

Dev Guide Book @Adview Android Unity

Contents

- I. Register and configure SDK-KEY**
- II.About AdView_Unity3D_Android_SDK-3.4.1**
- III. Add SDK**
- IV.Add Assets Files**
- V. AndroidManifest.xml text configuration**
- VI. Acquire ad configurations**
- VII. Create banner advertising**
- VIII. Create interstitial advertising**
- IX. Create opening screen ad**
- X. Create native advertising**
- XI. Create video advertising**
- XII. Adding Proguard-rules**
- XIII. Adding custom ad network**
- XIV. Contact us**

I. Register and configure SDK-KEY

1. Visit AdView website <http://adview.com> and complete the registration process.
2. After Login you will land on **My Products** page by default, select **"Publish App"**
3. Select **"Android"** follow the prompts to complete the relevant information About the application and click on **"Next"** button at the end where you will be redirected to **"APP management"** page
4. Under **"App managemnet"** page Click **"Next"** button at the bottom --> you will get the sole SDK key --> click **"Finish"** button at the bottom centre of the page and you will be redirected to the below page.

+ Add apps		Search by app name		Download Android SDK		Download iOS SDK	
Device	App name	Impressions (today/yesterday)▲	Clicks (today/yesterday)▲	Ad bidding and supplement switch on/off ?	Creation date	Operate	
iOS	call flash light	0/0	0/0	Pending review	2016-12-15	Edit	Configure Delete
iOS	call flash light	0/0	0/0	Pending review	2016-12-15	Edit	Configure Delete
iOS	call flash light	0/0	0/0	Pending review	2016-12-15	Edit	Configure Delete
Android	lucky rupee	0/0	0/0	Pending review	2016-11-23	Edit	Configure Delete

5. Now click on **"configure"** against your application, then you will be redirected to **"APP management"** page → under the Ad format you want to configure (Banner, interstitial, video, open screen and native) open the switch against Auction ads (AdView Ads) and in case you want to add any custom ad networks (like InMobi, AdMob etc..) Please refer **XIII. Adding custom ad network** for further steps, the cumulative percentage must be 100%, then click save button. Generally recommended number of platforms is 1-3.

AdViewBidSDKDemo
SDK-KEY
SDK20172102090332w8rthp5ep8p8r1n
Copy

Banner
Interstitial/Full screen
Open screen
Video ads
Native ads
Go to

Region optimization
ON
Ad remnant
ON
One key to configure

User configuration in China
+Add ad platform

Advertising platform	Scoring	Setting	On/off	Capacity	Priority	Delete platform
Auction ads	5.70	passed	<input checked="" type="checkbox"/>	100%		
Guang-point	1.83	not set	<input type="checkbox"/>	0%	+	Delete
Baidu	2.44	not set	<input type="checkbox"/>	0%	+	Delete
InMobi	1.33	not set	<input type="checkbox"/>	0%	+	Delete
Direct Advertising	1.23	Setting	<input type="checkbox"/>	0%	+	
Ad exchange	1.25	Setting	<input type="checkbox"/>	0%	+	

Total capacity: 100%

User configuration out of China

+Add ad platform

Advertising platform	Scoring	Setting	On/off	Capacity	Priority	Delete platform
Auction ads	5.70	Setting	<input checked="" type="checkbox"/>	100%		
AdMob	1.92	not set	<input type="checkbox"/>	0%	+	Delete

Total capacity: 100%

Save

6.For **video ad or native ad**,Please configure in AdView Dash board .Please click **under review or setting** label ,you will get dialog box.In the dialog box For Auction ads ,please click save button in dialog box and click save button in bottom of the page .For other platforms ,please configure those custom adNetworks ids .

Banner
Interstitial/Full screen
Open screen
Video ads
Native ads
Go to

Region optimization
ON
One key to configure

User configuration in China
+Add ad platform

Advertising platform	Scoring	Setting	On/off	Capacity	Priority	Delete platform
Auction ads	7.36	under review	<input checked="" type="checkbox"/>	100%		

