    1. UK, Ireland, and SportXR clients should use .UK
    2. Everyone else should use .US
5. Set your App/Team/Venue name
6. Set configuration on Presence SDK
7. Set configuration on Purchase SDK (if you are using the Purchase SDK)



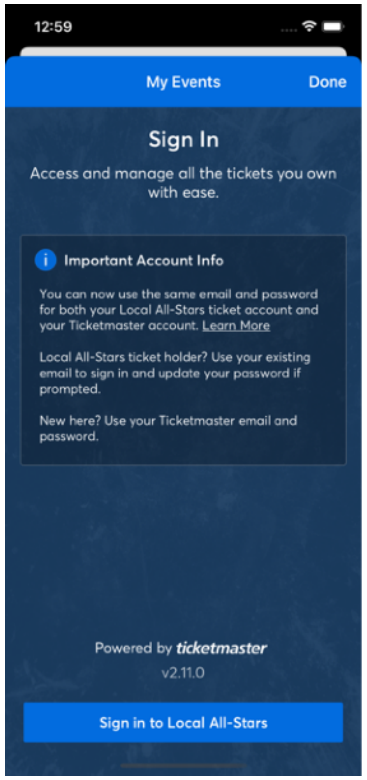
```swift
import UIKit
// 1. import Presence module
import PresenceSDK
// 2. import Purchase module (if needed)
import TicketmasterPurchase

class ViewController: UIViewController {

    override func viewDidLoad() {
        super.viewDidLoad()

        // 3. set your Ticketmaster API Key
        let apiKey = "12345" // from: https://developer-acct.ticketmaster.com/user/register

        // build configuration
        let presenceConfig = PSDK.Configuration(consumerKey: apiKey,
                                                hostEnvironment: .US, // 4. select US or UK
                                                displayName: "Local All-Stars") // 5. App/Team/Venue Name

        // 6. configure Presence SDK
        PSDK.shared.setConfiguration(presenceConfig)

        // 7. configure Purchase SDK (if needed)
        TMPurchase.shared.apiKey = apiKey
    }
```

4. Present the Presence SDK

Now that the SDKs are configured, it's time to present the SDK UI.
The Presence SDK is used to show Purchased Events (including Orders for those Events, Tickets, and Barcodes).

1. Create a PresenceViewController
2. Present PresenceViewController

```
@IBAction func presentPresenceSDK(_ sender: Any) {
    // 1. build Presence ViewController
    let presenceVC = PresenceViewController()

    // 2. present Presence SDK
    present(presenceVC, animated: true, completion: nil)
}
```



12:59

My Events Done

Sign In

Access and manage all the tickets you own with ease.

Important Account Info

You can now use the same email and password for both your Local All-Stars ticket account and your Ticketmaster account. Learn More

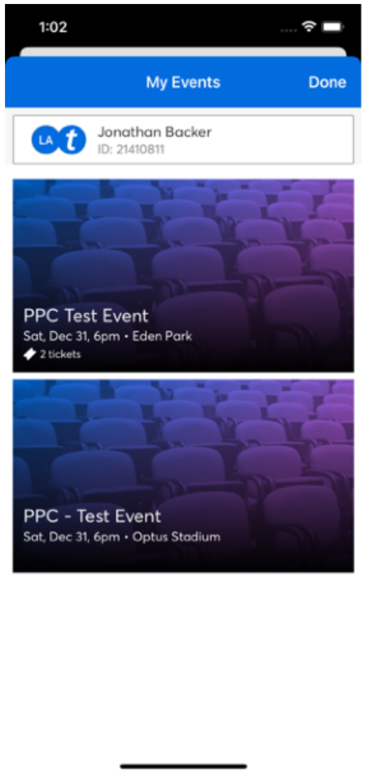
Local All-Stars ticket holder? Use your existing email to sign in and update your password if prompted.

New here? Use your Ticketmaster email and password.

