

MobFox Android SDK Integration Guide 4.1.6

The following document provides detailed instructions on how to integrate the MobFox Android SDK 4.1.6 into your Android Projects based on a simple Demo app. The MobFox SDK is capable of displaying both traditional Banner Ads and (Rich Media) Interstitials.

Step 1: Set Up your Application at the MobFox Dashboard

- Log into your MobFox Account or create an account at <http://www.mobfox.com/signup>
- At the MobFox Dashboard navigate to "Sites & Apps"
- Click on "Create New Site/App" and enter the details
- Once created you will be provided with a unique Publisher ID

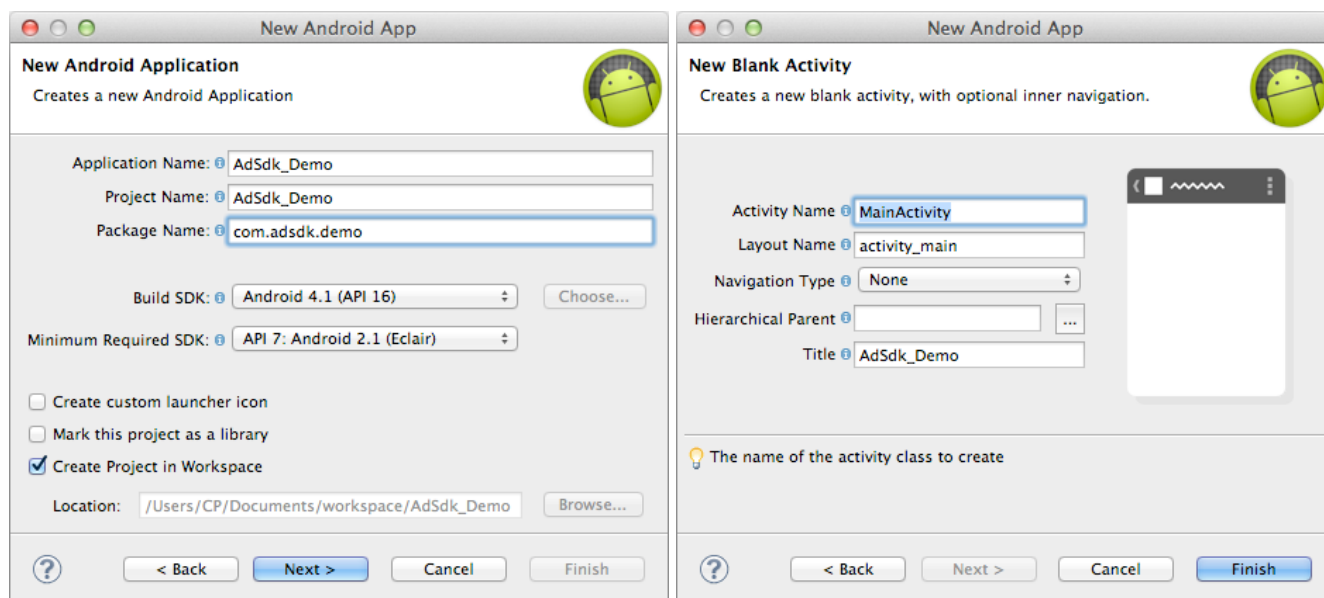
Step 2: Download the SDK

The downloaded ZIP contains the following files:

- PDF Documentation "MobFox Android SDK Setup Guide 4.1.6.pdf" (this document)
- The SDK Framework itself: AdSdk_4.1.6.jar
- Demo Application "AdSdk_Demo" folder:
 - Request Banner Ads and Video/Interstitial Ads with Buttons or using code
 - Based on Eclipse 4.2.2 with Android SDK Rev. 21.1, SDK Platform Android 4.2.2 Rev. 2 (API 17)
 - Some Coding and Customization options

