

## **LAMPIRAN**

### **1. SPESIFIKASI DEVICE APLIKASI**

Os android ICS (Ice Cream Sandwich 4.0) dan di atasnya

Ram 352 atau lebih

Resolution 364 x 768

Processor 1Ghz atau lebih

Mempunyai akses GPS dan internet

### **2. KODING**

#### **JSONParser.java**

```
package net.muhamadiqbal.petalokasi;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.UnsupportedEncodingException;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.impl.client.DefaultHttpClient;
import org.json.JSONException;
import org.json.JSONObject;

import android.util.Log;

public class JSONParser {

    static InputStream is = null;
    static JSONObject jObj = null;
    static String json = "";

    // constructor
    public JSONParser() {
```

```
}

public JSONObject getJSONObjectFromUrl(String url) {

    // Making HTTP request
    try {
        // defaultHttpClient
        DefaultHttpClient httpClient = new DefaultHttpClient();
        HttpPost httpPost = new HttpPost(url);

        HttpResponse httpResponse = httpClient.execute(httpPost);
        HttpEntity httpEntity = httpResponse.getEntity();
        is = httpEntity.getContent();

    } catch (UnsupportedEncodingException e) {
        e.printStackTrace();
    } catch (ClientProtocolException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }

    try {
        BufferedReader reader = new BufferedReader(new InputStreamReader(
            is, "iso-8859-1"), 8);
        StringBuilder sb = new StringBuilder();
        String line = null;
        while ((line = reader.readLine()) != null) {
            sb.append(line + "\n");
        }
        is.close();
        json = sb.toString();
    } catch (Exception e) {
        Log.e("Buffer Error", "Error converting result " + e.toString());
    }

    // try parse the string to a JSON object
    try {
        jObj = new JSONObject(json);
    } catch (JSONException e) {
        Log.e("JSON Parser", "Error parsing data " + e.toString());
    }
}
```

```
// return JSON String  
return jObj;
```

```
}
```

### MainActivityMenuUtama.java

```
package net.muhamadiqbal.petalokasi;  
  
import java.util.ArrayList;  
import java.util.HashMap;  
  
import org.json.JSONArray;  
import org.json.JSONException;  
import org.json.JSONObject;  
  
import android.os.AsyncTask;  
import android.os.Bundle;  
import android.app.Activity;  
import android.app.AlertDialog;  
import android.content.Intent;  
import android.view.View;  
import android.view.View.OnClickListener;  
import android.widget.AdapterView;  
import android.widget.Button;  
import android.widget.ListAdapter;  
import android.widget.ListView;  
import android.widget.SimpleAdapter;  
import android.widget.TabHost;  
import android.widget.TabHost.TabSpec;  
import android.widget.TextView;  
  
public class MainActivityMenuUtama extends Activity {  
  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.menu_utama);  
        Button btn=(Button)findViewById(R.id.button1);  
  
        btn.setOnClickListener(new OnClickListener() {  
  
            @Override
```

```
public void onClick(View v) {
    // TODO Auto-generated method stub
    Intent intent1= new
Intent(v.getContext(),MainActivityToko.class);
    startActivityForResult(intent1, 0);

}

});

Button btn1=(Button)findViewById(R.id.button2);
btn1.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        Intent intent2= new
Intent(v.getContext(),MainActivityResto.class);
        startActivityForResult(intent2, 0);

    }

});

Button btn2=(Button)findViewById(R.id.button3);
btn2.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        Intent intent2= new
Intent(v.getContext(),MainActivityLounge.class);
        startActivityForResult(intent2, 0);

    }

});

Button btn3=(Button)findViewById(R.id.button4);
btn3.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub

```

```
        Intent intent3= new
Intent(v.getContext(),MainActivity.class);
startActivityForResult(intent3, 0);

    }
});
```

