

## Avazu Native Ad SDK Setup for Android

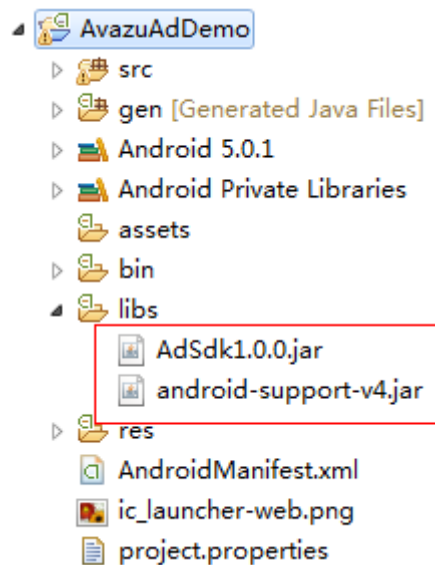
2015-01-14	1.0	Single banner ads Banner app list Button app wall App-installed ads Browser footer ads Desktop ads: lucky draw Desktop ads: recommended apps
------------	-----	--

<b>1. Development Preparation .....</b>	<b>3</b>
1) Import jar package.....	3
3) Configure AndroidManifest.xml.....	5
<b>2. Debug Switch (Optional).....</b>	<b>6</b>
<b>3. Main Functions.....</b>	<b>6</b>
1) SDK Initialization.....	6
2) Retrieve AdView .....	6
a) Instantiate AdViewController .....	6
b) Generate AdView.....	7
3) AdView Configuration .....	7
a) Customize App Elements .....	7
b) Customize Ad Color.....	7
c) Amount of Apps.....	8
d) Background Transparency .....	8
4) Activate the App-Installed Ads .....	8
5) Activate the Browser Footer Ads .....	8
6) Desktop Ads .....	8
a) Lucky Draw.....	8
b) Recommended Apps .....	8

# 1. Development Preparation

## 1) Import jar package

Copy “AdSdk1.0.0.jar” and “android-support-v4.jar” files to the “libs” directory of your project.

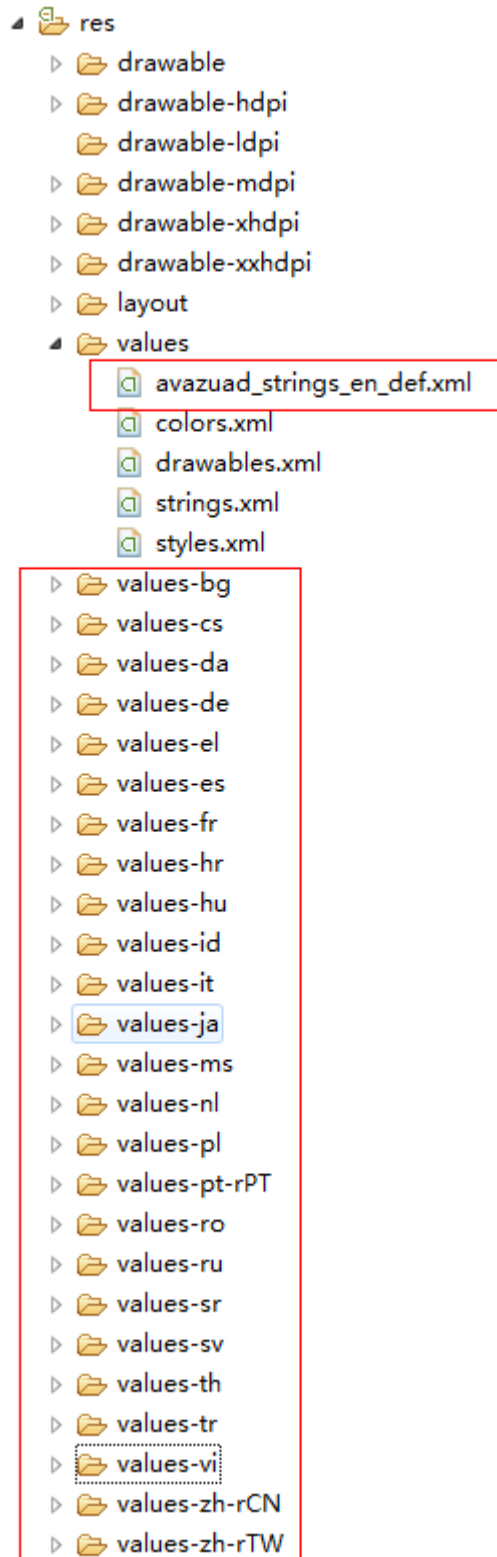


Note: If your project has already added “android-support-v4.jar”, do not add it twice.

## 2) Multiple Language Support

Add “avazuad\_strings\_en\_def.xml” to the “values” directory. This is the default language English.

Add other language resources to the “res” directory.



### 3) Configure AndroidManifest.xml

Add the following permissions to “AndroidManifest.xml”.

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.GET_TASKS"/>
<uses-permission android:name="com.android.launcher.permission.INSTALL_SHORTCUT"/>
<uses-permission android:name="com.android.launcher.permission.UNINSTALL_SHORTCUT"
/>
<uses-permission android:name="com.android.launcher.permission.READ_SETTINGS"/>
```

Add the following activities to “AndroidManifest.xml”.

