

## NativeX Pokkt Mediation

Pokkt SDK version used: **6.2.1**

NativeX SDK version used: **5.5.9**

## Common Steps for NativeX Pokkt Mediation

### Prerequisites

- Create account and login at <https://selfservice.nativex.com/>
- Add your application.
- Read instructions at <https://help.nativex.com/display/revenue/Android+SDK+Integration+Guide> to understand integration of Android application with NativeX.

### Step 1: Configure Pokkt Dashboard

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

The screenshot displays the Pokkt Dashboard interface. On the left, a sidebar menu includes 'Dashboard', 'Manage App', 'Notifications', 'Screens', 'App List', 'Ad Networks', and 'SDK'. A red arrow labeled 'Step 1' points to the 'Ad Networks' option. The main content area shows a list of ad networks: InMobi, Loopme, NativeX, Supersonic, Tapjoy, and Unity3d. A modal window titled 'NativeX' is open, showing configuration details. The modal has a title bar with 'NativeX' and a link 'How to configure this network?'. The configuration fields are as follows:

Field	Value
App Id	114334
Reward Amount	5
Class Name	com.pokkt.thirdparty.NativeXNetwork
ECPM	5
stageFail	Enter network
Incent	<input checked="" type="checkbox"/>
Non Incent	<input checked="" type="checkbox"/>
Reporting Api Key	182bb9c315f24975b9588b118a0c3e97

At the bottom of the modal are buttons for 'Cancel', 'Test', and 'Save'. A red arrow labeled 'Step 2' points to the 'Save' button.

- **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.NativeXNetwork.
- **App Id (required):** Login to your NativeX account. The AppID is located directly to the right of your app name in the App ID column.

**nativeX** Monetization

Overview  
Apps  
**Monetization**  
Integrate Monetization  
Advertising  
Reports  
Account  
Help  
Settings | Logout

### Manage Monetization

Select App to Manage Monetization  
▼ SampleApp (Android) - 114334

Manage Placements Define Rewards Block Ads

This step allows you to define placements, which are points in your app where ads will be displayed. You can choose where, how many, and what type of ads appear in your app. For more information on placements and ad formats, check out our Knowledge Base.

To configure or view placement settings, simply click the placement name.

[+ Add Placement](#) All On Off

Active	Placement Name	Placement Info	Ad Units Enabled	Performance - <a href="#">View Reports</a> In The Last 7 Days
On	<a href="#">Level Completed</a>	<ul style="list-style-type: none"> <li>Single Offer</li> <li>Reward</li> </ul>	<ul style="list-style-type: none"> <li>Video</li> <li>Interstitial Image</li> </ul>	Publisher Commission: \$0.00 eCPM: \$0.00

- **Virtual Currency(required):** Provide reward value which user will get after the reward ad. This is the same value which you provide in NativeX dashboard.
- **Placement (required):** Map Pokkt screen name with the NativeX Placements here which will appear on NativeX dashboard under created app. Please check NativeX documentation for more details of NativeX placements. In each NativeX placement you can configure single ad unit to be either Video or Interstitial or both.

levelClear

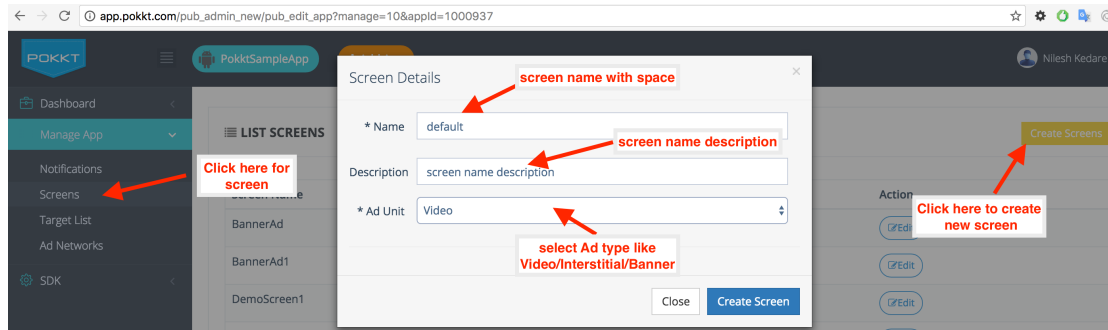
Level Completed

Incent ☒

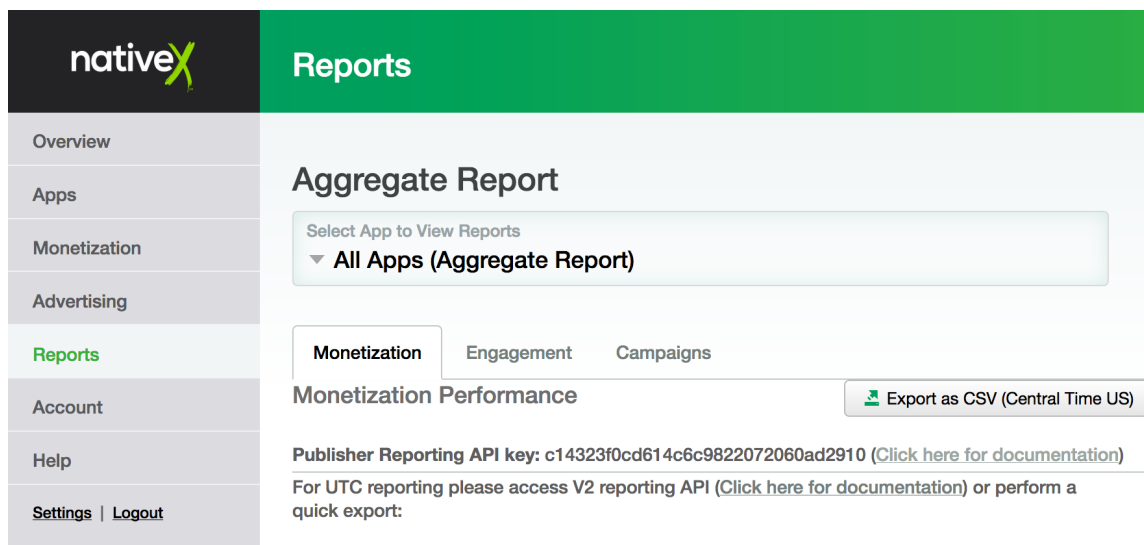
Non Incent ☒

Wednesday, 4 October 2017

- **Screen Creation:** For creating screen name, create screen name in Pokkt dashboard. Check below screen shot.



- **ECPM (required):** Put your expected value of ECPM which will in turn rank this network in comparison to other configured networks.
- Provide Reporting api key code and app id from NativeX account.



## Step 2: Configuration in Application (Native)

- Download the mediation project from POKKT and it is recommended to use the same class file **NativeXNetwork** (NativeXNetwork.java) in your project. You have the option to make changes to these classes, in-fact you can create your own custom adaptor for NativeX using our protocol, just make sure to mention the exact name of class in the “class name” box, refer the [Step 1](#).
- Add the “**NativeXMonetizationSDK\_v5.5.9.jar**” and “**gson-2.3.1.jar**” file into your project. Make sure to update your project's dependencies accordingly.
- Refer NativeX integration document at <https://help.nativex.com/display/revenue/Android+SDK+Integration+Guide>
- Add the necessary permissions for NativeX as suggested in NativeX integration document to your application manifest.
- Add the recommended **activities** to your **AndroidManifest.xml** as suggested in NativeX integration guide.
- Add the necessary proguard details as suggested in NativeX integration guide if your app needs it.