}

### **MainActivityMore.java**

```
package net.muhamadiqbal.petalokasi;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import org.w3c.dom.Document;

import android.content.Intent;
import android.location.Location;
import android.net.Uri;
import android.os.AsyncTask;
import android.os.Bundle;
import android.support.v4.app.FragmentActivity;
import android.util.Log;
import android.view.Menu;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import com.google.android.gms.common.ConnectionResult;
com.google.android.gms.common.GooglePlayServicesClient.ConnectionCallbacks;
com.google.android.gms.common.GooglePlayServicesClient.OnConnectionFailedLi
stener;
import com.google.android.gms.location.LocationClient;
import com.google.android.gms.location.LocationListener;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.GoogleMap.InfoWindowAdapter;
```

```
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.BitmapDescriptor;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.CameraPosition;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;

//implements class yang dibutuhkan
public class MainActivity extends FragmentActivity
    implements ConnectionCallbacks,
    OnConnectionFailedListener,
    LocationListener{
    private GoogleMap mMap;

    private LocationClient mLocationClient;
    //menyimpan JSON node name ke variabel
    private static final String Data = "data";
    private static final String Nama = "nama";
    private static final String Lokasi = "lokasi";
    private static final String Phone = "phone";
    private static final String Keterangan = "keterangan";
    private static final String Lintang = "lintang";
    private static final String Bujur = "bujur";
    private static final String Kode = "kode";
    private static final String url= "http://isgskripsi.esy.es/more.php";
    static boolean a=false;
    static int b=0;
    //buat menyimpan latitude longitude
    static double lat,lon;

    JSONArray data = null;
    // These settings are the same as the settings for the map. They will in fact give you
    updates
        // at the maximal rates currently possible.
    private static final LocationRequest REQUEST = LocationRequest.create()
        .setInterval(5000)      // 5 seconds
        .setFastestInterval(16)  // 16ms = 60fps
        .setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
```

```
try {
    // Loading map
    initializeMap();

} catch (Exception e) {
    e.printStackTrace();
}
}

private void initializeMap()
{

if (mMap == null) {
    //Try to obtain the map from the SupportMapFragment.
    mMap = ((SupportMapFragment)
getSupportFragmentManager().findFragmentById(R.id.map))
        .getMap();
    if (mMap != null) {
        Log.e("Sukses","sukses");
        //Menampilkan tombol my location pada peta
        mMap.setMyLocationEnabled(true);
        //Menampilkan marker
        new JSONParse().execute();
    }
}
}

@Override
protected void onResume() {
    super.onResume();
    initializeMap();
    setUpLocationClientIfNeeded();
    mLocationClient.connect();
}

@Override
public void onPause() {
    super.onPause();
    if (mLocationClient != null) {
        mLocationClient.disconnect();
    }
}

//jika lokasi belum didapat maka return lokasi
private void setUpLocationClientIfNeeded() {
    if (mLocationClient == null) {
```

```
mLocationClient = new LocationClient(  
    getApplicationContext(),  
    this, // ConnectionCallbacks  
    this); // OnConnectionFailedListener  
}  
}  
//method ketika lokasi user berubah  
@Override  
public void onLocationChanged(Location location) {  
    //simpan latitude dan longitude ke double  
    lat=location.getLatitude();  
    lon=location.getLongitude();  
    //deklarasi lokasi sebagai LatLng google maps  
    LatLng lokasi=new LatLng(lat,lon);  
    //setting lokasi, zoom, dan animasi kamera ketika awal masuk aplikasi  
    if(b==0)  
    {  
        CameraPosition cameraPosition = new CameraPosition.Builder()  
            .target(lokasi)    // Sets the center of the map to Mountain View  
            .zoom(17)         // Sets the zoom  
            .bearing(90)      // Sets the orientation of the camera to east  
            .tilt(30)         // Sets the tilt of the camera to 30 degrees  
            .build();         // Creates a CameraPosition from the builder  
        //animasi kamera langsung ke posisi user  
  
        mMap.animateCamera(CameraUpdateFactory.newCameraPosition(cameraPosition))  