Total capacity: 100%

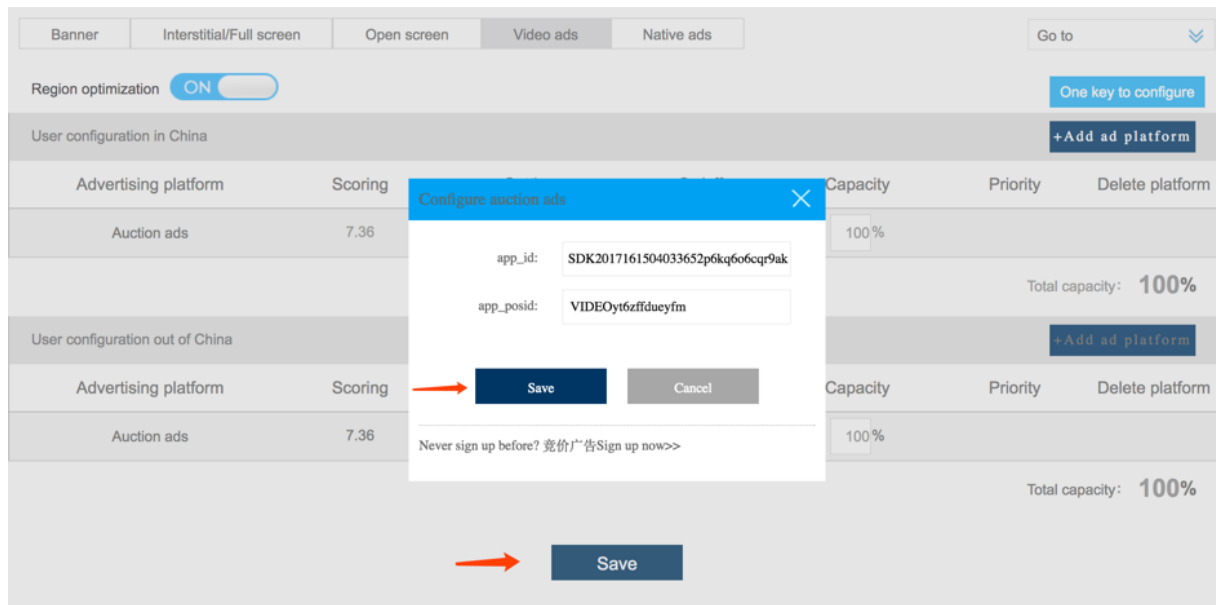
User configuration out of China

+Add ad platform

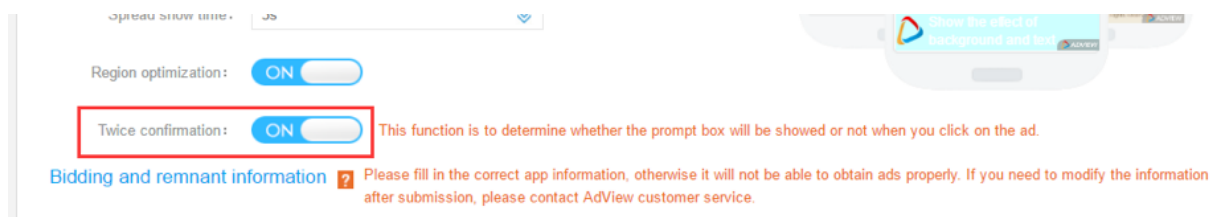
Advertising platform	Scoring	Setting	On/off	Capacity	Priority	Delete platform
Auction ads	7.36	under review	<input checked="" type="checkbox"/>	100%		

Total capacity: 100%

Save



7.(Optional) if you wish to show prompt when you click on the ad - Under **app management** --> select **"Edit"** against your app, Switch on "Twice confirmation" button under "Advertising text settings".



Notes:

1. we have provided you with Adview Android Unity3d SDK which gives you the freedom to choose your favourite ad network. in order to add custom ad networks please refer to this **XIII. Adding custom ad network** .
2. If you are fresher, you don't know much about ad platform, which ad platform to choose or which ad platform revenue is stable, we suggest you to use bidding first.
3. Bidding and remnant ads need to complement market information at background, if the status is **"Under review"** you will be receiving test ads. and you will get formal ads only after **"pass reviewed"** by the AdView team.
4. Ads will be shown of only those ad platforms for which the **switch** is on against them.
5. Only the **"capacity"** of those ad platforms for which the switch is on will be valid, the ad network with higher proportion will get prior request, for all ad platforms with status as ON, the cumulative should be **100%**. Other

wise the your priority can't be saved.

6. For Banner ad, full screen/interstitial, opening screen ,etc, there's a save button at the bottom of the page. You should click the save button every time you modify a ad format, otherwise the modification is invalid .
7. **Region optimization:** Region optimization function means mobile phone displays the regional configured ads when it's with in the region, while in foreign country it display foreign configured ads to meet the different demands to the maximum extent. When the region optimization function is closed, it does not distinguish between home and abroad.
8. In case you wish to show the same ad format more than once (like 2 banner ads) you need to create a new project (My product page --> publish app) with same credentials to get a new SDK key and the integration process remains same, you need not create a new project if the ad formats are different

II.About AdView_Unity3D_Android_SDK-3.4.1

In the **AdView_Unity3D_Android_SDK-3.4.1** package contains all files needed for smooth integration and some of the important fils include

AdViewTestDemoEN and libs.

AdViewTestDemoEN

AdViewTestDemoEN project which includes all types of ad format (banner,interstitial,video,native,open screen) sample code with explanation.you can run the application. you can see **test Ads** .

