

## Integration Guide of Apps Manager Ads SDK (Ad Optimized SDK)

1. Create or open your existing Android project in Android Studio.
2. import the `AdsSdk` module. Name the module "AdsSdk", for example.
3. Add the implementation module for the AppsManager Ads SDK to your module's app-level Gradle file, normally `app/build.gradle`:

```
dependencies {  
    implementation project(path: ':AdsSdk')  
}
```

4. In your project-level `build.gradle` file, include Google's Maven repository and Maven central repository in both your `buildscript` and `allprojects` sections:

```
buildscript {  
    repositories {  
        google()  
        mavenCentral()  
    }  
}  
  
allprojects {  
    repositories {  
        google()  
        mavenCentral()  
        maven { url 'https://jitpack.io' }  
  
        // If Use TopOn Ads  
        maven {  
            url "https://anythink.jfrog.io/artifactory/android_sdk"  
        }  
    }  
}
```

## Initializing the SDK in Android (Compulsory)

Before loading ads, have your app initialize the AppsManager Ads SDK by calling **ADSinit** which initializes and get response from server and calls back a **onSuccess**

### Example: SplashActivity

```
import com.pesonal.adsdk.ADS_SplashActivity;
import com.pesonal.adsdk.getDataListner;

import org.json.JSONObject;

import java.util.Arrays;

public class SplashActivity extends ADS_SplashActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);
        getWindow().setFlags(1024, 1024);
        setContentView(R.layout.activity_splash);

        ADSinit(SplashActivity.this,getCurrentVersionCode(), new
        getDataListner() {
            @Override
            public void onSuccess() {

                // Next Activity Intent
                startActivity(new Intent(SplashActivity.this,
                MainActivity.class));
                finish();
            }

            @Override
            public void onUpdate(String url) {
                Log.e("my_log", "onUpdate: "+url );
                showUpdateDialog(url);
            }

            @Override
            public void onRedirect(String url) {
                Log.e("my_log", "onRedirect: "+url );
                showRedirectDialog(url);
            }

            @Override
```



```
        public void onReload() {
            startActivity(new Intent(SplashActivity.this,
SplashActivity.class));
            finish();
        }

        @Override
        public void onGetExtradata(JSONObject extraData) {
            Log.e("my_log", "ongetExtradata: "+extraData.toString() );
        }
    });
}

public void showRedirectDialog(final String url) {

    final Dialog dialog = new Dialog(SplashActivity.this);
    dialog.setCancelable(false);
    View view = getLayoutInflater().inflate(R.layout.installnewappdialog,
null);
    dialog.setContentView(view);
    TextView update = view.findViewById(R.id.update);
    TextView txt_title = view.findViewById(R.id.txt_title);
    TextView txt_decription = view.findViewById(R.id.txt_decription);

    update.setText("Install Now");
    txt_title.setText("Install our new app now and enjoy");
    txt_decription.setText("We have transferred our server, so install
our new app by clicking the button below to enjoy the new features of app.");

    update.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            try {
                Uri marketUri = Uri.parse(url);
                Intent marketIntent = new Intent(Intent.ACTION_VIEW,
marketUri);
                startActivity(marketIntent);
            } catch (ActivityNotFoundException ignored1) {
            }
        }
    });

    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
        dialog.create();
    }

    dialog.show();
    Window window = dialog.getWindow();
    window.setLayout (LinearLayout.LayoutParams.MATCH_PARENT,
LinearLayout.LayoutParams.WRAP_CONTENT);
}
```





```
        window.setBackgroundDrawable(new
ColorDrawable(android.graphics.Color.TRANSPARENT));

    }

    public void showUpdateDialog(final String url) {

        final Dialog dialog = new Dialog(SplashActivity.this);
        dialog.setCancelable(false);
        View view = getLayoutInflater().inflate(R.layout.installnewappdialog,
null);
        dialog.setContentView(view);
        TextView update = view.findViewById(R.id.update);
        TextView txt_title = view.findViewById(R.id.txt_title);
        TextView txt_decription = view.findViewById(R.id.txt_decription);

        update.setText("Update Now");
        txt_title.setText("Update our new app now and enjoy");
        txt_decription.setText("");
        txt_decription.setVisibility(View.GONE);

        update.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                try {
                    Uri marketUri = Uri.parse(url);
                    Intent marketIntent = new Intent(Intent.ACTION_VIEW,
marketUri);
                    startActivity(marketIntent);
                } catch (ActivityNotFoundException ignored1) {
                }
            }
        });

        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
            dialog.create();
        }

        dialog.show();
        Window window = dialog.getWindow();
        window.setLayout(LinearLayout.LayoutParams.MATCH_PARENT,
LinearLayout.LayoutParams.WRAP_CONTENT);
        window.setBackgroundDrawable(new
ColorDrawable(android.graphics.Color.TRANSPARENT));

    }

    public int getCurrentVersionCode()
    {
        PackageManager manager = getPackageManager();
        PackageInfo info = null;
        try {
```



```
        info = manager.getPackageInfo(  
            getPackageName(), 0);  
        return info.versionCode;  
    } catch (PackageManager.NameNotFoundException e) {  
        e.printStackTrace();  
    }  
  
    return 0;  
}
```

