SmileSoft Ads Manager

With a single line of code, you can monetize your app using Google Admob ads.

Supported Ads -

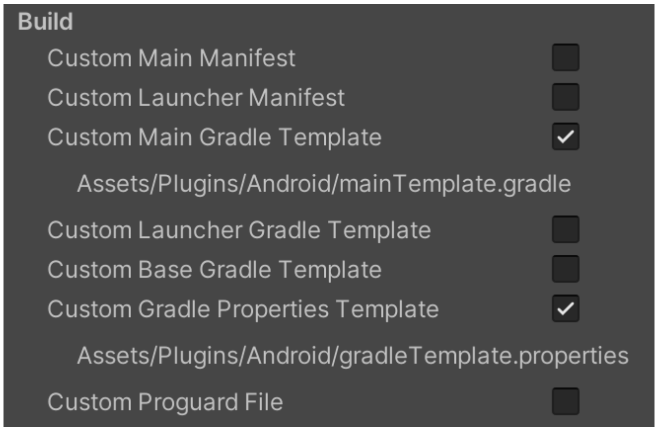
- Banner Ads
- Interstitial Ads
- Reward Ads

Project Setup -

1. Please install the latest Google Official Admob Plugin from here.

2. Go to **Project Settings > Player > Android > Publishing Settings > Build** and select

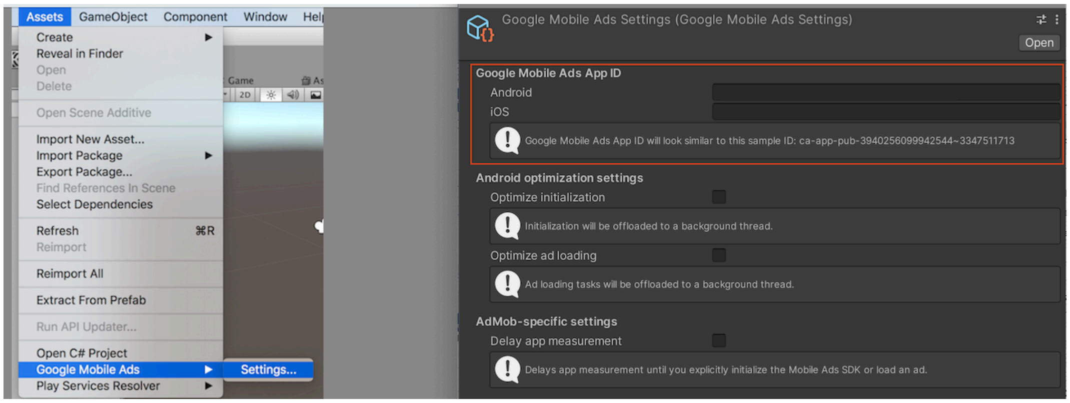
   — Custom Main Gradle Template
   – Custom Gradle Properties Template



Build

Custom Main Manifest ☐
Custom Launcher Manifest ☐
Custom Main Gradle Template ☑
Assets/Plugins/Android/mainTemplate.gradle
Custom Launcher Gradle Template ☐
Custom Base Gradle Template ☐
Custom Gradle Properties Template ☑
Assets/Plugins/Android/gradleTemplate.properties
Custom Proguard File ☐

3. In the Unity editor, select **Assets > Google Mobile Ads > Settings** from the menu and enter your Admob app ID for iOS and Android. Check the official documentation from here.



Google Mobile Ads Settings (Google Mobile Ads Settings)

Google Mobile Ads App ID

Android
iOS

Google Mobile Ads App ID will look similar to this sample ID: ca-app-pub-3940256099942544~3347511713

Android optimization settings

Optimize initialization ☐

Initialization will be offloaded to a background thread.

