

Configuring for Google Play Store

Introduction

This guide describes the process of establishing the digital records and relationships necessary for a Unity game to interact with an In-App Purchase Store. The [Unity IAP](#) purchasing API is targeted.

In-App Purchase (IAP) is the process of transacting money for digital goods. A platform's Store allows the purchase of Products, representing digital goods. These Products have an Identifier, typically of string datatype. Products have Types to represent their durability: *subscription*, *consumable* (capable of being rebought), and *non-consumable* (capable of being bought once) are the most common.

Google Play Store

Getting Started

1. Write a game implementing Unity IAP. See [Unity IAP Initialization](#) and [Integrating Unity IAP with your game](#).
2. Keep the game's product identifiers on-hand for [Google Play](#) Developer Console use later.

```
public class MyStoreClass : MonoBehaviour, IStoreListener {  
    void Start() {  
        var module = StandardPurchasingModule.Instance();  
        ConfigurationBuilder builder = ConfigurationBuilder.Instance(module);  
        builder.AddProduct("com.unity3d.storeguidedemo.100gold", ProductType.Consumable);  
        UnityPurchasing.Initialize(this, builder);  
    }  
}
```

3. Build a [signed non-Development Build Android APK](#) from your game.

TIP: Make sure you safely store your **keystore**

file. The original **keystore** is always required to update a published Google Play application.

TIP: Reuse the Bundle Version Code from your last uploaded **APK**

during local testing to permit side-loading without first being required to upload the changed **APK** to the Developer Console. See the settings for the [Android platform Player](#).

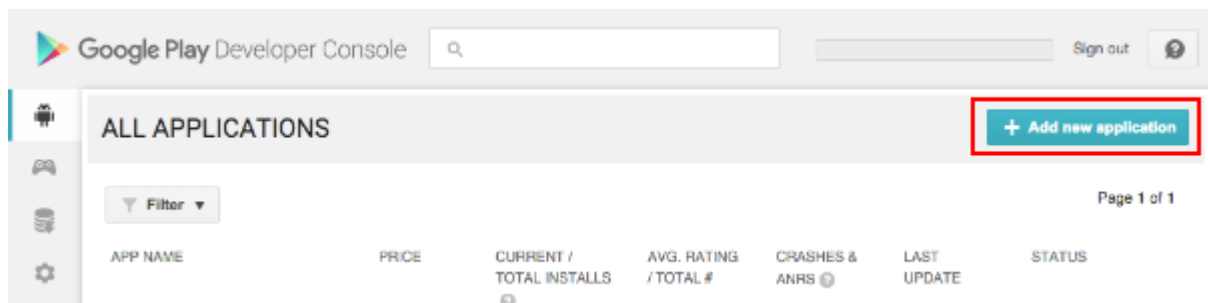
Register the Application

From the Google Account that will publish the game, register the Android application with the [Google Play Developer Console](#).

NOTE: This guide uses the [Google Play Alpha/Beta Testing approach](#) for testing in-app purchase integration.

NOTE: A restriction against using the publisher's Google Account for testing exists, because Google payments does not let you buy items from yourself. Therefore you will need to create a non-publisher Google Account for testing.

1. Choose **Add new application**.



2. Give the application a Title. For configuring application billing, choose “Upload APK” now. The Store Listing will be prepared and populated in a later step.

ADD NEW APPLICATION

Default language *

English (United States) – en-US

Title *

0 of 30 characters

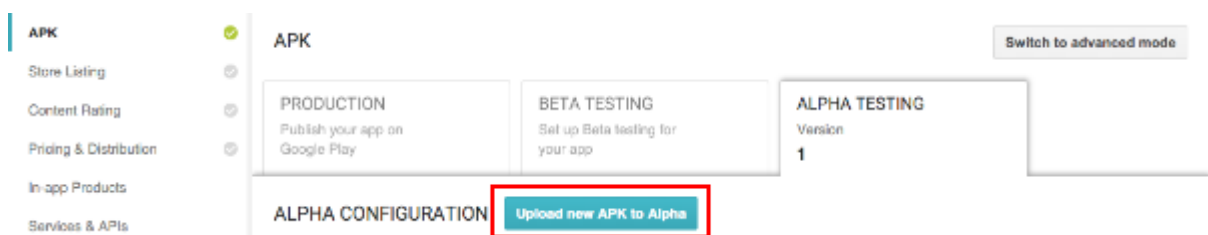
What would you like to start with?

