



APIs II

Curso: Desarrollando aplicaciones con Android

Contenidos

- ▶ **Sensores**

- ▶ Tipos
- ▶ Listar sensores
- ▶ Acceder a sensores

- ▶ **Google maps API**

- ▶ Crear clave
- ▶ Ejemplo
- ▶ Marcadores y eventos



Sensores en Android

- ▶ Android permite trabajar con los sensores internos del dispositivo usando las clases `Sensor`, `SensorEvent`, `SensorManager`, y la interfaz `SensorEventListener`, que se encuentran en el paquete `android.hardware`.
- ▶ La clase `Sensor` acepta ocho tipos de sensores.
- ▶ Aunque, los sensores disponibles varían en función del dispositivo utilizado.



Sensores soportados

Sensor	Android 4.0 (API Level 14)	Android 2.3 (API Level 9)	Android 2.2 (API Level 8)	Android 1.5 (API Level 3)
TYPE_ACCELEROMETER	Yes	Yes	Yes	Yes
TYPE_AMBIENT_TEMPERATURE	Yes	n/a	n/a	n/a
TYPE_GRAVITY	Yes	Yes	n/a	n/a
TYPE_GYROSCOPE	Yes	Yes	n/a ¹	n/a ¹
TYPE_LIGHT	Yes	Yes	Yes	Yes
TYPE_LINEAR_ACCELERATION	Yes	Yes	n/a	n/a
TYPE_MAGNETIC_FIELD	Yes	Yes	Yes	Yes
TYPE_ORIENTATION	Yes ²	Yes ²	Yes ²	Yes
TYPE_PRESSURE	Yes	Yes	n/a ¹	n/a ¹
TYPE_PROXIMITY	Yes	Yes	Yes	Yes
TYPE_RELATIVE_HUMIDITY	Yes	n/a	n/a	n/a
TYPE_ROTATION_VECTOR	Yes	Yes	n/a	n/a
TYPE_TEMPERATURE	Yes ²	Yes	Yes	Yes



5.7 Listar Sensores

```
public class SensoresActivity extends Activity {  
    private TextView salida;  
  
    @Override  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);  
  
        Sensalida = (TextView) findViewById(R.id.salida);  
        SensorManager sensorManager = (SensorManager)  
            getSystemService(SENSOR_SERVICE);  
        List<Sensor> listaSensores = sensorManager.  
            getSensorList(Sensor.TYPE_ALL);  
        for(Sensor sensor: listaSensores) {  
            salida.append(sensor.getName() + "\n");  
        }  
    }  
}
```



5.8a Escuchar eventos de sensor

- ▶ Tenemos que añadir un `SensorEventListener`
- ▶ (Si queremos probar este ejercicio tenemos que hacerlo en un teléfono o tablet, no podemos generar eventos desde el emulador)

```
public class SensoresActivity extends Activity implements SensorEventListener {  
    private TextView salida;  
  
    @Override  
    public void onCreate(Bundle savedInstanceState) {  
        ...  
        listaSensores = sensorManager.getSensorList(Sensor.TYPE_PROXIMITY);  
        if (!listaSensores.isEmpty()) {  
            Sensor proximitySensor = listaSensores.get(0);  
            sensorManager.registerListener(this, proximitySensor, SensorManager.SENSOR_DELAY_UI);}  
    }  
}
```



5.8b Escuchar eventos de sensor

► Añadir métodos a implementar

```
public class SensoresActivity extends Activity implements SensorEventListener {  
    ...  
    @Override  
    public void onSensorChanged(SensorEvent event) {  
        synchronized (this) {  
            switch(event.sensor.getType()) {  
                case Sensor.TYPE_PROXIMITY:  
                    for (int i=0 ; i<event.values.length ; i++) {  
                        salida.append("Proximidad "+i+": "+event.values[i] +"\n");  
                    }  
                    break;  
            }  
        }  
    }  
    @Override  
    public void onAccuracyChanged(Sensor sensor, int accuracy) {  
        // TODO Auto-generated method stub  
    }  
}
```