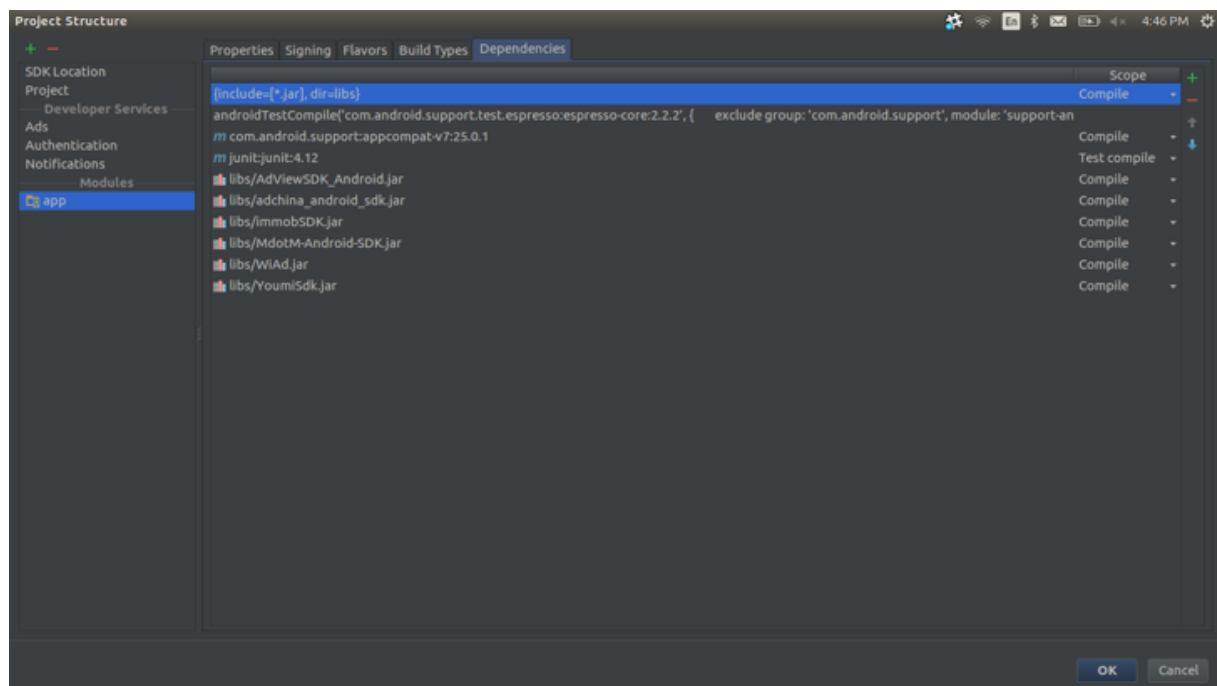
libs

It contains all the .jar file SDK needed for ad platform integration.
(Libinfo.pdf has the ad platform instructions corresponding to each jar.)

III. Add SDK

1. In the AdViewSDK_Android-3.4.1 folder contains libs folder ,it contains the SDK for all ad platforms. (Libinfo.pdf has the ad platform instructions corresponding to each jar.)

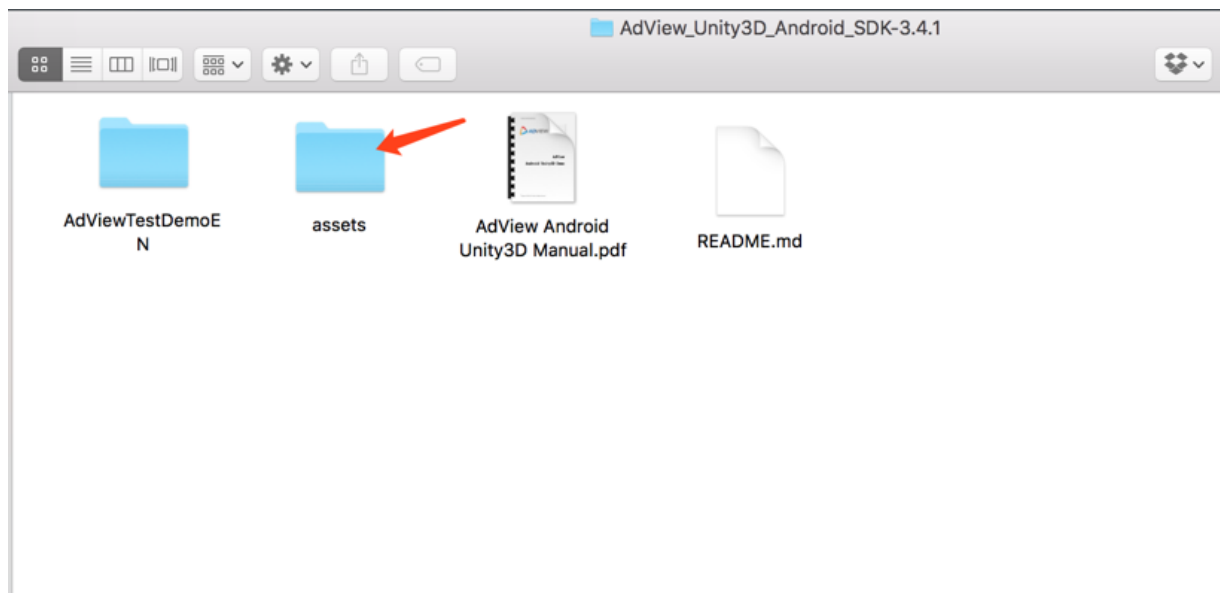
2. Please copy and paste **AdViewSDK_Android.jar, classes.jar and google-play-services.jar** into your application lib folder. you'll need to integrate the Google Play Services SDK into your app. This is mandatory; without Google Play Services, the SDK cannot function.
3. In order to add **new ad network** please copy the .jar file of that particular ad platform provided by AdView to your lib folder and follow the same for all other ad platforms you would like to integrate. In case of **InMobi** add the **InMobi.jar** and **Picasso.jar** files. Please refer for adding custom ad Network .



IV. Add Assets Files

Please Create assets Folder in Android Studio and copy the assets files from AdView_Unity3D_Android_SDK-3.4.1.

