
About EasyAds

EasyAds is a powerful plugin that allows you to easily integrate Ad Networks in your games/apps. It helps you to maximize your fill rate and increase your eCPM. EasyAds also sends ad requests to all the enabled networks to serve best ads to your users.

There's no coding required to use this plugin. You can easily monetize your game/apps with EasyAds. "Easy Ads" helps mobile apps/games developers to maximize their profit by monetizing multiple ad-networks with a simple click.

Supported Networks

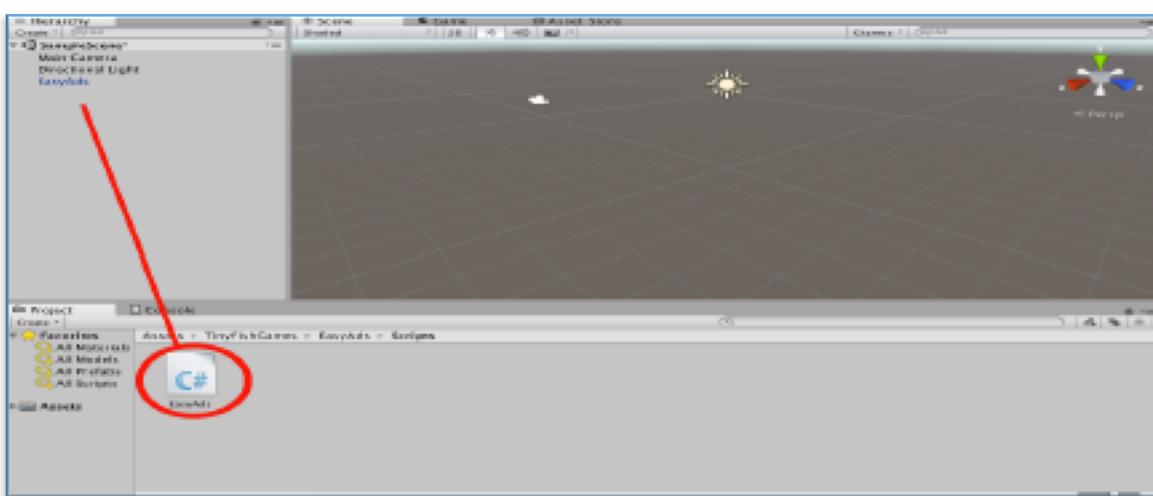
Network	Banner	Static Interstitial	Video Interstitial	Rewarded Video
Admob	✓	✓	✓	✓
Chartboost	X	✓	✓	✓
UnityAds	X	X	✓	✓
Leadbolt	X	✓	✓	✓
Vungle	X	X	✓	✓
RevMob	✓	✓	✓	✓

Supported Platforms

Platform	Supported
iOS	✓
Android	✓
Windows	✓

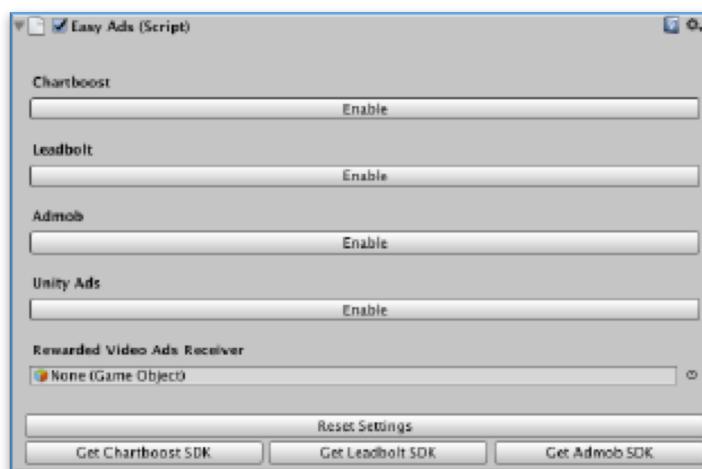
Plugin Configuration

1. **Download Ad SDK:** Download the SDK(s) of ad-networks, which you would like to use in your game. You can check the downloads section below which contains instructions how to download the required SDK(s).
2. **Enable Ad Networks:** After downloading the SDK, import ad-sdk to your unity project. After that, drag the prefab of EasyAds (*Assets->TinyFishGames->Prefabs->EasyAds.prefab*) to your scene.