Optimize ad loading ☐

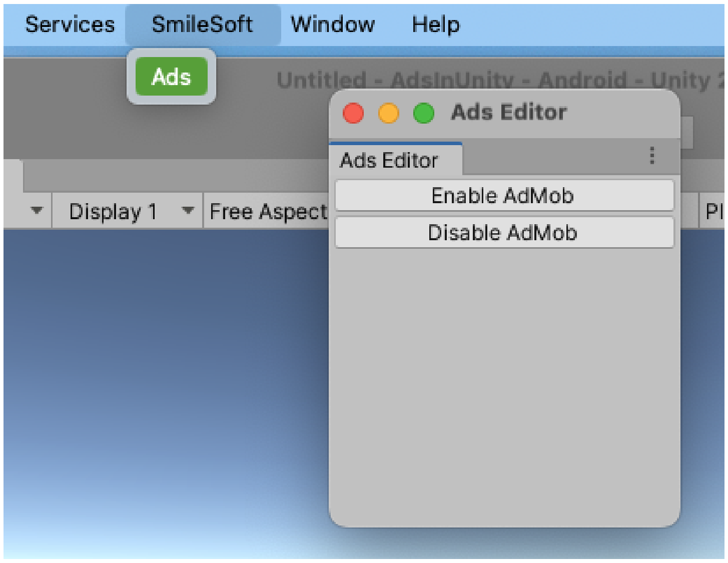
Ad loading tasks will be offloaded to a background thread.

AdMob-specific settings

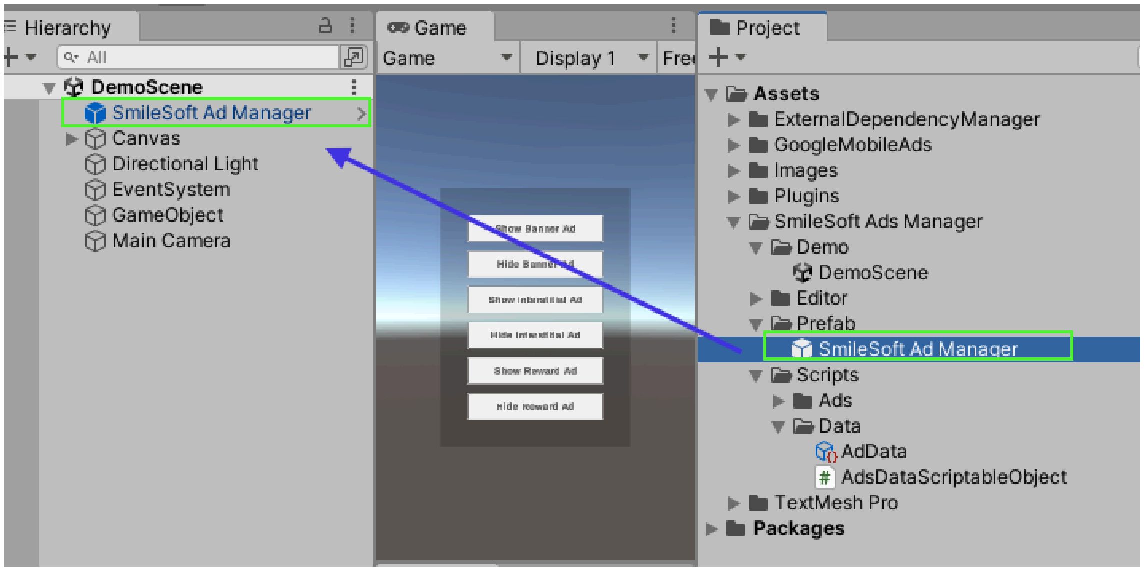
Delay app measurement ☐

Delays app measurement until you explicitly initialize the Mobile Ads SDK or load an ad.