AdView_Unity3D_Android_SDK-3.4.1 -->Assets



V. AndroidManifest.xml text configuration

5.1 Add permission code

Required permissions should be added .

please refer to **AndroidManifest** file in the **AdViewTestDemoEN** project.

```
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
```

Note:

-INTERNET: allow to visit network (required)

-ACCESS_NETWORK_STATE: allow to visit various status of mobile phone

(required)

-ACCESS_COARSE_LOCATION: allow a procedure to visit CellID or WIFI to get the rough position.

-ACCESS_FINE_LOCATION: allow a procedure to visit the accurate position (for example, GPS)

-ACCESS_WIFI_STATE: allow a procedure to visit WIFI status

-WRITE_EXTERNAL_STORAGE: allow a procedure to visit outside storage device and can cache ads.

-READ_EXTERNAL_STORAGE: allow a procedure to visit outside storage device

5.2 Add Activity declaration

The given code should be added to in AndroidManifest file for AdView, as some platforms need to declare activity to work normal, please refer to **AndroidManifest file** in **AdViewTestDemoEN** project.

Configurations that adview bidding ads should add:

```
<service android:name="com.kyview.DownloadService" />
<activity android:name="com.kyview.AdviewWebView" />
<activity android:name="com.kyview.AdActivity" />
<!-- Adview bidding video -->
<activity android:name="com.kuaiyou.video.vast.activity.VASTAdActivity"
    android:hardwareAccelerated="true"
    android:screenOrientation="landscape"/>
```


5.3 Appointed app channel

Please add the below code in the AndroidManifest file:

```
<meta-data android:name="AdView_CHANNEL" android:value="GFAN">
</meta-data>
```

(You must add the above code, otherwise your application won't be able to pass the review);

Note:

You can refer **AndroidManifest** file in **AdViewTestDemoEN** Project.

VI. Acquire ad configurations

Note:

1. InitConfiguration serve for the overall procedure, just need to transfer once only.
2. The set methods above are optional, not required.
3. From 3.2.4 version, SDK supports setting up multiple ad slots (SDK-KEY) in one application. Take 3 ad slots of demo keyset for example, some APP would like to set different ad slots in multiple Activities, thus to statistic the user visit amount of each Activity based on the amount of ad display. If one ad slot can meet the demand of APP, then there's no need to apply multiple ad slots.

```
// Be sure to initialize before requesting ads, otherwise the ads cannot be used
// set ad request configured parameter,
//you can use default configuration : InitConfiguration. create Default(this);
InitConfiguration initConfig = new InitConfiguration.Builder(this)
//real-time access to configuration, not required
.setUpdateMode(UpdateMode.EVERYTIME)
// banner switcher can be closed
```

```
.setBannerCloseble(BannerSwitcher.CANCLOSED)
//Interstitial Type
.setInstlDisplayType(InitConfiguration.InstlDisplayType.POPUPWI
NDOW_MODE).build();
```

```
//respectively request banner,interstitial,native, opening scre
en ad
//configuration for one SDK_KEY
AdViewBannerManager.getInstance(this).init(MainActivity.initCon
figuration,new String[]{MainActivity.SDK_KEY});
AdViewInstlManager.getInstance(this).init(MainActivity.initConf
iguration,new String[]{MainActivity.SDK_KEY});
AdViewNativeManager.getInstance(this).init(MainActivity.initCon
figuration,new String[]{MainActivity.SDK_KEY});
AdViewSpreadManager.getInstance(this).init(MainActivity.initCon
figuration,new String[]{MainActivity.SDK_KEY});
AdViewVideoManager.getInstance(this).init(MainActivity.initConf
iguration,new String[]{MainActivity.SDK_KEY});
```

```
// respectively request banner, interstitial, native, opening s
creen ad
//configuration, keyset can be one or more key.
//For Example If you need more than one banner placement while
//initialization this way .
AdViewBannerManager.getInstance(this).init(initConfig,MainActivi
ty.keySet);
AdViewInstlManager.getInstance(this).init(initConfig,MainActivi
ty.keySet);
AdViewNativeManager.getInstance(this).init(initConfig,MainActivi
ty.keySet);
AdViewVideoManager.getInstance(this).init(initConfig,MainActivi
ty.keySet);
```

Note:

You can refer to the code of **MainActivity** in **AdViewTestDemoEN** Project.

VII. Create banner advertising

7.1 Add ads through adding code

Add a banner code on your activity .

```
//Basic Initialization
InitConfiguration initConfiguration = new InitConfiguration.Builder(this)
    .setUpdateMode(InitConfiguration.UpdateMode.EVERYTIME)
    .setBannerCloseble(InitConfiguration.BannerSwitcher.CANCLOSED)
    .setInstlDisplayType(InitConfiguration.InstlDisplayType.POPUPWINDOW_MODE)
    .build();
```

```
//Initialization for Banner
AdViewBannerManager.getInstance(this).init(initConfiguration,new String[]{SDK_KEY});
```

```
//Showing Banner Ad
public void showBanner(){

    ((Activity) context).runOnUiThread(new Runnable() {

        @Override
        public void run() {
            //Code Layout Function for Banner Ad
            CodeLayout(SDK_KEY);

            //Creating Frame Layout Programmatically for Banner Ad
            FrameLayout.LayoutParams params = new FrameLayout.LayoutParams(
                FrameLayout.LayoutParams.WRAP_CONTENT,
                FrameLayout.LayoutParams.WRAP_CONTENT);
            params.gravity = Gravity.BOTTOM;
            addContentView(AdViewBannerManager.getInstance(
```

```

context)
        .getAdViewLayout(context, MainActivity.
SDK_KEY), params);
    }
    });
}