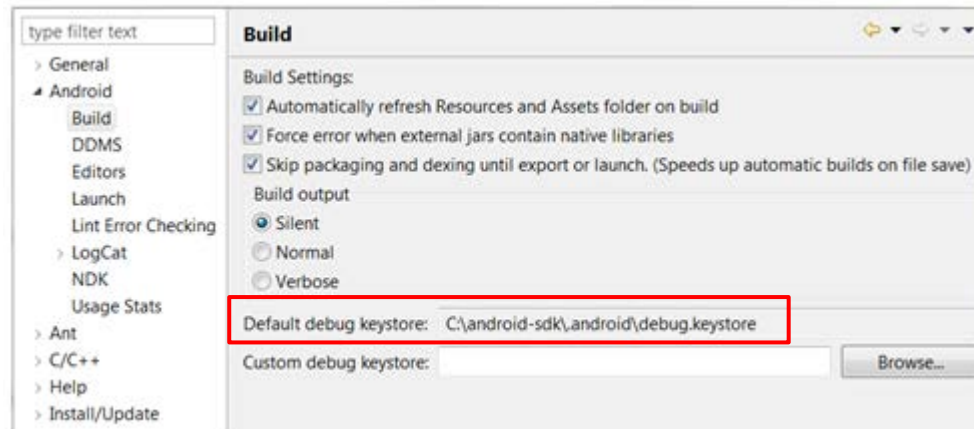


Google maps API



Paso 1: Buscar debug keystore

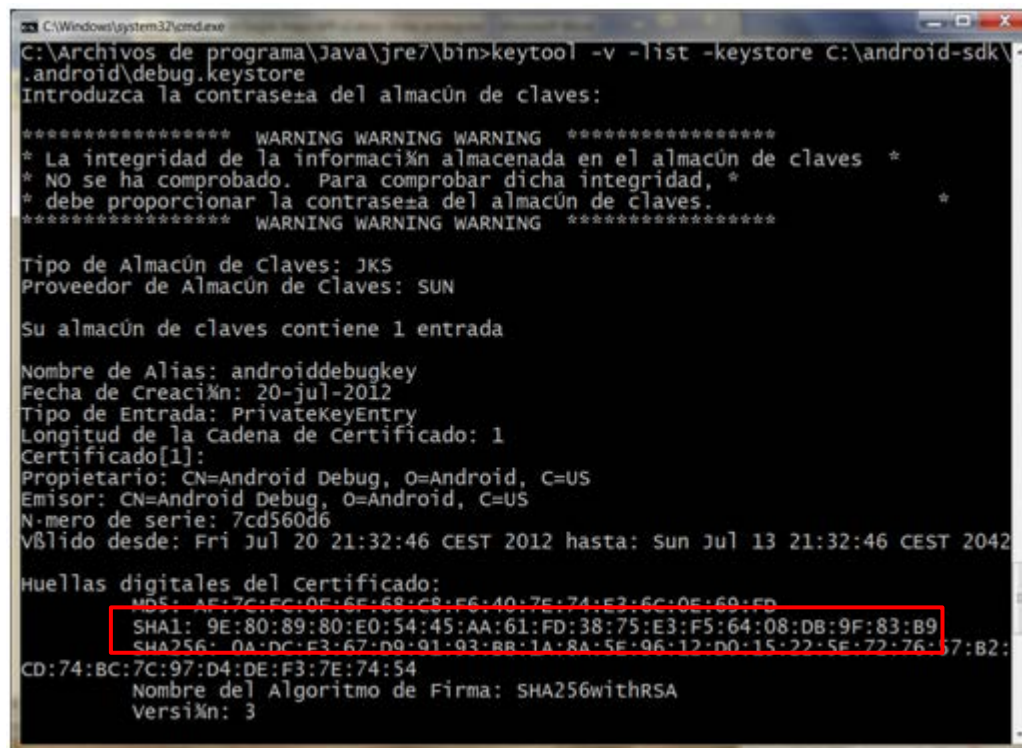
- ▶ Buscar el certificado digital de depuración. Utilizando el entorno Eclipse accede al menú *Windows > Preferences > Android > Build*. Aparecerá el siguiente cuadro de diálogo:



Paso 2: Certificado digital SHA1

- ▶ En línea de comandos:

`keytool -v -list -keystore [Tu debug.keystore path]`



```
C:\Windows\system32\cmd.exe
C:\Archivos de programa\Java\jre7\bin>keytool -v -list -keystore C:\android-sdk\
.android\debug.keystore
Introduzca la contraseña del almacén de claves:

***** WARNING WARNING WARNING *****
* La integridad de la información almacenada en el almacén de claves *
* NO se ha comprobado. Para comprobar dicha integridad, *
* debe proporcionar la contraseña del almacén de claves. *
***** WARNING WARNING WARNING *****

Tipo de Almacén de Claves: JKS
Proveedor de Almacén de Claves: SUN

Su almacén de claves contiene 1 entrada

Nombre de Alias: androiddebugkey
Fecha de Creación: 20-Jul-2012
Tipo de Entrada: PrivateKeyEntry
Longitud de la Cadena de Certificado: 1
Certificado[1]:
Propietario: CN=Android Debug, O=Android, C=US
Emisor: CN=Android Debug, O=Android, C=US
Número de serie: 7cd560d6
Válido desde: Fri Jul 20 21:32:46 CEST 2012 hasta: Sun Jul 13 21:32:46 CEST 2042

Huellas digitales del Certificado:
MD5: AF:7C:FC:0F:6F:68:C8:F6:40:7E:74:E3:6C:0F:69:FD
SHA1: 9E:80:89:80:E0:54:45:AA:61:FD:38:75:E3:F5:64:08:DB:9F:83:B9
SHA256: 0A:DC:E3:67:D9:91:93:BB:1A:8A:5E:96:12:D0:15:22:5E:72:76:57:B2:
CD:74:BC:7C:97:D4:DE:F3:7E:74:54
Nombre del Algoritmo de Firma: SHA256withRSA
Versión: 3
```

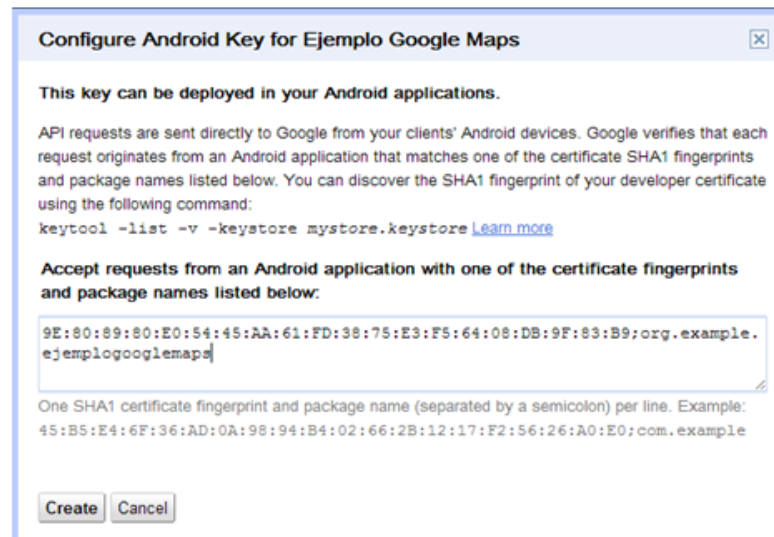
- ▶ Copia esta secuencia de dígitos. En Windows pulsa con el botón derecho sobre la barra superior de la ventana y selecciona *Marcar*.

Paso 3: Conseguir API key

- ▶ Ir a: <https://code.google.com/apis/console/>
- ▶ Crea un proyecto: Ejemplo google maps
- ▶ Activar la API de google maps Android