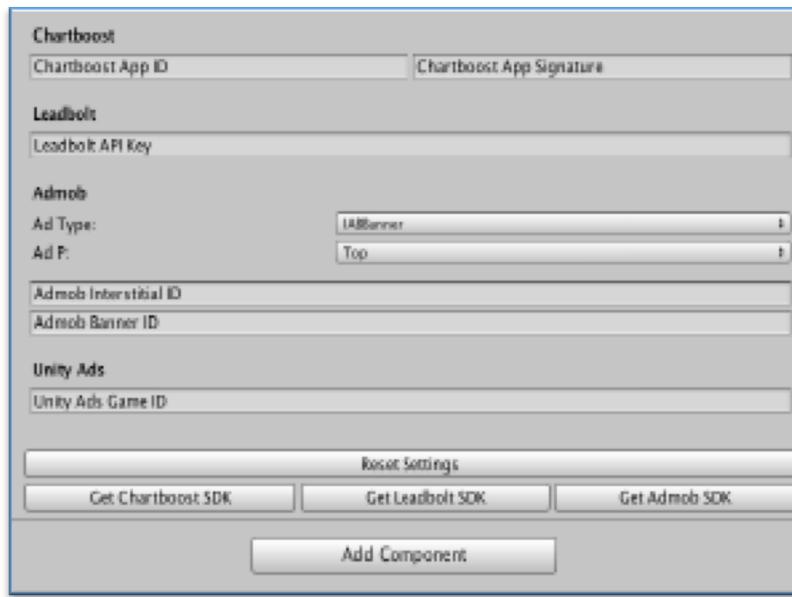
Note: EasyAds prefab is created only once and it is not destroyed when you change the scene.

Each ad network section contains an enable button, clicking that button will allow you to add your app id and chose ad positions.



Reset Settings: This button will remove all the settings and Ad ID's from EasyAds.

Note: It will not remove any SDK imported.



If you're using Revmob:

If you're using RevMob, then add IRevMobListener in EasyAds.cs file.
File Location: Assets -> TinyFishGames -> EasyAds -> Scripts -> EasyAds.cs

Eg.

```
public class EasyAds : MonoBehaviour, IRevMobListener
```

Calling Ad

Use the following code to call ads:

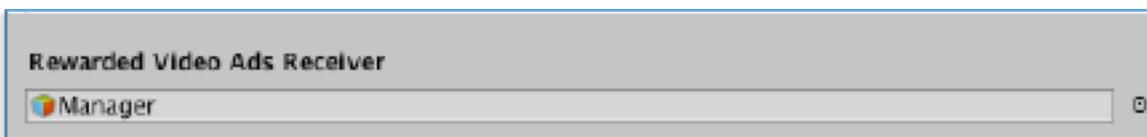
<code>EasyAds.Instance._showInterstitialAd();</code>	It will check for available ad-networks and will show the cached interstitial ad.
<code>EasyAds.Instance._showNonRewardedVideoAd();</code>	It will check for available ad-networks and will show the cached non-rewarded video ad.
<code>EasyAds.Instance._showRewardedVideoAd();</code>	It will check for available ad-networks and will show the cached rewarded video ad.
<code>EasyAds.Instance._showAdmobBanner();</code>	It will display Banner If available.
<code>EasyAds.Instance._hideAdmobBanner();</code>	It will Hide Banner Ad

Rewarded Video Ad Callback:

In order to know check if your user completely watched a rewarded video, add the following code to your class (any script):

```
public void rewardedVideoCompleted (string network) {  
    // Received Network Names...  
    // • UnityAds  
    // • Leadbolt  
    // • Chartboost  
}
```

You'll be required to pass the gameobject which contains the above code to EasyAds prefab.



Calling Specific Ad-Network:

Use the following code to call interstitial/video ad for specific ad-network [Chartboost, UnityAds, Leadbolt, Admob]

Chartboost:

You've to drag Chartboost prefab [Chartboost->Chartboost.prefab] to scene hierarchy in order to use chartboost sdk.

