MOBILE PROGRAMMING

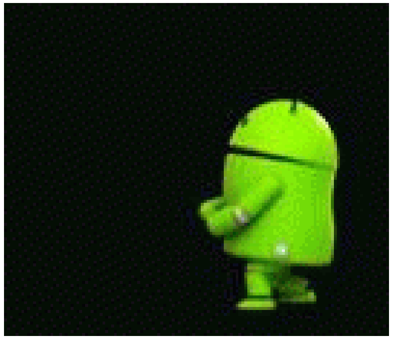
Chapter 5

ANDROID PROGRAMMING (ADVANCED)

12/2/2014

Contents

- ✓ Details of activities, dialogs
- ✓ Adapter for list/grid view
- ✓ SharedReferences
- ✓ SQLite
- ✓ Network/Internet (Http)
- Maps,
- Sms & Telephony
- Multimedia,
- Notification,...

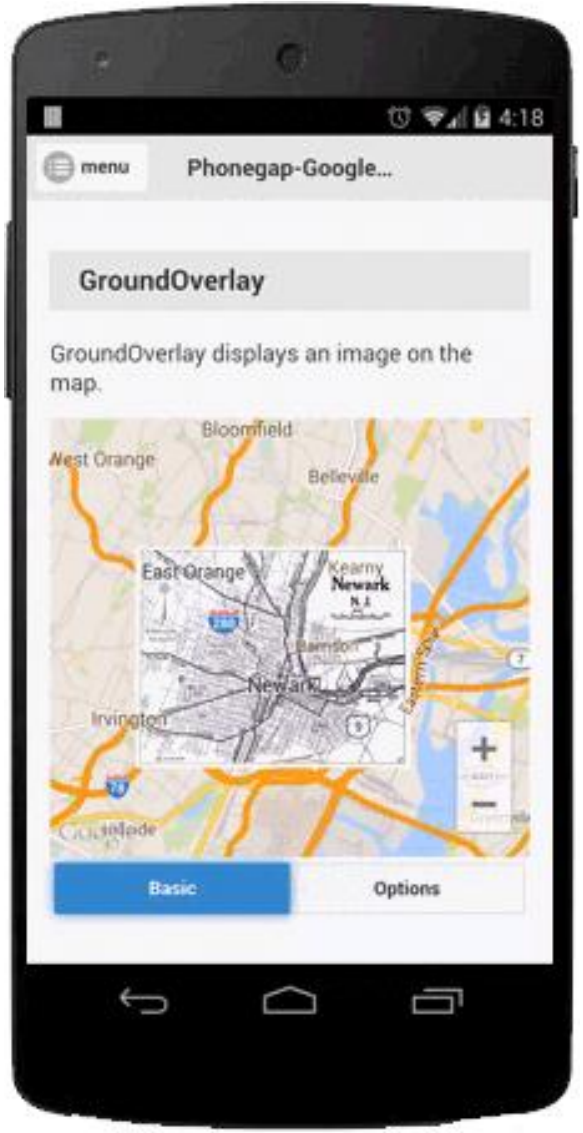


Intent
Activity
Service
Broadcast Receiver
Content Provider

5.6- MAPS

- **Map introduction**
  - Google play services

  - Android apps can use the Google play services



Phonegap-Google...

GroundOverlay

GroundOverlay displays an image on the map.

Basic | Options

5.6- MAPS

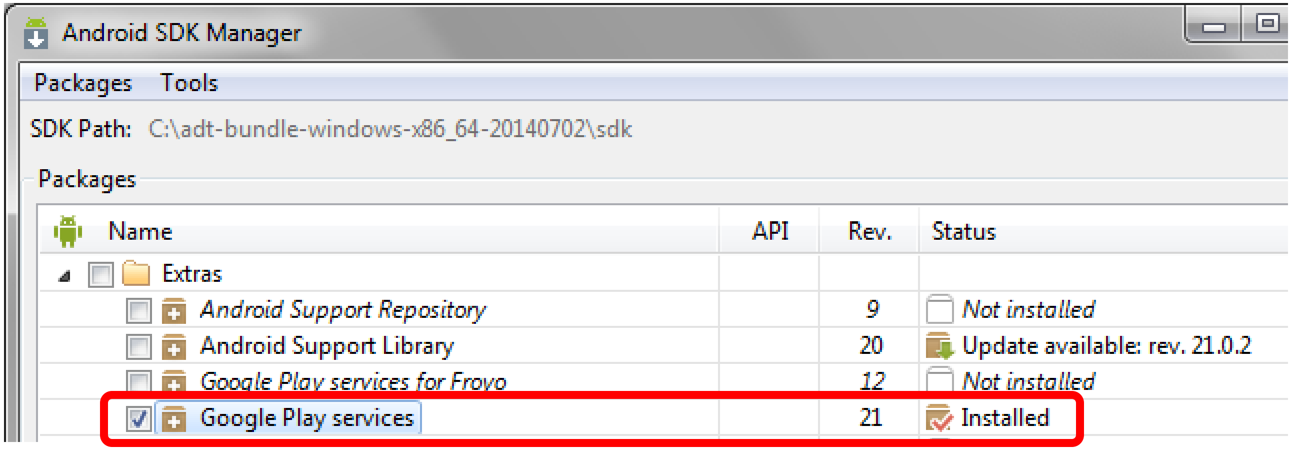
- **Install Google play services**

  http://developer.android.com/google/play-services/setup.html

  - Step1) Install “Google Play services” to SDK
  - Step2) Copy the service library to your Android project folder
  - Step3) Import the library project into Eclipse workspace
  - Step4) Reference your project to the library
  - Step5) Get API key
  - Step6) Add a declaration to app’s manifest

