Admob integration lib and other options
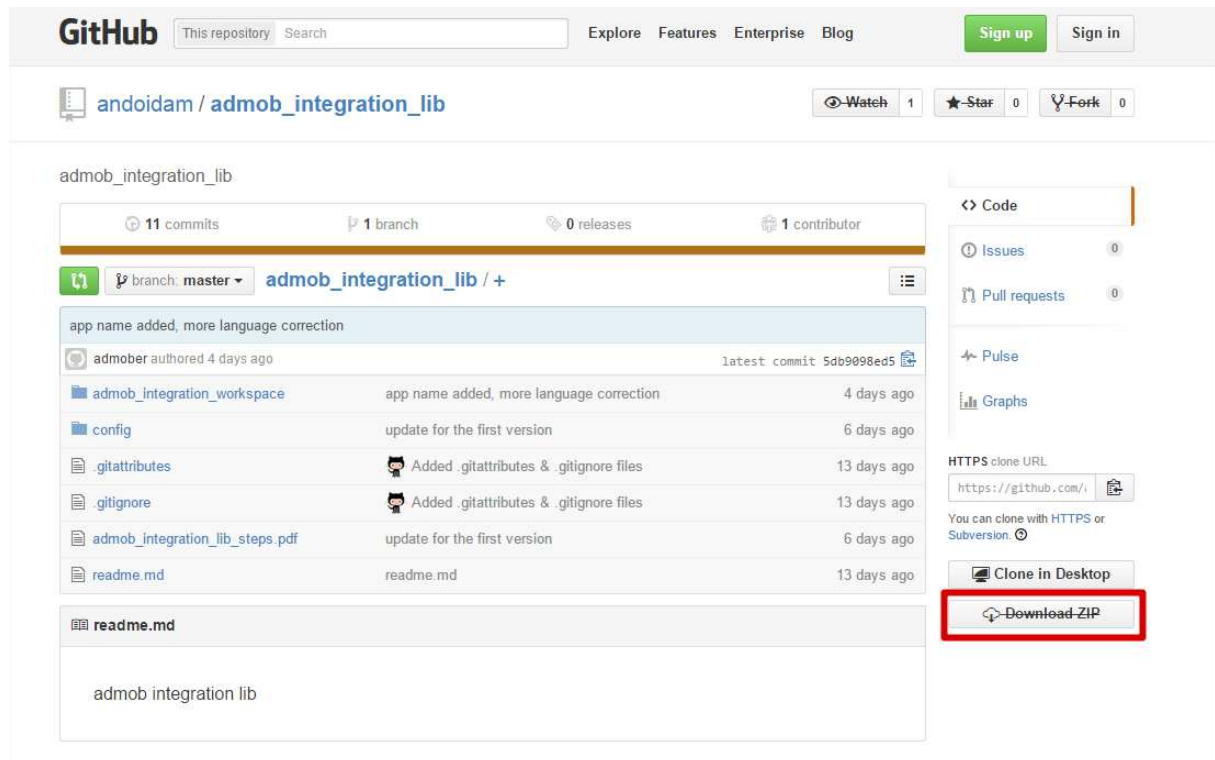
Version : 1.0

Download lib

To download the lib, go to this link

https://github.com/andoidam/admob_integration_lib

Click the **Download ZIP**

Extract the zip file:

Open your eclipse and use the workspace **admob_integration_workspace** as your workspace.

Import the project in which you want to integrate this library and follow the steps below.

Configuring your project

Step 1:

After the manifest tag (<manifest) add this permissions:

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

Step 2:

In the application tag add this line:

```xml
<application
    android:name="com.google.admob.integration.libs.Admob"
    android:allowBackup="true"
```

Like this:

```xml
<application
        android:name="com.google.admob.integration.libs.Admob"
        android:allowBackup="true"
```

Step 3 :

Before </application> add this lines :

```xml
<!-- ads -->
<meta-data android:name="com.google.android.gms.version"
android:value="@integer/google_play_services_version" />
<activity android:name="com.amazon.device.ads.AdActivity"
android:configChanges="keyboardHidden|orientation|screenSize" />
<!-- Include the AdActivity configChanges and theme. -->
<activity android:name="com.google.android.gms.ads.AdActivity"
android:configChanges="keyboard|keyboardHidden|orientation|screenLayout|uiMode|screenSize|smallestScreenSize" android:theme="@android:style/Theme.Translucent" />
```

Step 4 :

Configuring your unit add for admob banner, admob interstitial, analytics tacker_id, facebook and google play developer name

Open your strings.xml file, and add this line

```xml
<string name="admob_test">1</string>
<string name="admob_banner">admob_banner_code</string>
<string name="admob_interstitial">admob_interstitial_code</string>
```

```xml
<string name="google_analytics">google_tracker_id</string>
<string name="facebook_id">facebook_page_id</string>
<string name="user_play_name">your_google_play_developper_name</string>
```

How you can you this line : `<string name="admob_test">1</string>` ?

When the value of admob_test is 1, you can use admob in test mode, so you do not have to wear about clicking ads

When you want to export your application for production don't forget to change the value to 0 or to delete the hall line

Now your project is ready to use admob and to use other option of this lib

Using interstitial:

Call interstitial just one time

When you want to show an interstitial ad, you have just to call one instruction

```
Admob.get().skShowInterstitial();
```

Using interstitial with timer:

You can use this when you have an app with article to read or with picture to view...