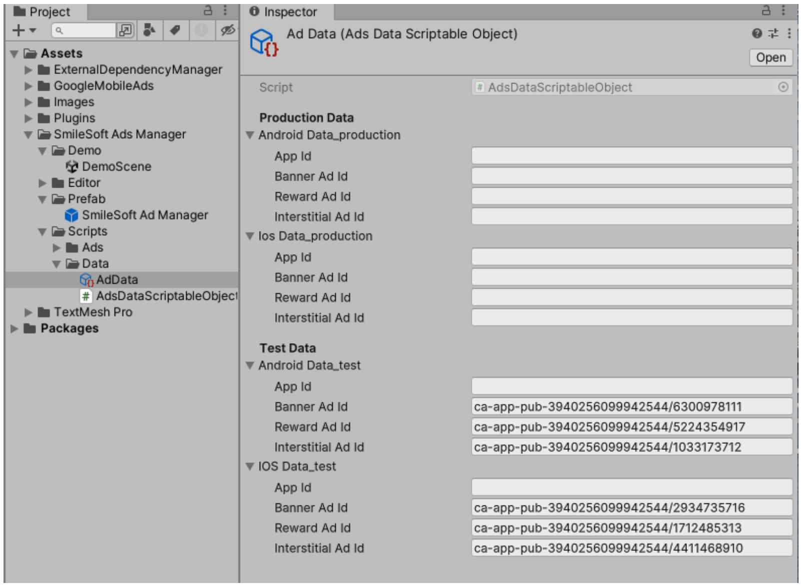
4. Now enable the Admob Ads from **SmileSoft > Ads > Enable Admob**.



5. Drag the SmileSoft **Ad Manager** game object from **Assets > SmileSoft Ads Manager > Prefab > SmileSoft Ad Manager** to your **scene** hierarchy.



6. Set all of your ads id from **Assets > SmileSoft Ads Manager > Scripts > Data > AdData** AdData has two parts one for production and the other for the test. Please put all of your ad IDs correctly. For the test, you can use the predefined test ad IDs but for the production please create your own ad IDs from here.



Ad Data (Ads Data Scriptable Object)

Script AdsDataScriptableObject

Production Data
Android Data_production
App Id
Banner Ad Id
Reward Ad Id
Interstitial Ad Id
Ios Data_production
App Id
Banner Ad Id
Reward Ad Id
Interstitial Ad Id

Test Data
Android Data_test
App Id
Banner Ad Id ca-app-pub-3940256099942544/6300978111
Reward Ad Id ca-app-pub-3940256099942544/5224354917
Interstitial Ad Id ca-app-pub-3940256099942544/1033173712
IOS Data_test
App Id
Banner Ad Id ca-app-pub-3940256099942544/2934735716
Reward Ad Id ca-app-pub-3940256099942544/1712485313
Interstitial Ad Id ca-app-pub-3940256099942544/4411468910

Great! You have completed the project setup successfully. Now just call some functions to show ads.

- ☐ **Banner Ads :**
  - ☐ **Show Banner Ad -> SmileSoftAdManager.instance.ShowBannerAd(AdSize size, AdPosition position);**

    Get available AdSize from here . Available **AdPositions** are **Top, Bottom, TopLeft, TopRight, BottomLeft, BottomRight, Center**.
  - ☐ **Hide Banner Ad -> BannerAdsController.instance.HideAD()**

- ☐ **Interstitial Ads :**
  - ☐ **Show Ad -> SmileSoftAdManager.instance.ShowInterstitialAd(Action<bool>)**

    This function takes a delegate as a parameter which will throw true if it can load the ads otherwise it will throw false.
  - ☐ **Hide Ad -> SmileSoftAdManager.instance.HideInterstitialAd()**

Example ->

```
SmileSoftAdManager.instance.ShowInterstitialAd(isSuccess =>  {     } );
```

- □ **Reward Ads :**

  - □ **Show Ad -> SmileSoftAdManager.instance.ShowAd(Action<string, Double, bool>)**

    — This function takes a delegate as a parameter.

    — This delegate will return a string ,double and boolean value. Here string is the reward type and double value is the reward amount.Third one will throw true if it can load the ads otherwise it will throw false.

  - □ **Hide Ad -> SmileSoftAdManager.instance.HideRewardAd()**

Please Request Ads first and after ads availability show the Ads.

Example ->

```
SmileSoftAdManager.instance.ShowRewardAd((receivedRewardType, receivedRewardAmount,
isSuccess) => {  } );
```

For better understanding please check the **DemoScene** from **Assets > SmileSoft Ads Manager > DemoScene** and **AdSceneManager.cs** from **Assets > SmileSoft Ads Manager > Scripts > Ads > AdSceneManager.cs**