5.6- MAPS

– Step1) Install “Google Play services” to SDK



Android SDK Manager

Packages Tools

SDK Path: C:\adt-bundle-windows-x86_64-20140702\sdk

Packages

Name	API	Rev.	Status
□ Extras			
□ *Android Support Repository*		9	*Not installed*
□ Android Support Library		20	Update available: rev. 21.0.2
□ *Google Play services for Froyo*		*12*	*Not installed*
☑ Google Play services		21	Installed

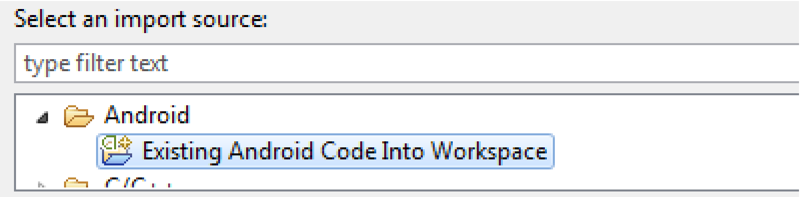
– Step2) Copy the service library to your Android project folder

<android-sdk>/extras/google/google_play_services/libproject/google-play-services_lib/

5.6- MAPS

– Step3) Import the library project into Eclipse workspace

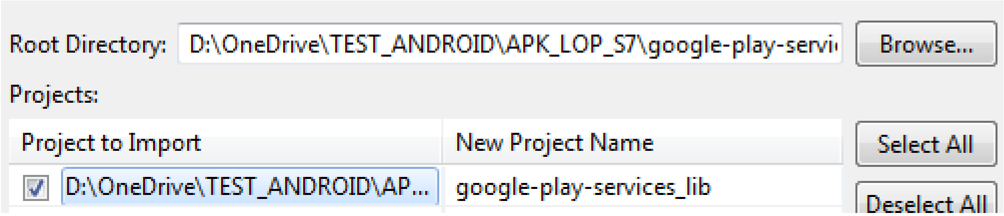
- Right-click on the Project → Import → Next



Select an import source:

type filter text

- Select the copied library → Finish

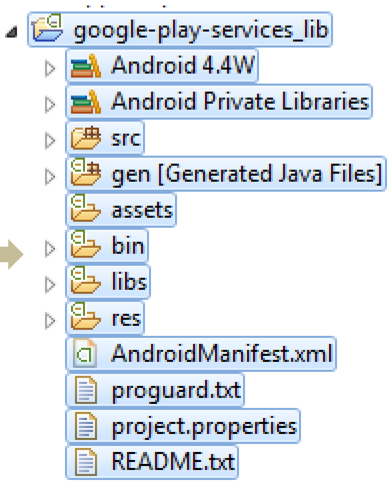


Root Directory: D:\OneDrive\TEST_ANDROID\APK_LOP_S7\google-play-servi

Projects:

Project to Import	New Project Name
☑ D:\OneDrive\TEST_ANDROID\AP...	google-play-services_lib

- In Eclipse workspace, we will see as →

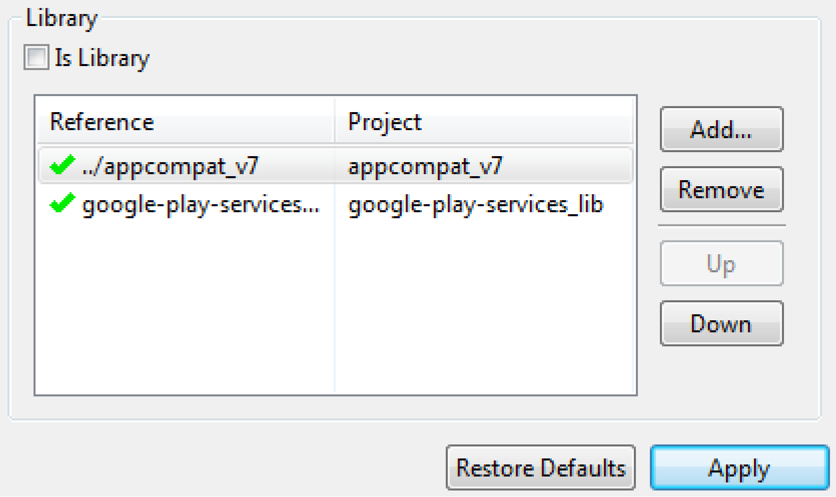


google-play-services_lib
Android 4.4W
Android Private Libraries
src
gen [Generated Java Files]
assets
bin
libs
res
AndroidManifest.xml
proguard.txt
project.properties
README.txt