If you want to show the interstitial every 10 seconds for example, just put the time you want like this:

```
Admob.get().skShowInterstitial(10);
```

Every 30 seconds:

```
Admob.get().skShowInterstitial(30);
```

In this case, don't forget to call Admob.get().closeInterstitial(); when you leaving your activity, like this :

```
@Override
protected void onStop() {
    Admob.get().skCloseInterstitial();
    super.onStop();
}
```

Using interstitial after N time :

If you want to show interstitial after a number of actions executed by user, just call showInterstitialNTime

Example: showing the interstitial every 3 time

```
Admob.get().skShowInterstitialNTime(3);
```

Using interstitial in cocos2d lib

to use interstitial in a code developed with cocos2d, follow this step :

- first find you main activity (you can find it in the manifest)

- look for the **onCreate** method and add the following code just after this line super.onCreate(savedInstanceState);

```
Admob.get().initCocos(this);
```

Exemple :

```java
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);

    Admob.get().initCocos(this);
```

- then look for the emplacement on witch you want to show interstitial and use this code :

```java
Admob.get().skShowInterstitialCocos();
```

Add Banner to your code

Now you do not have to use the xml admob ads banner declaration in your xml file, just in your code, you can add admob banner dynamically

First reach to area where you want to add you admob banner, and give him and id, the preference this area have to be a LinearLayout, so you can specify exactly where you want to add the banner, in the first place or in the last one

Then reach the id of your View and use it in this code instead of *LLMainApp*

For a Smart banner, use this :

```
LinearLayout ll = (LinearLayout) findViewById(R.id.llMainApp);
ll.addView(Admob.get().skSmartBanner(), 0);
```

For a normal banner use this:

```
LinearLayout ll = (LinearLayout) findViewById(R.id.llMainApp);
ll.addView(Admob.get().skBanner(), 0);
```

Add Banner to a code using cocos2d

look for the main activity (you can find it in the Manifest.xml)

look for a declaration like this **mGLSurfaceView** = new CCGLSurfaceView(this);

keep the **mGLSurfaceView** in mind because we will use it

find the setContentView in the code (you will find it in the onCreate method)

replace the hall line with this code :

for banner :

top left

```
setContentView(Admob.get().skBannerForCocos(mGLSurfaceView,0,0));
```

top right

```
setContentView(Admob.get().skBannerForCocos(mGLSurfaceView,0,1));
```

bottom left

```
setContentView(Admob.get().skBannerForCocos(mGLSurfaceView,1,0));
```

bottom right

```
setContentView(Admob.get().skBannerForCocos(mGLSurfaceView,0,1));
```

for smart banner

if you want to show banner in the top of your game, use this code :

```
setContentView(Admob.get().skSmartBannerForCocos(mGLSurfaceView,0));
```

 if you want to show banner in the bottom of your game, use this code :

```
setContentView(Admob.get().skSmartBannerForCocos(mGLSurfaceView,1));
```

Adding exit option

Add exit confirmation

To add just a exit confirmation to app preventing user of exiting (helpful when user play a game and touch return button by error), add this code to your main activity

```
@Override
public void onBackPressed() {
     Admob.get().skExit(this);
}
```

Add exit with options

In your main Activity (you will find it on the <application> in your manifest) add this code :

If you want all options, just add this one to your main activity :

```
@Override
public void onBackPressed() {
     Admob.get().skExitWithOption(this);
}
```

If you want to choose just some options or make options on an order that you prefer, use this code in your main activity, just comment or delete option that you don't want:

```
@Override
public void onBackPressed() {

     List<String> choice = new ArrayList<String>();

     choice.add(Constantes.RATE_APP);
     choice.add(Constantes.LIST_APP);
     choice.add(Constantes.FACEBOOK);
     choice.add(Constantes.SHARE_APP);
     choice.add(Constantes.RETURN_TO_APP);
     choice.add(Constantes.EXIT);

     Admob.get().skExitWithOption(this, choice);
}
```
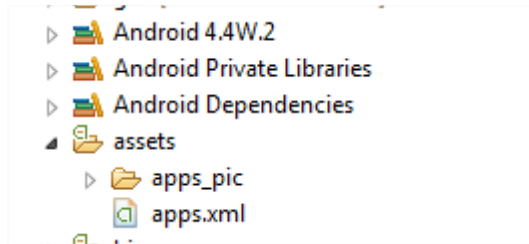
If you want just showing rate option, use this code:

```
@Override
```

```java
public void onBackPressed() {
    Admob.get().skRateApp(this);
}
```

Creating your app list

To create your app list to show them to the user, copy file and folder from the folder assets you will find in the config folder coming with this document to your asset folder like this:

Open the **apps.xml** file, and edit it by adding your applications

For each application you have to specify some informations :

name	The name of your application
description	A short description of your application
package	The package name of your application
icon	An icon of your application, you have to put an icon in the **assets/apps_pic** for each app, and put the nom of the picture in this tag

Add google analytics service:

To send tracking data of your app and getting statistics about the use of your app, you have just to call this line anywhere in your code:

```
Admob.get().skAnalytics("page_to_view");
```

Change the value page_to_view by the name of the activity you are tracking, for example: main_activity, game_activity, arcticle activity ...