```

```

//Banner Advertisement code
public void CodeLayout(String key) {

    View view = AdViewBannerManager.getInstance(context).ge
tAdViewLayout(
        context, key);
    view.setTag(key);
    if (null != view) {
        ViewGroup rootView = (ViewGroup) findViewById(andro
id.R.id.content);
        for (int i = 0; i < rootView.getChildCount(); i++)
        {
            if (rootView.getChildAt(i) == view) {
                rootView.removeView(view);
            }
        }
    }

    //Requesting For Banner Ad
    //To receive Banner ad events from AdViewBannerListener
    AdViewBannerManager.getInstance(context).requestAd(cont
ext, key,
        new AdViewBannerListener() {

            @Override
            public void onAdReady(String arg0) {
                Log.i(TAG, "onAdReady");
                Toast.makeText(AdBannerActivity.this, "
onAdReady",
                    Toast.LENGTH_SHORT).show();
            }

            @Override
            public void onAdFailed(String arg0) {
                Log.i(TAG, "onAdFailed");

```

```

        Toast.makeText(AdBannerActivity.this,
            "onAdFailed", Toast.LENGTH_SHOR
T).show();
    }

    @Override
    public void onAdDisplay(String arg0) {
        Log.i(TAG, "onAdDisplay");
        UnityPlayer.UnitySendMessage(DEFAULTCAM
ERA,
            "onDisplayAd", "");
    }

    @Override
    public void onAdClose(String key) {
        Log.i(TAG, "onAdClose");
        ViewGroup rootView = (ViewGroup) findVi
ewById(android.R.id.content);
        if (null != rootView)
            rootView.removeView(rootView.findVi
ewWithTag(key));
        UnityPlayer.UnitySendMessage(DEFAULTCAM
ERA,
            "onClosedAd", "");
    }

    @Override
    public void onAdClick(String arg0) {
        Log.i(TAG, "onAdClick");
        UnityPlayer.UnitySendMessage(DEFAULTCAM
ERA,
            "onAdClick", "");
    }
});
}

```

Note:

You can refer to the code of **AdBannerActivity** in **AdViewTestDemoEN** Project.

VIII. Create interstitial advertising

8.1 create interstitial

Note:

Since interstitial ad has a certain life cycle, Please do not wait too long after the request to call showAd method, so as to avoid invalid advertising.

Add the following code to your activity:

```
//Basic Initialization
InitConfiguration initConfiguration = new InitConfiguration.Builder(this)
    .setUpdateMode(InitConfiguration.UpdateMode.EVERYTIME)
    .setBannerCloseble(InitConfiguration.BannerSwitcher.CANCLOSED)
    .setInstlDisplayType(InitConfiguration.InstlDisplayType.POPUPWINDOW_MODE)
    .build();

//Initialization for interstitual advertisement
AdViewInstlManager.getInstance(this).init(initConfiguration, new String[]{SDK_KEY});
```

```
//Requesting Interstitual
public void requestInstl() {

    ((Activity) context).runOnUiThread(new Runnable() {
        @Override
        public void run() {

            // interstitial ad request after initialization
            ,
            //To receive Banner ad events from AdViewInstlListener

            AdViewInstlManager.getInstance(context).requestAd(context, SDK_KEY,
```

```

new AdViewInstllListener() {

    @Override
    public void onAdRecieved(String arg
0) {

        Log.i(TAG, "onReceivedAd");
        UnityPlayer.UnitySendMessage(DE
FAULTCAMERA,

            "onReceivedAd", "");

        Toast.makeText(AdInstlActivity.this,"onReceivedAd"
,Toast.LENGTH_SHORT).show();
    }

    @Override
    public void onAdFailed(String arg0)
    {
        Log.i(TAG, "onReceivedAdFailed"
);
        UnityPlayer.UnitySendMessage(DE
FAULTCAMERA,

            "onReceivedAdFailed", "
");

        Toast.makeText(AdInstlActivity.this,"onReceivedAd"
,Toast.LENGTH_SHORT).show();
    }

    @Override
    public void onAdDisplay(String arg0
) {
        Log.i(TAG, "onDisplayAd");
        UnityPlayer.UnitySendMessage(DE
FAULTCAMERA,

            "onDisplayAd", "");
    }

    @Override
    public void onAdDismiss(String arg0
) {
        Log.i(TAG, "onAdDismiss");
        UnityPlayer.UnitySendMessage(DE
FAULTCAMERA,

            "onAdDismiss", "");
    }
}

```

```

        @Override
        public void onAdClick(String arg0)
        {
            Log.i(TAG, "onClickAd");
            UnityPlayer.UnitySendMessage(DE
                "onClickAd", "");
        }
    });
});
}

```

```
//Show Interstitial
public void showInstl() {

    ((Activity) context).runOnUiThread(new Runnable() {
        @Override
        public void run() {
            // After ad request succeed , call the display
            ad
            AdViewInstlManager.getInstance(context).showAd(
context, MainActivity.SDK_KEY);
        }
    });

}
```

Note:

You can refer to the code of **AdInstlActivity** in **AdViewTestDemoEN**. Project.