5.6- MAPS

– Step4) Reference your project to the library

- Right-click on Project → Properties → Android



Library

□ Is Library

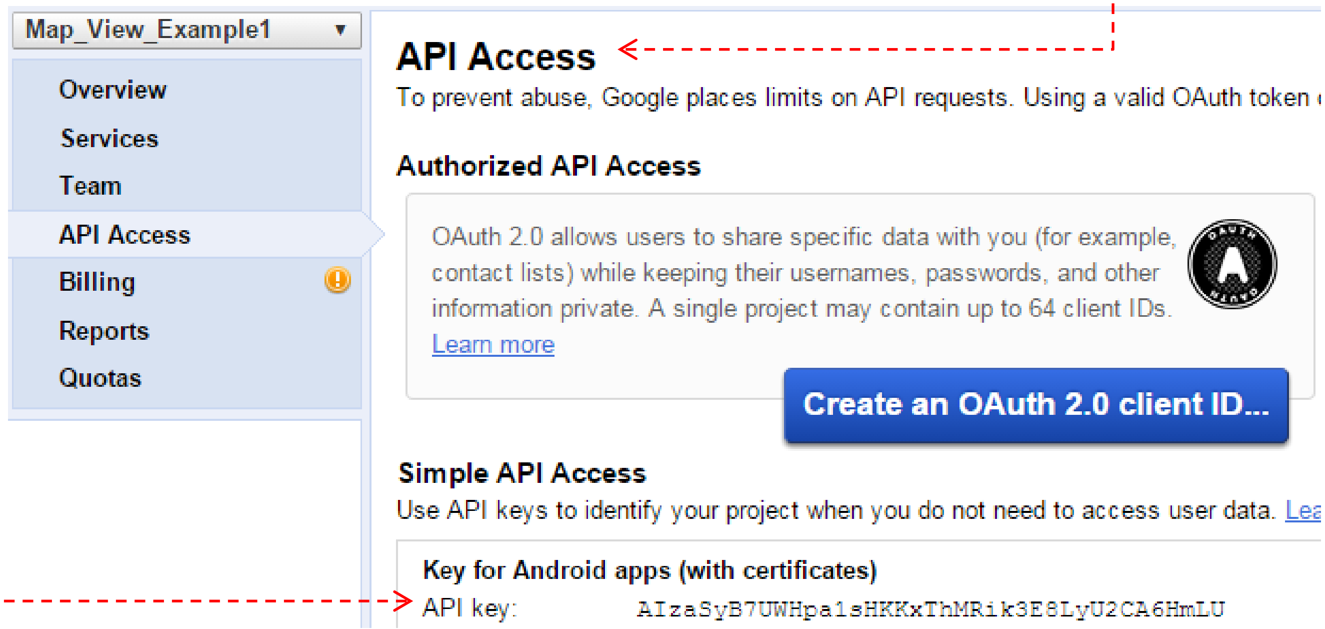
Reference	Project
✓ ../appcompat_v7	appcompat_v7
✓ google-play-services...	google-play-services_lib

- Add → choose the copied library → Apply

5.6- MAPS

– Step5) Get API key

1. Navigate to your project in the Google APIs Console.
2. In the **Services** page, verify that the "Google Maps Android API v2" is enabled.
3. In the left navigation bar, click **API Access**.
4. In the resulting page, click **Create New Android Key...**.
5. In the resulting dialog, enter the SHA-1 fingerprint, then a semicolon, then your application's package name.



API Access

To prevent abuse, Google places limits on API requests. Using a valid OAuth token

Authorized API Access

OAuth 2.0 allows users to share specific data with you (for example, contact lists) while keeping their usernames, passwords, and other information private. A single project may contain up to 64 client IDs. Learn more

Simple API Access

Use API keys to identify your project when you do not need to access user data. Lea

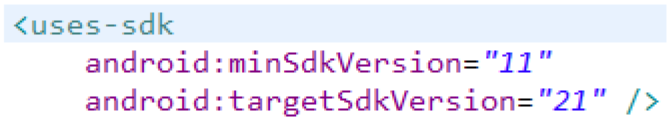
Key for Android apps (with certificates)

API key: AIzaSyB7UWHpa1sHKKxThMRik3E8LyU2CA6HmLU

5.6- MAPS

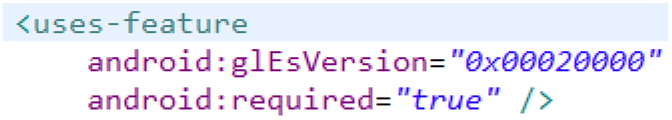
– Step6) Add declarations to app’s manifest

- Min SDK version is 11



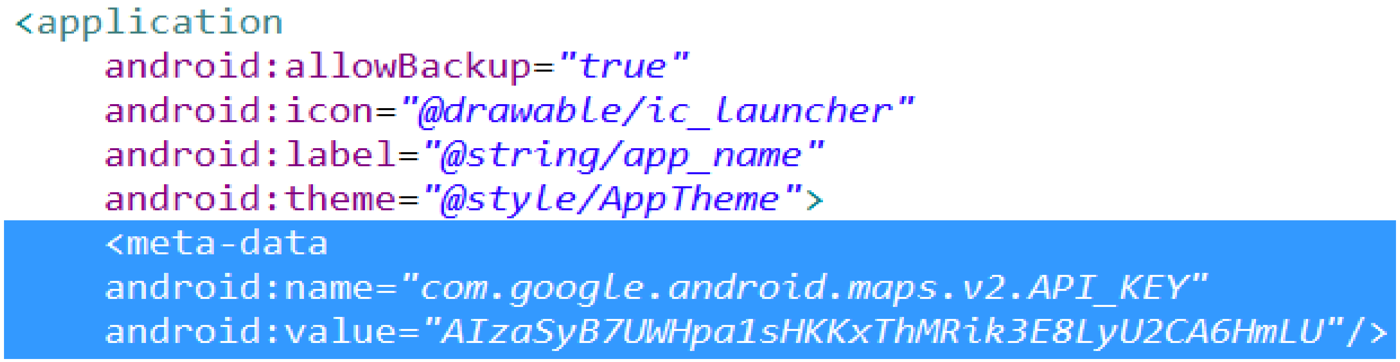
```
<uses-sdk
    android:minSdkVersion="11"
    android:targetSdkVersion="21" />
```

