



# Emediate Android SDK Developer Guide

---

For Android SDK Version 0.4

## Table of Contents

Overview.....	3
Introduction.....	3
Specification.....	3
Development Requirement.....	3
SDK Contents.....	3
SDK Installation Instructions.....	3
Installation of Android SDK .....	3
Installation of Emediate Android SDK.....	4
Emediate Mobile Ads URL.....	5
URL Example.....	5
Model.....	5
SDK API References.....	6
Class EmediateView.....	6
Overview.....	6
Global functions.....	6
Class Param.....	7
Overview.....	7
Global functions.....	7
Usages of SDK.....	7
Basic Class import.....	7
Invoke EmediateView.....	7
Important Thing.....	8

## Overview

This document indicates the following descriptions:

- Introduction
- Specification
- Development Requirement
- SDK Contents
- SDK Installation Instructions
- SDK API References
- Usages of this SDK

## Introduction

Emediate Android SDK is facilitating Android native mobile apps development with Emediate Mobile Ads, and speed both software development as well as integration/design issues.

## Specification

Emediate Android SDK is a MRAID v1.0 and ORMMA level 1 compliant SDK. Reference implementations for MRAID and ORMMA can be found as follows:

- <http://code.google.com/p/ormma/>
- <http://www.iab.net/mraid>

## Development Requirement

- Android SDK --- Platform 2.1+(API Level 7+)
- Eclipse 3.5 or later version

## SDK Contents

- EmediateSDK.jar --- Emediate SDK JAR Library file
- EmediateSDKTester --- Emediate SDK Test Demo Applications

## SDK Installation Instructions

### ***Installation of Android SDK***

Official Android SDK download url: <http://developer.android.com/sdk/index.html>

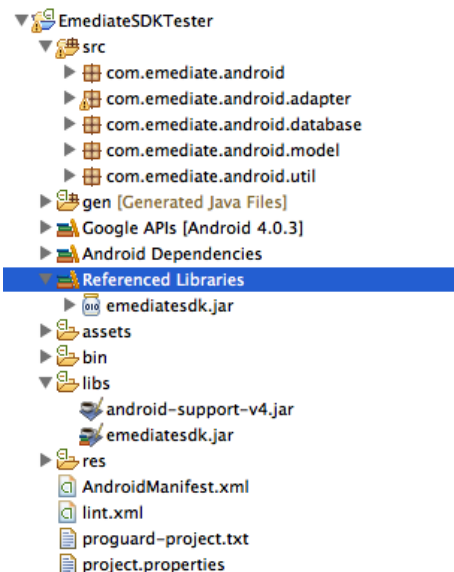
Download the Android SDK, and set up your development environment according to the installation instruction supported by above link.

## Installation of Emediate Android SDK

The Emediate Android SDK is one compiled Java jar library, which means the developer need to import this SDK jar lib into the target Android project, and also some relevant project properties will need to be configured.

Here is one instruction as follows:

1. Create your Android project into your Workspace by Eclipse.
2. Create library folder
  - a. Choose your project in Eclipse, and right click on your project, then, select “New->Folder”
  - b. Name the new folder “libs”, and click “Finish”. You will find the library folder under project root directory.
3. Import emediatedsdk.jar into your project as a library
  - a. Copy emediatedsdk.jar from the SDK\_Package.
  - b. Paste the jar file into “libs” folder.
  - c. Right click on the jar file, and select “Build Path -> Add to Build Path”, then you will see that one new Dependencies Folder called “Referenced Libraries” has been created.



*Figure 1: Added emediatedsdk.jar file into project as library*

4. Updating project AndroidManifest.xml file
  - a. To make sure the project will support Android SDK 2.1+, you will need to include the “minSdkVersion”:  
`<uses-sdk android:minSdkVersion="7" android:targetSdkVersion="15" />`
  - b. Include the following EmediateSDK Android Activities:
    - i. Embedded Browser Activity --- This Activity will be called/started

when the Emediate Mobile Ad has any intent to open a link. This Embedded Browser will work with the normal browser functions: “Forward”, “Backward”, “Refresh” and “Close”. Add the following fragment:

```
<activity android:name="com.emediate.view.Browser" />
```

ii. MraidActionHandler Activity --- This Activity will be called/started if there is any video/audio to be played within the Emediate Mobile Ads. Include the following fragment:

```
<activity android:name="com.emediate.view.MraidActionHandler" />
```

c. Setup relevant Android Device security permissions

Permission	Description & Manifest XML fragment
INTERNET	Access internet. Required for Emediate Ad content fetching. <uses-permission android:name="android.permission.INTERNET" />
ACCESS_NETWORK_STATE	Access information about networks <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
ACCESS_COARSE_LOCATION	Access coarse(eg. Cell-ID, WIFI) location <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
ACCESS_FINE_LOCATION	Access fine(e.g. GPS) location <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />

5. Clean Project by selecting “Project->Clean...” in the Eclipse top menu.

## Emediate Mobile Ads URL

### URL Example

*http://stage.emediate.eu/eas?  
cu=123&cre=mu&eas\_uid=555&EASTSDK=1&EASTorientation=landscape&EASTscreen  
=bi g&EASTsection=sports*

### Model

Emediate Mobile Ads URL includes a certain number of URL parameters which declare the specification of the ads.

