

Documentación

Guía Integración Librería de publicidad con Real Media para Android

Código: 1 **Edición:** 01.00

Fecha: 13/12/2011

Historial de Revisión

Fecha	Versión	Descripción	Autores
13/12/2011	1.0	Versión inicial	Antonio Ochoa, Sergio Girao, Arturo Gutiérrez

Documentos Asociados

Fecha	Versión	Descripción	Autor
13/12/2011	1.0	Librería Android	
13/12/2011	1.0	Aplicación de ejemplo	

Tabla de contenidos

HISTORIAL DE REVISIÓN	2
DOCUMENTOS ASOCIADOS	2
1. INTRODUCCIÓN	4
2. POSICIONES PUBLICITARIAS	5
2.1.1 <i>Notas generales sobre posiciones:</i>	5
2.1.2 <i>Descripción de posiciones publicitarias ANDROID</i>	5
3. INTEGRACIÓN ANDROID	6
3.1.1 <i>Requerimientos:</i>	6
3.1.2 <i>Incorporando el SDK</i>	6
3.1.3 <i>Añadiendo un com.prisadigital.realmedia.adlib.AdView</i>	10
3.1.4 <i>Añadiendo un AdView en forma de XML</i>	11
3.1.5 <i>Interstitial</i>	11

1. Introducción

En este documento se describen las posiciones publicitarias disponibles para ANDROID, los pasos para la integración de la librería y ejemplos para incluirla en las aplicaciones de ANDROID.

Junto con esta documentación se distribuye la librería con el código y el código fuente de una aplicación de ejemplo con todas las posiciones publicitarias integradas.

La gestión de memoria de la librería se realiza desde la propia librería.

El código de la librería es propiedad de Prisa Digital, a excepción del código encargo del soporte para GIF animado y las modificaciones realizadas sobre él (ver comentarios para mas detalles).

2. Posiciones Publicitarias

2.1.1 Notas generales sobre posiciones:

Es tarea del integrador colocar los anuncios en la posición deseada dentro de su aplicación.

El tiempo que tardan los anuncios en desaparecer se controla desde el AdServer.

2.1.2 Descripción de posiciones publicitarias ANDROID

Plataforma	ANDROID	Nombre	Facilitado por Real Media.
Localización	A elección del integrador.	Formatos aceptados	JPG, GIF, GIF Animados, MP4.
Tamaño máximo	320x50 / 320x300 en MDPI. Escalado para el resto de resoluciones.		
Comportamiento	<ul style="list-style-type: none">- Banner que muestra una imagen o un video.- Puede ser expandible a otra imagen o video.		
Comentarios			

Plataforma	ANDROID	Nombre	Facilitado por Real Media.
Localización	Interstitial	Formatos aceptados	JPG, GIF, GIF Animados, MP4.
Tamaño máximo	320x460 MDPI. Escalado para el resto de resoluciones.		
Comportamiento	<ul style="list-style-type: none">- Aparece ocupando toda la pantalla.- Puede contener video.- Posee botón de cerrar.		
Comentarios	Se ejecuta en una Activity aparte.		

3. Integración ANDROID

A continuación se explica como integrar la librería de publicidad en las aplicaciones Android.

IMPORTANTE (Antes de pasar a producción solicitar a Prisa BS tanto el ID de la aplicación correspondiente como el ID de sus posiciones)

3.1.1 Requerimientos:

RealMedia SDK requiere Android 1.6 o superior. Asegúrate que estás compilando contra al menos Android 1.6 (pon `target` igual `android-4` en `default.properties`)

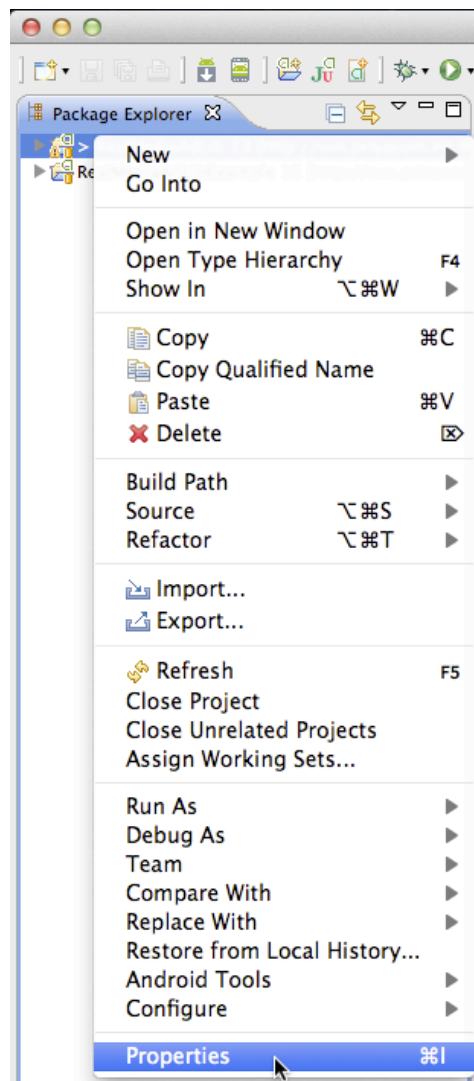
3.1.2 Incorporando el SDK

Efectúa los siguiente tres pasos:

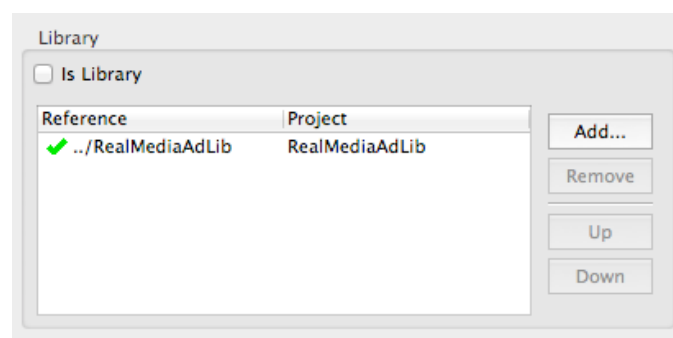
1. Añade el SDK JAR a tu proyecto de Eclipse.
2. Declara `com.prisadigital.realmedia.adlib.AdActivity` dentro de `AndroidManifest.xml`.
3. Añade los permisos requeridos en el manifest.

