

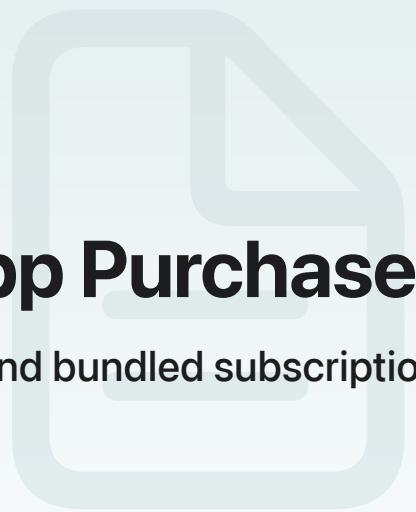
Documentation

[Advanced Commerce API](#) / Creating SKUs for your In-App Purchases

Article

Creating SKUs for your In-App Purchases

Define and manage one-time charges, subscriptions, and bundled subscriptions within your app.



Overview

For each purchase you plan to offer, create a product identifier, or SKU, in your system along with corresponding text such as display name and price. The App Store displays this information on the payment sheet when a customer initiates a purchase, which helps them understand the transaction before they confirm their purchase. The App Store also uses this information in additional communication to the customer, including in email receipts and in the Apple Account Settings under Subscriptions.

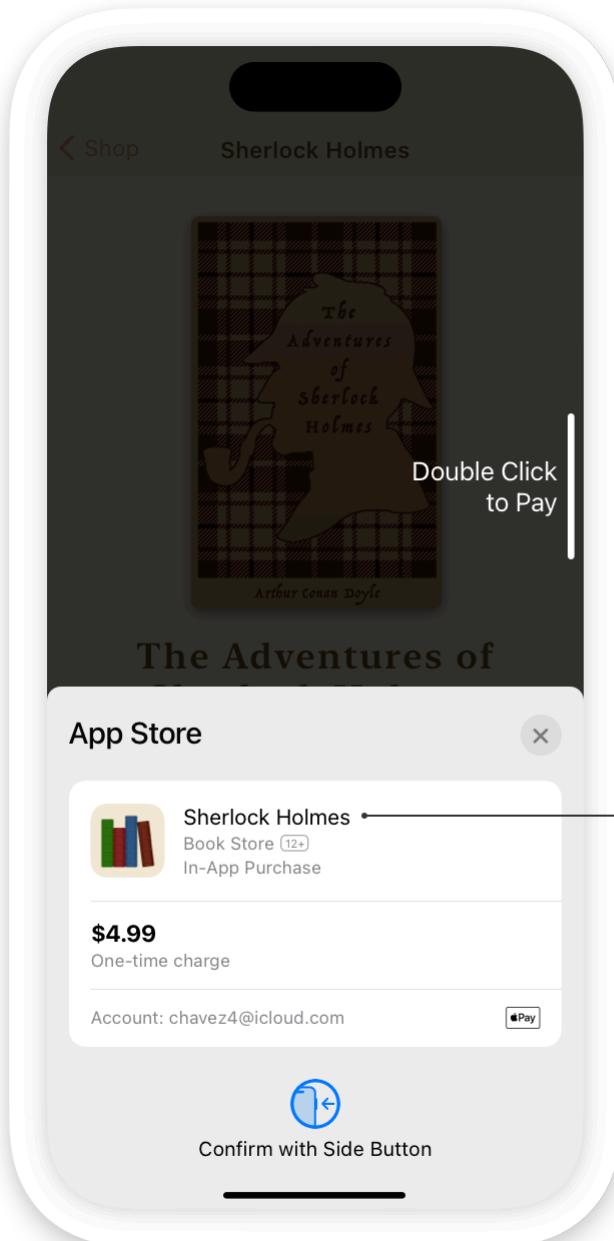
To ensure your purchases display properly and provide a quality experience, follow these guidelines:

- Be clear, concise, and descriptive.
- Use proper capitalization and punctuation, and avoid using all capitals.
- Create localized information for any regions where your app is available.
- Consider how you use special characters (for example, hyphens, periods, and underscores) and diacritics. You can use special characters, but avoid using them excessively or beginning strings with them.
- Don't use markup language, emoticons, diacritics, or control characters (for example, null, new lines, carriage returns, escape, or other invisible characters) that cause strings to exceed a single line.