- Required OpenGL ES 2.0. for Maps V2



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

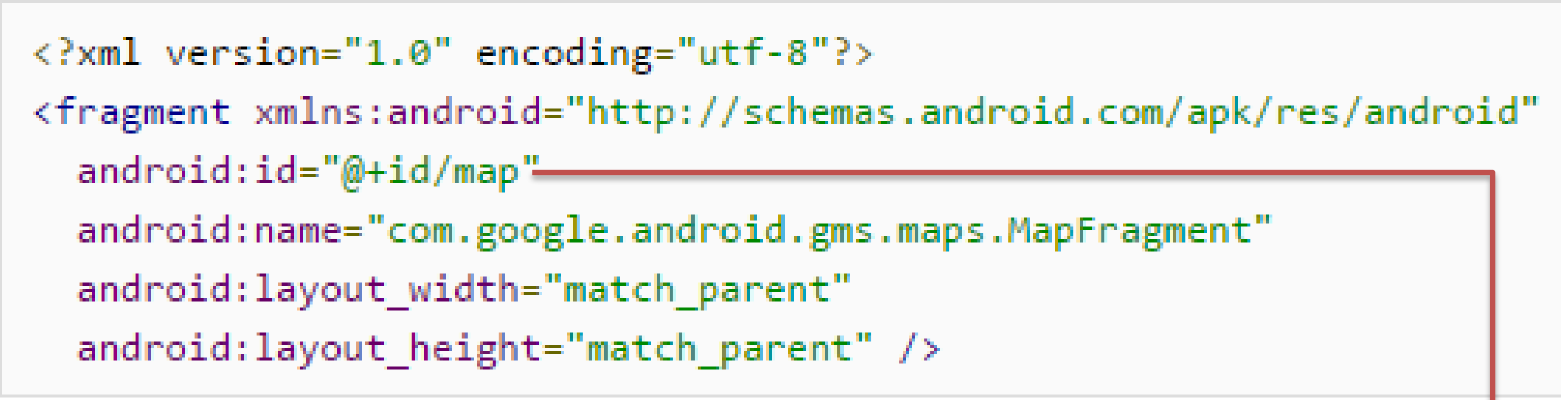
- Add Google API_KEY



```
<application
    android:allowBackup="true"
    android:icon="@drawable/ic_launcher"
    android:label="@string/app_name"
    android:theme="@style/AppTheme">
    <meta-data
    android:name="com.google.android.maps.v2.API_KEY"
    android:value="AIzaSyB7UWHpa1sHKKxThMRik3E8LyU2CA6HmLU"/>
```

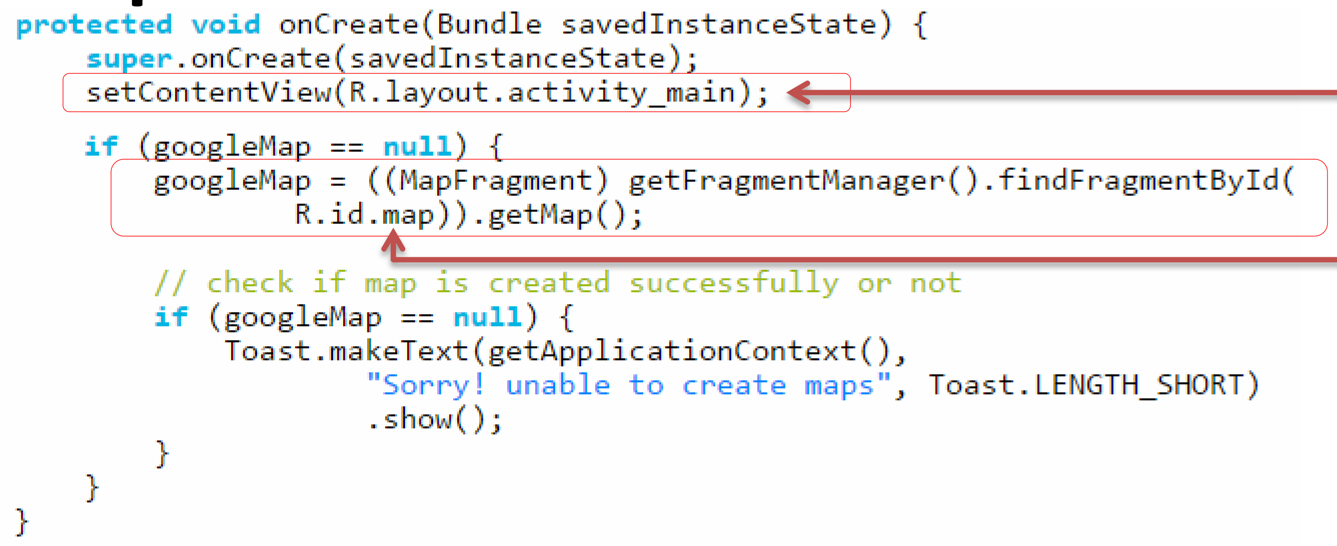
5.6- MAPS

- Create layout of MapView



```xml
<?xml version="1.0" encoding="utf-8"?>
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
  android:id="@+id/map"
  android:name="com.google.android.gms.maps.MapFragment"
  android:layout_width="match_parent"
  android:layout_height="match_parent" />
```

- View map



```java
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    if (googleMap == null) {
        googleMap = ((MapFragment) getFragmentManager().findFragmentById(
                R.id.map)).getMap();

        // check if map is created successfully or not
        if (googleMap == null) {
            Toast.makeText(getApplicationContext(),
                    "Sorry! unable to create maps", Toast.LENGTH_SHORT)
                    .show();
        }
    }
}
```

