

## Facebook Pokkt Mediation

Pokkt SDK version used: - 7.5.1

Facebook SDK version used: - 5.2.1

1. **Common Steps for Facebook Pokkt Mediation:** This is common documentation which is useful for all plugin.
2. **Native Integration:** Check the Steps Mentioned for Native.
3. **Xamarin Integration:** Check the Steps Mentioned for Xamarin.
4. **AIR Integration:** Check the Steps Mentioned for AIR.
5. **Marmalade Integration:** Check the Steps Mentioned for Marmalade.
6. **ShiVa Integration:** There is no separate documentation for this. Please follow "Common Facebook Pokkt Mediation" page.
7. **Cocos2d-x Integration:** There is no separate documentation for this. Please follow "Common Facebook Pokkt Mediation" page.
8. **Unity3D Integration:** There is no separate documentation for this. Please follow "Common Facebook Pokkt Mediation" page.
9. **Titanium Integration:** There is no separate documentation for this. Please follow "Common Facebook Pokkt Mediation" page.

# Common Facebook Pokkt Mediation

## Prerequisites

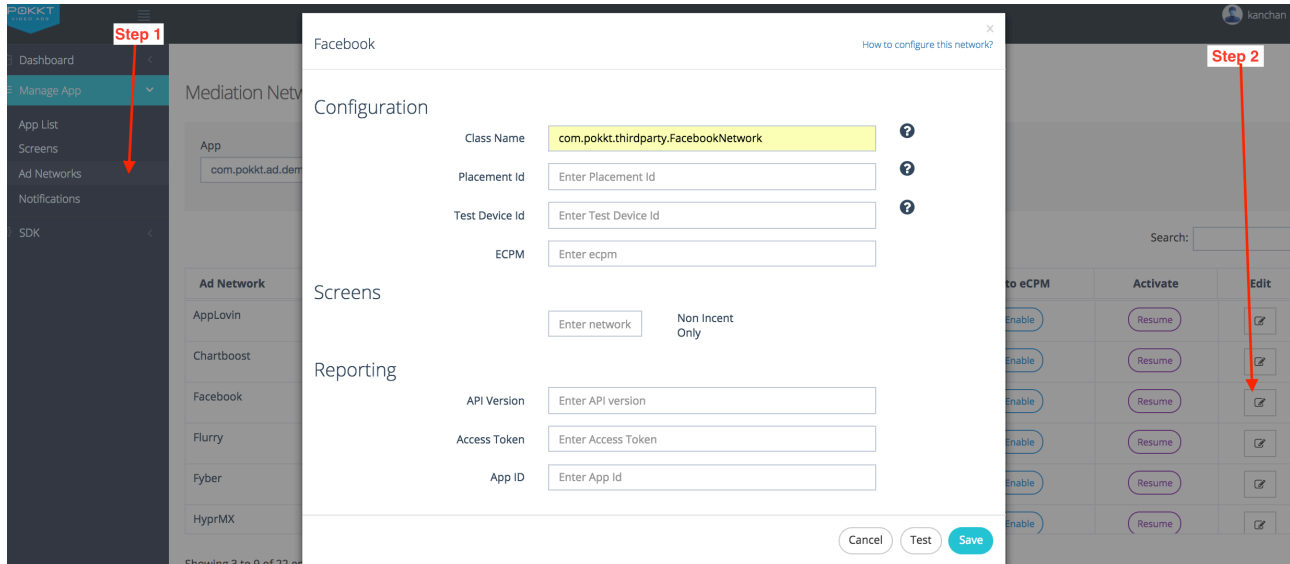
- Create account and login at <https://developers.facebook.com>
- Add an app in Facebook account and create placement under Audience network.
- Android Min SDK API 15 and up.

## GDPR

- Facebook will not serve ads if User has denied the GDPR consent to get personalised ads.