IX. Create opening screen ad

9.1 Create opening screen ad

Add the following code to your activity:

```
final RelativeLayout layout = new RelativeLayout(this);
    layout.setLayoutParams(new LayoutParams(LayoutParams.MATCH_PARENT,
        LayoutParams.MATCH_PARENT));

    setContentView(layout);
```

```
//Basic Initialization
InitConfiguration initConfiguration = new InitConfiguration.Builder(this)
    .setUpdateMode(InitConfiguration.UpdateMode.EVERYTIME)
    .setBannerCloseble(InitConfiguration.BannerSwitcher.CANCLOSED)
    .setInstlDisplayType(InitConfiguration.InstlDisplayType.POPUPWINDOW_MODE)
    .build();
```

```
//Intialization for Open Screen ad
AdViewSpreadManager.getInstance(this).init(initConfiguration, new String[]{SDK_KEY});
```

```
// Set the logo at the bottom of opening screen (not required), you can also upload local images or images
AdViewSpreadManager.getInstance(this).setSpreadLogo(R.drawable.spread_logo);

// Set background color of opening screen( not required)
```

```

        AdViewSpreadManager.getInstance(this).setSpreadBackgroudColor( Color.WHITE);

        // Request opening screen ads
        AdViewSpreadManager.getInstance(this).request(this, SDK_KEY,layout, this);

```

9.2 Ad Opening screen Event Handling

To receive events from ad, **you should implement an event listener interface `AdViewSpreadListener`.**

After you implement this listener you will get the following methods.

```

public interface AdViewSpreadListener {
    /**
     * This function is called when the ad is displayed.
     */
    public void onAdDisplay(String key);

    /**
     * This function is called when the ad requestsucceeds.
     */
    public void onAdReceived(String key);

    /**
     * Click to callback .
     */
    public void onAdClick(String key);

    /**
     * This function is called when the ad request failed.
     */
    public void onAdFailed(String key);

    /**
     * This function is called when the ad is closed.
     */
    public void onAdClose(String key);

    /**
     * Custom callback
     */
    public void onAdNotifyCustomCallback(String key,ViewGro

```

```
up view,int ruleTime,int delayTime);  
  
}
```

Note:

Please refer **SpreadScreenActivity** in **AdViewTestDemoEN** Project.

9.3 Custom notification on top of opening screen with countdown options

```
// Skip button will appears on the top after settings  
AdViewSpreadManager.getInstance(this).setSpreadNotifyType(this, AdSpreadManager.NOTIFY_COUNTER_NUM);  
  
// Defaults, none notification will be displayed  
public final static int NOTIFY_COUNTER_NULL = 0;  
  
// Countdown will be shown after settings  
public final static int NOTIFY_COUNTER_NUM = 1;  
  
// Skip button will be shown on the top after settings,but  
it will appear only after specified times.  
public final static int NOTIFY_COUNTER_TEXT = 2;  
  
// Will call this after settings:onAdNotifyCustomCallback(String  
key,ViewGroup view,intruleTime,int delayTime) interface,  
you can  
//custom notification styles  
public final static int NOTIFY_COUNTER_CUSTOM = 3;
```

Note:

For opening advertising please make sure the exposure time is sufficient, otherwise it will affect the ad revenue.

You can refer to the code of **SpreadScreenActivity** in **AdViewTestDemoEN** Project.

X. Create native advertising

10.1 create native advertising

Add a RecyclerView to layout file,

```
<android.support.v7.widget.RecyclerView
    android:id="@+id/list"
    android:layout_width="match_parent"
    android:layout_height="match_parent" />
```

Add the following code to your activity:

```
//Basic Initialization
InitConfiguration initConfiguration = new InitConfiguration.Builder(this)
    .setUpdateMode(InitConfiguration.UpdateMode.EVERTIME)
    .setBannerCloseble(InitConfiguration.BannerSwitcher.CANCLOSED)
    .setInstlDisplayType(InitConfiguration.InstlDisplayType.POPUPWINDOW_MODE)
    .build();

//Intialization for Native advertisement
AdViewNativeManager.getInstance(this).init(initConfiguration, new String[]{SDK_KEY});
```

```
//Initialized native ads should custom ad layout in advance  
, and apply native ad ID at app background  
AdViewNativeManager.getInstance(this).requestAd(this, SDK_KEY, 1, this);
```

```
/** Please add this code in the onAdReceived call back.  
 * For creating this call back method you can refer 9.2  
 */  
// Ad successfully Received  
@Override  
public void onAdRecieved(String arg1, ArrayList arg0) {  
    for (int i = 0; i < arg0.size(); i++) {  
        Data data = new Data();  
        NativeAdInfo nativeAdInfo = (NativeAdInfo) arg0.get(i);  
        data.title = ((NativeAdInfo) arg0.get(i)).getTitle();  
        data.descript = nativeAdInfo.getDescription();  
        data.icon = nativeAdInfo.getIconUrl();  
        data.image= nativeAdInfo.getImageUrl();  
        data.adInfo = (NativeAdInfo) arg0.get(i);  
        ((NativeAdInfo) arg0.get(i)).getIconHeight();  
        data.setAd(true);  
        data.setType(STREAM_AD);  
        Log.i("Native info: ", "data.title:" + data.title +  
"\ndata.descript: " + data.descript + "\ndata.icon: "  
            + data.icon + "\ndata.image: " + data.image);  
        list.add(3, data);  
        ((NativeAdInfo) arg0.get(i)).onDisplay(new View(  
            AdNativeActivity.this));  
    }  
    mAdapter.notifyDataSetChanged();  
}
```

Note:

Please refer to the code of **AdNativeActivity** in **AdViewTestDemoEN** Project.

please add **DividerItemDecoration** ,**MyItemClickListener** classes for Native ad. you can get it from **AdViewTestDemoEN** Project .

