

[StoreKit](#) / External Purchase

API Collection

# External Purchase

Enable qualifying apps to offer external purchases.

## Overview

If you develop a qualifying app, you may complete a request for optional entitlements that allow you to offer external purchases. For more information about whether your app qualifies, requesting an entitlement, and implementation requirements, see:

- [Communication and promotion of offers on the App Store in the EU](#)
- [Distributing dating apps in the Netherlands](#)
- [Distributing apps in Russia that provide an external purchase link](#)
- [Distributing apps using a third-party payment provider in South Korea](#)
- [Distributing music streaming apps in the EEA that provide an external purchase link](#)

## Offer external purchases using alternative payment service providers

If your account receives the StoreKit External Purchase entitlement, implement the following to offer external purchases within your app:

- Configure the entitlement for your app. For more information, see [com.apple.developer.storekit.external-purchase](#).
- Configure the `SKExternalPurchase` property list key.
- Use the `ExternalPurchase` type's `canPresent` property to determine whether external purchase is available.

- Call the `presentNoticeSheet()` method and use the external purchase token you receive from the `ExternalPurchase.NoticeResult.continuedWithExternalPurchaseToken(token:)` result to record transactions.
- From your server, report the external purchase tokens and the transactions associated with the tokens by using the [External Purchase Server API](#).

## Communicate and promote offers

If your account receives the StoreKit External Purchase Link Entitlement (EU), your app can communicate and promote offers for digital goods or services available for purchase or download, in a distribution channel of your choice. Implement the following:

- Configure the `com.apple.developer.storekit.external-purchase-link` entitlement for your app.
- Configure the `SKExternalPurchaseCustomLinkRegions` property list key. Include the country code for each permitted region where your app communicates and promotes offers.
- Check the `isEligible` property of the `ExternalPurchaseCustomLink` API to determine whether the API is available at runtime.
- At launch and before every potential transaction, call the `token(for:)` function to request the external purchase tokens, using the token types ACQUISITION and SERVICES. Associate these tokens with a customer account on your server.
- Call the `showNotice(type:)` function after a deliberate customer interaction, such as tapping a button, and before communicating and promoting offers.
- From your server, report the external purchase tokens and the transactions associated with the tokens by using the [External Purchase Server API](#).

## Communicate and promote offers for music-streaming apps

If your account receives the Music Streaming Services Entitlement (EEA), your music-streaming app can use the `ExternalPurchaseCustomLink` API to communicate and promote offers for music streaming services. Custom links can be to a distribution channel of your choice. Implement the following:

- Configure the `com.apple.developer.storekit.external-purchase-link-streaming` entitlement for your app.
- Configure the `SKExternalPurchaseLinkStreamingRegions` property list key. Include the country code for each permitted region where your app communicates and promotes offers for

music streaming.

- Check the `isEligible` property of the `ExternalPurchaseCustomLink` API to determine whether external purchase is available at runtime.
- At launch and before every potential transaction, call the `token(for:)` function to request the external purchase tokens, using the token types `ACQUISITION` and `SERVICES`. Associate these tokens with a customer account on your server.
- Call the `showNotice(type:)` function after a deliberate customer interaction, such as tapping a button, and before communicating and promoting offers.
- From your server, report the external purchase tokens and the transactions associated with the tokens by using the `External Purchase Server API`.

## Offer external purchase link outs

If your account receives the StoreKit External Purchase Link entitlement, implement the following to offer external purchase links:

- Configure the entitlement for your app. For more information, see `com.apple.developer.storekit.external-purchase-link`.
- Configure the `SKExternalPurchaseMultiLink` property list key. Provide one or more external purchase links for each permitted country code.
- Use the `ExternalPurchaseLink` type's `eligibleURLs` array to determine whether one or more external purchase links are available, then select one of those eligible URLs.
- Call the `open(url:)` method with the URL you select. StoreKit appends the external purchase token to your website's URL. Use this token to record purchases.
- From your server, report the external purchase tokens and the transactions associated with the tokens by using the `External Purchase Server API`.

## Offer a single external purchase link out for each country code

Use the `SKExternalPurchaseLink` property list key if your app has a minimum OS version prior to iOS 17.5, macOS 14.5, watchOS 10.5, or tvOS 17.5, or visionOS 1.2 and receives the StoreKit External Purchase Link entitlement.

Implement the following to offer an external purchase link when your app runs on these prior operating systems:

- Configure the entitlement for your app. For more information, see [com.apple.developer.storekit.external-purchase-link](https://developer.apple.com/documentation/storekit/external-purchase-link).
- Configure the [SKExternalPurchaseLink](https://developer.apple.com/documentation/storekit/skexternalpurchase-link) property list key.
- Use the [ExternalPurchaseLink](https://developer.apple.com/documentation/storekit/externalpurchase-link) type's [canOpen](https://developer.apple.com/documentation/storekit/externalpurchase-link/canopen) property to determine whether external purchase link is available.
- Call the [open\(\)](https://developer.apple.com/documentation/storekit/externalpurchase-link/open) method. StoreKit appends the external purchase token to your website's URL. Use this token to record purchases.
- From your server, report the external purchase tokens and the transactions associated with the tokens by using the [External Purchase Server API](https://developer.apple.com/documentation/storekit/external-purchase-server-api).