Review the Human Interface Guidelines for additional best practices for [writing](#) and designing your [in-app purchases](#). If you offer subscriptions, get additional best practices for [clearly describing subscriptions](#) in your paywalls and payment sheets.

Create one-time charges

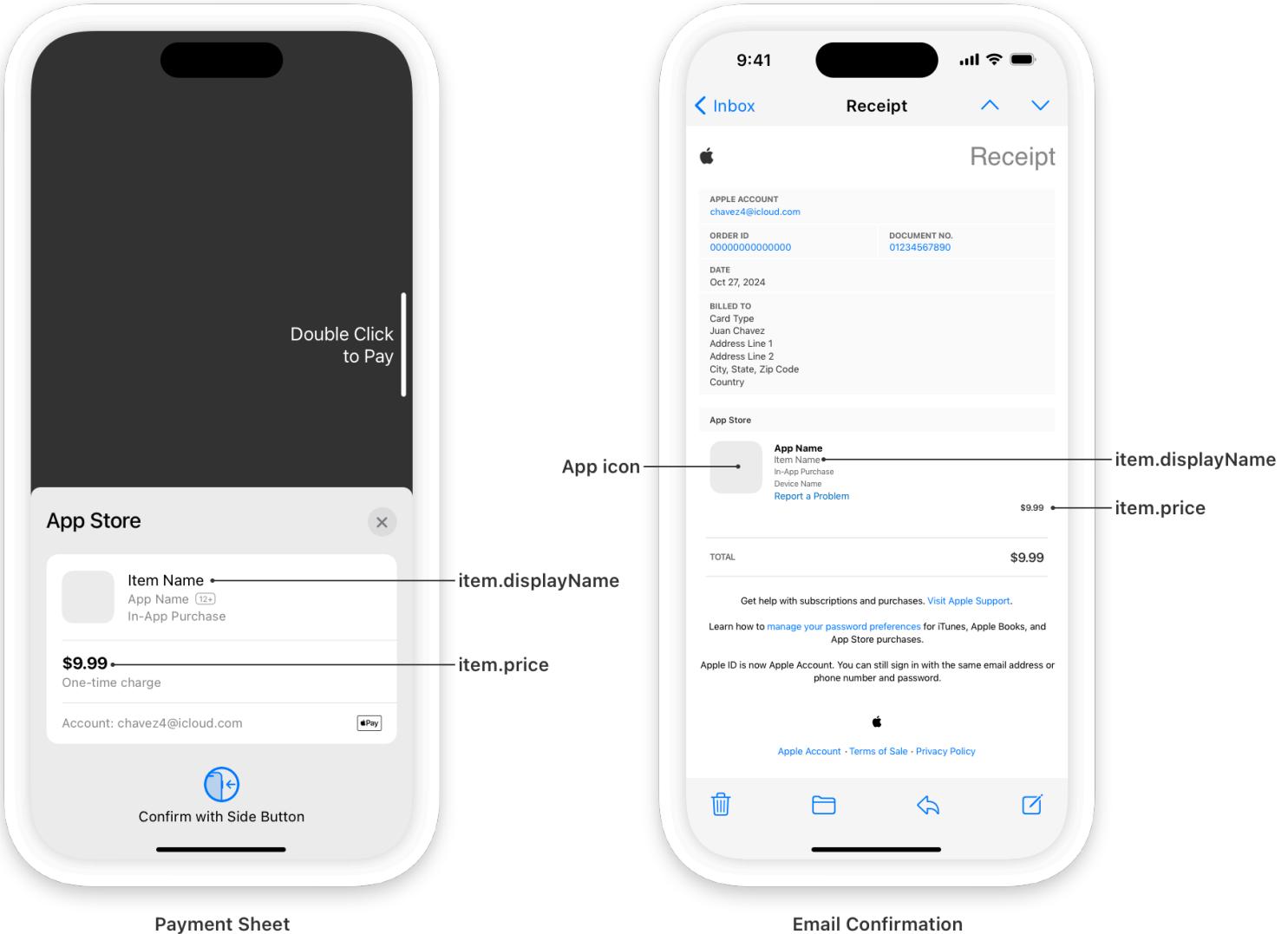
Manage one-time purchases — such as one-time rentals, books, or courses — by using the [OneTimeChargeCreateRequest](#) API to provide information to the App Store when the customer initiates a purchase. After a one-time charge is complete, customers receive an email receipt from Apple.



