

## Ads only :: SuperG SDK installation

### Step 1:

Unzip the SDK package and keep the directories 'ThumbSDK' and 'TestApp' OR YOUR APPLICATION within the same directory.

### Step 2:

Add a reference to the Thumbr SDK library project. Follow these steps:

1. Import the Thumbr SDK project into Eclipse
2. In the Package Explorer, right-click YOUR project and select Properties.
3. In the Properties window, select the "Android" properties group at left and locate the Library properties at right.
4. Click Add to open the Project Selection dialog.
5. From the list of available library project, select the Thumbr SDK project and click OK.
6. When the dialog closes, click Apply in the Properties window.
7. Click OK to close the Properties window As soon as the Properties dialog closes, Eclipse rebuilds your project.

### Step 3:

Update your AndroidManifest.xml with at least the settings below:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.yourcompany.yourgame"
    android:versionCode="1"
    android:versionName="1.0" >
    <uses-sdk android:targetSdkVersion="17" android:minSdkVersion="8" />

    <uses-permission android:name="android.permission.GET_ACCOUNTS" />
    <uses-permission android:name="android.permission.READ_PHONE_STATE" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.READ_CALENDAR" />
    <uses-permission android:name="android.permission.WRITE_CALENDAR" />
    <uses-permission android:name="android.permission.CAMERA" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.WAKE_LOCK" />
    <uses-permission android:name="android.permission.CALL_PHONE" />
    <uses-permission android:name="android.permission.SEND_SMS" />
    <uses-permission android:name="android.permission.READ_PHONE_STATE" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />

    <application
        android:icon="@drawable/ic_launcher"
        android:debuggable="true"
        android:label="YourLabel" >
    </application>

</manifest>
```

### Step 4:

Add a view to your layout that will contain the advertisement

```
<RelativeLayout
    android:id="@+id/ad_view"
    android:layout_width="fill_parent"
    android:layout_height="0dp"
```

```

        android:layout_alignParentBottom="false"
        android:layout_alignParentTop="true"
        android:layout_centerInParent="true"
        android:layout_gravity="bottom"
        android:background="@color/transparent" >
</RelativeLayout>

```

## Step 5 ACTIVITY FILE:

Update the *activity file*, where Thumbr will be called (usually your main activity)

```
package com.yourcompany.yourapp.youractivity;
```

### Import at least these libraries

```

import android.annotation.SuppressLint;
import android.app.Activity;
import android.content.Context;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.util.Log;
import android.view.Menu;
import android.widget.RelativeLayout;
import android.widget.Toast;
import com.gkxim.android.thumbsdk.FunctionThumbrSDK;
import com.gkxim.android.thumbsdk.FunctionThumbrSDK.OnInterstitialCloseListener;

```

### Open your activity class and enter your app specific settings (provided by your Thumbr Game Manager)

```
public class yourActivity extends Activity implements OnClickListener,OnDismissListener{
```

```
//AD SERVING SETTINGS
```

```

private int updateTimeInterval = 0;//Initial number of seconds before Ad refresh (setting will be updated from server)
private int showCloseButtonTime = 6;//Number of seconds before the Ad close button appears
private String tablet_Inline_zoneid = "1356888057";
private String tablet_Inline_secret = "20E1A8C6655F7D3E";
private String tablet_Overlay_zoneid = "3356907052";
private String tablet_Overlay_secret = "ADAA22CB6D2AFDD3";
private String tablet_Interstitial_zoneid = "7356917050";
private String tablet_Interstitial_secret = "CB45B76FE96C8896";
private String phone_Inline_zoneid = "0345893057";
private String phone_Inline_secret = "04F006733229C984";
private String phone_Overlay_zoneid = "7345907052";
private String phone_Overlay_secret = "AEAAA69F395BA8FA";
private String phone_Interstitial_zoneid = "9345913059";
private String phone_Interstitial_secret = "04B882960D362099";

```

### Define the ‘thumbr’ object and add / edit onPause and onResume functions

```
FunctionThumbrSDK thumbr;
```

```

@Override
protected void onPause() {
    thumbr.pause();
    super.onPause();
}
@Override
protected void onResume(){
    thumbr.resume();
    super.onResume();
}

```

## Modify your onCreate function

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    // SET VIEW FOR YOUR APPLICATION
    super.onCreate(savedInstanceState);
    setContentView(R.layout.YourMainActivityView);

    //SET AD SERVER SETTINGS
    SharedPreferences settings = this.getSharedPreferences("ThumbrSettings", Context.MODE_PRIVATE);
    settings.edit().putInt("updateTimeInterval", updateTimeInterval).commit();
    settings.edit().putInt("showCloseButtonTime", showCloseButtonTime).commit();
    settings.edit().putString("tablet_Inline_zoneid", tablet_Inline_zoneid).commit();
    settings.edit().putString("tablet_Inline_secret", tablet_Inline_secret).commit();
    settings.edit().putString("tablet_Overlay_zoneid", tablet_Overlay_zoneid).commit();
    settings.edit().putString("tablet_Overlay_secret", tablet_Overlay_secret).commit();
    settings.edit().putString("tablet_Interstitial_zoneid", tablet_Interstitial_zoneid).commit();
    settings.edit().putString("tablet_Interstitial_secret", tablet_Interstitial_secret).commit();
    settings.edit().putString("phone_Inline_zoneid", phone_Inline_zoneid).commit();
    settings.edit().putString("phone_Inline_secret", phone_Inline_secret).commit();
    settings.edit().putString("phone_Overlay_zoneid", phone_Overlay_zoneid).commit();
    settings.edit().putString("phone_Overlay_secret", phone_Overlay_secret).commit();
    settings.edit().putString("phone_Interstitial_zoneid", phone_Interstitial_zoneid).commit();
    settings.edit().putString("phone_Interstitial_secret", phone_Interstitial_secret).commit();
    //CALL THE THUMBR LIBRARY
    thumbr = new FunctionThumbrSDK(this, 2);
    thumbr.adInit();//initialize Ads

    /* Interstitial advertisement close listener
       If you use the thumbr.adInterstitial(ad_view) method, you can use this listener to know when it closes.
    */
    thumbr.setInterstitialCloseListener(new OnInterstitialCloseListener(){
        public void onEvent(){
            Log.i("ThumbrSDK","The interstitial advertisement was closed. You can resume your application.");
            Toast.makeText(getApplicationContext(), "The application has been notified about closing the interstiti
            //resumeYourApp();
        }
    });

    final RelativeLayout ad_view = (RelativeLayout) findViewById(R.id.ad_view);

    /*Load an inline advertisement*/
    thumbr.adInline(ad_view);

    /*Load an overlay advertisement*/
    //thumbr.adOverlay(ad_view);

    /*Load an interstitial advertisement*/
    //thumbr.adInterstitial(ad_view);
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}

}
```

## Closing the class

