

Pokkt integration to AdMob

Pokkt SDK version required: - 7.4.0

AdMob SDK version used: - 17.1.2

AdMob consent library used: 1.0.6

Step 1: Getting Started

The following instructions assume you have already integrated Google Mobile Ads SDK into your application.

Step 2: Adding Your Application to Your Pokkt Publisher's Account

- Create Account and log-in at <http://www.pokkt.com>
- Add your application and get its application Id and Security key.

Step 3: Pokkt Integration

GDPR

- Pokkt Solution adheres to GDPR requirement as per AdMob doc. Please refer section “Publisher-managed consent collection” at <https://developers.google.com/admob/android/eu-consent> and https://developers.google.com/admob/android/eu-consent#forward_consent_to_the_google_mobile_ads_sdk for understanding.
- GDPR consent must be taken by publishers before calling any ad related api. For more information on GDPR please refer <https://www.eugdpr.org/> and <https://www.eugdpr.org/gdpr-faqs.html>.

Dependencies

- Add PokktSDK_v7.4.0.aar or PokktSDK_v7.4.0.jar to your project.
- Add pokktsdk360ext.jar to your project.
- Download PokktCustomBanner, PokktCustomInterstitial and PokktCustomRewardedVideo from Pokkt site and change package as per your project structure. OR Download our Wrapper jar and place in your project.
- Add AdMob consent library to your project.

Manifest

Permissions Declarations

Add the following permissions to your project manifest

1. Mandatory permissions.

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

- android.permission.INTERNET = Required for SDK communication with server.
- android.permission.ACCESS_NETWORK_STATE = Required to detect changes in network, like if WIFI is available or not.

2. Optional permissions.

```
<uses-permission android:name="android.permission.WAKE_LOCK" />  
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />  
<uses-permission android:name="android.permission.WRITE_CALENDAR" />  
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />  
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />  
<uses-permission android:name="android.permission.CALL_PHONE" />  
<uses-permission android:name="android.permission.SEND_SMS" />  
<uses-permission android:name="android.permission.VIBRATE" />
```

- android.permission.WAKE_LOCK = Required to prevent device from going into the sleep mode during video play.
- android.permission.WRITE_EXTERNAL_STORAGE = Required to store media files related to ads in external SD card, if not provided we will use app cache folder to store media files, which will result in unnecessary increase in application's size. It is recommended to ask for this permission as low end devices generally have less internally memory available.
- android.permission.WRITE_CALENDAR = Some Ads create events in calendar.
- android.permission.ACCESS_COARSE_LOCATION" = Some Ads show content based on user's location.
- android.permission.ACCESS_FINE_LOCATION = Some Ads show content based on user's location.
- android.permission.CALL_PHONE = Some Ads are interactive and they provide you a way to call directly from the content.
- android.permission.SEND_SMS = Some Ads are interactive and they provide you a way to send message.
- android.permission.VIBRATE = Some Ads provide haptic feedback, so as to maintain their behaviour we need this permission

Activity Declaration

Add the following activity in your AndroidManifest for Pokkt SDK integration.

```
<activity  
  android:name="com.pokkt.sdk.PokktAdActivity"  
  android:configChanges="keyboard|keyboardHidden|navigation|orientation|screenLayout|uiMode|  
  screenSize|smallestScreenSize"  
  android:hardwareAccelerated="true"  
  android:label="Pokkt"  
  android:screenOrientation="landscape"  
  android:windowSoftInputMode="stateAlwaysHidden|adjustUnspecified" />
```

You can change the android screenOrientation landscape to portrait of your choice, the way you want to display the ads.

Service Declaration

Add the following service in your AndroidManifest for receiving InApp notifications.

```
<service  
  android:name="com.pokkt.sdk.notification.NotificationService"  
  android:exported="false"  
  android:label="PokktNotificationService" />
```

Step 4: Adding a Custom Event

1. Sign in to your AdMob account at <https://apps.admob.com>.
2. Click the Monetize tab.
3. Under All apps on the left-hand side, select the app you want to update.
4. Click the link in the Mediation column to the right of the ad unit you want to modify.