    };  
    b=1;  
}else {};  
//just console log, to see if im wrong.. :)  
//Log.e is useful for debugging bro  
Log.e("Location = ", location.toString());  
}  
  
@Override  
public void onConnected(Bundle connectionHint) {  
    mLocationClient.requestLocationUpdates(  
        REQUEST,  
        this); // LocationListener  
}  
  
@Override  
public void onDisconnected() {
```

```
}

@Override
public void onConnectionFailed(ConnectionResult result) {
    // Do nothing
}
//JSON parser, mengambil data dari web service,
private class JSONParse extends AsyncTask<String, String, JSONObject> {
@Override
protected void onPreExecute(){

}
//menjalankan proses di background, tidak mengganggu proses lain
@Override
protected JSONObject doInBackground(String... args)
{
//Membuat JSON Parser instance
JSONParser jParser = new JSONParser();

//mengambil JSON String dari url
JSONObject json = jParser.getJSONFromUrl(url);
if(json==null)
{
    a=false;
}
else a=true;
return json;
}
protected void onPostExecute(JSONObject json) {
if(a==true)
{
try{
    Log.e("status",a+"");

    data = json.getJSONArray(Data);
    for(int i=0; i<data.length();i++)
    {
        JSONObject a = data.getJSONObject(i);
        //simpan di variable
        String nama = a.getString(Nama);
        String lokasi = a.getString(Lokasi);
        String phone = a.getString(Phone);
        String keterangan = a.getString(Keterangan);
    }
}
}
}
}
```

```
String lintang = a.getString(Lintang);
String bujur = a.getString(Bujur);
String kode = a.getString(Kode);
Log.e("nama",nama);

BitmapDescriptor icon =null;
if(kode.equals("ap")){
    icon = BitmapDescriptorFactory.fromResource(R.drawable.ap);
}else if(kode.equals("bni")){
    icon = BitmapDescriptorFactory.fromResource(R.drawable.bni);
}else if(kode.equals("cgk")){
    icon = BitmapDescriptorFactory.fromResource(R.drawable.cgk);
}else if(kode.equals("prk")){
    icon = BitmapDescriptorFactory.fromResource(R.drawable.prk);
}else if(kode.equals("xtr")){
    icon = BitmapDescriptorFactory.fromResource(R.drawable.xtr);
}

String sampah = ("/" +keterangan)+("/" +phone)+("/" +lokasi);
String banget = ("/" +lintang)+("/" +bujur);
//konversi data dari String ke double
//data dari web service bertipe string
Double lat=Double.parseDouble(lintang.toString());
Double longi=Double.parseDouble(bujur.toString());

//add marker ke peta

mMap.addMarker(new MarkerOptions().position(new LatLng(lat, longi))
    .icon(icon)
    .title(nama).snippet(sampah+banget));

mMap.setInfoWindowAdapter(iwa);

}

}catch(JSONException e)
{
e.printStackTrace();
}

//Jika tidak ada koneksi atau server down, keluarkan message error
else {Toast.makeText(getApplicationContext(), "error getting data",
Toast.LENGTH_SHORT).show();}
```

```
        Log.e("status",a+"");
    }
}

InfoWindowAdapter iwa=new InfoWindowAdapter() {
    @Override
    public View getInfoWindow(Marker arg0) {
        return null;
    }

    @Override
    public View getInfoContents(Marker arg0) {

        setContentView(R.layout.info_window);
        final String[] hasil=arg0.getSnippet().split("/");

        TextView tv=(TextView) findViewById(R.id.view1);
        TextView tv4=(TextView) findViewById(R.id.view4);
        TextView tv5=(TextView) findViewById(R.id.view5);
        TextView tv7=(TextView) findViewById(R.id.view7);
        TextView tvL=(TextView) findViewById(R.id.textView1);
        TextView tvB=(TextView) findViewById(R.id.textView2);

        Button btn = (Button) findViewById(R.id.button2);
        Button btn1 = (Button) findViewById(R.id.button1);

        tv.setText(arg0.getTitle());
        tv7.setText(hasil[1]);
        tv5.setText(hasil[2]);
        tv4.setText(hasil[3]);
        tvL.setText(hasil[4]);
        tvB.setText(hasil[5]);
        final String latt = hasil[4];
        final String lonn = hasil[5];
        btn.setOnClickListener(new OnClickListener() {
            public void onClick(View v){
                Intent intent = new
                Intent(android.content.Intent.ACTION_VIEW,
                Uri.parse("http://maps.google.com/maps?saddr="+lat+","+lon+"&daddr="+latt+","+lon));
                startActivity(intent);
            }
        });
    }
}
```

```
});

btn1.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        Intent intent1 = new Intent(v.getContext(),
                MainActivity.class);
        startActivityForResult(intent1, 0);

    }
});
```

