



AdFalcon Android SDK 1.0.4 Developer's Guide

AdFalcon Mobile Ad Network
Product of
Noqoush Mobile Media Group



Table of Contents

1	Introduction	3
	Supported Android version	3
	Permissions	3
2	AdFalcon SDK Classes and Enums	4
	ADFAAdSize enum	4
	ADFErrorCode enum	4
	ADFListener interface.....	4
	ADFTargetingParams class	5
	ADFView class	6
3	AdFalcon Android SDK Integration.....	8
	Step 1: Add JARs.....	8
	Step 2: Add permissions.....	8
	Step 3: Add activity to manifest	9
	Step 4: Add ADFView	9
	A – Without XML layout	9
	B – With XML layout	11
	Step 5 (optionally): implements ADFlistener	12
4	More Information:	13



1 Introduction

AdFalcon Android SDK developer's guide contains all the information needed by an Android developer to integrate with the AdFalcon Network. The guide will also show examples and code snippets on how to perform the integration.

Supported Android version

This SDK is designed to work on all Android platforms starting from Android version 1.6 to the last version 4.

Permissions

AdFalcon SDK requires the following Android permissions to allow access to the Internet and to allow access to device information.

- 1 - android.permission.INTERNET
- 2 - android.permission.ACCESS_NETWORK_STATE
- 3 - android.permission.ACCESS_FINE_LOCATION (Optionally)



2 AdFalcon SDK Classes and Enums

ADFAAdSize enum

ADFAAdSize Enum defines the ad units supported by the AdFalcon.

Enum	Name	Size	Devices
AD_UNIT_320x48	Standard	320 x 48	All
AD_UNIT_300x250	Medium Rectangle	300 x 250	All
AD_UNIT_468x60	Full Banner	468 x 60	Tablet
AD_UNIT_728x90	Leaderboard	728 x 90	Tablet
AD_UNIT_120x600	Skyscraper	120 x 600	Tablet

ADFErroCode enum

ADFErroCode Enum defines AdFalcon SDK and Ad Server error codes.

Enum	Description
GENERIC_SDK_ERROR	Error occurred in the SDK.
INTERNAL_SERVER_ERROR	Error occurred in AdFalcon server.
COMMUNICATION_ERROR	SDK can not reach the AdFalcon server; mostly for issues related to the phone internet connection.
NO_AD_AVAILABLE	No ad is available in the AdFalcon store that meets the ad request parameters.
INVALID_PARAM	An invalid ad request parameter has been set.
MISSING_PARAM	A required ad request parameter has not been set.

ADFListener interface

ADSListener is an interface that is used to get feedback from the AdFalcon SDK.

Method	Description
<code>public void willLoadAd(ADFView view)</code>	SDK will load a new ad.
<code>public void didLoadAd(ADFView view)</code>	SDK has loaded an ad.
<code>public void onError(ADFView view, ADFErroCode error, String message)</code>	<p>SDK has encountered an error during the loading of an ad.</p> <p>Params view: AdFalcon view</p>



	error: type of error message: description for error
<code>public void willPresentAdScreen(ADFView view)</code>	User clicks on an ad view which will display an ad screen.
<code>public void didPresentAdScreen(ADFView view)</code>	Ad screen is displayed.
<code>public void willDismissAdScreen(ADFView view)</code>	User clicks to close the ad screen.
<code>public void didDismissAdScreen(ADFView view)</code>	Ad screen has closed.
<code>public void applicationWillEnterBackground()</code>	Application will enter to background mode

ADFTargetingParams class

The ADFTargetingParams Class contains all the needed parameters about any given user and application in order to help adFalcon network to send most related and targeted ads to the user and application. All parameters are optional

Variable	Required	Description	Values
Language	No	A two-characters string parameter indicating Ad language	ar, en
Postal code	No	A string parameter containing the user's postal/ZIP code	11121
Area code	No	A string parameter containing the user's area code	06
Age	No	An integer parameter containing the user's age	27
Keywords	No	List of keywords in comma separated format. AdFalcon's ads selector engine will search for ads containing these keywords.	ex. sport, news, lifestyle, ...etc.
Gender	No	A parameter containing the user's gender	NONE GENDER_MALE GENDER_FEMALE



Country code	No	Country code of the end user in ISO 3166-1 alpha-2 format code (two-letter code)	JO, SA ...etc.
Birthdate	No	Birthdate of application user in format dd.MM.yyyy	
Additional Info	No	A Map of keys and values to add additional parameters.	
Location: Latitude, Longitude	No	The geolocation information of the device. The location information is divided into two double values; latitude and longitude.	35.654, 34.6598

ADFView class

ADFView is the main SDK class and it is responsible for initializing, loading and rendering an ad.

Method	Description
<pre>public void initialize(String siteID, ADFAdSize adSize, ADFTargetingParams params, ADFListener listener, boolean autorefresh)</pre>	<p>This is the main method in ADFView class and is used to initialize the SDK, load the first Ad and start the auto refresh timer.</p> <p>Parameters</p> <ul style="list-style-type: none"> site ID: You can get this ID from AdFalcon website and it is unique for each application. adSize: The size of the needed ad. params: contains all the needed params to help AdFalcon network to detect and better target an Ad. listener: SDK feedback. autorefresh: used to enable auto refresh for ads.



