Android mOcean SDK

Developer Guide

For Android SDK Version 2.x

Table of Contents

[Overview 3](#_Toc283903126)

[System requirements: 3](#_Toc283903127)

[SDK contents: 3](#_Toc283903128)

[Installation instructions: 3](#_Toc283903129)

[SDK API reference: 6](#_Toc283903130)

[Global functions: 6](#_Toc283903131)

[AdServerView arguments 6](#_Toc283903132)

[Sample usage 11](#_Toc283903133)

## Overview

This document provides a description of the following:

* System requirements
* Installation Instructions
* SDK contents
* SDK API reference
* Sample of usage

### System requirements:

* Android SDK (version 1.5 or later)
* Eclipse 3 or later
* 10 Mb free disk space

### SDK contents:

Documentation - SDK documentation

Lib - SDK library files

Sample - sample usage/test app

### Installation instructions:

*Installing Android SDK*

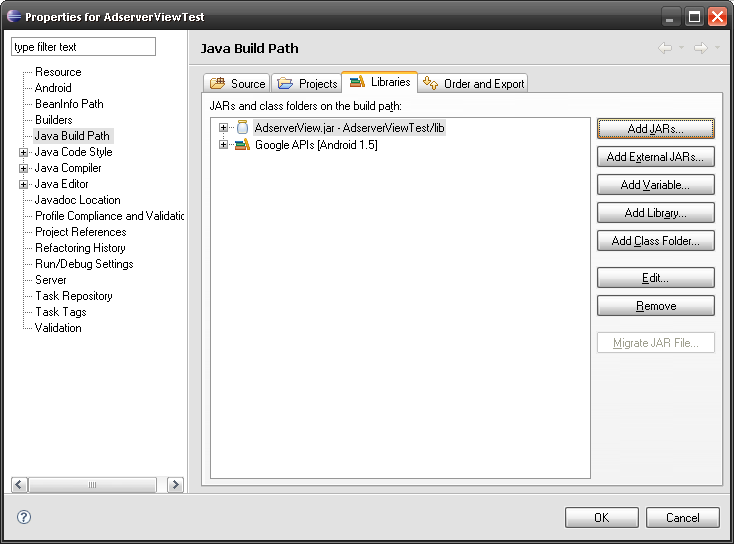
Download and install Android SDK (http://developer.android.com/sdk/index.html).

Follow installation instructions provided by Android SDK. Once SDK has been installed follow to the next step to install Android mOcean SDK.

*Installing Android mOcean SDK*

The SDK has been developed as a jar library. To add the library to a project, the developer must configure the project properties to indicate the location of library file, as well as the names of library dependencies.

First, select Project -> Properties and add the library to the dialog under Java Build Path -> Libraries. Click "Add JARs..." (or "Add External JARs...") and select Android mOcean SDK:



After installing the library, open the provided test project, AdserverViewTest (AdserverView\test), and attempt to build the project. If unable to compile, please review the Eclipse documentation on configuring library file paths to ensure changes were properly made. If unable to execute, you may be experiencing library version issues. Please refer to the developer forums for more information.

*Add “*minSdkVersion*” parameter in project manifest file (AndroidManifest.xml)*

Example: <uses-sdk android:minSdkVersion="3" />

*To set the security permissions in your manifest file (AndroidManifest.xml)*

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

<uses-permission android:name=*"android.permission.ACCESS\_NETWORK\_STATE"*></uses-permission>

<uses-permission android:name=*"android.permission.ACCESS\_FINE\_LOCATION"*></uses-permission>

<uses-permission android:name=*"android.permission.READ\_PHONE\_STATE"*></uses-permission>

<uses-permission android:name=*"android.permission.READ\_CALENDAR"*></uses-permission>

<uses-permission android:name=*"android.permission.WRITE\_CALENDAR"*></uses-permission>

<uses-permission android:name=*"android.permission.CALL\_PHONE"*></uses-permission>

<uses-permission android:name=*"android.permission.SEND\_SMS"*></uses-permission>

<uses-permission android:name=*"android.permission.CAMERA"*></uses-permission>

<uses-permission android:name=*"android.permission.WRITE\_EXTERNAL\_STORAGE"*></uses-permission>

<uses-permission android:name=*"android.permission.WAKE\_LOCK"* />

Add the following in the <application> section, replacing <YOUR\_APPLICATION\_PACKAGE> with a

package identifier that is unique to your application. This Content Provider manages local storage of ad content, while the Activity manages ad display.

