

# Admob Ads Callbacks Part-3

## Introduction

In this video, we will learn how to set up **AdMob Ads Callbacks** in your mobile game using **Mobile Monetization Pro**.

## Admob Ads - Banner & Interstitial Callback Event Example

Mobile Monetization Pro_Admob Ads Initializer (Script)	
Script	MobileMonetizationPro_AdmobAdsInitializer
Use GDPR Consent	<input checked="" type="checkbox"/>
<b>Ad unit Id's For Android</b>	
Android Banner Id	ca-app-pub-8589236751418541/9586352578
Android Interstitial Id	ca-app-pub-8589236751418541/8538786796
Android Rewarded Id	ca-app-pub-8589236751418541/1140886340
Android Native Id	ca-app-pub-8589236751418541/8758278641
Android App Open Id	ca-app-pub-8589236751418541/6407447478
Android Rewarded Interstitial ID	ca-app-pub-8589236751418541/1071360318
<b>Ad unit Id's For iOS</b>	
iOS Banner Id	ca-app-pub-8589236751418541/6881132330
iOS Interstitial Id	ca-app-pub-8589236751418541/6323924674
iOS Rewarded Id	ca-app-pub-8589236751418541/5112755099
iOS Native Id	ca-app-pub-8589236751418541/7046575875
iOS App Open Id	ca-app-pub-8589236751418541/3217219491
iOS Rewarded Interstitial ID	ca-app-pub-8589236751418541/2941887327
<b>Ads Settings</b>	
Show Banner Ads In Start	<input checked="" type="checkbox"/>
Choose Banner Position	Bottom
Use Adaptive Banner Size	<input checked="" type="checkbox"/>
Use Custom Adaptive Banner Width	<input type="checkbox"/>
Custom Adaptive Banner Width	0
Adaptive Banner Ad Size Options	Portrait
Enable Timed Interstitial Ads	<input checked="" type="checkbox"/>
Interstitial Ad Interval Seconds	10
Reset Interstitial Ad Timer On Rewarded Ad	<input checked="" type="checkbox"/>
<b>AppOpen Ads Settings</b>	
App Opens To Check Before Showing App Op...	3
Delay Show App Open Ad	2
<button>Add Component</button>	

```
```csharp using System.Collections; using UnityEngine;
```

```
public class AdmobAdEvents : MonoBehaviour { private  
MobileMonetizationPro.MobileMonetizationPro_AdmobAdsInitializer admobInitializer;
```

```
private void Start()  
{
```

```
    admobInitializer = MobileMonetizationPro.MobileMonetizationPro_AdmobAd
```

```

        if (admobInitializer != null)
        {
            StartCoroutine(ShowInterstitialAutomatically());
        }
        else
        {
            Debug.LogWarning("AdmobAdsInitializer instance not found!");
        }
    }

    private void OnTriggerEnter(Collider other)
    {
        if (other.CompareTag("Player") && admobInitializer != null)
        {
            admobInitializer.HideBanner();
        }
    }

    private void OnTriggerExit(Collider other)
    {
        if (other.CompareTag("Player") && admobInitializer != null)
        {
            admobInitializer.ShowBanner();
        }
    }

    private IEnumerator ShowInterstitialAutomatically()
    {
        yield return new WaitForSeconds(5f);
        if (admobInitializer != null)
        {
            admobInitializer.ShowInterstitialAd(true);
            admobInitializer.ResetInterstitialAdTimer();
        }
    }
}'''

```

## Admob Ads - Rewarded Callback Event Example

```
``csharp using UnityEngine; using UnityEngine.SceneManagement; using
UnityEngine.UI;
```

```
public class ProgressionBar : MonoBehaviour { public Image fillImage; // Image
component to fill public float fillSpeed = 0.1f; // Speed at which the fill completes
```

```
public Button LoadLevelButton;
```

```
private void Start()
```

```
{
```

```
    if (fillImage == null)
```

```
    {
```

```
        Debug.LogError("Fill Image is not assigned!");
```

```
        return; // Exit if no image assigned
```

```
    }
```

```
    fillImage.fillAmount = 0f; // Ensure starting from 0
```

```
    StartFilling(); // Start filling normally if assigned
```

```
}
```

```
private void Update()
```

```
{
```

```
    if(MobileMonetizationPro.MobileMonetizationPro_AdmobAdsInitializer.ins
```

```
    {
```

```
        // give reward
```

```
        LoadLevelButton.gameObject.SetActive(true);
```

```
        MobileMonetizationPro.MobileMonetizationPro_AdmobAdsInitializer.ins
```

```
    }
```

```
}
```

```
public void StartFilling()
```

```
{
```

```
    StartCoroutine(FillProgressionBar());
```

```
}
```

```
private System.Collections.IEnumerator FillProgressionBar()
```

```
{
```

```
    while (fillImage.fillAmount < 1f)
```

```
{
    fillImage.fillAmount += fillSpeed * Time.deltaTime; // Increment
    fillImage.fillAmount = Mathf.Clamp01(fillImage.fillAmount); // Cla
    yield return null; // Wait for next frame
}

// Fill completed
Debug.Log("Fill Completed!");

    if (MobileMonetizationPro.MobileMonetizationPro_AdmobAdsInitializer.i
        MobileMonetizationPro.MobileMonetizationPro_AdmobAdsInitializer.ins
}

}

...
```