<i>EasyAds.Instance._isChartboostInterstitialAvailable()</i>	Check availability of Interstitial Ad
<i>EasyAds.Instance._isChartboostMoreAppsAvailable()</i>	Check availability of More Apps
<i>EasyAds.Instance._isChartboostRewardedVideoAdAvailable()</i>	Check availability of Rewarded Video Ad
<i>EasyAds.Instance._showChartboostInterstitial()</i>	Display Chartboost Interstitial
<i>EasyAds.Instance._showChartboostMoreApps()</i>	Display Chartboost More Apps
<i>EasyAds.Instance._showChartboostRewardedVideo()</i>	Display Chartboost Rewarded Video
<i>EasyAds.Instance._showChartboostVideo()</i>	Display Chartboost Non-Rewarded Video

Leadbolt:

For Android: You've to drag Leadbolt prefab [Plugins->Android->AppTrackerAndroid->AppTrackerAndroid.prefab] to scene hierarchy in order to use Leadbolt sdk.

For iOS: You've to drag Leadbolt prefab [Plugins->iOS->AppTrackeriOS->AppTrackeriOS.prefab] to scene hierarchy in order to use Leadbolt sdk.

<i>EasyAds.Instance._isLeadboltInterstitialAvailable()</i>	Check availability of Interstitial Ad
<i>EasyAds.Instance._isLeadboltVideoAvailable()</i>	Check availability of Video Ad
<i>EasyAds.Instance._showLeadboltInterstitial()</i>	Display Leadbolt Interstitial
<i>EasyAds.Instance._showLeadboltRewardedVideo()</i>	Display Leadbolt Rewarded Video
<i>EasyAds.Instance._showLeadboltVideoAd()</i>	Display Leadbolt Non Rewarded Video

Unity Ads:

<code>EasyAds.Instance._isUnityAdsAvailable()</code>	Check availability of Ad from UnityAds
<code>EasyAds.Instance._showUnityAdsRewardedVideo()</code>	Display UnityAds Rewarded Video Ad
<code>EasyAds.Instance._showUnityAdsVideoAd()</code>	Display UnityAds Non-Rewarded Video Ad

Admob:

<code>EasyAds.Instance._showAdmobBanner()</code>	Display Admob Banner Ad
<code>EasyAds.Instance._hideAdmobBanner()</code>	Hide Admob Banner Ad
<code>EasyAds.Instance._isAdmobInterstitialAvailable()</code>	Check availability of Admob Interstitial
<code>EasyAds.Instance._showAdmobInterstitial()</code>	Display Admob Interstitial Ad

Revmob:

If you're using RevMob, then add `IRevMobListener` in EasyAds.cs file.

File Location: Assets -> TinyFishGames -> EasyAds -> Scripts -> EasyAds.cs

Eg.

```
public class EasyAds : MonoBehaviour, IRevMobListener
```

<code>EasyAds.Instance._isRevmobInterstitialAvailable ()</code>	Check availability of RevMob Static Interstitial Ad
<code>EasyAds.Instance._showRevmobInterstitial ()</code>	Display RevMob static interstitial Ad
<code>EasyAds.Instance._isRevmobVideoAdAvailable ()</code>	Check availability of RevMob Video Ad
<code>EasyAds.Instance._showRevmobVideoAd ()</code>	Display RevMob Video Ad
<code>EasyAds.Instance._isRevmobRewardedVideoAvailable ()</code>	Check availability of RevMob Rewarded Video Ad
<code>EasyAds.Instance._showRevmobRewardedVideo ()</code>	Display RevMob Rewarded Video Ad
<code>EasyAds.Instance._showRevmobBanner ()</code>	Show RevMob Banner Ad
<code>EasyAds.Instance._hideRevmobBanner ()</code>	Hide RevMob Banner Ad

Vungle:

EasyAds.Instance._isVungleAdAvailable ()	Check the availability of Vungle Video Ad
EasyAds.Instance._showVungleVideoAd ()	Show Non-Rewarded Vungle Video Ad
EasyAds.Instance._showVungleRewardedVideo ()	Show Vungle Rewarded Video Ad