```
<activity
    android:name="com.avazu.ad.AvazuActy"
    android:theme="@android:style/Theme.Dialog"
    android:screenOrientation="portrait"
    android:launchMode="singleInstance"
    android:excludeFromRecents="true" >
    <intent-filter >
        <action android:name="YOUR_PACKAGE_NAME.SHORT_CUT"/>
        <category android:name="android.intent.category.DEFAULT"/>
    </intent-filter>
</activity>
<service android:name="com.avazu.ad.AvazuWorkingService"></service>
<activity
    android:name="com.avazu.ad.AvazuDummyActy"
    android:excludeFromRecents="true"
    android:theme="@android:style/Theme.NoDisplay">
    <intent-filter >
        <action android:name="YOUR_PACKAGE_NAME.SHORT_CUT"/>
        <category android:name="android.intent.category.DEFAULT"/>
    </intent-filter>
</activity>
```

Note: Replace **YOUR\_PACKAGE\_NAME** with your own package name

## 2. Debug Model Switch (Optional)

Activate the debug model:

```
AvazuAdSdk.DEBUG = true;
```

## 3. Main Functions

### 1) SDK Initialization

This initialization function must be called at the startup of your application.

```
AvazuAdSdk.initialize(Context context, String sourceId);
```

sourceId: Traffic Source ID

### 2) Retrieve AdView

#### a) Instantiate AdViewController

```
AdViewController(int width_dip, int height_dip, int show_type, boolean  
should_show_loading)
```

width\_dip: Ad width in dip

height\_dip: Ad height in dip

show\_type: One of the following ad format types

AdViewController.TYPE\_RECT\_WALL: Multiple-line rectangle app wall

AdViewController.TYPE\_RECT\_SINGLE: Single-line rectangle app wall

AdViewController.TYPE\_BANNER\_WALL: Banner app wall

AdViewController.TYPE\_BANNER\_SINGLE: Single banner

AdViewController.TYPE\_BANNER\_TRANSPARENT: Single banner with transparent background

should\_show\_loading: The switch to show loading effect or not when the AdView is loading

## b) Generate AdView

Use the instantiated AdViewController object to generate AdView.

```
AdView adView = AdViewFactory.createAdView(Context context,  
AdViewController controller);
```

Place the generated AdView to the proper position in your application.

## 3) AdView Configuration

AdViewController also provides configuration options.

### a) Customize App Elements

Show app icon or not

```
AdViewController.setNeedIcon(boolean needIcon)
```

Show app title or not

```
AdViewController.setNeedTitle(boolean needTitle)
```

Show app ranking score or not

```
AdViewController.setNeedRating(boolean needRating)
```

Show app category or not

```
AdViewController.setNeedCat(boolean needCat)
```

Show app package size or not

```
AdViewController.setNeedSize(boolean needSize)
```

Show download button or not

```
AdViewController.setNeedBtn(boolean needBtn)
```

Show app installation number or not

```
AdViewController.setNeedInstalls(boolean needInstalls)
```

Show app review number or not

```
AdViewController.setNeedReviewNum(boolean needReviewNum)
```

### b) Customize Ad Color

All color value must be “#” followed by 6 hexadecimal RGB **digitals**. For example, #FFFFFF is the white color.

Set the background color of app wall

```
AdViewController.setMainBackColor(String mainBackColor)
```

Set the background color of the ad block

```
AdViewController.setBlockBackColor(String blockBackColor)
```

Set the font color of the app title

```
AdViewController.setAppTitleColor(String appTitleColor)
```

Set the font color of the download button

```
AdViewController.setButtonTextColor(String buttonTextColor)
```

Set the background color of the download button

```
AdViewController.setButtonBackColor(String buttonBackColor)
```

### c) Amount of Apps

For `TYPE_RECT_WALL`, `TYPE_RECT_SINGLE`, `TYPE_BANNER_WALL` only

Set the maximum amount of apps to show in the ad

```
AdViewController.setAppCount(int appCount)
```

### d) Background Transparency

For `TYPE_BANNER_TRANSPARENT` only

Set the transparency ratio of the background

```
AdViewController.setAlpha(int alpha)
```

alpha: should be between 0 (completely transparent) and 100 (completely opaque)

## 4) Activate the App-Installed Ads

```
AvazuAdSdk.setInstallHijackEnable(Context context, boolean enable)
```

enable: set "true" to activate the app-installed ads and "false" to deactivate it

## 5) Activate the Browser Footer Ads

```
AvazuAdSdk.setAppOpenHijackEnable(Context context, boolean enable)
```

enable: set "true" to activate the browser footer ads and "false" to deactivate it

## 6) Desktop Ads

### a) Lucky Draw

```
AvazuAdSdk.setTryLuck(Context context, String show_name, Bitmap show_icon)
```

show\_name: the title of the desktop shortcut. Default is "Lucky Draw"

show\_icon: the icon of the desktop shortcut. Default is a gift icon.

### b) Recommended Apps

```
AvazuAdSdk.setRecommendation(Context context, String
```



`show_name)`

`show_name`: the title of the desktop shortcut. Default is  
"Recommended Apps"