return null;

```
}
```

### **MainActivityToko.java**

```
package net.muhamadiqbal.petalokasi;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import org.w3c.dom.Document;

import android.content.Intent;
import android.location.Location;
import android.net.Uri;
import android.os.AsyncTask;
import android.os.Bundle;
import android.support.v4.app.FragmentActivity;
import android.util.Log;
import android.view.Menu;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import com.google.android.gms.common.ConnectionResult;
```

```
com.google.android.gms.common.GooglePlayServicesClient.ConnectionCallbacks;
com.google.android.gms.common.GooglePlayServicesClient.OnConnectionFailedLi
stener;
import com.google.android.gms.location.LocationClient;
import com.google.android.gms.location.LocationListener;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.GoogleMap.InfoWindowAdapter;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.BitmapDescriptor;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.CameraPosition;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;

//implements class yang dibutuhkan
public class MainActivity extends FragmentActivity
    implements ConnectionCallbacks,
    OnConnectionFailedListener,
    LocationListener{
    private GoogleMap mMap;

    private LocationClient mLocationClient;
    //menyimpan JSON node name ke variabel
    private static final String Data = "data";
    private static final String Nama = "nama";
    private static final String Lokasi = "lokasi";
    private static final String Phone = "phone";
    private static final String Keterangan = "keterangan";
    private static final String Lintang = "lintang";
    private static final String Bujur = "bujur";
    private static final String Kode = "kode";
    private static final String url= "http://isgskripsi.esy.es/jtoko.php";
    static boolean a=false;
    static int b=0;
    //buat menyimpan latitude longitude
    static double lat,lon;

    JSONArray data = null;
    // These settings are the same as the settings for the map. They will in fact give you
    updates
```

```
// at the maximal rates currently possible.
private static final LocationRequest REQUEST = LocationRequest.create()
    .setInterval(5000)          // 5 seconds
    .setFastestInterval(16)     // 16ms = 60fps
    .setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    try {
        // Loading map
        initializeMap();

    } catch (Exception e) {
        e.printStackTrace();
    }
}

private void initializeMap()
{
    if (mMap == null) {
        //Try to obtain the map from the SupportMapFragment.
        mMap = ((SupportMapFragment)
            getSupportFragmentManager().findFragmentById(R.id.map))
            .getMap();
        if (mMap != null) {
            Log.e("Sukses","sukses");
            //Menampilkan tombol my location pada peta
            mMap.setMyLocationEnabled(true);
            //Menampilkan marker
            new JSONParse().execute();
        }
    }
}
@Override
protected void onResume() {
    super.onResume();
    initializeMap();
    setUpLocationClientIfNeeded();
    mLocationClient.connect();
}
@Override
```

```
public void onPause() {
    super.onPause();
    if (mLocationClient != null) {
        mLocationClient.disconnect();
    }
}
//jika lokasi belum didapat maka return lokasi
private void setUpLocationClientIfNeeded() {
    if (mLocationClient == null) {
        mLocationClient = new LocationClient(
            getApplicationContext(),
            this, // ConnectionCallbacks
            this); // OnConnectionFailedListener
    }
}
//method ketika lokasi user berubah
@Override
public void onLocationChanged(Location location) {
    //simpan latitude dan longitude ke double
    lat=location.getLatitude();
    lon=location.getLongitude();
    //deklarasi lokasi sebagai LatLng google maps
    LatLng lokasi=new LatLng(lat,lon);
    //setting lokasi, zoom, dan animasi kamera ketika awal masuk aplikasi
    if(b==0)
    {
        CameraPosition cameraPosition = new CameraPosition.Builder()
            .target(lokasi)    // Sets the center of the map to Mountain View
            .zoom(17)          // Sets the zoom
            .bearing(90)       // Sets the orientation of the camera to east
            .tilt(30)          // Sets the tilt of the camera to 30 degrees
            .build();          // Creates a CameraPosition from the builder
        //animasi kamera langsung ke posisi user
        mMap.animateCamera(CameraUpdateFactory.newCameraPosition(cameraPosition));
    }
    b=1;
}else {};
//just console log, to see if im wrong.. :)