Your app may configure both the [SKExternalPurchaseLink](https://developer.apple.com/documentation/storekit/skexternalpurchase-link) and [SKExternalPurchaseMultiLink](https://developer.apple.com/documentation/storekit/skexternalpurchase-multilink) property list keys.

## Record and report external purchase tokens and associated transactions

An external purchase token is a unique string that your app or website receives when your app's customer chooses to view your external purchase offerings. Record all tokens in your system and report them and their associated transactions using the [External Purchase Server API](https://developer.apple.com/documentation/storekit/external-purchase-server-api). For more information about tokens, see [Receiving and decoding external purchase tokens](https://developer.apple.com/documentation/storekit/receiving-and-decoding-external-purchase-tokens).

## Check API availability

The External Purchase APIs, including [ExternalPurchaseLink](https://developer.apple.com/documentation/storekit/externalpurchase-link), [ExternalPurchase](https://developer.apple.com/documentation/storekit/externalpurchase), and [ExternalPurchaseCustomLink](https://developer.apple.com/documentation/storekit/externalpurchasecustomlink) provide *external purchase tokens* that you use to report transactions to Apple. [ExternalPurchaseLink](https://developer.apple.com/documentation/storekit/externalpurchase-link) and [ExternalPurchase](https://developer.apple.com/documentation/storekit/externalpurchase) are available starting in iOS 17.4, iPadOS 17.4, macOS 14.4, Mac Catalyst 17.4, tvOS 17.4, visionOS 1.1, and watchOS 10.4. [ExternalPurchaseCustomLink](https://developer.apple.com/documentation/storekit/externalpurchasecustomlink) is available starting in iOS 18.1, iPadOS 18.1, Mac Catalyst 18.1, and macOS 15.1.

For apps that run on iOS 15.4 through 17.3 and iPadOS 15.4 through 17.3, the External Purchase APIs have the following behavior:

- [ExternalPurchase](https://developer.apple.com/documentation/storekit/externalpurchase) and [ExternalPurchaseLink](https://developer.apple.com/documentation/storekit/externalpurchase-link) APIs throw errors or return `false` for compatible iPad or iPhone apps running in visionOS, on a Mac with Apple silicon, or on a Mac app built with Mac Catalyst.
- The APIs don't provide external purchase tokens.
- The APIs are available in iOS and iPadOS only.

---

# Topics

## Offering external purchases

`enum ExternalPurchase`

Enables qualifying apps to offer external purchases within the app.

`com.apple.developer.storekit.external-purchase`

A Boolean value that indicates whether your app can offer external purchases.

`SKExternalPurchase`

A string array of country codes that indicates your app supports external purchases.

## Communicating and promoting offers

`enum ExternalPurchaseCustomLink`

Enables qualifying apps to offer custom links for external purchases and use alternative payment service providers.

`struct Token`


A token you use with the External Purchase custom link API.

`com.apple.developer.storekit.external-purchase-link`

A Boolean value that indicates whether your app can include a link that directs people to a website to make an external purchase.

`SKExternalPurchaseCustomLinkRegions`

An array of country code strings that indicate the regions where your app supports custom links for the communication and promotion of offers.

 Testing transactions that use custom link tokens

Recognize custom link tokens that your app receives in the sandbox testing environment, and use them to test reporting transactions.

## Communicating and promoting offers for music streaming services

`enum ExternalPurchaseCustomLink`

Enables qualifying apps to offer custom links for external purchases and use alternative payment service providers.

`com.apple.developer.storekit.external-purchase-link-streaming`

An entitlement that grants a qualifying music-streaming app the ability to communicate and promote offers.

`SKExternalPurchaseLinkStreamingRegions`

A list of country codes that indicate the regions where your music-streaming app communicates and promotes offers.

## Offering external purchase links

`enum ExternalPurchaseLink`

Enables qualifying apps to offer external purchase links.

`com.apple.developer.storekit.external-purchase-link`

A Boolean value that indicates whether your app can include a link that directs people to a website to make an external purchase.

`SKExternalPurchaseMultiLink`

A dictionary that contains an array of URLs to websites where people using your app can make external purchases.

`SKExternalPurchaseLink`

A dictionary that contains URLs to websites where people using your app can make external purchases for supported regions.

## Managing external purchase tokens

 Receiving and decoding external purchase tokens

Receive tokens for external purchases that you use to report transactions to Apple.