## Banner Ads

### Add Layout

#### Example: activity\_second.xml

```
<LinearLayout  
    android:id="@+id/banner_container"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_alignParentBottom="true"  
    android:orientation="vertical" />
```

### Load & Show Ad

1. If you want to take random Ads unit Id

```
AppManage.getInstance(SecondActivity.this).showBanner((ViewGroup)  
findViewById(R.id.banner_container1));
```

2. If you want to do activity page wise separately ad unit id then

```
AppManage.getInstance(SecondActivity.this).showBanner((ViewGroup)  
findViewById(R.id.banner_container), AppManage.ADMOB_B[0],  
AppManage.FACEBOOK_B[0], AppManage.TOPON_B[0]);
```

## NativeBanner Ads

### Add Layout

#### Example:activity\_main.xml

```
<LinearLayout
    android:id="@+id/banner_container"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:orientation="vertical" />
```

### Load & Show Ad

#### 1. If you want to take random Ads unit Id

```
AppManage.getInstance(MainActivity.this).showNativeBanner((ViewGroup)
    findViewById(R.id.banner_container));
```

#### 2. If you want to do activity page wise separately ad unit id then

```
AppManage.getInstance(MainActivity.this).showNativeBanner((ViewGroup)
    findViewById(R.id.banner_container), AppManage.ADMOB_B[0],
    AppManage.FACEBOOK_NB[0], AppManage.TOPON_B[0]);
```

## Native Ads

### Add Layout

#### Example: activity\_main.xml

```
<LinearLayout
    android:id="@+id/native_container"
    android:layout_width="match_parent"
```



```
android:layout_height="250dp"  
android:orientation="vertical" />
```

## Load & Show Ad

### 1. If you want to take random Ads unit Id

```
AppManage.getInstance(MainActivity.this).showNative((ViewGroup)  
findViewById(R.id.native_container));
```

### 2. If you want to do activity page wise separately ad unit id then

```
AppManage.getInstance(MainActivity.this).showNative((ViewGroup)  
findViewById(R.id.native_container), AppManage.ADMOB_N[0],  
AppManage.FACEBOOK_N[0], AppManage.TOPON_N[0]);
```

## Interstitial Ads

### Load Ad

#### 1. If you want to random Ads unit Id do it

**Note:** If you use random ads unit id, then you do not need to `loadInterstitialAd` in all activities, you have to `loadInterstitialAd` only once.

#### Example: StartActivity

```
AppManage.getInstance(MainActivity.this).loadInterstitialAd(this);
```

#### 2. If you want to do activity page wise separately ad unit id then

```
AppManage.getInstance(SecondActivity.this).loadInterstitialAd(this,  
AppManage.ADMOB_I[0], AppManage.FACEBOOK_I[0], AppManage.TOPON_I[0]);
```



## Show Ad

### *showInterstitialAd()* Function Argument :

1. context
2. callback
3. If you want to show ads of specific platform then you have to write platform name Like... **AppManage.FACEBOOK** otherwise blank Like... ""
4. ads on every [ X ] click `AppManage.app_mainClickCntSwAd` **OR**  
`AppManage.app_innerClickCntSwAd`

### Example:

#### 1. Default

```
AppManage.getInstance(SecondActivity.this).showInterstitialAd(SecondActivity.this, new AppManage.MyCallback() {  
    public void callbackCall() {  
  
        Toast.makeText(SecondActivity.this, "Hello Perform Anything Here After Ad Show", Toast.LENGTH_SHORT).show();  
  
    }  
}, "", AppManage.app_mainClickCntSwAd);
```

#### 2. Interstitial with ads on every [inner click count] click

```
AppManage.getInstance(SecondActivity.this).showInterstitialAd(SecondActivity.this, new AppManage.MyCallback() {  
    public void callbackCall() {  
  
        Toast.makeText(SecondActivity.this, "Interstitial With Click Count", Toast.LENGTH_SHORT).show();  
  
    }  
}, "", AppManage.app_innerClickCntSwAd);
```