//Log.e is useful for debugging bro
Log.e("Location = ", location.toString());
}
```

```
@Override
public void onConnected(Bundle connectionHint) {
    mLocationClient.requestLocationUpdates(
        REQUEST,
        this); // LocationListener
}

@Override
public void onDisconnected() {
}

@Override
public void onConnectionFailed(ConnectionResult result) {
    // Do nothing
}
//JSON parser, mengambil data dari web service,
private class JSONParse extends AsyncTask<String, String, JSONObject> {
@Override
protected void onPreExecute(){
}

//menjalankan proses di background, tidak mengganggu proses lain
@Override
protected JSONObject doInBackground(String... args)
{
//Membuat JSON Parser instance
JSONParser jParser = new JSONParser();

//mengambil JSON String dari url
JSONObject json = jParser.getJSONFromUrl(url);
if(json==null)
{
    a=false;
}
else a=true;
return json;
}
protected void onPostExecute(JSONObject json) {
if(a==true)
{
try{
    Log.e("status",a+"");
}
```

```
data = json.getJSONArray(Data);
for(int i=0; i<data.length();i++)
{
    JSONObject a = data.getJSONObject(i);
    //simpan di variable
    String nama = a.getString(Nama);
    String lokasi = a.getString(Lokasi);
    String phone = a.getString(Phone);
    String keterangan = a.getString(Keterangan);
    String lintang = a.getString(Lintang);
    String bujur = a.getString(Bujur);
    String kode = a.getString(Kode);
    Log.e("nama",nama);

    BitmapDescriptor icon =null;
    if(kode.equals("ap")){
        icon = BitmapDescriptorFactory.fromResource(R.drawable.ap);
    }else if(kode.equals("bni")){
        icon = BitmapDescriptorFactory.fromResource(R.drawable.bni);
    }else if(kode.equals("cgk")){
        icon = BitmapDescriptorFactory.fromResource(R.drawable.cgk);
    }else if(kode.equals("prk")){
        icon = BitmapDescriptorFactory.fromResource(R.drawable.prk);
    }else if(kode.equals("xtr")){
        icon = BitmapDescriptorFactory.fromResource(R.drawable.xtr);
    }

    String sampah = ("/" +keterangan)+("/" +phone)+("/" +lokasi);
    String banget = ("/" +lintang)+("/" +bujur);
    //konversi data dari String ke double
    //data dari web service bertipe string
    Double lat=Double.parseDouble(lintang.toString());
    Double longi=Double.parseDouble(bujur.toString());

    //add marker ke peta

    mMap.addMarker(new MarkerOptions().position(new LatLng(lat, longi))
        .icon(icon)
        .title(nama).snippet(sampah+banget));

    mMap.setInfoWindowAdapter(iwa);
}
```

```
        }catch(JSONException e)
    {
        e.printStackTrace();
    }
}

//Jika tidak ada koneksi atau server down, keluarkan message error
else {Toast.makeText(getApplicationContext(), "error getting data",
Toast.LENGTH_SHORT).show();
Log.e("status",a+"");
}

InfoWindowAdapter iwa=new InfoWindowAdapter() {
    @Override
    public View getInfoWindow(Marker arg0) {
        return null;
    }

    @Override
    public View getInfoContents(Marker arg0) {

        setContentView(R.layout.info_window);
        final String[] hasil=arg0.getSnippet().split("/");

        TextView tv=(TextView) findViewById(R.id.view1);
        TextView tv4=(TextView) findViewById(R.id.view4);
        TextView tv5=(TextView) findViewById(R.id.view5);
        TextView tv7=(TextView) findViewById(R.id.view7);
        TextView tvL=(TextView) findViewById(R.id.textView1);
        TextView tvB=(TextView) findViewById(R.id.textView2);

        Button btn = (Button) findViewById(R.id.button2);
        Button btn1 = (Button) findViewById(R.id.button1);

        tv.setText(arg0.getTitle());
        tv7.setText(hasil[1]);
        tv5.setText(hasil[2]);
        tv4.setText(hasil[3]);
        tvL.setText(hasil[4]);
        tvB.setText(hasil[5]);
    final String latt = hasil[4];
    final String lonn = hasil[5];
}
```

```
        btn.setOnClickListener(new OnClickListener() {
            public void onClick(View v){
                Intent intent = new
                Intent(android.content.Intent.ACTION_VIEW,
                Uri.parse("http://maps.google.com/maps?addr="+lat+","+lon+"&daddr="+latt+","+
                lonn+""));
                startActivity(intent);
            }
        });

        btn1.setOnClickListener(new OnClickListener() {

            @Override
            public void onClick(View v) {
                // TODO Auto-generated method stub
                Intent intent1 = new Intent(v.getContext(),
                    MainActivity.class);
                startActivityForResult(intent1, 0);

            }
        });

        return null;
    }
};
```