<provider android:name=*"com.greystripe.android.sdk.AdContentProvider"*

android:authorities=*"<YOUR\_APPLICATION\_PACKAGE>.AdContentProvider"*

android:multiprocess=*"true"*

android:exported=*"false"*

/>

<service android:name=*"com.medialets.advertising.AdManagerService"* />

<activity android:name=*"com.greystripe.android.sdk.AdView"*

android:configChanges=*"keyboard|keyboardHidden|orientation"* >

<intent-filter>

<category android:name=*"android.intent.category.LAUNCHER"* />

</intent-filter>

</activity>

<activity android:name=*"com.admob.android.ads.AdMobActivity"*

android:theme=*"@android:style/Theme.NoTitleBar.Fullscreen"*

android:configChanges=*"orientation|keyboard|keyboardHidden"*

/>

<activity android:name=*"com.millennialmedia.android.MMAdViewOverlayActivity"*

android:theme=*"@android:style/Theme.Translucent.NoTitleBar"*>

</activity>

<activity android:name=*"com.millennialmedia.android.VideoPlayer"*

android:theme=*"@android:style/Theme.NoTitleBar.Fullscreen"*

android:configChanges=*"keyboardHidden|orientation|keyboard"* >

</activity>

It is necessary to add following parameters

<string name=*"app\_id"*>764f0842b9025416f78b6b99891cb70302bd8c50</string>

<string name=*"use\_test\_servers"*>true</string>

to strings.xml.

## SDK API reference:

### Global functions:

**AdLog.setDefaultLogLevel**(int logLevel)

AdLog.LOG\_LEVEL\_NONE none

AdLog.LOG\_LEVEL\_1 only errors

AdLog.LOG\_LEVEL\_2 +warning

AdLog.LOG\_LEVEL\_3 +server traffic

**AdLog.*setFileLog***(String fileName)file to output

**AdServerView**(Context context, java.lang.String site, java.lang.String zone)

Creation of viewer of advertising.

**AdServerInterstitialView**(Context context, java.lang.String site, java.lang.String zone)

Creation of interstitial viewer of advertising.

void **show**()

Show Interstitial advertising.

void SetLogLevel(int logLevel).

All settings can be set by means of methods get and set (for example: getCity() and setCity(String city)).

java.lang.String getAdserverURL()

Optional.

java.lang.String getAdvertiserId()

Optional.

java.lang.String getArea()

Optional.

java.lang.String getBackgroundColor()

Optional.

java.lang.String getCarrier()

Optional.

java.lang.String getCity()

Optional.

java.lang.String getCountry()

Optional.

java.util.Hashtable<java.lang.String,java.lang.String> getCustomParameters()

Optional.

java.lang.Integer getDefaultImage()

Optional.

java.lang.String getGroupCode()

Optional.

boolean getInternalBrowser()

Get the flag which operates advertising opening.

java.lang.String getKeywords()

Optional.

java.lang.String getLatitude()

Optional.

java.lang.String getLongitude()

Optional.

java.lang.Integer getMaxSizeX()

Optional.

java.lang.Integer getMaxSizeY()

Optional.

java.lang.String getMetro()

Optional.

java.lang.Integer getMinSizeX()

Optional.

java.lang.Integer getMinSizeY()

Optional.

AdServerViewCore.OnAdClickListener getOnAdClickListener()

Get interface for advertising opening.