Powered by *ticketmaster*

v2.11.0

Sign in to Local All-Stars



1:02

My Events Done

Jonathan Backer
ID: 21410811

PPC Test Event
Sat, Dec 31, 6pm • Eden Park
2 tickets

PPC - Test Event
Sat, Dec 31, 6pm • Optus Stadium

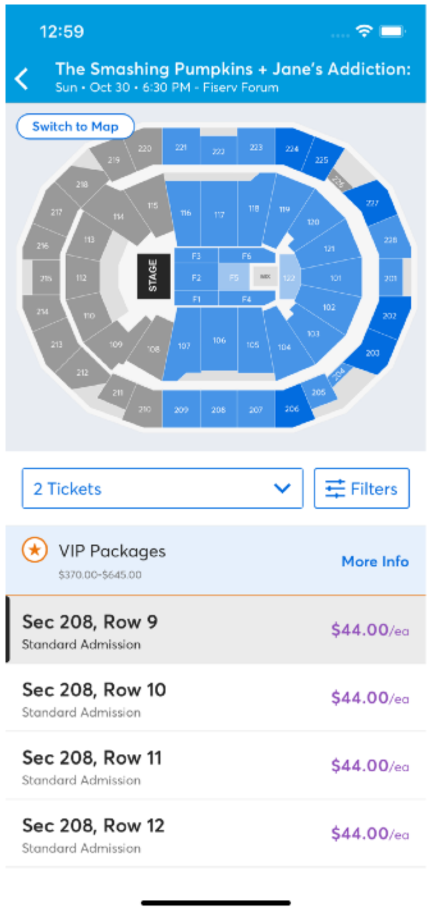
5. Present the Purchase SDK

Now that the SDKs are configured, it's time to present the SDK UI.
The Purchase SDK is used to show new Events to be purchased (including Smart Queue, Special Offers, Delivery Options, and the user's Payment Wallet).

1. Determine which Event ID to purchase.
    1. This ID can be hard-coded, or fetched from the Discovery API
    2. This works for both classic Host IDs ("07005CA5DEBD61A6") and Discovery IDs ("vvG1jZ919rRpkJ")
2. Create a TMPurchaseWebsiteConfiguration (aka Club Site Configuration)
3. Create a TMPurchaseNavigationController
4. Connect Purchase SDK to Presence SDK (for login purposes)
5. Present TMPurchaseNavigationController

```swift
@IBAction func presentPurchaseSDK(_ sender: Any) {
    // 1. determine Event ID
    // The Smashing Pumpkins, Fiserv Forum (Oct 30, 2022):
    let eventID = "07005CA5DEBD61A6" // or "vvG1jZ919rRpkJ"

    // 2. configure Purchase SDK for this particular Event
    let clubSiteConfig = TMPurchaseWebsiteConfiguration(eventID: eventID)

    // 3. build Purchase ViewController
    let purchaseVC = TMPurchaseNavigationController(configuration: clubSiteConfig)

    // 4. connect Presence SDK login to Purchase SDK
    purchaseVC.oauthDelegate = PSDK.shared.oauthProvider

    // 5. present Purchase SDK
    present(purchaseVC, animated: true, completion: nil)
}
```

Step 4 is very important, as it connects the Presence SDK's Login UI/State to the Purchase SDK.



The Smashing Pumpkins + Jane's Addiction:
Sun • Oct 30 • 6:30 PM • Fiserv Forum

VIP Packages
$370.00-$645.00

Sec 208, Row 9 Standard Admission	$44.00/ea
Sec 208, Row 10 Standard Admission	$44.00/ea
Sec 208, Row 11 Standard Admission	$44.00/ea
Sec 208, Row 12 Standard Admission	$44.00/ea

6. User Logout

The Purchase SDK and Presence SDK will prompt the user to login when needed. However, you may want to allow the user to logout as well (usually to switch accounts).

1. Call PSDK.shared.logout( )

```
@IBAction func logoutUser(_ sender: Any) {
    // 1. log user out of Presence SDK (includes Purchase SDK)
    PSDK.shared.logOut { hostSuccess, hostError, teamSuccess, teamError, sportXRSuccess, sportXRError in
        if hostSuccess && teamSuccess && sportXRSuccess {
            // logout worked!
        } else {
            // OAuth 2.0 error
            // user is still logged in
        }
    }
}
```

OAuth 2.0 protocol requires a logout operation to complete, logging the user out of his account across all applications and websites.

This function can safely be called multiple times as needed, even if the user is not logged in.

7. Next Steps

This Quick Start guide has provided you with the bare minimum steps to integrate the Ticketmaster Purchase and Presence SDKs.

However, there is more optional parts to learn:

1. Additional Purchase and Presence configuration options allowing you to customize the user experience inside these SDKs.

2. Delegate callbacks and Notifications to inform your App when the user performs interesting actions, like purchasing Tickets, or viewing Orders.

These options are documented in:

1. the public source-code interfaces

2. The source code of the Purchase Sample App & the Presence Demo App
    1. available at: https://developer.ticketmaster.com/products-and-docs/sdks/presence-sdk/

3. full documentation:
    1. available at: https://developer.ticketmaster.com/products-and-docs/sdks/presence-sdk/