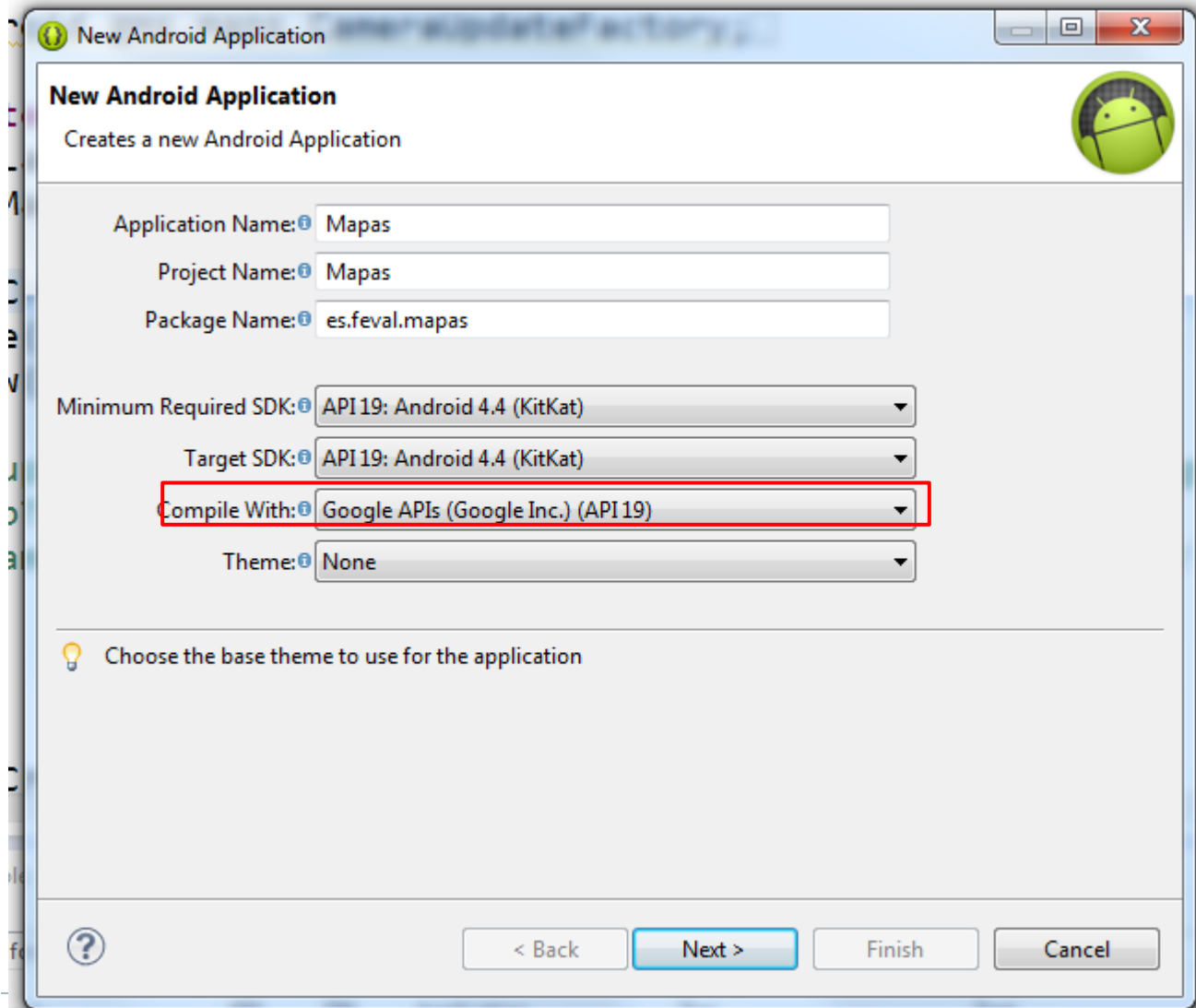


- ▶ Create new key -> Android key
- ▶ Pegar la clave SHA1 seguido del nombre del paquete: Ej: ;es.feval.mapas



- ▶ Copiar API key

Paso 4: Crear proyecto



New Android Application
Creates a new Android Application

Application Name: Mapas

Project Name: Mapas

Package Name: es.feval.mapas

Minimum Required SDK: API 19: Android 4.4 (KitKat)

Target SDK: API 19: Android 4.4 (KitKat)

Compile With: Google APIs (Google Inc.) (API 19)