Upload APK

Prepare Store Listing

Cancel

3. Navigate to **ALPHA TESTING** and choose **Upload your first APK to Alpha**. Select your APK and upload it.



4. Resolve any issues listed in the **Why can't I publish?** popup first, then click **Publish app** to publish your Alpha application.

TIP: There may be a delay of up to 24 hours while the published Alpha application is made available for download from the Google Play Store. Similarly, there may be a 24-hour delay to published changes to an application's In-App Purchases (see below).

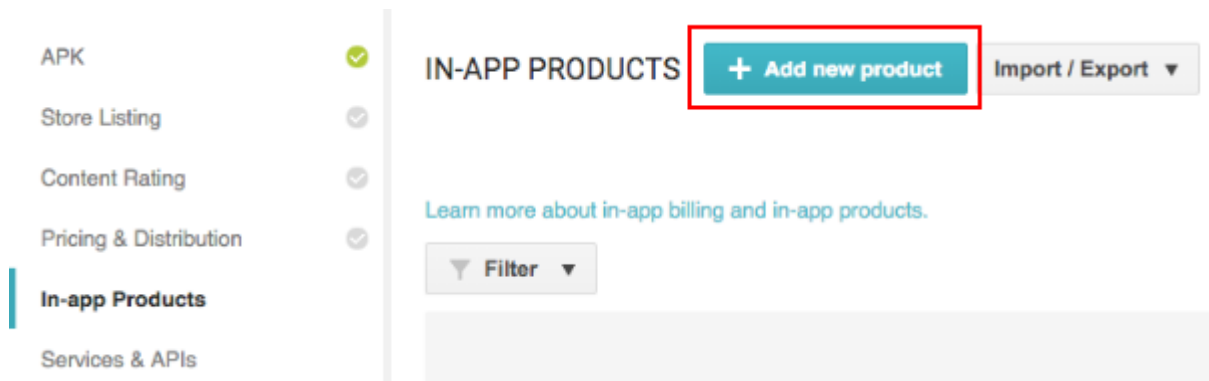
WARNING: Make sure the published application is the Alpha version. If the Production version is published, the general public will have access to the untested application.



Add In-App Purchases

In the Developer Console, add one or more in-app purchases for the game.

1. Navigate to **In-app Products** and choose **Add new product**.



2. Define the **Product ID** and choose **Continue**.

You can specify a consumable or non-consumable Product Type in **Managed product**. **Subscription** is also supported by Unity IAP.

NOTE: The "Product ID" here is the same identifier used in the game source code, added to the [Unity IAP ConfigurationBuilder](#) instance via `AddProduct()` or `AddProducts()`.

ADD NEW PRODUCT

What type of product would you like to add? *

Managed product

Subscription

Managed items that can be purchased only once per user account on Google Play. Google play permanently stores the transaction information for each item on a per-user basis.

[Learn more](#)

Product ID *

com.unity3d.storeguidedemo.100gold

34 of 136 characters

Please note that you can NOT change the product type and product ID later and that you cannot re-use the product ID again. [Learn more](#)

Continue

Cancel

1. Populate the IAP's **Title**, **Description**, and **Default price**. Then enable it by changing the **Inactive** button to **Active**.

MANAGED PRODUCT DETAILS

Fields marked with * need to be filled before activating.

English (United States) – en-US
Add translations

Title *
English (United States) – en-US
0 of 55 characters

Description *
English (United States) – en-US
0 of 80 characters

PRICING

Default price *
USD
 This price excludes tax.

Result:

APK	✓	IN-APP PRODUCTS + Add new product Import / Export ▾				
Store Listing	⊙	Learn more about in-app billing and in-app products.				
Content Rating	⊙	Filter ▾ Page 1 of 1				
Pricing & Distribution	⊙					
In-app Products						
Services & APIs						
Optimization Tips						
		NAMEID	PRICE	TYPE	LAST UPDATE	STATUS
		100 Gold (com.unity3d.storeguicedemo.100gold)	USD 0.99	Managed product	Dec 3, 2015	✓ Active
		Page 1 of 1				