AdServerViewCore.OnAdDownload getOnAdDownload()

Get interface for advertising downloading.

java.lang.Integer getPremium()

Optional.

java.lang.String getRegion()

Optional.

java.lang.String getSite()

Get the id of the publisher site.

java.lang.String getState()

java.lang.Boolean getTest()

Optional.

java.lang.String getTextColor()

Optional.

java.lang.Integer getUpdateTime()

Optional.

java.lang.String getZip()

Optional.

java.lang.String getZone()

Get the id of the zone of publisher site.

void setAdserverURL(java.lang.String adserverURL)

Optional.

void setAdvertiserId(java.lang.String advertiserId)

Optional.

void setArea(java.lang.String area)

Optional.

void setBackgroundColor(java.lang.String backgroundColor)

Optional.

void setCarrier(java.lang.String carrier)

Optional.

void setCity(java.lang.String city)

Optional.

void setCountry(java.lang.String country)

Optional.

void setCustomParameters(java.util.Hashtable<java.lang.String,java.lang.String> customParameters)

Optional.

void setDefaultImage(java.lang.Integer defaultImage)

Optional.

void setGroupCode(java.lang.String groupCode)

Optional.

void setInternalBrowser(boolean internalBrowser)

Set the flag which operates advertising opening.

void setKeywords(java.lang.String keywords)

Optional.

void setLatitude(java.lang.String latitude)

Optional.

void setLongitude(java.lang.String longitude)

Optional.

void setMaxSizeX(java.lang.Integer maxSizeX)

Optional.

void setMaxSizeY(java.lang.Integer maxSizeY)

Optional.

void setMetro(java.lang.String metro)

Optional.

void setMinSizeX(java.lang.Integer minSizeX)

Optional.

void setMinSizeY(java.lang.Integer minSizeY)

Optional.

void setOnAdClickListener(AdServerViewCore.OnAdClickListener adClickListener)

Set interface for advertising opening.

void setOnAdDownload(AdServerViewCore.OnAdDownload adDownload)

Set interface for advertising downloading.

void setPremium(java.lang.Integer premium)

Optional.

void setRegion(java.lang.String region)

Optional.

void setSite(java.lang.String site)

Required.

void setTest(java.lang.Boolean enabled)

Optional.

void setTextColor(java.lang.String textColor)

Optional.

void setUpdateTime(java.lang.Integer updateTime)

Optional.

void setZip(java.lang.String zip)

Optional.

void setZone(java.lang.String zone)

Required.

void update()

Immediately update banner contents.

## Sample usage

See AdserverViewTest (AdserverView\test) in SDK contents.

To create viewer of advertising and to add it to the form.

Is possible two variants of creation and object placing:

1. Dynamic creation, is created in a form **class**.

For example:

AdServerView adserverView = new AdServerView(this, "5441", "9312");

adserverView.setLayoutParams(new ViewGroup.LayoutParams(ViewGroup.LayoutParams.FILL\_PARENT, 250));

adserverView.setTest(false);

adserverView.setPremium(AdServerView.PREMIUM\_STATUS\_PREMIUM);

adserverView.setKeywords("test");

adserverView.setMinSizeX(200);

adserverView.setMinSizeY(40);

adserverView.setMaxSizeX(320);

adserverView.setMaxSizeY(50);

adserverView.setBackgroundColor("33CCFF");

adserverView.setTextColor("11CC22");

adserverView.setInternalBrowser(false);

linearLayout.addView(adserverView);

2. Template creation, is created in XML a template.

For example:

<com.adserver.adview.AdServerView

android:id="@+id/adViewer1"

android:layout\_width="fill\_parent"

android:layout\_height="50dip"

site="5441"

zone="6365"

isTestModeEnabled="false"

defaultImage="@drawable/test\_banner"

/>

To show Viewer of interstitial advertising.

AdServerInterstitialView interstitialView = new AdServerInterstitialView(context, "6235", "9312");

interstitialView.setShowCloseButtonTime(3);

interstitialView.setAutoCloseInterstitialTime(20);

interstitialView.show();

-- END OF DOCUMENT --