Vungle - For Windows Phone Only:

For Windows, install the Windows SDK before following the rest of these instructions to install the Vungle Unity Plugin. Windows development can only be performed on a Windows PC. Make sure you have the Windows SDK installed for the platform version you are using in development:

- Download Windows SDK 8.1 - <https://developer.microsoft.com/en-us/windows/downloads/windows-8-1-sdk>
- Download Windows SDK 10 - <https://developer.microsoft.com/en-us/windows/downloads/windows-10-sdk>

Under the **Project** tab, select the appropriate Vungle SDK. Confirm that the following parameters under the **Inspector** tab, in the **Select platforms for plugin** and **Platform settings** sections are properly set:

- For Assets/Plugins/metro/VungleSDKProxy.winmd:
Platform: WSAPlayer
SDK: Any SDK
Placeholder: Assets/Plugins/VungleSDKProxy.dll;
- For Assets/Plugins/metro/VungleSDK.winmd:
Platform: WSAPlayer
SDK: SDK81
- For Assets/Plugins/metro/UWP/VungleSDK.winmd:
Platform: WSAPlayer
SDK: UWP
- For Assets/Plugins/metro/WindowsPhone81/VungleSDK.winmd:
Platform: WSAPlayer
SDK: Phone SDK81

Important! If you are using Unity version 5.3.1 or higher to build the **Universal 8.1** or **Phone 8.1** project, you must disable the use of Assets/Plugins/metro/UWP/VungleSDK.winmd in one of these two ways:

- Either select Assets/Plugins/metro/UWP/VungleSDK.winmd and, under the **Inspector** tab, in the **Select platforms for plugin** section, clear the **WSAPlayer** option
- Or delete Assets/Plugins/metro/UWP/VungleSDK.winmd

After you publish the Windows project in Unity, make sure that your project has the **internetClient** capability in Unity environment. Open Build Settings (shift + ctrl + b) --> Select **Windows Store** under **Platform** --> click **Player Settings**. Under **Publishing Settings**, in the **Capabilities** menu, select the **InternetClient** option

Platform Settings - Android

- Include the following permissions in your AndroidManifest.xml file [Plugins->Android->AndroidManifest.xml]

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

- If you're using Leadbolt, then add the following code in AndroidManifest.xml file [Plugins->Android->AndroidManifest.xml]

```
<activity android:configChanges="keyboard|keyboardHidden|orientation|screenSize"
    android:name="com.apptracker.android.module.AppModuleActivity"
    android:hardwareAccelerated="false">
    </activity>
    <service android:name="com.apptracker.android.track.AppTrackerService" />
```

- If you're using Revmob Banner, then add the following code in AndroidManifest.xml file [Plugins->Android->AndroidManifest.xml]

```
<activity android:name="com.unity3d.player.UnityPlayerNativeActivity"
    android:label="@string/app_name"
    android:configChanges="keyboard|keyboardHidden|orientation|screenSize">
    <meta-data android:name="android.app.lib_name" android:value="unity" />
    <meta-data android:name="unityplayer.ForwardNativeEventsToDalvik"
        android:value="true" />
    </activity>
```

- If you're using Revmob Static Interstitial, then add the following code in AndroidManifest.xml file
[Plugins->Android->AndroidManifest.xml]

```
<activity android:name="com.unity3d.player.UnityPlayerNativeActivity"  
    android:label="@string/app_name"  
    android:configChanges="keyboard|keyboardHidden|orientation|screenSize">  
    <meta-data android:name="android.app.lib_name" android:value="unity" />  
    <meta-data android:name="unityplayer.ForwardNativeEventsToDalvik"  
        android:value="true" />  
</activity>
```

- If you're using Revmob Video Interstitial OR Rewarded Video Interstitial, then add the following code in AndroidManifest.xml file
[Plugins->Android->AndroidManifest.xml]

```
<application>  
    <activity  
        android:name="com.revmob.FullscreenActivity"  
        android:configChanges="keyboardHidden|orientation" >  
    </activity>  
</application>
```