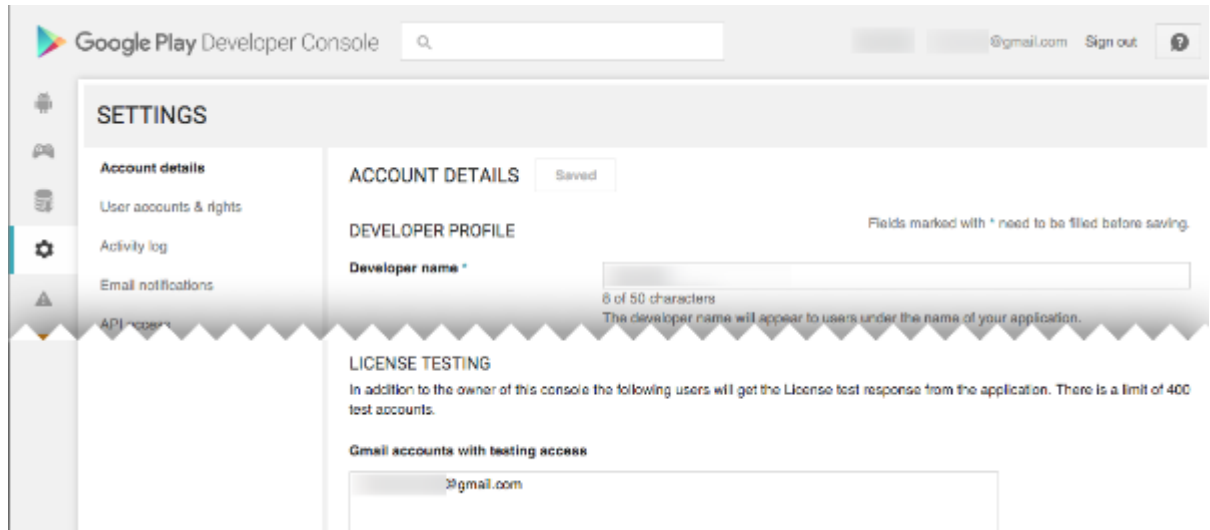
Test IAP

Follow these instructions to assign “test users” to be granted special license testing permissions, allowing them to purchase any IAP for free for this application, and register them as Alpha Testers.

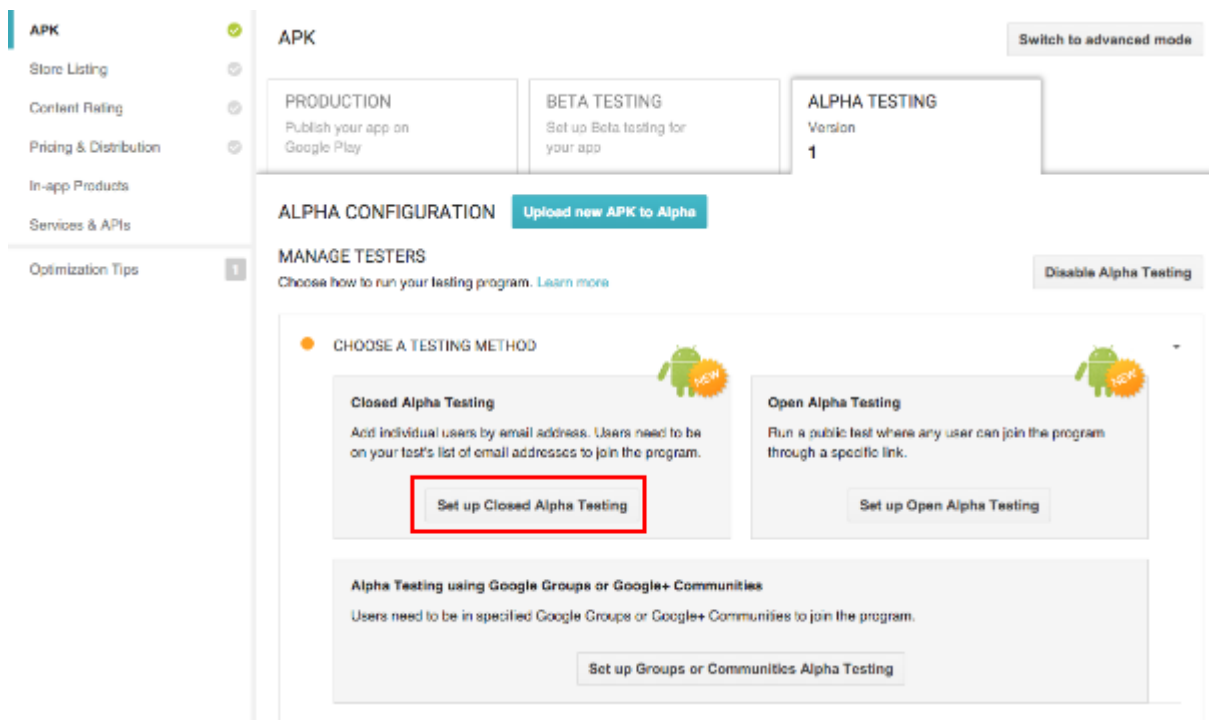
NOTE: Here we used [Closed Alpha Testing](#). Other tester groupings are available.