Theme: None

Choose the base theme to use for the application

< Back Next > Finish Cancel

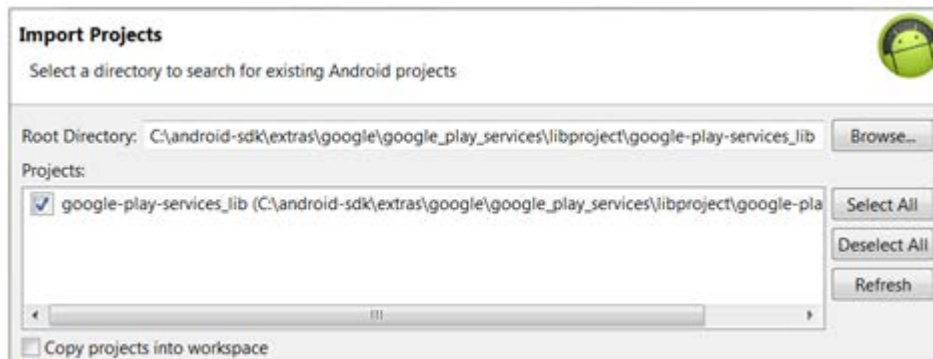
Paso 5: Instalar play services

► Abrir Android SDK manager

▲	Extras		
<input type="checkbox"/>	Android Support Repository	5	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Android Support Library	19.1	<input checked="" type="checkbox"/> Installed
<input type="checkbox"/>	Google Analytics App Tracking SDK	3	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Google Play services for Froyo	12	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Google Play services	16	<input checked="" type="checkbox"/> Installed
<input type="checkbox"/>	Google Repository	7	<input type="checkbox"/> Not installed
<input type="checkbox"/>	Google Play APK Expansion Library	3	<input type="checkbox"/> Not installed

Paso 6: Importar google play services

- ▶ En Eclipse, entra a *File > Import > Android > Existing Android Code Into Workspace*.
- ▶ Dentro de la carpeta *android-sdk/extras/google/*
- ▶ *google_play_services/libproject/google-play-services_lib*

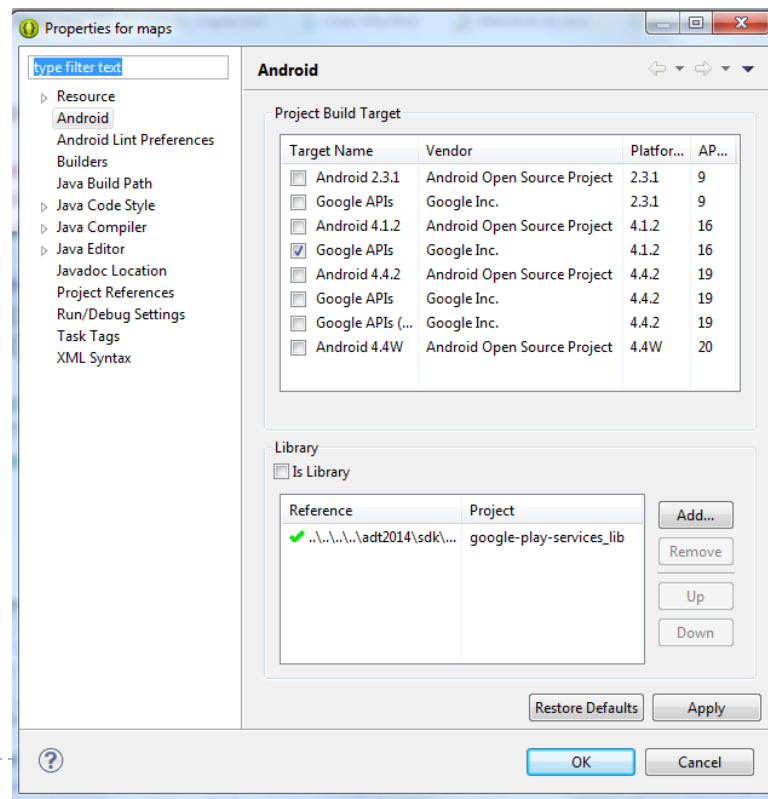


- ▶ *Aparecerá como un proyecto más*
-



Paso 7: Añadir la librería a nuestro proyecto

- ▶ Botón derecho y selecciona *Properties* > *Android*.
- ▶ Add library
 - ▶ *google-play-services_lib*



Paso 8: AndroidManifest.xml

- ▶ Dentro del elemento `<application>`, añade las siguientes líneas:

```
<meta-data
    android:name="com.google.android.gms.version"
    android:value="@integer/google_play_services_version"/>
<meta-data
    android:name="com.google.android.maps.v2.API_KEY"
    android:value="AlzaSyCXV7RcLaKvNaOpJuViKIJ9kbKHoucJ7Zg"/>
```

- ▶ *Cambiando la API, por la obtenida en el paso 3.*



Paso 9: AndroidManifest.xml II

► Añadir permisos:

```
<permission
    android:name="org.example.ejemplogooglemaps.permission.MAPS_RECEIVE"
    android:protectionLevel="signature"/>
<uses-permission
    android:name="org.example.ejemplogooglemaps.permission.MAPS_RECEIVE"/>
<uses-permission
    android:name="com.google.android.providers.gsf.permission.READ_GSERVICES"/>

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

► Poner el nombre del paquete: es.feval.mapas en las líneas rojas



Paso 10: AndroidManifest.xml III

► Añadir OpenGL:

```
<uses-feature  
    android:glEsVersion="0x00020000"  
    android:required="true" />
```



Paso 11:Activity_main.xml

► Sustituir por el siguiente:

```
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <fragment
        android:id="@+id/map"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        class="com.google.android.gms.maps.SupportMapFragment"/>
</RelativeLayout>
```

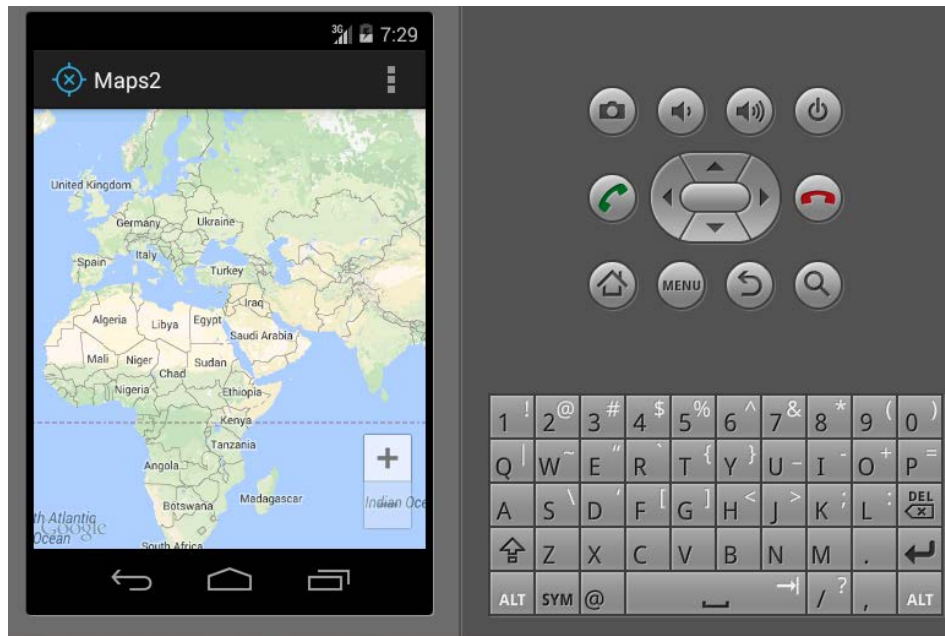


Paso 12: MainActivity.java

► Extender de la clase `FragmentActivity`:

```
public class MainActivity extends FragmentActivity {
```

► Probar en un nuevo emulador con google Api 19



Ejercicio 5.9

- Centrar el mapa en un punto.

```
public class MainActivity extends FragmentActivity {  
    private final LatLng Uned = new LatLng(39.2138386,-6.096301);  
    private GoogleMap mapa;  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        mapa =  
        ((SupportMapFragment)getSupportFragmentManager().findFragmentById(R.id.map)).getMap();  
        mapa.setMapType(GoogleMap.MAP_TYPE_SATELLITE);  
        mapa.moveCamera(CameraUpdateFactory.newLatLngZoom(Uned, 15));  
    }  
}
```



Ejercicio 5.10

► Añadir marcador.

....

```
mapa.setMyLocationEnabled(true);  
mapa.getUiSettings().setZoomControlsEnabled(false);  
mapa.getUiSettings().setCompassEnabled(true);  
mapa.addMarker(new MarkerOptions()  
    .position(Uned)  
    .title("UNED Mérida")  
    .snippet("Descripcion de Uned")  
    .icon(BitmapDescriptorFactory  
        .fromResource(R.drawable.ic_launcher))  
    .anchor(0.5f, 0.5f));
```

....



Ejercicio 5.10a

► Eventos sobre el mapa.

```
public class MainActivity extends FragmentActivity implements  
OnMapClickListener  
{ ....
```

```
    mapa.setOnMapClickListener((OnMapClickListener) this);  
  
}
```

```
    public void onMapClick(LatLng puntoPulsado) {  
        mapa.addMarker(new MarkerOptions().position(puntoPulsado).  
            icon(BitmapDescriptorFactory  
                .defaultMarker(BitmapDescriptorFactory.HUE_YELLOW)));  
    }  
    ....
```