Añadiendo el SDK

1. Click derecho en el proyecto de tu aplicación en Eclipse y elige **Properties**.



2. Selecciona **Android** y en la sección **Library** clicas en **Add...** para añadir el SDK de RealMedia como un proyecto librería. Si no te sale para elegir necesitas incorporar primero el proyecto SDK de RealMedia a tu workspace. Debe quedar algo cómo:



AdActivity

El SDK requiere que `com.prisadigital.realmedia.adlib.AdActivity` sea declarado en tu `AndroidManifest.xml`:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.company"
    android:versionCode="1"
    android:versionName="1.0.0">

    <uses-sdk android:minSdkVersion="4" />

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <!-- Optional to cache the video ads in SD -->
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />

    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".RealMediaAdLibExampleActivity"
            android:label="@string/app_name"
            android:configChanges="keyboard|keyboardHidden|orientation">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity android:name="com.prisadigital.realmedia.adlib.AdActivity"
            android:configChanges="keyboard|keyboardHidden|orientation|screenLayout"
            android:screenOrientation="portrait"/>
    </application>
```


Permisos

El SDK requiere que los permisos INTERNET y ACCESS_NETWORK_STATE sean declarados en el Manifest. El permisos WRITE_EXTERNAL_STORAGE es opcional para cachear los video anuncios en la SD del terminal.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.company"
    android:versionCode="1"
    android:versionName="1.0.0">

    <uses-sdk android:minSdkVersion="4" />

    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <!-- Optional to cache the video ads in SD -->
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />

    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".RealMediaAdLibExampleActivity"
            android:label="@string/app_name"
            android:configChanges="keyboard|keyboardHidden|orientation">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity android:name="com.prisadigital.realmedia.adlib.AdActivity"
            android:configChanges="keyboard|keyboardHidden|orientation|screenLayout"/>

    </application>
</manifest>
```

3.1.3 Añadiendo un `com.prisadigital.realmedia.adlib.AdView`

Las aplicaciones de Android están compuestas por objetos View, instancias Java que el usuario ve como áreas de texto, botones u otros controles. AdView es simplemente una View más que muestra anuncios y responde a la interacción con el usuario.

Como cualquier otra View, una AdView puede ser creada puramente por código o en XML.

Las siguientes 4 líneas de código añade un banner:

1. Importa `com.prisadigital.realmedia.adlib.*`
2. Declara una instancia de `AdView`
3. Créala especificando el ID de aplicación y el ID de posición (proporcionados por RealMedia).
4. Añádela a tu Interfaz.

```
import com.prisadigital.realmedia.adlib.*;

public class RealMediaAdLibExampleActivity {
    private AdView      adView;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.main);

        // Create the adView
        adView = new AdView(this, AdView.BANNER_SMARTPHONE, MY_ID_APP,
MY_AD_POSITION);

        // Get main layout
        RelativeLayout rlMain = (RelativeLayout) findViewById(R.id.main_layout);

        // Add to UI
        rlMain.addView(adView);
    }
}
```

3.1.4 Añadiendo un AdView en forma de XML

Ejemplo de añadir un AdView desde un XML Layout. Nótese en negrita las líneas a añadir necesarias:

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:ads="http://schemas.android.com/apk/res/com.prisadigital.realmedia.adlib"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:id="@+id/main_layout">

    <com.prisadigital.realmedia.adlib.AdView
        android:id="@+id/adView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        ads:adType="banner_smartphone"
        ads:adKey="MY_ID_APP"
        ads:adPosition="MY_AD_POSITION"
    />

</RelativeLayout>
```

3.1.5 Interstitial

Para cargar un interstitial únicamente necesitas llamar a la Activity del SDK encargado de mostrarlo. Ten en cuenta que se llama en otra Activity diferente por lo que ten en mente en que parte de tu flujo de ejecución encaja mejor.

```
InterstitialAd iad = new InterstitialAd(this, MY_ID_APP, MY_ID_POSITION);
// Show
iad.loadAd();
```

3.1.6 Preroll

La implementación del preroll es diferente a lo anteriormente visto. Debido a la inmensa cantidad de diferentes implementaciones de video que puede utilizar el usuario se ha optado por implementar únicamente la comunicación con el AdServer dejando al usuario la implementación de la visualización del anuncio. El AdServer devolverá la URL del anuncio y la URL que tiene que lanzar si el usuario hace click en el video.

```
// Query for a PreRoll
PreRollVideo.queryForVideo(this, MY_APP_ID, MY_ID_POSITION,
    new PreRollVideoListener() {

        @Override
        public void onVideoRetreived(String videoUrl, String clickUrl) {
            // Show the video ad
        }

        @Override
        public void onVideoNotFound() {
            // No video ad, show your video
        }
    });
```