### **MainActivityLounge.java**

```
package net.muhamadiqbal.petalokasi;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import org.w3c.dom.Document;

import android.content.Intent;
import android.location.Location;
import android.net.Uri;
import android.os.AsyncTask;
import android.os.Bundle;
```

```
import android.support.v4.app.FragmentActivity;
import android.util.Log;
import android.view.Menu;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.GooglePlayServicesClient.ConnectionCallbacks;
import com.google.android.gms.common.GooglePlayServicesClient.OnConnectionFailedListener;
import com.google.android.gms.location.LocationClient;
import com.google.android.gms.location.LocationListener;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.GoogleMap.InfoWindowAdapter;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.BitmapDescriptor;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.CameraPosition;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;

//implements class yang dibutuhkan
public class MainActivity extends FragmentActivity
    implements ConnectionCallbacks,
    OnConnectionFailedListener,
    LocationListener{
    private GoogleMap mMap;

    private LocationClient mLocationClient;
    //menyimpan JSON node name ke variabel
    private static final String Data = "data";
    private static final String Nama = "nama";
    private static final String Lokasi = "lokasi";
    private static final String Phone = "phone";
    private static final String Keterangan = "keterangan";
    private static final String Lintang = "lintang";
    private static final String Bujur = "bujur";
    private static final String Kode = "kode";
```

```
private static final String url= "http://isgskripsi.esy.es/jlounge.php";
static boolean a=false;
static int b=0;
//buat menyimpan latitude longitude
static double lat,lon;

JSONArray data = null;
// These settings are the same as the settings for the map. They will in fact give you
updates
// at the maximal rates currently possible.
private static final LocationRequest REQUEST = LocationRequest.create()
    .setInterval(5000)      // 5 seconds
    .setFastestInterval(16)  // 16ms = 60fps
    .setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    try {
        // Loading map
        initializeMap();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

private void initializeMap()
{
    if (mMap == null) {
        //Try to obtain the map from the SupportMapFragment.
        mMap = ((SupportMapFragment)
getSupportFragmentManager().findFragmentById(R.id.map))
            .getMap();
        if (mMap != null) {
            Log.e("Sukses","sukses");
            //Menampilkan tombol my location pada peta
            mMap.setMyLocationEnabled(true);
            //Menampilkan marker
            new JSONParse().execute();
        }
    }
}
```

```
}

@Override
protected void onResume() {
    super.onResume();
    initializeMap();
    setUpLocationClientIfNeeded();
    mLocationClient.connect();
}

@Override
public void onPause() {
    super.onPause();
    if (mLocationClient != null) {
        mLocationClient.disconnect();
    }
}

//jika lokasi belum didapat maka return lokasi
private void setUpLocationClientIfNeeded() {
    if (mLocationClient == null) {
        mLocationClient = new LocationClient(
            getApplicationContext(),
            this, // ConnectionCallbacks
            this); // OnConnectionFailedListener
    }
}

//method ketika lokasi user berubah
@Override
public void onLocationChanged(Location location) {
    //simpan latitude dan longitude ke double
    lat=location.getLatitude();
    lon=location.getLongitude();
    //deklarasi lokasi sebagai LatLng google maps
    LatLng lokasi=new LatLng(lat,lon);
    //setting lokasi, zoom, dan animasi kamera ketika awal masuk aplikasi
    if(b==0)
    {
        CameraPosition cameraPosition = new CameraPosition.Builder()
            .target(lokasi)    // Sets the center of the map to Mountain View
            .zoom(17)          // Sets the zoom
            .bearing(90)        // Sets the orientation of the camera to east
            .tilt(30)           // Sets the tilt of the camera to 30 degrees
            .build();           // Creates a CameraPosition from the builder
        //animasi kamera langsung ke posisi user
    }
}
```

```
mMap.animateCamera(CameraUpdateFactory.newCameraPosition(cameraPosition));
);
b=1;
}else {};
//just console log, to see if im wrong.. :)
//Log.e is useful for debugging bro
Log.e("Location = ", location.toString());
}

@Override
public void onConnected(Bundle connectionHint) {
    mLocationClient.requestLocationUpdates(
        REQUEST,
        this); // LocationListener
}

@Override
public void onDisconnected() {
}

@Override
public void onConnectionFailed(ConnectionResult result) {
    // Do nothing
}
//JSON parser, mengambil data dari web service,
private class JSONParse extends AsyncTask<String, String, JSONObject> {
@Override
protected void onPreExecute(){
}

//menjalankan proses di background, tidak mengganggu proses lain
@Override
protected JSONObject doInBackground(String... args)
{
//Membuat JSON Parser instance
JSONParser jParser = new JSONParser();

//mengambil JSON String dari url
JSONObject json = jParser.getJSONFromUrl(url);
if(json==null)
{
    a=false;
```

```
        }
    else a=true;
    return json;
}
protected void onPostExecute(JSONObject json) {
if(a==true)
{
try{
    Log.e("status",a+"");
    data = json.getJSONArray(Data);
    for(int i=0; i<data.length();i++)
    {
        JSONObject a = data.getJSONObject(i);
        //simpan di variable
        String nama = a.getString(Nama);
        String lokasi = a.getString(Lokasi);
        String phone = a.getString(Phone);
        String keterangan = a.getString(Keterangan);
        String lintang = a.getString(Lintang);
        String bujur = a.getString(Bujur);
        String kode = a.getString(Kode);
        Log.e("nama",nama);
        BitmapDescriptor icon =null;
        if(kode.equals(" ap")){
            icon = BitmapDescriptorFactory.fromResource(R.drawable.ap);
        }else if(kode.equals("bni")){
            icon = BitmapDescriptorFactory.fromResource(R.drawable.bni);
        }else if(kode.equals("cgk")){
            icon = BitmapDescriptorFactory.fromResource(R.drawable.cgk);
        }else if(kode.equals("prk")){
            icon = BitmapDescriptorFactory.fromResource(R.drawable.prk);
        }else if(kode.equals("xtr")){
            icon = BitmapDescriptorFactory.fromResource(R.drawable.xtr);
        }
        String sampah = (" /"+keterangan)+(" /"+phone)+(" /"+lokasi);
        String banget = (" /"+lintang)+(" /"+bujur);
        //konversi data dari String ke double
        //data dari web service bertipe string
        Double lat=Double.parseDouble(lintang.toString());
        Double longi=Double.parseDouble(bujur.toString());
    }
}
```

```
//add marker ke peta

mMap.addMarker(new MarkerOptions().position(new LatLng(lat, longi))
    .icon(icon)
    .title(nama).snippet(sampah+banget));

mMap.setInfoWindowAdapter(iwa);

}

}catch(JSONException e)
{
e.printStackTrace();
}

//Jika tidak ada koneksi atau server down, keluarkan message error
else {Toast.makeText(getApplicationContext(), "error getting data",
Toast.LENGTH_SHORT).show();
Log.e("status",a+"");
}

}

InfoWindowAdapter iwa=new InfoWindowAdapter() {
    @Override
    public View getInfoWindow(Marker arg0) {
        return null;
    }

    @Override
    public View getInfoContents(Marker arg0) {

        setContentView(R.layout.info_window);
        final String[] hasil=arg0.getSnippet().split("/");

        TextView tv=(TextView) findViewById(R.id.view1);
        TextView tv4=(TextView) findViewById(R.id.view4);
        TextView tv5=(TextView) findViewById(R.id.view5);
        TextView tv7=(TextView) findViewById(R.id.view7);
        TextView tvL=(TextView) findViewById(R.id.textView1);
        TextView tvB=(TextView) findViewById(R.id.textView2);

        Button btn = (Button) findViewById(R.id.button2);
```

```
Button btn1 = (Button) findViewById(R.id.button1);

tv.setText(arg0.getTitle());
tv7.setText(hasil[1]);
tv5.setText(hasil[2]);
tv4.setText(hasil[3]);
tvL.setText(hasil[4]);
tvB.setText(hasil[5]);
final String latt = hasil[4];
final String lonn = hasil[5];
btn.setOnClickListener(new OnClickListener() {
    public void onClick(View v){
        Intent intent = new
Intent(android.content.Intent.ACTION_VIEW,
Uri.parse("http://maps.google.com/maps?saddr="+lat+","+lon+"&daddr="+latt+","+lonn+""));
        startActivity(intent);
    }
});

btn1.setOnClickListener(new OnClickListener() {

    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        Intent intent1 = new Intent(v.getContext(),
            MainActivity.class);
        startActivityForResult(intent1, 0);

    }
});

return null;
}
});
```

