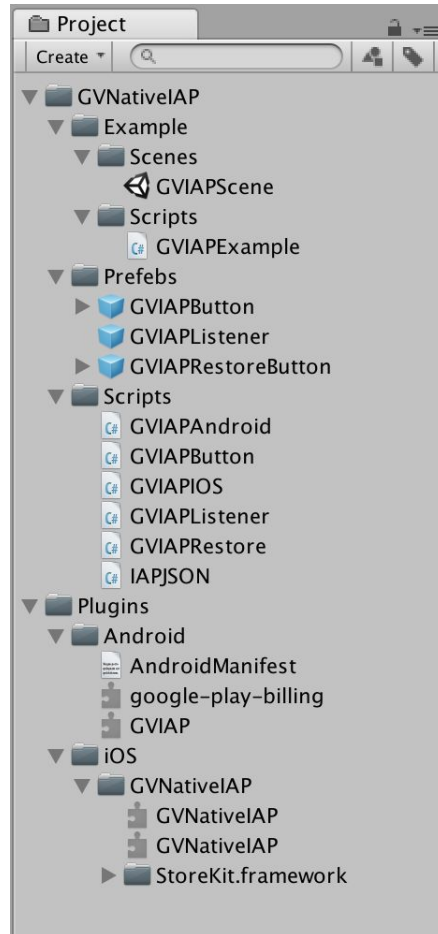


GVNativeIAP

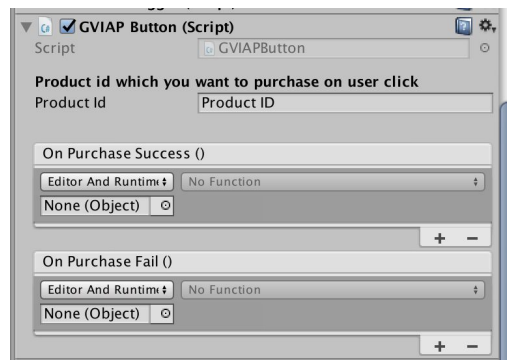
Steps to implement IAP (in-app purchase)

1. Add "Assets/GVNativeIAP/Prefabs/GVIAPListener" prefab into your splash screen (Or any starting screen)

(Note: Ad "GVIAPListener" prefab in that screen which only load one time during the lifetime of game. So recommended that, ad this prefab in splash screen)

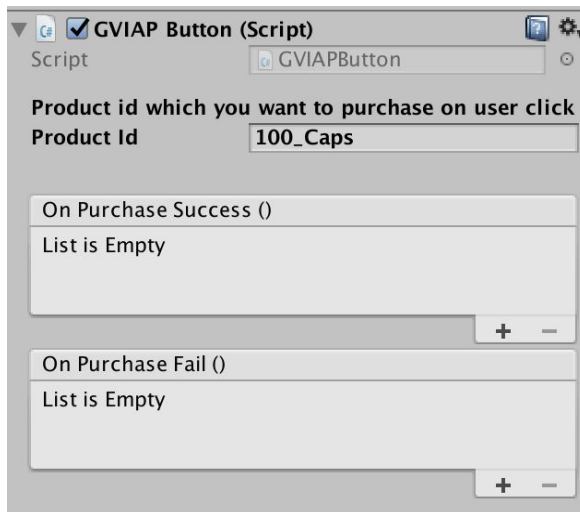


For Purchase Product

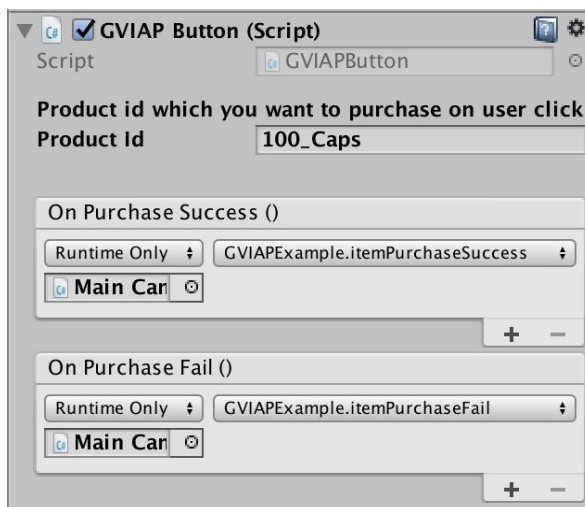


1. Add "Assets/GVNativeIAP/Prefabs/GVIAPListener/GVIAPButton" into your scene.

2. Change your purchasing product id under “GVIAP Button”



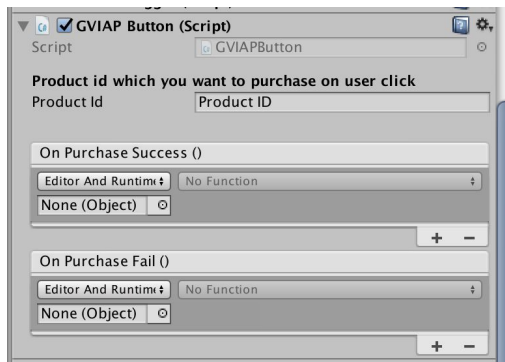
3. Add function which you want to execute when user purchase product successfully on “On Purchase Success”.
4. Add function which you want to execute when user can't purchase product successfully on “On Purchase Fail”.