Item a customer is purchasing

The following details appear in the App Store payment sheet as well as your email receipt; you set both of these values in the [OneTimeChargeItem](#) of the [OneTimeChargeCreateRequest](#):

- `item.displayName`: The name of the item the customer is purchasing.
- `item.price`: The price of the item the customer is purchasing.

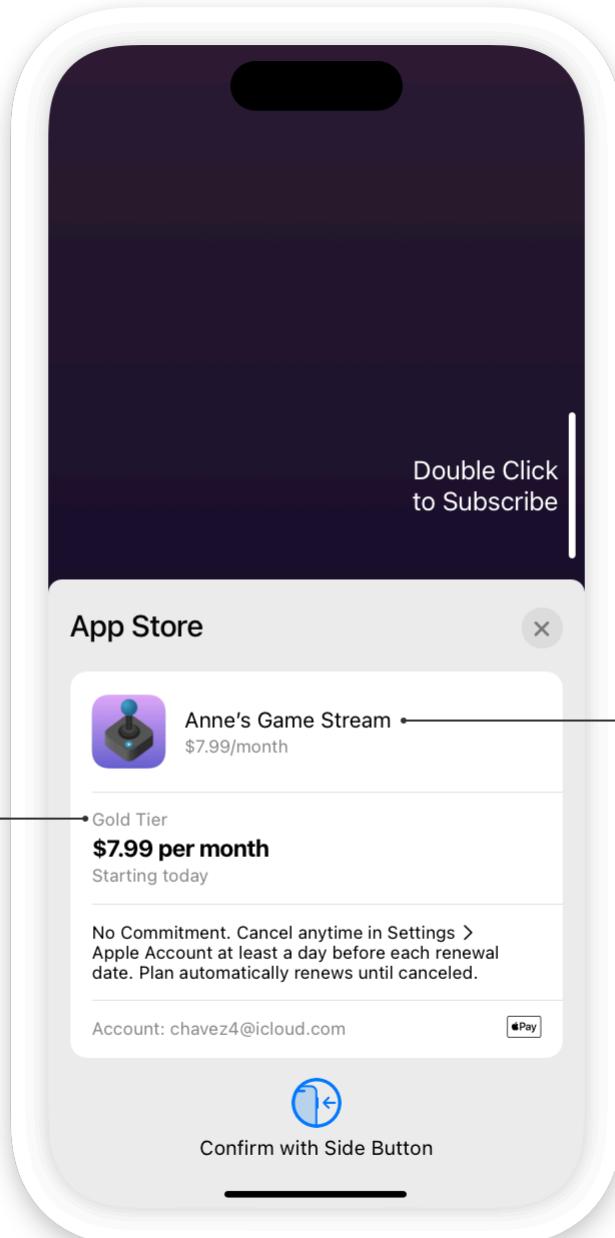


Note

You can set up an image that displays in the payment sheet instead of the app icon. For more information, see the "Set up an image for the payment sheet to display" section in [Setting up your project for Advanced Commerce API](#).