## MainActivityResto.java

```
package net.muhamadiqbal.petalokasi;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import org.w3c.dom.Document;

import android.content.Intent;
import android.location.Location;
import android.net.Uri;
import android.os.AsyncTask;
import android.os.Bundle;
import android.support.v4.app.FragmentActivity;
import android.util.Log;
import android.view.Menu;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.GooglePlayServicesClient.ConnectionCallbacks;
import com.google.android.gms.common.GooglePlayServicesClient.OnConnectionFailedListener;
import com.google.android.gms.location.LocationClient;
import com.google.android.gms.location.LocationListener;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.InfoWindowAdapter;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.BitmapDescriptor;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.CameraPosition;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;

//implements class yang dibutuhkan
public class MainActivity extends FragmentActivity
    implements ConnectionCallbacks,
```

```
OnConnectionFailedListener,
LocationListener{
private GoogleMap mMap;

private LocationClient mLocationClient;
//menyimpan JSON node name ke variabel
private static final String Data = "data";
private static final String Nama = "nama";
private static final String Lokasi = "lokasi";
private static final String Phone = "phone";
private static final String Keterangan = "keterangan";
private static final String Lintang = "lintang";
private static final String Bujur = "bujur";
private static final String Kode = "kode";
private static final String url= "http://isgskripsi.esy.es/jresto.php";
static boolean a=false;
static int b=0;
//buat menyimpan latitude longitude
static double lat,lon;

JSONArray data = null;
// These settings are the same as the settings for the map. They will in fact give you
updates
    // at the maximal rates currently possible.
    private static final LocationRequest REQUEST = LocationRequest.create()
        .setInterval(5000)      // 5 seconds
        .setFastestInterval(16)  // 16ms = 60fps
        .setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    try {
        // Loading map
        initializeMap();
    } catch (Exception e) {
        e.printStackTrace();
    }
}

private void initializeMap()
{
```

```
if (mMap == null) {
    //Try to obtain the map from the SupportMapFragment.
    mMap = ((SupportMapFragment)
getSupportFragmentManager().findFragmentById(R.id.map))
        .getMap();
    if (mMap != null) {
        Log.e("Sukses","sukses");
        //Menampilkan tombol my location pada peta
        mMap.setMyLocationEnabled(true);
        //Menampilkan marker
        new JSONParse().execute();
    }
}
@Override
protected void onResume() {
    super.onResume();
    initializeMap();
    setUpLocationClientIfNeeded();
    mLocationClient.connect();
}
@Override
public void onPause() {
    super.onPause();
    if (mLocationClient != null) {
        mLocationClient.disconnect();
    }
}
//jika lokasi belum didapat maka return lokasi
private void setUpLocationClientIfNeeded() {
    if (mLocationClient == null) {
        mLocationClient = new LocationClient(
            getApplicationContext(),
            this, // ConnectionCallbacks
            this); // OnConnectionFailedListener
    }
}
//method ketika lokasi user berubah
@Override
public void onLocationChanged(Location location) {
    //simpan latitude dan longitude ke double
    lat=location.getLatitude();
```

```
lon=location.getLongitude();
//deklarasi lokasi sebagai LatLng google maps
LatLng lokasi=new LatLng(lat,lon);
//setting lokasi, zoom, dan animasi kamera ketika awal masuk aplikasi
if(b==0)
{
    CameraPosition cameraPosition = new CameraPosition.Builder()
        .target(lokasi)      // Sets the center of the map to Mountain View
        .zoom(17)           // Sets the zoom
        .bearing(90)         // Sets the orientation of the camera to east
        .tilt(30)            // Sets the tilt of the camera to 30 degrees
        .build();            // Creates a CameraPosition from the builder
    //animasi kamera langsung ke posisi user

mMap.animateCamera(CameraUpdateFactory.newCameraPosition(cameraPosition));
};

b=1;
}else {};
//just console log, to see if im wrong.. :(
//Log.e is useful for debugging bro
Log.e("Location = ", location.toString());
}

@Override
public void onConnected(Bundle connectionHint) {
    mLocationClient.requestLocationUpdates(
        REQUEST,
        this); // LocationListener
}

@Override
public void onDisconnected() {
}

@Override
public void onConnectionFailed(ConnectionResult result) {
    // Do nothing
}
//JSON parser, mengambil data dari web service,
private class JSONParse extends AsyncTask<String, String, JSONObject> {
@Override
protected void onPreExecute(){
```

```
        }
        //menjalankan proses di background, tidak mengganggu proses lain
        @Override
    protected JSONObject doInBackground(String... args)
    {
        //Membuat JSON Parser instance
        JSONParser jParser = new JSONParser();

        //mengambil JSON String dari url
        JSONObject json = jParser.getJSONFromUrl(url);
        if(json==null)
        {
            a=false;
        }
        else a=true;
        return json;
    }
    protected void onPostExecute(JSONObject json) {
        if(a==true)
        {
            try{
                Log.e("status",a+"");
                data = json.getJSONArray(Data);
                for(int i=0; i<data.length();i++)
                {
                    JSONObject a = data.getJSONObject(i);
                    //simpan di variable
                    String nama = a.getString(Nama);
                    String lokasi = a.getString(Lokasi);
                    String phone = a.getString(Phone);
                    String keterangan = a.getString(Keterangan);
                    String lintang = a.getString(Lintang);
                    String bujur = a.getString(Bujur);
                    String kode = a.getString(Kode);
                    Log.e("nama",nama);

                    BitmapDescriptor icon =null;
                    if(kode.equals("ap")){
                        icon = BitmapDescriptorFactory.fromResource(R.drawable.ap);
                    }else if(kode.equals("bni")){
                        icon = BitmapDescriptorFactory.fromResource(R.drawable.bni);
                    }else if(kode.equals("cgk")){