5.6- MAPS

- **Some other actions**
  - Add marker

```
// latitude and longitude
double latitude = ?;
double longitude = ?;
 // create marker
MarkerOptions  ♣ = new MarkerOptions().position(
                  new LatLng(latitude, longitude)).title("Hello Maps ");
// adding marker
googleMap.addMarker(♣);
```

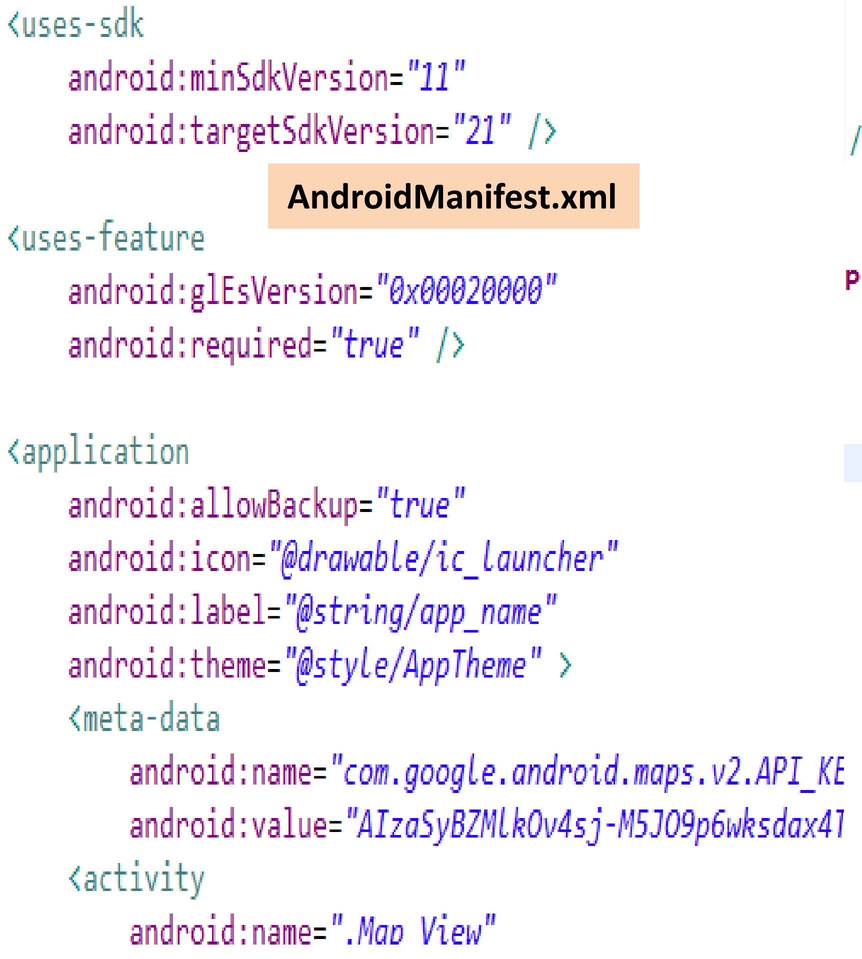
.icon(?) / …

  - Moving camera

```
CameraPosition  ♥ = new CameraPosition.Builder().target(
             new LatLng( lat , lng )).zoom(12).build();
googleMap.animateCamera(CameraUpdateFactory.newCameraPosition(♥));
```