Create subscriptions

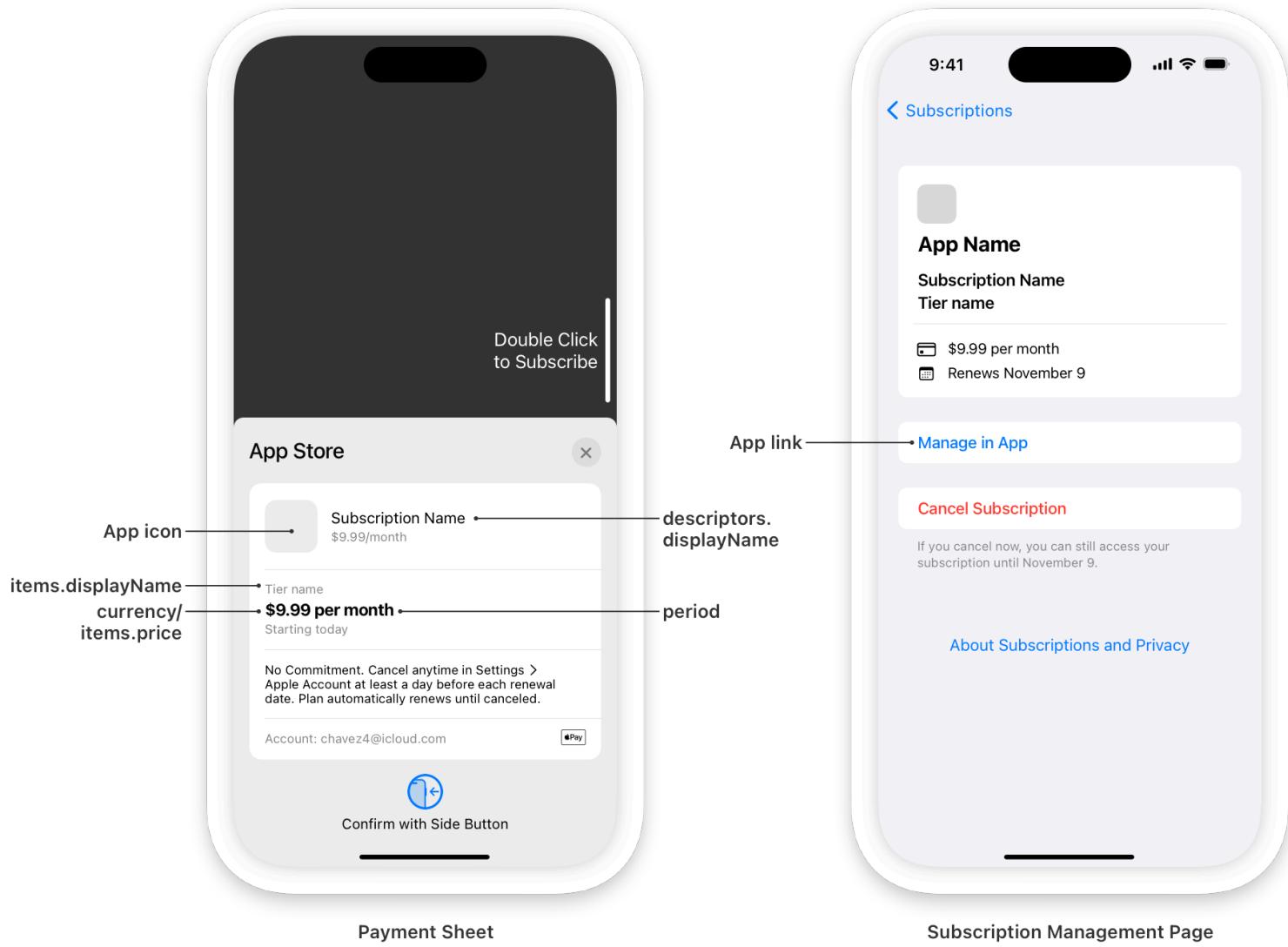
Use the [SubscriptionCreateRequest](#) API to provide information to the App Store when the customer initiates a purchase for each subscription SKU you offer. After a successful purchase, Apple sends customers a subscription confirmation email and a receipt. Customers can go to their Apple Account Subscription Settings to manage their subscription at any time.



The following details appear in the App Store payment sheet, email communications from Apple, and the customer's Apple Account Subscription Settings; you set the values of these details in [SubscriptionCreateRequest](#):