Step 3: Create and define a new Android Project in Eclipse

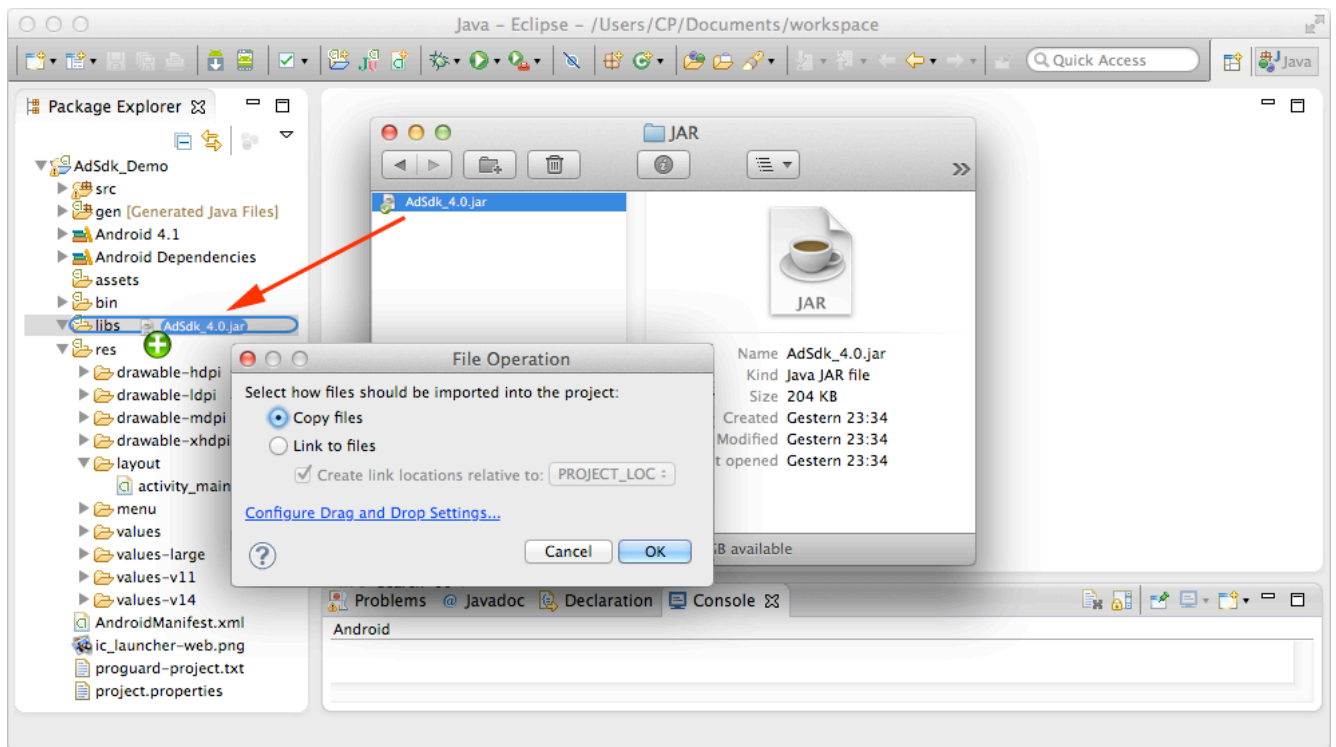
Download and open the latest version of Eclipse. Update the Android SDK Tools and Revisions listed at the Android SDK Manager. Now open the "Android Application Project" wizard and enter all needed fields as shown below. Set the minimum SDK to API 8 (Androids v2.2).



Step 4: Add the SDK to your project "libs" folder

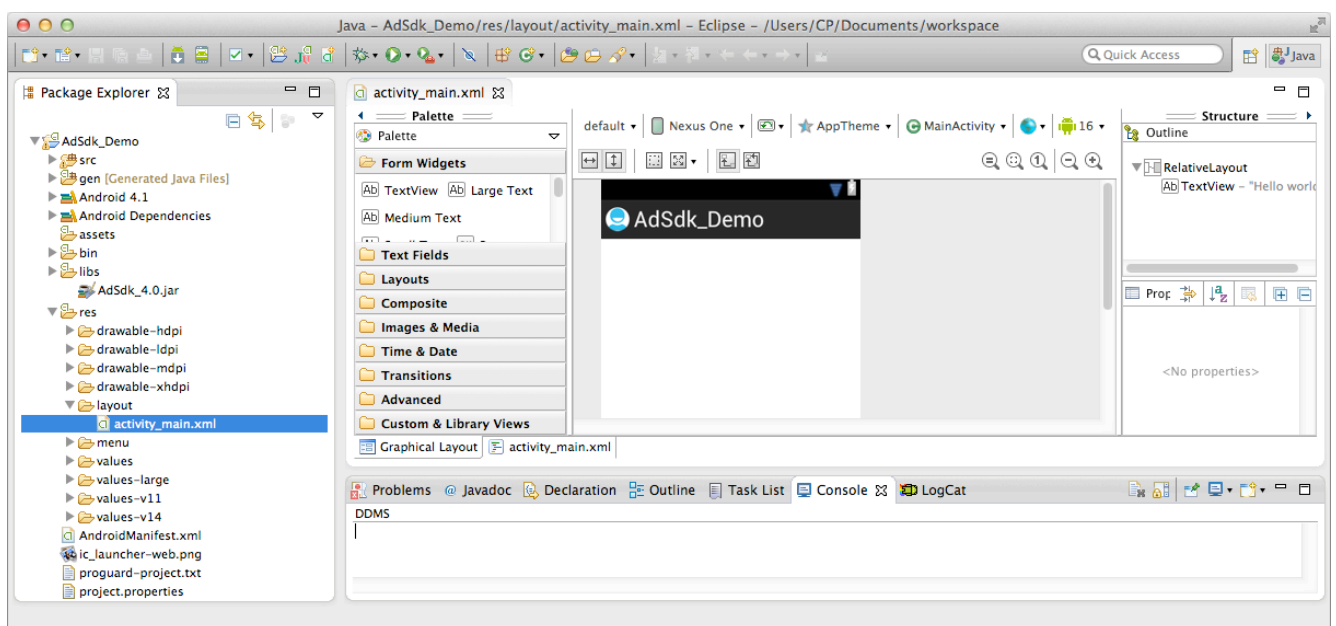
If you don't have a "libs" folder, create one by right clicking on the Project Root and select "New>Folder".