1. Add Google Account email addresses for testers who will be granted a license to purchase any IAP for free via the Developer Console's **Settings > Account details > LICENSE TESTING** section.

NOTE: There may be a delay of 15 minutes for the license testing setting to become effective.



2. Establish a list of Alpha Tester Google Account email addresses, from your list of license testing users.



Add each Google Account email address and name the list.

Create testers list

List name

Tester emails

Enter comma separated emails to add to the list.

Upload new CSV file

Save

Cancel

3. When available (after a multi-hour delay starting from publishing the APK) share the **Opt-in URL** with the Alpha Testers and have them install the application from the store.

NOTE: To test updates retaining permission to purchase IAP's for free, one may side-load applications, updating the existing store-downloaded APK install.

PUBLISHED Unpublish app

Statistics
User Acquisition
Finance
Ratings & Reviews
Crashes & ANRs
Optimization Tips
Cloud Test Lab

APK
Store Listing
Content Rating
Pricing & Distribution
In-app Products
Services & APIs

APK

PRODUCTION
Publish your app on Google Play

BETA TESTING
Set up Beta testing for your app

ALPHA TESTING
Version **1**

ALPHA CONFIGURATION [Upload new APK to Alpha](#)

MANAGE TESTERS
Choose how to run your testing program. [Learn more](#)

CLOSED ALPHA TESTING [Switch to another type of alpha testing](#)

[Create list](#)

After you create a list, you can reuse the list for Closed Testing with any of your published apps.

ACTIVE	LIST NAME	NUMBER OF TESTERS
<input checked="" type="checkbox"/>		1

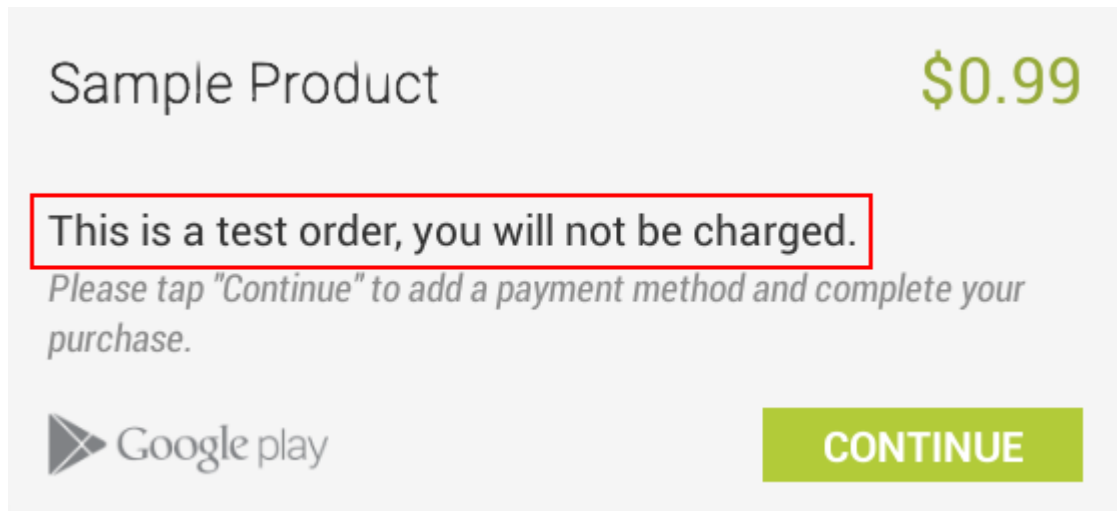
Feedback Channel [?](#)

Opt-in URL
Share this opt-in link with your testers.

<https://play.google.com/apps/testing/>

- To test the IAP, make a purchase on a device logged in with an Alpha Tester Google Account. A modified purchase dialog box appears to confirm the fact this product is under test and is free.

WARNING: If this dialog box does not appear, then the Alpha Tester Google Account will be charged real money for the product.



Did you find this page useful? Please give it a rating:

• • • • • What kind of problem would you like to report?

- This page needs code samples
- Code samples do not work
- Information is missing
- Information is incorrect
- Information is unclear or confusing
- There is a spelling/grammar error on this page
- Something else

Is something described here not working as you expect it to? It might be a **Known Issue**. Please check with the Issue Tracker at issuetracker.unity3d.com.

@