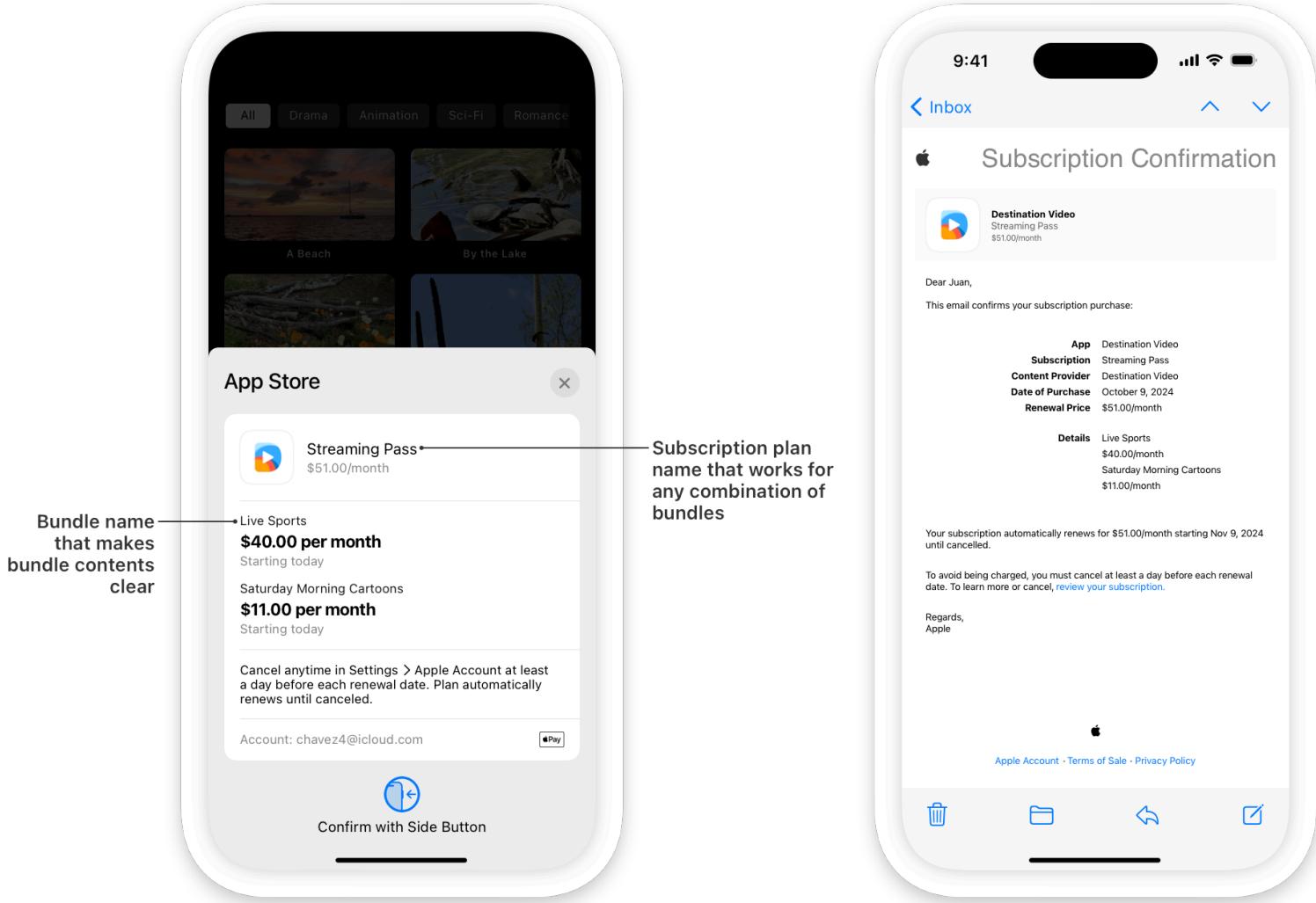
- descriptors.displayName: The name of the subscription the customer is purchasing.
- items.displayName: The tier of the subscription service the customer is purchasing, which needs to be different from your descriptors.displayName. For example, if your app includes creator subscriptions and a creator offers multiple subscription tiers, the displayName represents the tier the customer is purchasing.
- items.price: The price of the item the customer is purchasing.
- period: The duration of the subscription's billing cycle.
- currency: The currency that your app uses to charge the customer.

- App link: A dedicated link for customers to manage their subscription within your app. This link appears in the customer's Apple Account Subscription Settings as a "Manage in [Your App Name]" button. For setup information, see [Setting up a link to manage subscriptions](#). You must also provide a way for customers to [resubscribe](#) to a subscription that has expired or has automatic renewals turned off.