#### 3. specific platform





```
AppManage.getInstance(SecondActivity.this).showInterstitialAd(SecondActivity.this, new AppManage.MyCallback() {  
    public void callbackCall() {  
  
        Toast.makeText(SecondActivity.this, "specific platform",  
Toast.LENGTH_SHORT).show();  
    }  
}, AppManage.FACEBOOK, AppManage.app_mainClickCntSwAd);
```

## App Open Ads

### Load Ad

```
AppManage.getInstance(MainActivity.this).loadAdmobAppOpenAd(this, AppManage.ADMOB_APP_OPEN[0]);
```

### Show Ad

```
AppManage.getInstance(context).showAppOpenAd(context, new MyCallback() {  
    @Override  
    public void callbackCall() {  
  
        Toast.makeText(SecondActivity.this, "Hello",  
Toast.LENGTH_SHORT).show();  
  
    }  
});
```



```
// Show Apps Manager Custom Ad
```

```
AppManage.getInstance(SecondActivity.this).showCustomAppOpenAd(SecondActivity
.this, new AppManage.MyCallback() {
    @Override
    public void callbackCall() {
        finish();
    }
});
```

## Show Back Press Ads

```
AppManage.getInstance(MainActivity.this).showBackPressAd(MainActivity.this,
new AppManage.MyCallback() {
    @Override
    public void callbackCall() {

        finish();
    }
});
```

## Load & Show Ads (Interstitial)

If you want to load the ads while showing the ads

### 1. Default

```
AppManage.getInstance(LoadAndShowDemoActivity.this).loadAndShowInterstitialAd
(LoadAndShowDemoActivity.this,true, new AppManage.MyCallback() {
    public void callbackCall() {

        Toast.makeText(LoadAndShowDemoActivity.this,"Load and Show
Interstitial",Toast.LENGTH_SHORT).show();
    }
}, "", AppManage.app_mainClickCntSwAd);
```

## 2. Specific Platform

```
AppManage.getInstance(LoadAndShowDemoActivity.this).loadAndShowInterstitialAd
(LoadAndShowDemoActivity.this,true, new AppManage.MyCallback() {
    public void callbackCall() {

        Toast.makeText(LoadAndShowDemoActivity.this,"Load and Show
Interstitial Specific Platform",Toast.LENGTH_SHORT).show();

    }
},AppManage.FACEBOOK,AppManage.app_mainClickCntSwAd);
```

## 3. Specific Ads Unit ID

```
AppManage.getInstance(LoadAndShowDemoActivity.this).loadAndShowInterstitialAd
(LoadAndShowDemoActivity.this,true, new AppManage.MyCallback() {
    public void callbackCall() {

        Toast.makeText(LoadAndShowDemoActivity.this,"Load and Show
Interstitial Specific Ad Unit ID",Toast.LENGTH_SHORT).show();

    }
},"",AppManage.app_mainClickCntSwAd,AppManage.ADMOB_I[0],AppManage.FACEBOOK_I
[0],AppManage.TOPON_I[0]);
```

# Ads Analytics

If you want to track ads like... Total REQ, REQ\_LOAD, REQ\_FAIL, AD\_SHOW, AD\_IMP, ERROR\_FAIL, ERROR\_LOG

### Example: ThirdActivity

```
new
AdsAnalysis(ThirdActivity.this).request_adAnalysis(this.getClass().getSimpleName());
```

### Example: SixthActivity

```
new
AdsAnalysis(SixthActivity.this).request_adAnalysis(this.getClass().getSimpleName());
```



**Note:**

1. You have written **request\_adAnalysis()** function in ThirdActivity, then from SplashActivity to ThirdActivity Ads have been counted, they will be sent to server side.
2. If you have done **request\_adAnalysis()** function in both ThirdActivity and SixthActivity then the ads count from SplashActivity to ThirdActivity and from ThirdActivity to SixthActivity both ads count will be sent to server side.

## MoreApp

```
//GET SPLASH MORE APP DATA  
  
List<MoreApp_Data> splash_more_data =  
AppManage.getInstance(MainActivity.this).get_SPLASHMoreAppData();  
  
//GET EXIT MORE APP DATA  
  
List<MoreApp_Data> exit_more_data =  
AppManage.getInstance(MainActivity.this).get_EXITMoreAppData();
```

**Note:**

- AdsSdk of Apps Manager is open source that's why you can change and edit the source as per your requirement.
- If you want to use your AdsSdk with Apps Manager System is also possible.
- Still, if you are confused about how to integrate it, you can contact to apps manager team via support & which will help you anytime.