The screenshot shows the AdMob Monetize interface for the app 'PokktSampleApp' (Android). The top navigation bar includes 'HOME', 'MONETIZE' (selected), 'CAMPAIGNS', and 'ANALYZE'. A left sidebar contains a '+ MONETIZE NEW APP' button, 'Performance reports', and 'All apps' with 'PokktSampleApp' selected. The main content area has tabs for 'AD UNITS (3)', 'ALLOW & BLOCK ADS', and 'SETTINGS'. Below the 'AD UNITS (3)' tab, there is a '+ NEW AD UNIT' button and a table of existing ad units.

<input type="checkbox"/>	↑ Ad unit	Ad format	Mediation
<input type="checkbox"/>	Banner Ad unit ID: ca-app-pub- XXXXXXXXXXXXXXXXXXXX	Banner	2 ad sources
<input type="checkbox"/>	Interstitial Ad unit ID: ca-app-pub- XXXXXXXXXXXXXXXXXXXX	Interstitial	2 ad sources
<input type="checkbox"/>	Video Ad unit ID: ca-app-pub- XXXXXXXXXXXXXXXXXXXX	Rewarded video	2 ad sources

5. Click New Ad Network.

The screenshot shows the AdMob Monetization dashboard. The left sidebar contains a red button '+ MONETIZE NEW APP', 'Performance reports', and 'All apps' with a list of apps including 'PokktSampleApp' (Android). The main content area is titled 'Mediation' and shows the app 'PokktSampleApp' (Android) configured for 'Video' (Rewarded video). Below this, there are buttons for 'SAVE', 'CANCEL', and 'VIEW MEDIATION REPORT'. A section for 'Campaigns (0)' is collapsed. The 'Ad sources ordered by eCPM (2)' section is expanded, showing a table with two ad sources: 'AdMob Network (Optimized)' and 'Pokkt'. The 'Pokkt' source is selected and shows its class name and JSON string.

<input type="checkbox"/>	Ad source
<input type="checkbox"/>	AdMob Network (Optimized)
<input checked="" type="checkbox"/>	Pokkt Class Name: com.pokkt.wrapper.thirdparty.AdMob.PokktCustomRewardedVideo {"POKKT_SCREEN_NAME":"Test","POKKT_APP_ID":"a2717a45b835b5e9f50284e"}

6. Click Custom event, and provide the following details:

- **Class Name:** Enter the name of the custom event class that you will implement in your app code.

For Rewarded Video: Enter your package.PokktCustomRewardedVideo

For Interstitial: Enter your package.PokktCustomInterstitial

For Banner: Enter your package.PokktCustomBanner

If you are using our Wrapper jar, Custom event class name will be :

For Rewarded Video: com.pokkt.wrapper.thirdparty.AdMob.PokktCustomRewardedVideo

For Interstitial: com.pokkt.wrapper.thirdparty.AdMob.PokktCustomInterstitial


For Banner: com.pokkt.wrapper.thirdparty.AdMob.PokktCustomBanner

- **Label:** Enter the name for the custom event. This is used in reporting and cannot be changed later.

- **Parameter(Mandatory):** Enter a JSON string as follows :

```
{
  "POKKT_SCREEN_NAME":"Screen Created on Pokkt Dashboard",
  "POKKT_APP_ID":"Pokkt App Id",
  "POKKT_SEC_KEY":"Pokkt App Security Key",
  "POKKT_THIRD_PARTY_USERID":"Unique user id as in your app",
  "POKKT_REWARD_NAME":"Virtual currency As per your App",
  "POKKT_DEBUG":true/false
}
```

}

 AdMob

HOME

MONETIZE

CAMPAIGNS

ANALYZE

 PokktSampleApp
Android

>

Video
Rewarded video

7. Click Continue.

Step 6: Testing your application

Congratulation - that's it! You may now run your app and see Pokkt ads in action.

Note : It will take around 6 hours time to reflect the changes made in the AdMob dashboard.