5.6- MAPS

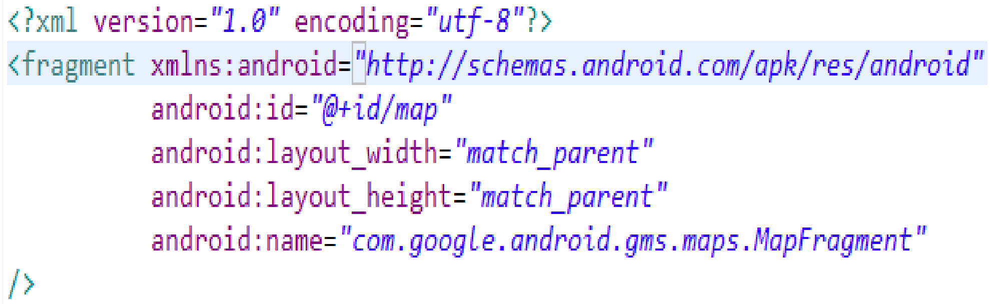
- **Example**



AndroidManifest.xml

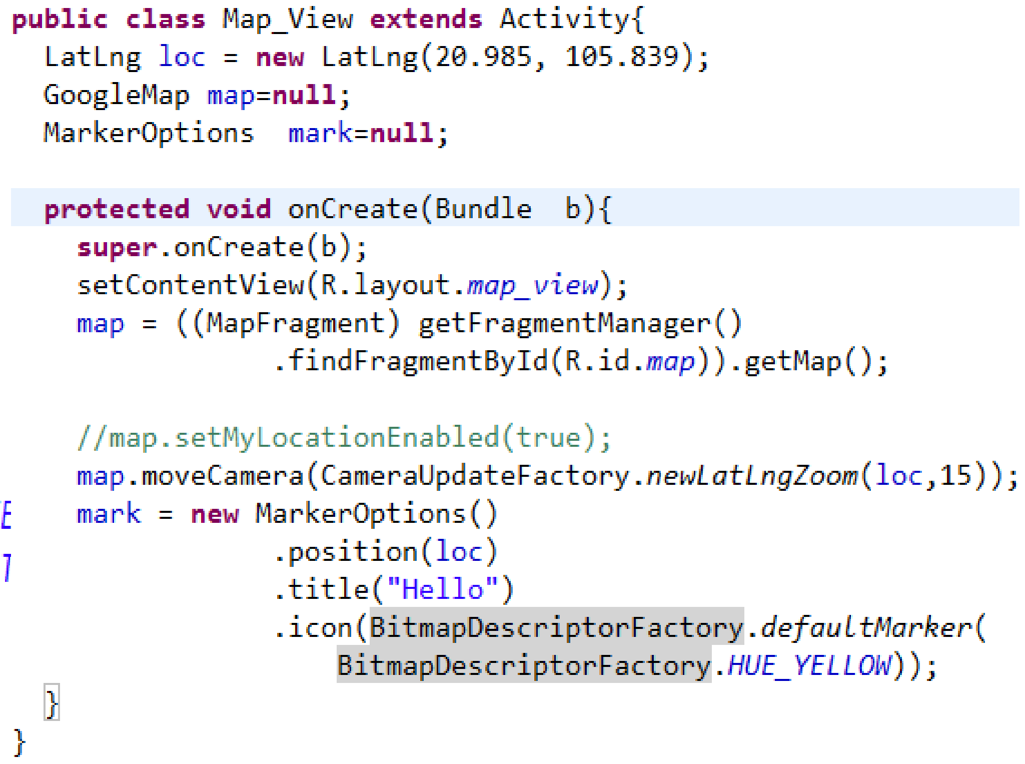
```
<uses-sdk
    android:minSdkVersion="11"
    android:targetSdkVersion="21" />

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

<application
    android:allowBackup="true"
    android:icon="@drawable/ic_launcher"
    android:label="@string/app_name"
    android:theme="@style/AppTheme" >
    <meta-data
        android:name="com.google.android.maps.v2.API_KE
        android:value="AIzaSyBZMlkOv4sj-M5JO9p6wksdax4l
    <activity
        android:name=".Map View"
```

map_view.xml



```
<?xml version="1.0" encoding="utf-8"?>
<fragment xmlns:android="http://schemas.android.com/apk/res/android"
        android:id="@+id/map"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:name="com.google.android.gms.maps.MapFragment"
/>
```

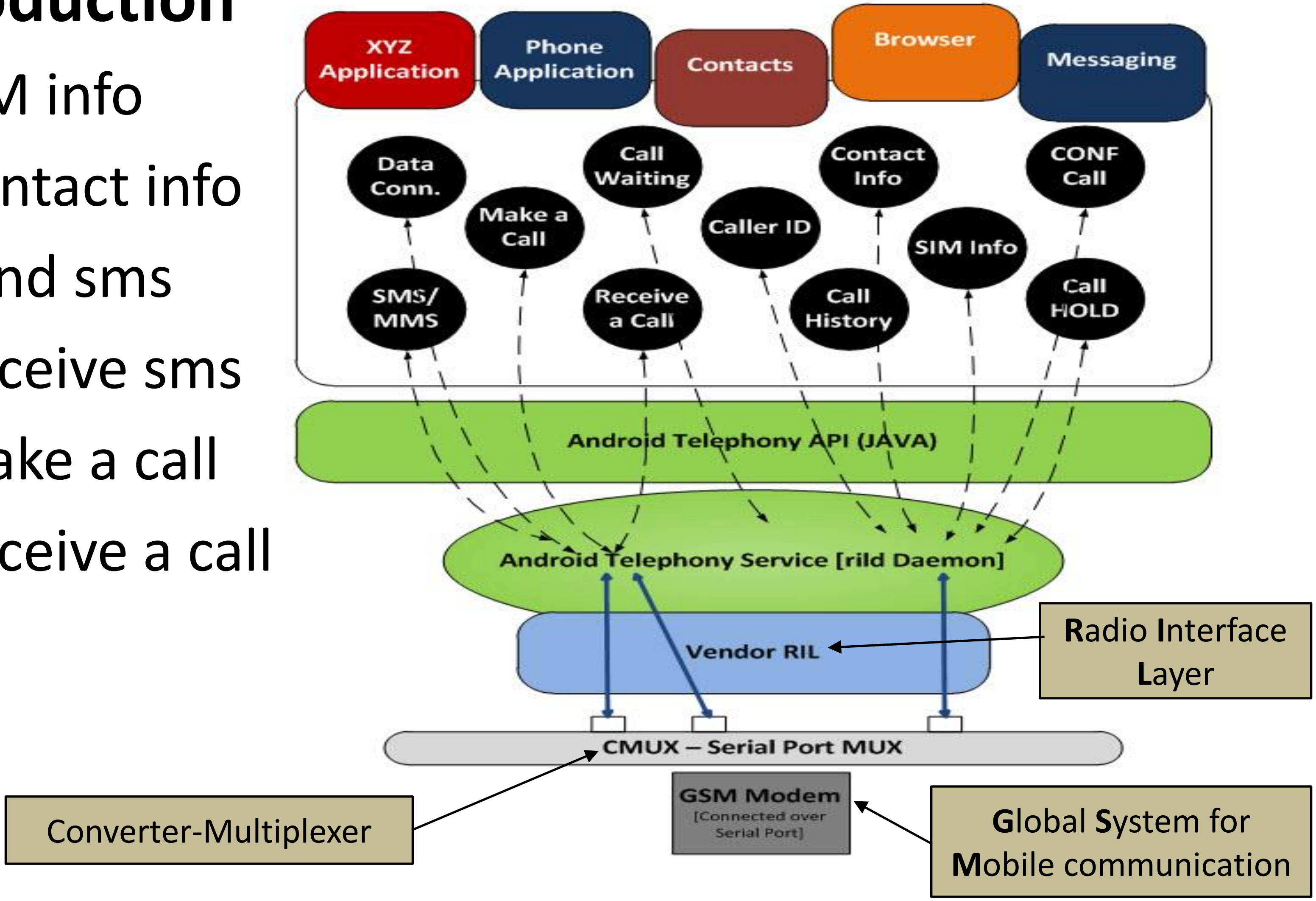
Map_View.java



```
public class Map_View extends Activity{
  LatLng loc = new LatLng(20.985, 105.839);
  GoogleMap map=null;
  MarkerOptions  mark=null;

  protected void onCreate(Bundle  b){
    super.onCreate(b);
    setContentView(R.layout.map_view);
    map = ((MapFragment) getFragmentManager()
                .findFragmentById(R.id.map)).getMap();

    //map.setMyLocationEnabled(true);
    map.moveCamera(CameraUpdateFactory.newLatLngZoom(loc,15));
    mark = new MarkerOptions()
                .position(loc)
                .title("Hello")
                .icon(BitmapDescriptorFactory.defaultMarker(
                    BitmapDescriptorFactory.HUE_YELLOW));
  }
}
```