Copy the AdSdk_4.1.6.jar into your "libs" folder and select "Copy files" as shown below.



Note: If you are using SDK Rev 16 or earlier, you must additionally right click on Project, then Properties, go to "Java Build Path", select "Add JARs...", browse to your libs folder and select the AdSdk_4.1.6.jar. But we highly recommend to update your development tools to the latest version!

Please ensure the folder name is "libs". Your eclipse project should look as follows on SDK Rev 17 or later.



To display Banner Ads please continue with "Step 5".

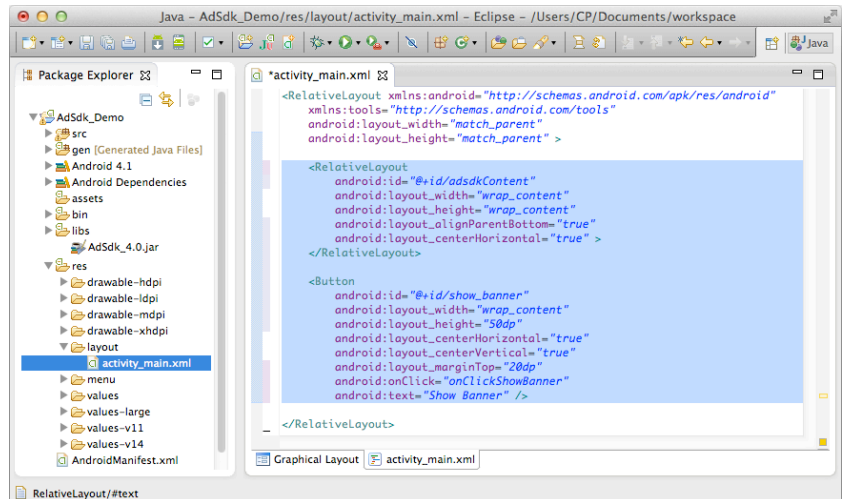
To display vAds, our high-paying Premium Ad Format, please jump to "Step 6".

Step 5: Request and display Banner Ads

Open `activity_main.xml` and add the Banner without adding additional code as shown in `AdSdk_XML_Demo`.

Alternatively create a button to call "onClickShowBanner" (copy/paste `activity_main.xml` in `AdSdk_Demo`)

Or instead of using a button display a Banner directly in `MainActivity.java`:

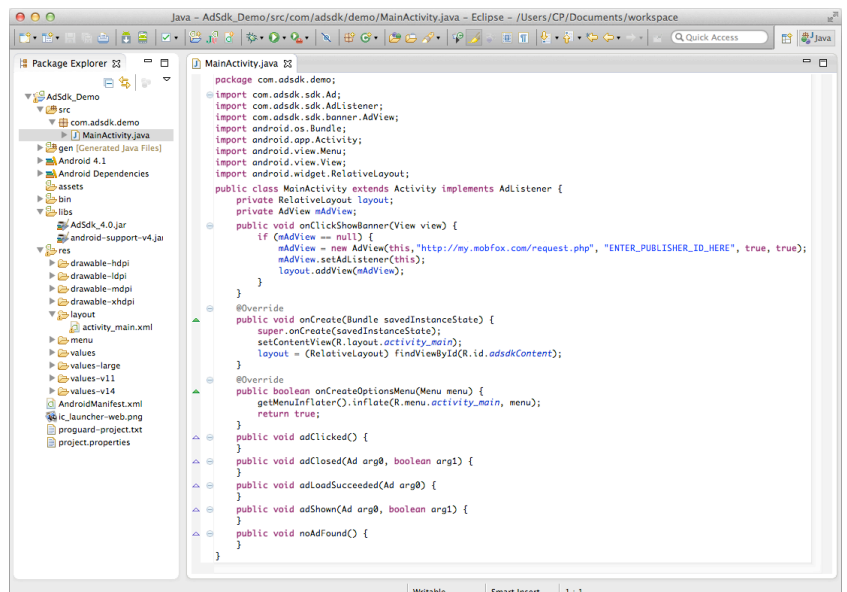


```
if (mAdView == null) {  
    mAdView = new AdView(this, "http://my.mobfox.com/request.php", "ENTER_PUBLISHER_ID_HERE", true, true);  
    mAdView.setAdListener(this);  
    layout.addView(mAdView);  
}
```