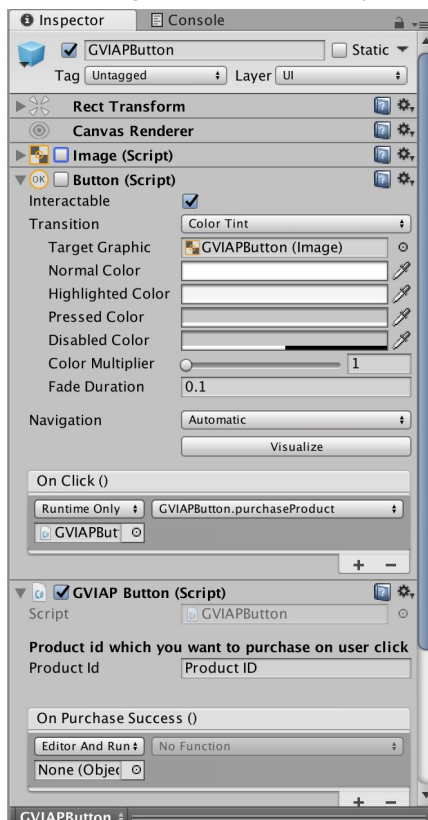
Custom Purchase Button

If you already added your custom purchase button and due to some reasons you cannot replace your custom purchase button with GVIAPButton, then follow following steps

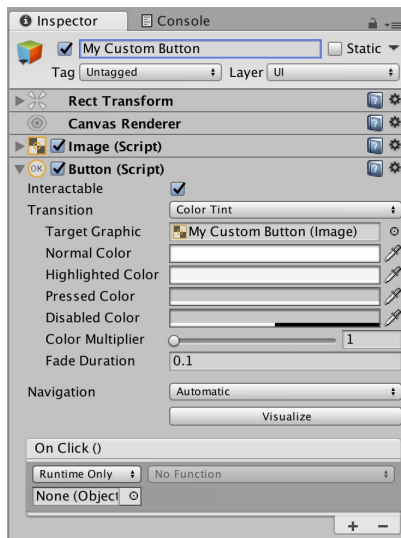
1. Add “Assets/GVNativeIAP/Prefabs/GVIAPListener/GVIAPButton” into your scene.



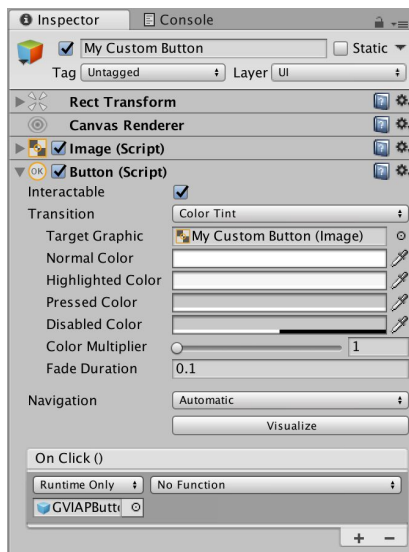
2. Change your purchasing product id under “GVIAP Button”.
3. Add function which you want to execute when user purchase product successfully on “On Purchase Success”.
4. Add function which you want to execute when user can’t purchase product successfully on “On Purchase Fail”.
5. Disable Image and Button property of GVIAPButton.



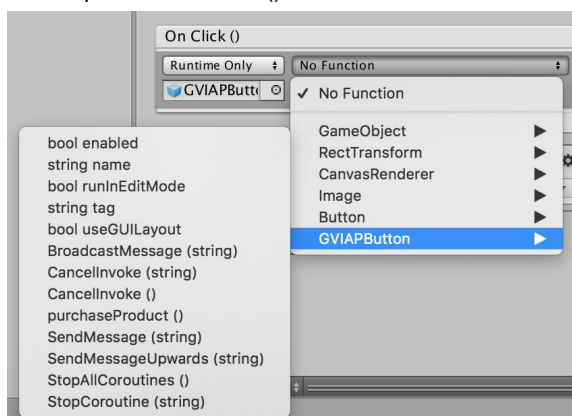
6. Click on '+' button of "OnClick()" event under Button property of your custom button.



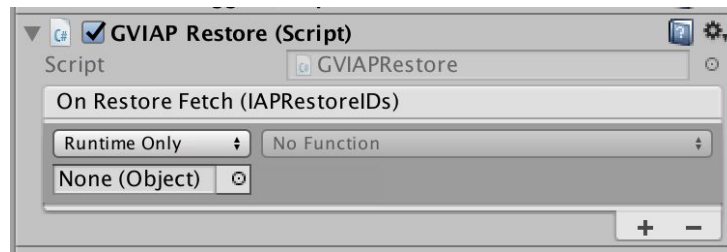
7. Drag GVIAPButton on "OnClick()" event function.