<code>public void refreshAd()</code>	Used to get a new ad.
<code>public void setTestMode(boolean test)</code>	Used to inform AdFalcon network if your application is working in test or production mode.
<code>public void setRefreshDuration(int duration)</code>	Determine the refresh duration in seconds among the receivedAds.
<code>public void setEnableAutoRefresh(boolean enable)</code>	Used to enable/disable the auto refresh for the received ads.
<code>public boolean isEnableAutoRefresh()</code>	Used to know if auto refresh timer is running or paused.
<code>public void setEnableAnimation(boolean enable)</code>	Used to Enable/Disable animation.
<code>public boolean isEnableAnimation()</code>	Returns true if the animation is enabled otherwise disabled.
<code>public void setListener(ADFListener listener)</code>	Set ADFListener.
<code>public ADFListener getListener()</code>	Get ADFListener.

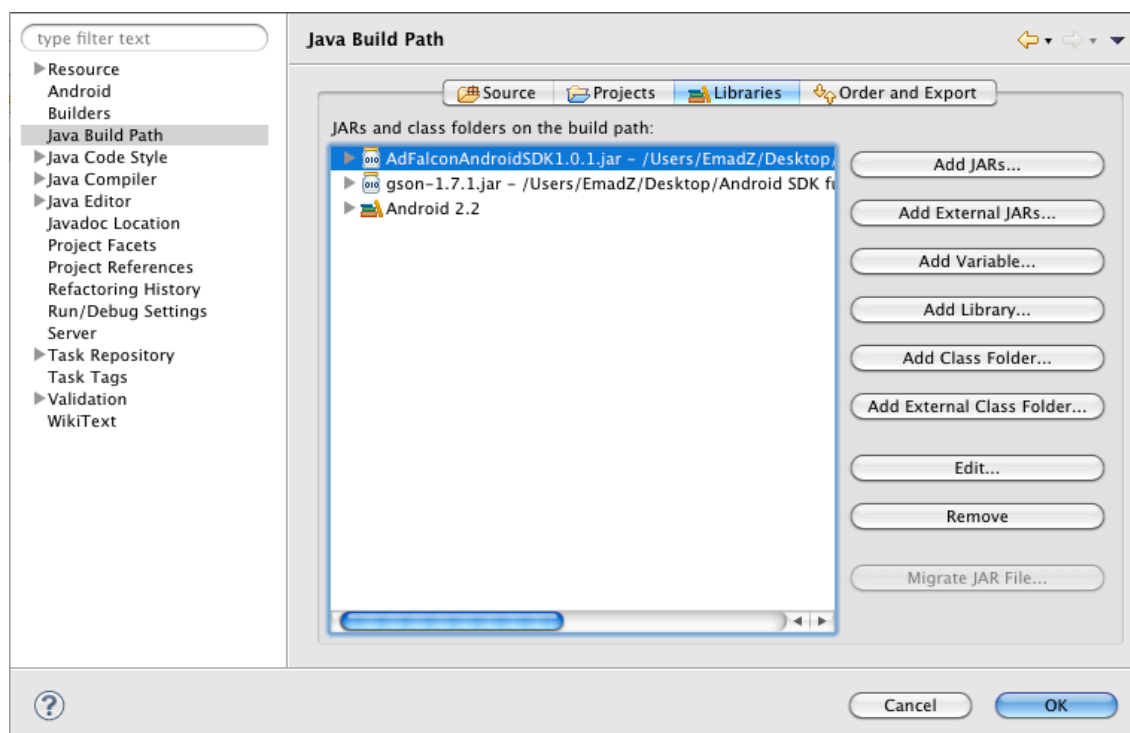


3 AdFalcon Android SDK Integration

This section will explain how to integrate the AdFalcon Android SDK within your application.

Step 1: Add JARs

Add **AdFalconAndroidSDK1.0.4.jar** and **gson-1.7.1.jar** to your application's JAR Libraries.



Step 2: Add permissions

Add permissions to your application **manifest.xml**

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

Optionally: if you want to get geolocation automatically; add location permission in manifest.xml. otherwise you could add geolocation manually in ADFTargetingInfo class.

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




Step 3: Add activity to manifest

Add Canvas activity to your application **manifest.xml**

```
<activity android:name="com.noqoush.adfalcon.android.sdk.ADFCanvas"
    android:theme="@android:style/Theme.Translucent">
    <intent-filter>
        <action android:name="android.intent.action.VIEW"/>
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
```

Step 4: Add ADFView

You have two ways adding ADFView to your activity class, these ways are:

Note: Ensure Test Mode Parameter is set to false before uploading your application to the android market.