```

```
        icon = BitmapDescriptorFactory.fromResource(R.drawable.cgk);
    }else if(kode.equals("prk")){
        icon = BitmapDescriptorFactory.fromResource(R.drawable.prk);
    }else if(kode.equals("xtr")){
        icon = BitmapDescriptorFactory.fromResource(R.drawable.xtr);
    }

    String sampah = ("/"+keterangan)+("/"+phone)+("/"+lokasi);
    String banget = ("/"+lintang)+("/"+bujur);
    //konversi data dari String ke double
    //data dari web service bertipe string
    Double lat=Double.parseDouble(lintang.toString());
    Double longi=Double.parseDouble(bujur.toString());

    //add marker ke peta

    mMap.addMarker(new MarkerOptions().position(new LatLng(lat, longi))
        .icon(icon)
        .title(nama).snippet(sampah+banget));

    mMap.setInfoWindowAdapter(iwa);

}

}catch(JSONException e)
{
e.printStackTrace();
}

//Jika tidak ada koneksi atau server down, keluarkan message error
else {Toast.makeText(getApplicationContext(), "error getting data",
Toast.LENGTH_SHORT).show();
Log.e("status",a+"");
}

}

InfoWindowAdapter iwa=new InfoWindowAdapter() {
    @Override
    public View getInfoWindow(Marker arg0) {
        return null;
    }

    @Override
```

```
public View getInfoContents(Marker arg0) {  
  
    setContentView(R.layout.info_window);  
    final String[] hasil=arg0.getSnippet().split("/");  
  
    TextView tv=(TextView) findViewById(R.id.view1);  
    TextView tv4=(TextView) findViewById(R.id.view4);  
    TextView tv5=(TextView) findViewById(R.id.view5);  
    TextView tv7=(TextView) findViewById(R.id.view7);  
    TextView tvL=(TextView) findViewById(R.id.textView1);  
    TextView tvB=(TextView) findViewById(R.id.textView2);  
  
    Button btn = (Button) findViewById(R.id.button2);  
    Button btn1 = (Button) findViewById(R.id.button1);  
  
    tv.setText(arg0.getTitle());  
    tv7.setText(hasil[1]);  
    tv5.setText(hasil[2]);  
    tv4.setText(hasil[3]);  
    tvL.setText(hasil[4]);  
    tvB.setText(hasil[5]);  
    final String latt = hasil[4];  
    final String lonn = hasil[5];  
    btn.setOnClickListener(new OnClickListener() {  
        public void onClick(View v){  
            Intent intent = new  
            Intent(android.content.Intent.ACTION_VIEW,  
            Uri.parse("http://maps.google.com/maps?saddr="+lat+","+lon+"&daddr="+latt+","+  
            lon+""));  
            startActivity(intent);  
        }  
    });  
  
    btn1.setOnClickListener(new OnClickListener() {  
  
        @Override  
        public void onClick(View v) {  
            // TODO Auto-generated method stub  
            Intent intent1 = new Intent(v.getContext(),  
                MainActivity.class);  
            startActivityForResult(intent1, 0);  
        }  
    });  
}
```

```
        }  
    } );  
    return null;  
}  
};  
}
```