## Step 1: Configure Pokkt Dashboard

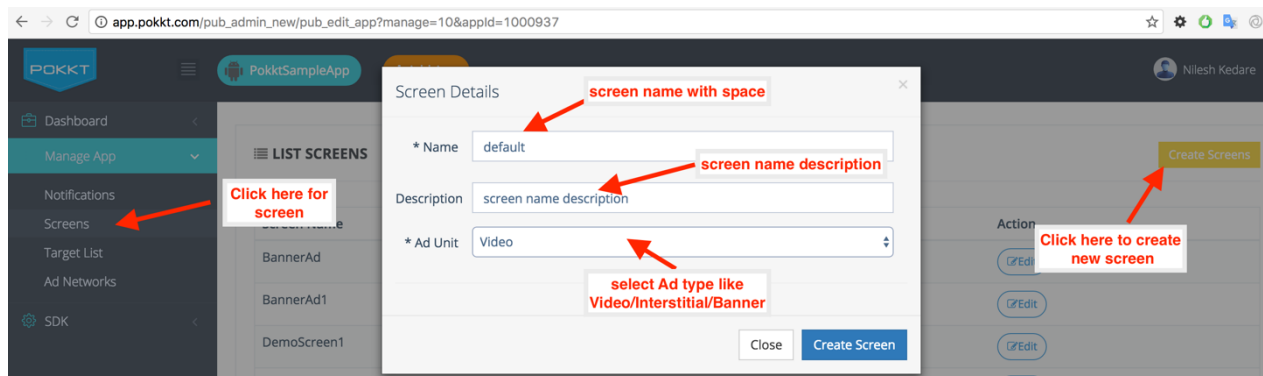
- Create Account and login at <http://www.pokkt.com>
- Register your app here if not registered.
- To use Facebook Mediation please enter the following details on Pokkt Dashboard:



- **Class Name (required):** This is the class name that you define inside your application to set up the network. Classname for the network can be set during SDK integration. If you are using Pokkt's default adaptors for mediation which is given in PokktMediation.jar, then class name will be com.pokkt.thirdparty.FacebookNetwork.
- **Test Device Id (optional):** For testing Facebook and if you want to see test ad for interstitial then please provide id of the device to use test mode, can be obtained from

debug log. You can add multiple device Id with comma (,) separated without any space.

- **ECPM** : Put your expected value of ECPM which will in turn rank this network in comparison to other configured networks.
- **Access Token** : Put App token from your Facebook account. App Access token can be found from <https://developers.facebook.com/tools/accesstoken/>
- **App Id** : Put Read only API key from your Facebook account dashboard.
- **API Version**: Put API version from your Facebook account dashboard. It will be mostly v2.5
- **Screens (required)**: Here need to provide “**Facebook Audience Network Placement Id**”. First create screen name in Pokkt dashboard which will reflect here. Now get the “**Facebook Audience Network Placement Id**”. Create new app in Facebook developer dashboard and open that app. There is a option “**Audience Network**” in left side so click that option and create new placement. Once placement has been created then you will see one placement Id which you need to provide here. This screen name will map with the placement Id.



**Notes:** For More Detail, Please follow the below link:

- For more details about Facebook mediation network integration with POKKT: [http://app.pokkt.com/mediation\\_help/Facebook.html](http://app.pokkt.com/mediation_help/Facebook.html)
- <https://developers.facebook.com/docs/android/getting-started>
- <https://developers.facebook.com/docs/audience-network/android>

---

## Step 2: Configuration in Application (Native)

- Download the mediation project from POKKT and it is recommended to use the same class file **FacebookNetwork** (FacebookNetwork.java, FacebookBannerManager.java and FacebookInterstitialManager.java) in your project. And if you want to do some changes please do those changes in same file. You have the option to make changes to these classes, in-fact you can create your own custom adaptor for Facebook using our protocol, just make sure to mention the exact name of class in the “class name” box, refer the [Step 1](#).
- Add '**com.facebook.android:audience-network-sdk:4.+**' to your gradle. Make sure to update your project's dependencies accordingly.
- Refer Facebook audience network integration documentation at <https://developers.facebook.com/docs/audience-network/android>
- Add the necessary permissions for Facebook as suggested in Facebook integration document to your application manifest.
- Add the recommended **activities** and **meta data** to your **AndroidManifest.xml** as suggested in Facebook integration guide.

## Xamarin Android

Follow the below steps to integrate Mediation in Xamarin android project:

- Add downloaded file **PokktMediation.dll** in your project under reference section.
- You need to download Facebook extension for Xamarin.
- Follow above all process to complete integration.

## AIR Android

Follow the below steps to integrate Mediation in AIR android project:

- Add downloaded file **PokktMediation.ane** in your project.
- Download Facebook ANE from below link and add this one in your project. This is just an idea if you want to use this otherwise you can find different .ane from other place.  
<https://github.com/Heyzap/facebook-audience-network-ane/releases>
- Google play service .ane file also required and few of the AdNetwork .ane comes with google play service .ane so be careful with this. We are giving PokktMediation.ane separate GooglePlayService.ANE separate.
- Follow above all process to complete integration and also check our given example specially app.xml file.

## Marmalade Android

Follow the below steps to integrate Mediation in Marmalade android project:

- Check downloaded file which has one mediation project “***PokktMediationExtension***” and this need to be added in your project as subproject in .mkb file.
- Facebook or Marmalade is not providing marmalade extension project so you need to create your own extension or you can take our help. We have created dummy extension for this testing purpose. We can help you on this.
- Add created Facebook extension project in your project as subproject.