5.7- SMS & Telephony

- **Introduction**
  - – SIM info
  - – Contact info
  - – Send sms
  - – Receive sms
  - – Make a call
  - – Receive a call
  - – …



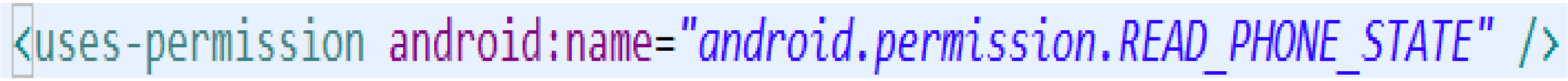
Radio Interface Layer

Converter-Multiplexer

Global System for Mobile communication

5.7- SMS & Telephony

- **Get SIM info**
  - Permission (AndroidManifest.xml)



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

  - TelephonyManager object

```
∆ = (TelephonyManager)getSystemService(TELEPHONY_SERVICE);
```

  - Some methods:

Stt	Methods	Usage
1	getDeviceId()	IMEI
2	getLine1Number()	Phone number
3	getSimCountryIso()	Country
4	getSimOperatorName()	
5	getSimSerialNumber()	…

5.7- SMS & Telephony

- **Make phone call**
  - Permission

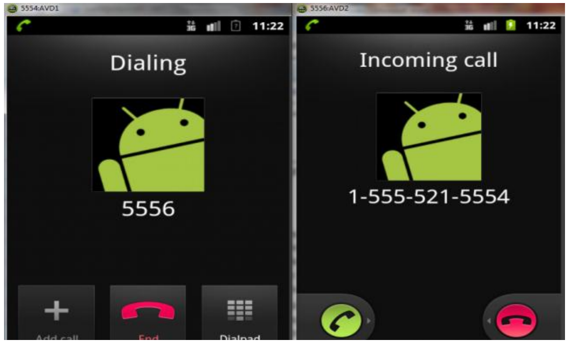
    <uses-permission   android:name="**android.permission.CALL_PHONE**"/>

  - Use “Intent” to make phone call
    - Step1) Create “**Uri**” object that contain phone number

      ◊ = **Uri.parse**("***tel***:"+“*phone_number*”);

    - Step2) Start an activity for new Intent

      startActivity(new **Intent**(**Intent.ACTION_CALL**, ◊));



5.6- SMS & Telephony

- **Listen incomming calls**

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

  – Create "PhoneStateListener" object (♥):

```
public  void  onCallStateChanged(int  state, String  incomingNumber) {
      switch (state) {
          case  TelephonyManager.CALL_STATE_RINGING:
                // … DO SOMETHING HERE … //
                break;
          case…
      }
 }
```

states of call changed:
CALL_STATE_RINGING,
CALL_STATE_OFFHOOK,
CALL_STATE_IDLE

  – Register the listener to "TelephonyManager"

```
Δ.listen( ♥ , PhoneStateListener.LISTEN_CALL_STATE );
```

LISTEN_NONE
for removing

5.6- SMS & Telephony

- **Listen outgoing calls**

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

  - Create “BroadcastReceiver” object listener (♥):

```
public  void  onReceive( Context context, Intent intent ) {
    String number = intent.getStringExtra(Intent.EXTRA_PHONE_NUMBER);
    //… DO SOMETHING HERE …//
}
```

  - Register the listener to “Context” object (Δ)

```
♦ = new  IntentFilter(Intent.ACTION_NEW_OUTGOING_CALL);
Δ.registerReceiver( ♥ , ♦ );
```

This only runs with the current Activity, and it’s disabled when lose the Activity → create **Service**!