A – Without XML layout

Set the following snap of code in your activity to add, initialize and load ad. You do not need using XML layout.

```
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Vector;
import android.widget.LinearLayout;
import com.noqoush.adfalcon.android.sdk.ADFListener;
import com.noqoush.adfalcon.android.sdk.ADFTargetingParams;
import com.noqoush.adfalcon.android.sdk.ADFView;
import com.noqoush.adfalcon.android.sdk.constant.ADFAdSize;
import com.noqoush.adfalcon.android.sdk.constant.ADFErrorCode;
.
.
.
// Implements ADFListener in case you want to use it.

public class ... extends Activity implements ADFListener

// Add private member variable of type ADFView

private ADFView adFalconView;

public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);
try {

    // create new instance of ADFView

    adFalconView = new ADFView(this);

    // create reference to singleton ADFTargetingParams class. all
    // params are optional

    ADFTargetingParams params = new ADFTargetingParams();

    // Add age or birthdate
```



```

        //params.setAge(27);
        //params.setBirthdate(new SimpleDateFormat("dd.MM.yyyy")
        //    .parse("21.11.1984"));

        //params.setGender(ADFTargetingParams.GENDER_MALE);
        //params.setLanguage("ar");

        //params.setAreaCode("962");
        //params.setPostalCode("11121");
        //params.setCountryCode("JO");

        //if you set location permission, the SDK will get location info
        // automatically.
        // in case you want to pass location info manually, use the
        // following methods

        //params.setLocationLatitude(31.956641);
        //params.setLocationLongitude(35.847037);

        // Add keywords of ad

        //Vector<String> keywords = new Vector<String>();
        //keywords.add("Sport");
        //params.setKeywords(keywords);

        //Ensure test mode is setting to false when your app is uploaded in the marketing

        adFalconView.setTestMode(true);

        // initialize the view by pass publisher id, ad unit size, params,
        // listener and enable auto refresh.
        // then load first ad

        adFalconView.initialize("xxxxxxx", ADAdSize.AD_UNIT_320x48, params, this, true);

        // Add ADFView to this activity.
        // we consider there is a layout naming linearLayout and
        // you want to add ADFView in this layout

        ((LinearLayout) findViewById(R.id.linearLayout)).addView(adFalconView);

    } catch (Exception ex) {
    }
}

```



B – With XML layout

- Add ADFView to activity XML layout

```
<com.noqoush.adfalcon.android.sdk.ADFView
    android:id="@+id/adFalconView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>
```

- Add the following snape of code in your activity class to initilize, load and render ads

```
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Vector;
import android.widget.LinearLayout;
import com.noqoush.adfalcon.android.sdk.ADFListener;
import com.noqoush.adfalcon.android.sdk.ADFTargetingParams;
import com.noqoush.adfalcon.android.sdk.ADFView;
import com.noqoush.adfalcon.android.sdk.constant.ADFAdSize;
import com.noqoush.adfalcon.android.sdk.constant.ADFErrorCode;
.
.
.
// Implements ADFListnsner in case you want to use it.

public class ... extends Activity implements ADFListener

// Add private member variable of type ADFView

private ADFView adFalconView;

public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);
try {

    // create new instance of ADFView
    adFalconView = ((ADFView)findViewById(R.id.adFalconView));

    // create reference to singleton ADFTargetingParams class. all
    // params are optional
    ADFTargetingParams params = new ADFTargetingParams();

    // Add age or birthdate

    //params.setAge(27);
    //params.setBirthdate(new SimpleDateFormat("dd.MM.yyyy")
    //    .parse("21.11.1984"));

    //params.setGender(ADFTargetingParams.GENDER_MALE);
    //params.setLanguage("ar");

    //params.setAreaCode("962");
    //params.setPostalCode("11121");
    //params.setCountryCode("JO");

    //if you set location permission, the SDK will get location info
    // automatically.
    // in case you want to pass location info manually, use the
    // following methods

    //params.setLocationLatitude(31.956641);
    //params.setLocationLongitude(35.847037);

    // Add keywords of ad
```



```
//Vector<String> keywords = new Vector<String>();  
//keywords.add("Sport");  
//params.setKeywords(keywords);  
  
//Ensure test mode is setting to false when your app is uploaded in the marketing  
adFalconView.setTestMode(true);  
  
// initialize the view by pass publisher id, ad unit size, params,  
// listener and enable auto refresh.  
// then load first ad  
  
adFalconView.initialize("xxxxxxx", ADFAdSize.AD_UNIT_320x48, params, this, true);  
  
} catch (Exception ex) {  
}  
}
```

Step 5 (optionally): implements ADFlisterner

ADFlisterner provides ability monitoring AdFalcon SDK cycle. Add the following abstract methods to the activity in case you have implemented ADFlisterner.

```
public void willLoadAd(ADFView view) {  
}  
  
public void didLoadAd(ADFView view) {  
}  
  
public void onError(ADFView view, ADFErrorCode error,  
String message) {  
}  
  
public void willPresentAdScreen(ADFView view) {  
}  
  
public void didPresentAdScreen(ADFView view) {  
}  
  
public void willDismissAdScreen(ADFView view) {  
}  
  
public void didDismissAdScreen(ADFView view) {  
}  
  
public void applicationWillEnterBackground() {  
}
```



4 More Information:

You can find more information in the sample project within the downloaded zip file.

For any SDK integration queries, please send us an email to support@adfalcon.com along with your login id.