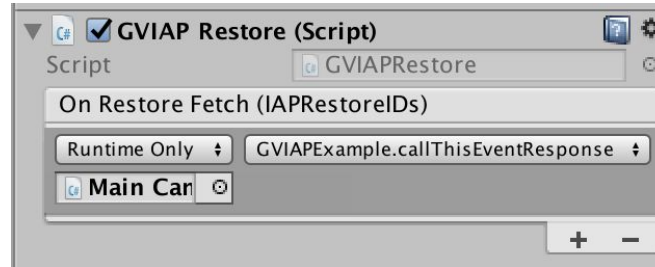
8. Select "purchaseProduct()" under GVIAPButton.



For Restore Products



1. Add "Assets/GVNativeIAP/Prefabs/GVIAPListener/GVIAPRestoreButton" into your scene.
2. Add function which you want to execute to restore purchased products on "On Restore Fetch".



(Note: Function which you want to execute when restore callbacks arrive with purchased product ids should have one parameter "IAPRestoreIDs")

```
public void callThisEventResponse(IAPRestoreIDs obj){  
    responseText.text = obj.ToString ();  
}
```

For Android Build

For Android build add following lines in "Assets/Plugins/AndroidManifest.xml"

1. Add Billing permission under <Manifest> tag and outside of <Application> tag.

<!-- Billing Permission -->

<uses-permission android:name="com.android.vending.BILLING" />

2. Add Billing Activity under <Application> tag

<activity

android:name="com.android.billingclient.api.ProxyBillingActivity"

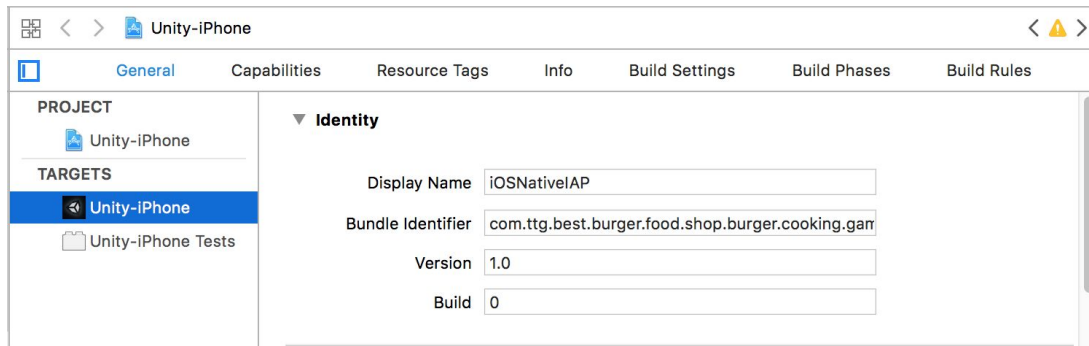
android:label="@string/app_name">

</activity>

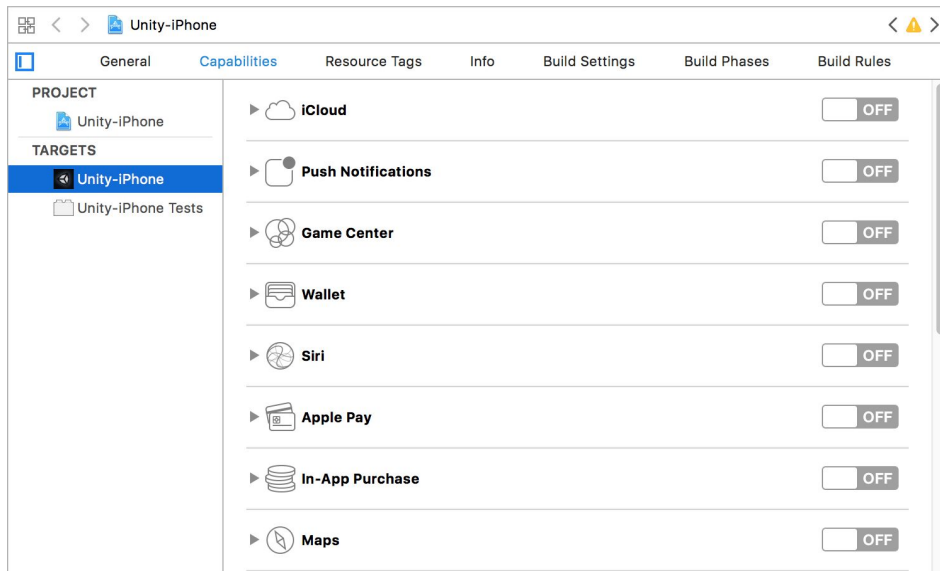
For iOS Build

For iOS build follow following steps

1. Click on "Unity-iPhone" under Targets. (Unity-iPhone is my project name, in your case may be something else).



2. Click on Capabilities.



3. Click on In-App Purchase.



Steps: ✓ Link StoreKit.framework
✓ Add the In-App Purchase feature to your App ID.