- **DEVICE\_ID**
  - KeyName: eas\_uid
  - ValueType: unique 64bit unsigned integer
  - IsMandatory: Yes
  - Description: This will be proceeded by SDK. This should be generated in such way that the same number is always generated on the same device. The native device ID must not be used. I.e. hash the unique device id to generate the DEVICE\_ID.
- **SYSTEM\_URL**
  - ValueType: String

- IsMandatory: Yes
- DefaultValue: <http://stage.emediate.eu/eas>
- **CONTENT\_UNIT\_ID**
  - KeyName: cu
  - ValueType: unique unsigned integer
  - IsMandator: Yes
  - Description: Generate and store the Unique ID for current device, if not stored already.
- **Optional parameters**
  - KeyNameType: must be 16 chars or less
  - ValueType: must be 32 chars or less
  - Format: *EAST{KeyName}={Value}*

## SDK API References

### Class *EmediateView*

#### Overview

This is the view that should be placed into the layout when implementing MARID/ORMMA functionality. It can be used either via xml or programmatically. It is a subclass of the standard WebView which brings with it all the standard functionality as well as the inherent bugs on some os versions. WebViews have a tendency to leak on orientation in older versions of the Android OS. This can be minimized by using an application context, but this will break the launching of SubWindows (such as alert calls from javascript).

#### Global functions

**EmediateView**(Context context)

Emediate Mobile Ad Container.

**EmediateView**(Context context, AttributeSet set)

Emediate Mobile Ad Container.

void **setAdsBaseUrl**(String adsBaseUrl)

Set Emediate Mobile Ad Base URL.

The default value is: <http://stage.emediate.eu/eas>

void **setAdsRefreshRate**(int adsRefreshRate)

Set Emediate Mobile Ad Refresh rate.

The Unit of rate is seconds, if adsRefreshRate = 10, the actual refresh rate is 10\*1000 milliseconds.

The Default RefreshRate is 60\*1000 milliseconds.

## Class Param

### Overview

This class includes two public properties:

Name	Description	Type
key	the key-name of URL Parameter	String
value	the value of the relevant URL parameter's key	String

### Global functions

void **setAdsParams**(ArrayList<Param> adsParams)

Set Emediate Mobile Ads URL Parameters.

void **fetchCampaignNormal**()

Fetch Emediate Mobile Ads(also called Campaign).

When this api is called, the pre set Emediate Mobile Ad's URL and parameters will be reassembled by the SDK and generate one final adsURL, which will be used to fetch the Ads.

void **fetchCampaignByFinalUrl**(String adsUrl)

Fetch Emediate Mobile Ad

However, this API only requires one completely Emediate Mobile Ad's URL as parameter, without initialize setAdsParams(), setAdsBaseUrl().

## Usages of SDK

As illustrated, the “EmediateView” is the Mobile Ad container, which will be invoked within your project source code. Similarly, “EmediateView” is extended as Android WebView. It can also be added into source code by two ways: Dynamically using Java Programming Code, and statically using the Android XML Layout Render.

### Basic Class import

```
import com.emediate.controller.model.Param;
import com.emediate.view.EmediateView;
```

### Invoke EmediateView

- Dynamically

```
EmediateView mAdView = new EmediateView(this);
LinearLayout.LayoutParams layoutParams = new
    LinearLayout.LayoutParams(LinearLayout.LayoutParams.MATCH_PARENT,
        (int) (50 * getResources().getDisplayMetrics().density));
mAdView.setLayoutParams(layoutParams);
```

```

mAdView.removeAllViews();
// Set Ads Base URL
mAdView.setAdsBaseUrl("http://stage.emediate.eu/eas");
// Set Ads Parameters
ArrayList<Param> mParams = new ArrayList<Param>();
// Example
// Add Param cu, kw1
Param mParam = new Param();
mParam.key = "cu";
mParam.value = "512";
mParams.add(mParam);

mParam = new Param();
mParam.key = "kw1";
mParam.value = "expand";
mParams.add(mParam);
mAdView.setAdsParams(mParams);

// Set Ads Refresh Rate 120*1000 seconds
mAdView.setAdsRefreshRate(120);
// Fetch Campaign Ads
mAdView.fetchCampaignByFinalUrl("http://stage.emediate.eu/eas?
                                cu=512&kw1=expand");

mAdView.fetchCampaignNormal();

LinearLayout.addView(mAdView);

```

- **Statically**

```

<com.emediate.view.EmediateView
    android:id="@+id/ad"
    android:layout_width="wrap_content"
    android:layout_height="50dp" />
<!-- Initil the Emeidate Mobile Ad's height as 50Dp -->

// Declare EmediateView in your source code
EmediateView mAdView = (EmediateView) findViewById(R.id.ad);

```

## ***Important Thing***

No matter if the EmediateView is invoked in a dynamic or static way, we need to set the Emediate Mobile Ad's height to its default state.