# All In One - Simple Mobile Ads (adMob – AppLovin – ChartBoost)

Overview:

This package provides an easy solution for the implementation of the three most popular mobile ad-networks.

By using All In One you will never spend countless hours trying to implement the ad network plugins without getting compatibility errors.

You can setup the mobile ad networks in 10 minutes, without any coding skills, it is easy as it sounds.
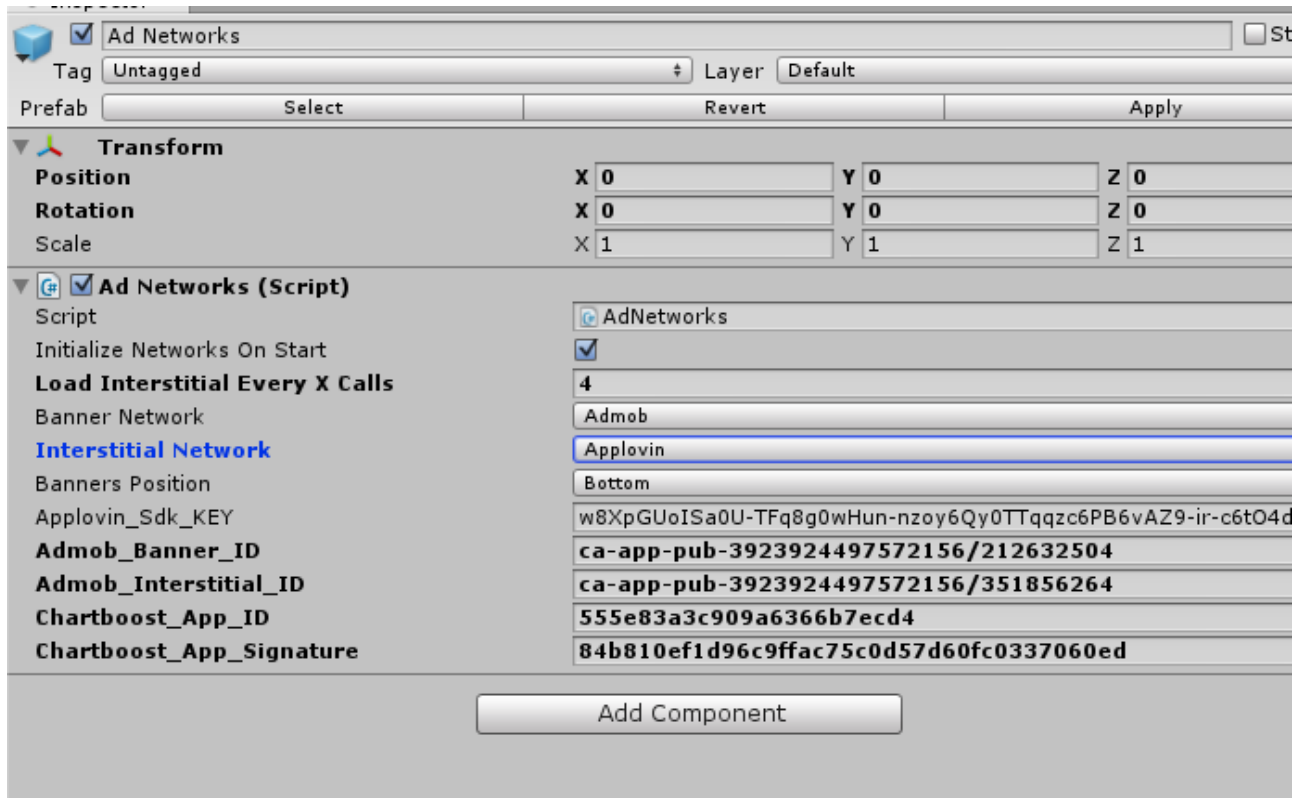
How to setup:

1) Create an empty GameObject and insert AdNetworks script inside.

2) Select your banner network and interstitial network.

3) Fill the ID fields.

  Example:

If you use only adMob just insert your adMob bannerID and your adMob interstitial ID.

Calling the functions:

The mobile ads will be shown only if you build and test your application on a mobile device, if an AD doesn't show please check your ID and the configuration on the respective ad network platform.

1) In the class where you want to use the ad networks just Initialize a new AdNetworks object like the screenshot below.

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class test : MonoBehaviour {
5     AdNetworks adNetworks;
6     // Use this for initialization
7     void Start () {
8         adNetworks = FindObjectOfType<AdNetworks> ();
9     }
```

2) **The functions available are:**

- **ShowBanner ();**

- **HideBanner();**

- **ShowInterstitial();**

Example:

To show an interstitial at the game over, just place adNetwork.ShowInterstitial() inside the function that handles your game over routine.

```
public void gameOver(){
    adNetworks.ShowInterstitial ();
    //YOUR GAMEOVER CODE HERE
}
```

You can also ShowInterstitial() every X calls. That means the Interstitial will be called only after a certain amount of requestes.

By default the variable is setted to 1.

In the example screenshot we setted it to 4 so the interstitial ad will be shown every 4 calls.

The class handles this for you.

Load Interstitial Every X Calls	4

Show the Banner Ads:

If you want to Show a Banner right after the initialization, it's important to add a Timer to make sure the Banner has been fully initialized. Expecially if you are planning to show it at the start of your game. For example you can add a Timer of 5 seconds like the screenshot below.

```
1 using UnityEngine;
2 using System.Collections;
3
4 public class test : MonoBehaviour {
5     AdNetworks adNetworks;
6     float bannerTimer = 5;
7     // Use this for initialization
8     void Start () {
9         adNetworks = FindObjectOfType<AdNetworks> ();
10    }
11
12    // Update is called once per frame
13    void Update () {
14
15        if (bannerTimer > 0) {
16        bannerTimer -= Time.deltaTime;
17            if (bannerTimer<=0)
18                adNetworks.ShowBanner();
19                        }
20
21    }
22
23
24
25 }
26
```

If you call the function right after FindObjectOfType<AdNetworks>(); You will get a Null Reference Exception because the Banners are not Initialized yet.

Conclusions:

That's it and thank you again for buying my products, if you have any questions please contact me at obliusgames@gmail.com