Add some code now to your Activity (copy/paste from `MainActivity.java` provided in the `AdSdk_Demo` ZIP).

This will initialize an instance of the Ad view with your Publisher ID, init the button and set a listener on it to request an Ad.

Listeners for the Ad view will handle events e.g. `adLoadSucceeded`,...



Add these Banner Ad specific lines to the Manifest (copy/paste from `AndroidManifest.xml`):

Both READ_PHONE_STATE and LOCATION Permissions are OPTIONAL

```
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.READ_PHONE_STATE" />  
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />  
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />  
  
<activity  
    android:name="com.adsdk.sdk.banner.InAppWebView"  
    android:configChanges="keyboard|keyboardHidden|orientation|screenLayout  
        uiMode|screenSize|smallestScreenSize" />  
  
<activity  
    android:name="com.adsdk.sdk.mraid.MraidActivity"  
    android:configChanges="keyboard|keyboardHidden|orientation|screenLayout  
        uiMode|screenSize|smallestScreenSize" />
```

Add the following to your activity with the banner (see `MainActivity` at the `AdSdk_Demo` ZIP):

```
android:configChanges="keyboard|keyboardHidden|orientation|screenLayout  
    uiMode|screenSize|smallestScreenSize"
```

If you don't want to add vAd Premium Ads please jump to "Step 7".

Step 6: Request and display vAds, our high-paying Premium full-screen format.

Open `activity_main.xml` and create a button which calls when pressed the method `"onClickShowVideoInterstitial"` (copy/paste from `activity_main.xml`)

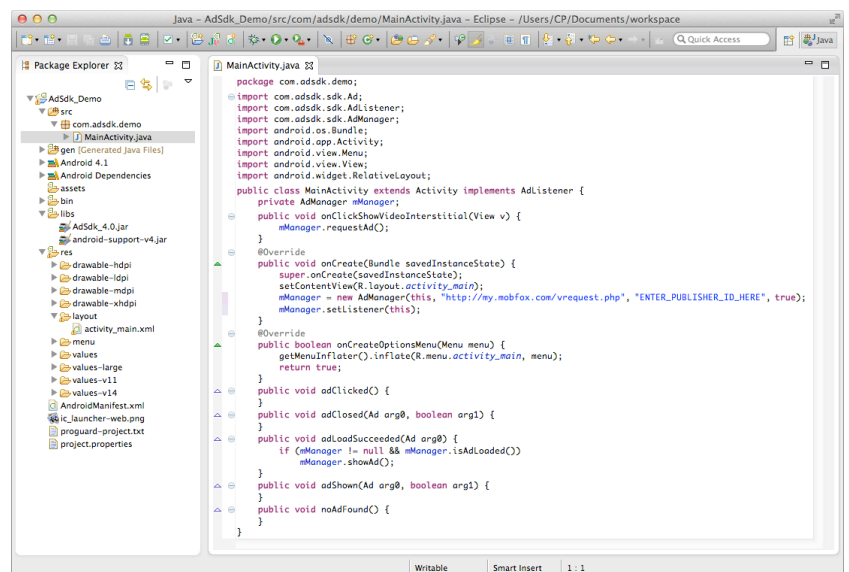
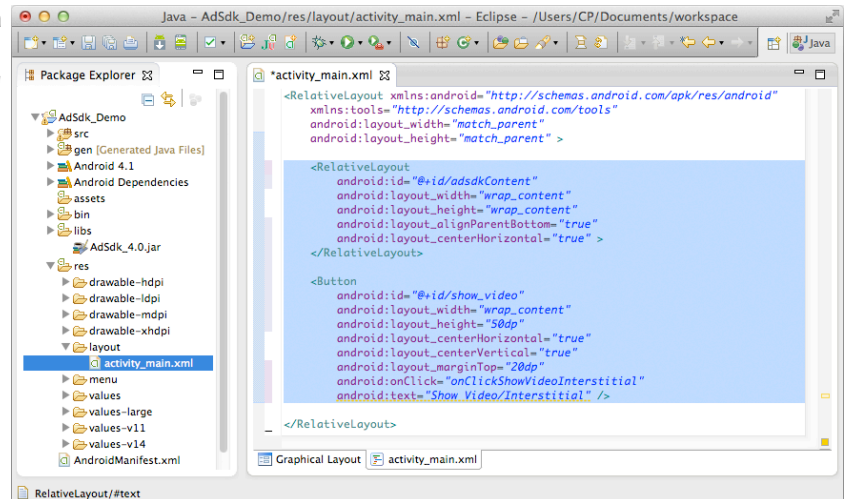
Or instead of using a button display a vAd directly in `MainActivity.java`:

```
mManager = new AdManager(this, "http://my.mobfox.com/vrequest.php", "ENTER_PUBLISHER_ID_HERE", true);  
mManager.setListener(this);  
mManager.requestAd();
```