10.2 Ad Native Event Handling

To receive events from ad, **you should implement an event listener interface **AdViewNativeListener**.**

After you implement this listener you will get the following methods.

```
public interface AdViewNativeListener {

    /**
     * This function is called when the ad request succeed.
     */
    public void onAdReceived(String key, List<NativeAdInfo>
adMaps);

    /**
     * This function is called when ad request failed.
     */
    public void onAdReceived(String key);

    /**
     * When the ad status changed.
     */
    public void onAdStatusChanged(String key, int status);

}
```

XI . Create video advertising

11.1 create video advertising

Add the following code in activity,

```
//Basic Initialization
InitConfiguration initConfiguration = new InitConfiguration.Builder(this)
    .setUpdateMode(InitConfiguration.UpdateMode.EVERYTIME)
    .setBannerCloseble(InitConfiguration.BannerSwitcher.CANCLOSED)
    .setInstlDisplayType(InitConfiguration.InstlDisplayType.POPUPWINDOW_MODE)
    .build();

//Initialization For Video ads
AdViewVideoManager.getInstance(this).init(initConfiguration, new String[]{SDK_KEY});
```

```
//Requesting video Ad Function
//Request video ads after initialization.
public void requestVideo() {

    ((Activity) context).runOnUiThread(new Runnable() {
        @Override
        public void run() {
            //To receive Video Ad events from AdViewVideoListener

            //Requesting video ads
            AdViewVideoManager.getInstance(context).requestAd(context, SDK_KEY,
                new AdViewVideoListener() {

                    @Override
                    public void onAdClose(String arg0)
```

```

{
    Log.i(TAG, "onAdClose");
    UnityPlayer.UnitySendMessage(DE
FAULTCAMERA,
        "onAdClose", "");
}

@Override
public void onAdFailed(String arg0)
{
    Log.i(TAG, "onAdFailed");
    UnityPlayer.UnitySendMessage(DE
FAULTCAMERA,
        "onAdFailed", "");

    Toast.makeText(AdVideoActivity.this, "onAdF
ailed", Toast.LENGTH_SHORT).show();

}

@Override
public void onAdPlayEnd(String arg0
, Boolean arg1) {
    Log.i(TAG, "onAdPlayEnd");
}

@Override
public void onAdPlayStart(String ar
g0) {
    Log.i(TAG, "onAdPlayStart");
    UnityPlayer.UnitySendMessage(DE
FAULTCAMERA,
        "onAdPlayStart", "");
}

@Override
public void onAdRecieved(String arg
0) {
    Log.i(TAG, "onAdRecieved");
    UnityPlayer.UnitySendMessage(DE
FAULTCAMERA,
        "onAdRecieved", "");
}

```



```

        Toast.makeText(AdVideoActivity.this,"onReceivedAd",Toast.LENGTH_SHORT).show();

    }

});

});

}

/**
 * Showing Ad Video Ad
 */
public void showVideo() {
    ((Activity) context).runOnUiThread(new Runnable() {
        @Override
        public void run() {
            AdViewVideoManager.getInstance(context).playVideo(context, MainActivity.SDK_KEY);
        }
    });
}

```

Note:

You can refer to the code of **AdVideoActivity** in **AdViewTestDemoEN** Project.

XII. Adding Proguard-rules

If you have a ProGuard configuration file please add the below lines of code in proguard-rules.pro file

```

-keep public class com.kyview.** {*; }
-keep public class com.kuaiyou.** {*; }

```

In case you add other adnetworks (like InMobi, AdMob etc..) add their proguard rules to proguard-rules.pro file.

For example In case of InMobi add the below lines of code to the proguard-rules.pro.

```
-keep class com.google.android.gms.ads.identifier.AdvertisingIdClient{
    public *;
}

-keep class com.google.android.gms.ads.identifier.AdvertisingIdClient$Info{
    public *;
}

-keep class com.inmobi.**
{ *; }

-dontwarn com.inmobi.**
```

XIII. Adding custom ad network

AdView has integrated various ad networks (InMobi, AdMob, etc..) where you are given flexibility to add the ad network of your choice and prioritize accordingly, there are various advantages of integrating more than one ad networks to monetize your app, some them include

- > Good fill rates - we suggest to include on an avrage of 3 ad netwroks for good fill rates
- > More revenue
- > You can prioritize according to the ad platform performance

1. In your publisher dashboard <http://adview.com/user/appreport> click on the

"App Management" tab --> click "Configure" against your app name which you want to add custom ad networks, then click on "Add Platform" button (out of China or within China depending on your region) --> you will be redirected to the popup as shown below where you can choose ad network of your choice.