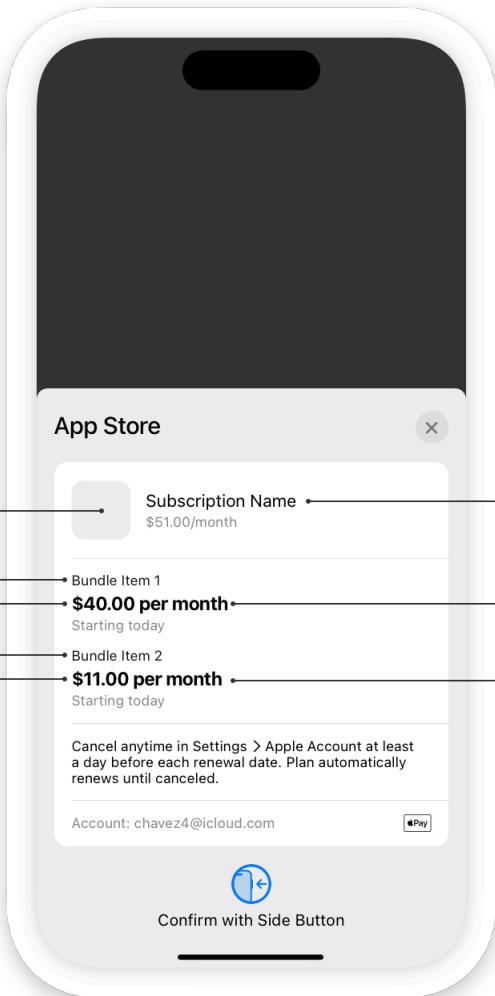
Create bundled subscriptions

For subscription-specific services that are bundled with additional add-on content or services (all of which auto-renews as a single subscription), create a SKU for each service or content offering. Use the [SubscriptionCreateRequest](#) API to provide information to the App Store when the customer initiates a purchase. After a successful purchase, Apple sends customers a subscription confirmation email and a receipt. Customers can go to their Apple Account Subscription Settings to manage their subscription at any time.

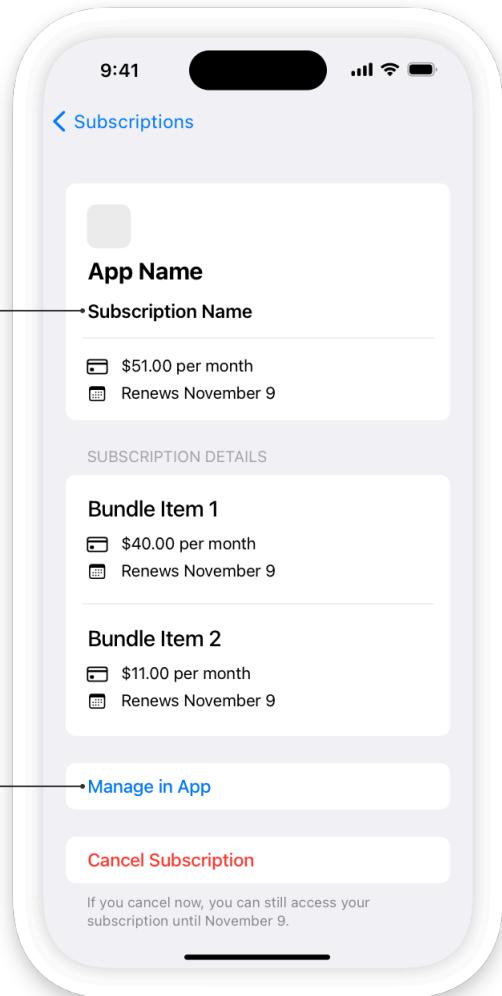


The following details appear in the App Store payment sheet, email communications from Apple, and customer's Subscription Settings; you set the values of these details in [Subscription CreateRequest](#):

- descriptors.displayName: The name of the subscription SKU a customer is purchasing. This needs to represent your overall subscription as well as be relevant for any combination of SKUs someone might choose to bundle.
- items.displayName: The name of the add-on a customer is purchasing. For example, for a streaming app with multiple channel add-ons, this string might be "Live Sports".
- items.price: The price for each add-on within the bundled subscription purchase. When your app or email displays multiple bundle items, it sorts them from highest to lowest price.
- period: The duration of the subscription's billing cycle.
- currency: The currency that your app uses to charge the customer.
- App link: A dedicated link for customers to manage their subscription within your app. This link appears in the customer's Apple Account Subscription Settings as a "Manage in [Your App Name]" button. For setup information, see [Setting up a link to manage subscriptions](#). You must also provide a way for customers to [resubscribe](#) to a subscription that has expired or has automatic renewals turned off.



Payment Sheet



Subscription Management Page

See Also

Essentials

- 📄 Setting up your project for Advanced Commerce API
Configure your app in App Store Connect, set up your server, and prepare your SKUs.
- 📄 Setting up a link to manage subscriptions
Create a deep link to a subscription-management page for your app.
- 📄 Advanced Commerce API changelog
Learn about new features and updates in the Advanced Commerce API.