5.6- SMS & Telephony

- **SMS sending**

```
<uses-permission   android:name="android.permission.SEND_SMS"/>
<uses-permission   android:name="android.permission.WRITE_SMS"/>
<uses-permission   android:name="android.permission.READ_SMS"/>
```

  – Get “SmsManager” object (Δ):

```
Δ = SmsManager.getDefault();
```

  – Send a message to a phone number

```
Δ.sendTextMessage(SỐ_ĐT_NHẬN, null, NỘI_DUNG_TIN, null, null );
```

5.7- SMS & Telephony

- **SMS receving**

```
<uses-permission   android:name="android.permission.RECEIVE_SMS"/>
<uses-permission   android:name="android.permission.WRITE_SMS"/>
<uses-permission   android:name="android.permission.READ_SMS"/>
```

  – Create “BroadcastReceiver” object listener (♥):

```
public   void   onReceive( Context context, Intent intent ) {
     //… DO SOMETHING HERE …//
}
```

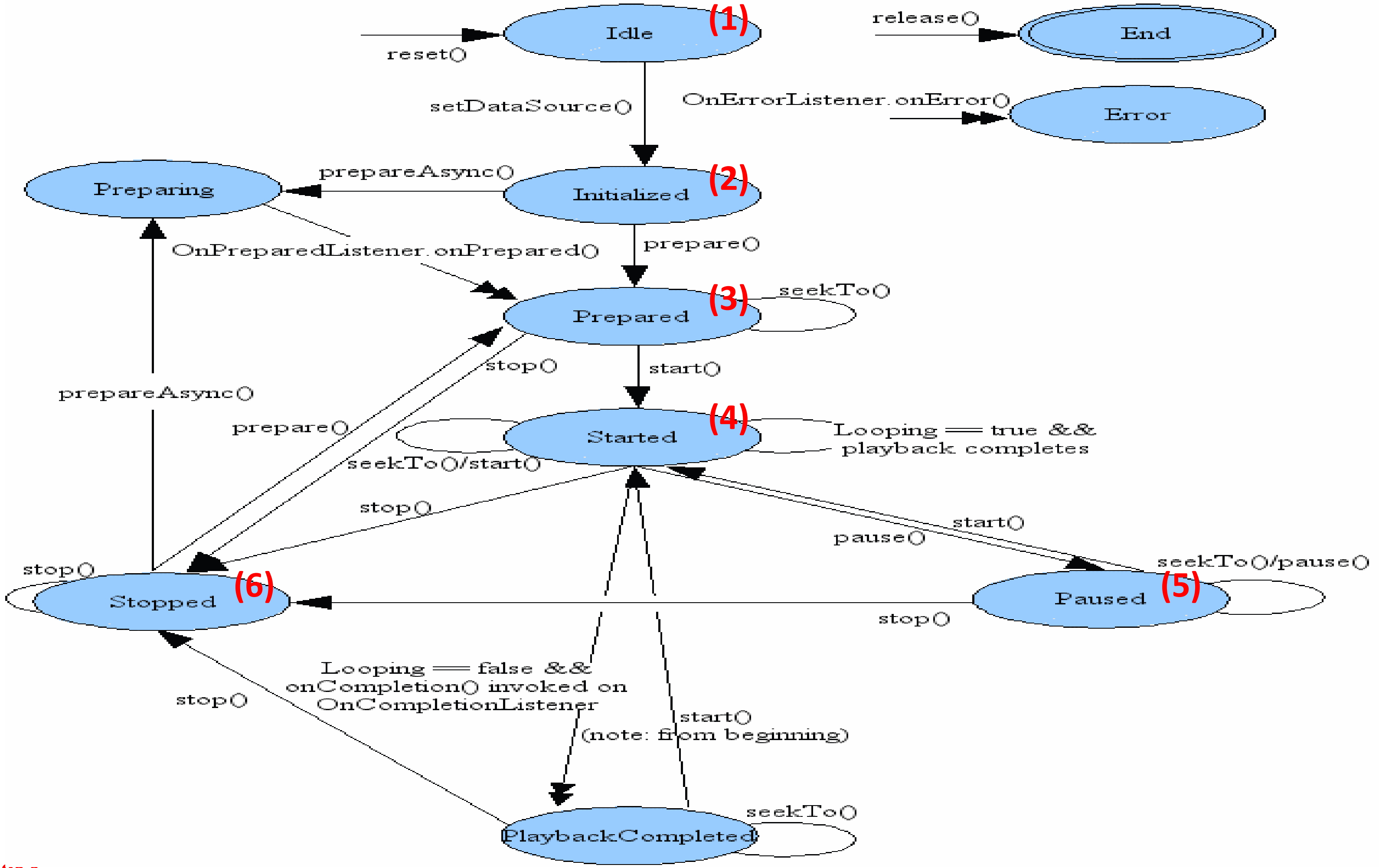
  – Register the listener to “Context” object (Δ)

```
♦ = new   IntentFilter(“android.provider.Telephony.SMS_RECEIVED”);
Δ.registerReceiver( ♥ , ♦ );
```

This only runs with the current Activity, and it’s disabled when lose the Activity → create **Service**!

5.8- Multimedia

- **Play audio**



5.8- Multimedia

- **Play audio**
  - No need view for audio
  - Step1) Create "MediaPlayer" object

```
∆ = new  MediaPlayer();
```

```
MediaPlayer.create( Context , R.raw.---);
```

  - Step2) Set data source (if not *.create(...) in Step1)

```
∆.setDataSource( source_path );
```

```
"http:// …xyz… "
```

  - Step3) Prepare & start to play

```
∆.prepare();
∆.start();
```

```
∆.seekTo(  miliSec );
∆.stop();
∆.pause();
∆.setVolume( leftVol,  rightVol );
```

5.8- Multimedia

- **Play video**
  - Need "VideoView" for video (in layout)
  - Step1) Get "VideoView" object (∆) from layout

```
∆ = (VideoView)findViewById( R.id.--- );
```

  - Step2) Set URI source of video

"http:// …xyz… "

```
∆.setVideoURI( Uri.parse( "android.resource://" +
                         getPackageName()+"/"+R.raw.---)));
```

  - Step3) Prepare & start to play

```
∆.requestFocus();
∆.start();
```

```
∆.seekTo(  miliSec );
∆.isPlaying();
∆.getDuration();
∆.pause();
```