Please select the ad platform to add

☐ select all

Ad platform in China

☐ Wow stick
☐ Posture
☐ Micro Cloud
☐ Easy Media
☐ MadNetwork Chi Road

☐ Dorman
☐ Vpon
☐ AdTouch
☐ Anwar
☐ Curtain one thousand

☐ AppMedia
☐ Atrium
☐ Baidu
☐ Win Advertisement
☐ One hundred million ZAP

☐ Aidesiqi
☐ Friends of the Union
☐ Fei
☐ US forces
☐ Poly win

☐ Heat
☐ Momarken
☐ Chang Si (touch)
☐ Pointing
☐ Guang-point

☐ Tanx Mobile
☐ 360 Poly-efficient
☐ More push
☐ Easy integration
☐ Should search

☐ Zhuo Yi
☐ Jingdong
☐ Hikarion
☐ Hee Mobile Fun
☐ 9 Asahi

☐ Shun fly
☐ European friends
☐ Direct Advertising
☐ Ad exchange
☐ Auction ads

☐ Custom advertising platform

Ad platform out of China

☐ MillennialMedia
☐ Greystripe
☒ InMobi
☐ DOTM
☐ ZestADZ

☐ Smaato
☐ DoubleClick
☐ Adlantis
☐ MobFox

Add
Cancel

2. In this example we will be adding InMobi as the custom Ad Network, select InMobi in the popup and click on "Add" button, this will redirect you to the AdView dashboard --> here turn on the switch again InMobi give the **priority(%)** of your choice then click on "not set" in InMobi row you will get "Configure InMobi" popup with Signup link at the bottom which will redirect you to the below login page, complete the login procedure

Guang-point	2.31	not set	<input type="checkbox"/>	0%	+	Delete
Baidu	3.82	not set	<input type="checkbox"/>	0%	+	Delete
Custom table plaque	1.20	not set	<input type="checkbox"/>	0%	+	Delete
Direct Advertising	1.73	Setting	<input type="checkbox"/>	0%	+	
Ad exchange	1.20	not set	<input type="checkbox"/>	0%	+	

Configure InMobi ✕

ACCOUNT_ID:

PLACEMENT_ID:

Never sign up before? InMobiSign up now>>

User configuration out of China


Advertising platform	Scoring
AdMob	2.53
InMobi	1.45
MobFox	1.20


Total capacity: **100%**

[+Add ad platform](#)

Capacity	Priority	Delete platform
100%	+	Delete
100%	+	Delete
0%	+	Delete

Total capacity: **200%**



Log In Log in with 

[Forgot Password?](#)

[Resend Activation Mail](#)

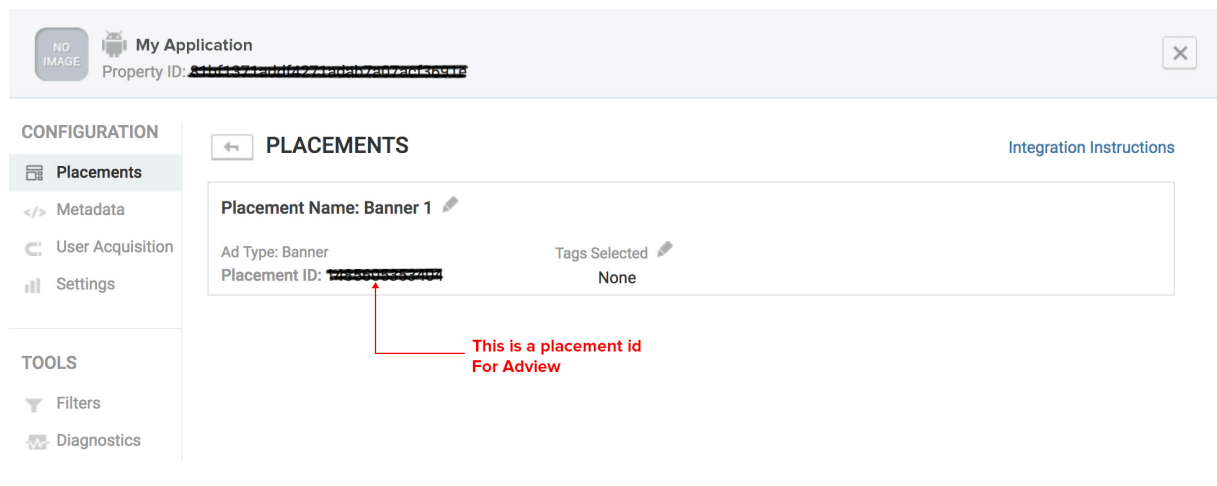
Don't have an account? [Sign Up Now!](#)

Privacy Policy | Terms | Help Center | Contact Us | Copyright © 2015, All rights reserved.

Please add the SDK of InMobi--> please refer **add sdk** .

3.After login you will get the the account ID as shown in the below image, click on Add an App.

6. Please copy the placement ID and Account ID which you need to add it in AdView dashboard as shown in the 2nd image



In the same way you can configure different custom Ad Networks to your app (different Ad networks might demand different ID's to be configured)

XIV. Contact us

Users can login Adview, there are service E-mail, service contact number and enterprise QQ customer service at the bottom of the homepage