Add some code now to your Activity (copy/paste from `MainActivity.java` provided in the `AdSdk_Demo` ZIP).

The `AdManager` initialized with your Publisher ID and the listener will request an Ad.

Listeners for the Ad view will handle events e.g. `adLoadSucceeded`,...



Add these vAd specific lines to the Manifest (copy/paste from `AndroidManifest.xml`):

Both READ_PHONE_STATE and LOCATION Permissions are OPTIONAL

```
<uses-permission android:name="android.permission.INTERNET" />  
<uses-permission android:name="android.permission.READ_PHONE_STATE" />  
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />  
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />  
  
<activity  
    android:name="com.adsdk.sdk.video.RichMediaActivity"  
    android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|  
        uiMode|screenSize|smallestScreenSize"  
    android:hardwareAccelerated="false" />
```

Optionally and for vAds only you can create an "anim" folder within your "res" folder. This will allow us to animate the transitions for vAds. Uncompress `animations.zip`, drag and drop the files to an "anim" folder and select "Copy files".

Step 7: Finished!

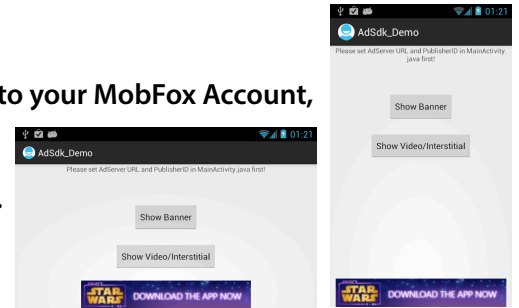
You will see test ads. Once ready to receive live ads, please log into your MobFox Account, navigate to „Sites & Apps“, and click on „Request Activation“

We will review and activate your app for live ads within 24 hours.

Note: Please always clear the heap before showing Ads and restore your apps orientation after an Ad is shown.

Publisher IDs for testing:

If you would like to test your SDK integration with one of our Test Publisher IDs, please click [here](#).



Explanation of Manifest

Internet Access Permission (mandatory and needed for the SDK to work)

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

Read Phone State Permission (optional)

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

Network State Permission (recommended for vAds to identify the connection type)

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

Location Permissions (to target ads for better customer experience - optional)

```
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
```

Explanation of Methods

The Banner Ad constructor argument "Location" let's the SDK use the current location for better targeting Ads and "animation" enables transition animations between Banner Ads:

```
AdView(Context, "http://my.mobfox.com/request.php", "PUBLISHER_ID", Location, animation);
```

The vAd constructor argument "Location" let's the SDK use the current location for better Ad targeting:

```
AdManager(Context, "http://my.mobfox.com/vrequest.php", "PUBLISHER_ID", Location);
```

Once a new Ad has loaded, this AdListener method will be called on the main User Interface thread.

```
public void adLoadSucceeded(Ad advertisement) {}
```

If the Ad couldn't load, this AdListener method will be called and your can proceed to the next screen.

```
public void noAdFound() {}
```

For vAds Ads you will be notified when the RichMediaActivity has finished to resume your activity.

```
public void adClosed(Ad advertisement, boolean completed) {}
```

After a vAd has shown and your activity has been paused, you will also be notified.

```
public void adShown(Ad advertisement, boolean succeeded) {}
```