**addUser.java**

package com.example.win8.myapplication;

import android.content.Context;

import android.database.sqlite.SQLiteDatabase;

import android.os.Bundle;

import android.support.design.widget.FloatingActionButton;

import android.support.design.widget.Snackbar;

import android.support.v7.app.AppCompatActivity;

import android.support.v7.widget.Toolbar;

import android.view.Menu;

import android.view.MenuItem;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

public class addUser extends AppCompatActivity {

private EditText editTextName;

private EditText editTextphn;

private Button btnAdd;

private Button btnView;

private Button btnPrev;

private Button btnNext;

private Button btnSave;

private Button btnDelete;

private SQLiteDatabase db;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_add\_user);

// Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);

// setSupportActionBar(toolbar);

createDatabase();

editTextName = (EditText) findViewById(R.id.name);

editTextphn = (EditText) findViewById(R.id.phone);

}

protected void createDatabase(){

db=openOrCreateDatabase("PersonDB", Context.MODE\_PRIVATE, null);

db.execSQL("CREATE TABLE IF NOT EXISTS persons(id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, name VARCHAR,phn\_no VARCHAR);");

}

protected void insertIntoDB(){

String name = editTextName.getText().toString().trim();

String add = editTextphn.getText().toString().trim();

if(name.equals("") || add.equals("")){

Toast.makeText(getApplicationContext(), "Please fill all fields", Toast.LENGTH\_LONG).show();

return;

}

String query = "INSERT INTO persons (name,phn\_no) VALUES('"+name+"', '"+add+"');";

db.execSQL(query);

Toast.makeText(getApplicationContext(),"Saved Successfully", Toast.LENGTH\_LONG).show();

// db.close();

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.menu\_main, menu);

return true;

}

@Override

public boolean onOptionsItemSelected(MenuItem item) {

// Handle action bar item clicks here. The action bar will

// automatically handle clicks on the Home/Up button, so long

// as you specify a parent activity in AndroidManifest.xml.

int id = item.getItemId();

//noinspection SimplifiableIfStatement

if (id == R.id.action\_settings) {

return true;

}

return super.onOptionsItemSelected(item);

}

// @Override

public void add(View v) {

// if (v == btnAdd) {

insertIntoDB();

editTextphn.setText("");

editTextName.setText("");

// }

}

public void cancel(View v)

{

finish();

}

}

**GPSTracker.java**

package com.example.win8.myapplication;

import android.Manifest;

import android.app.AlertDialog;

import android.app.Service;

import android.content.Context;

import android.content.DialogInterface;

import android.content.Intent;

import android.content.pm.PackageManager;

import android.location.Location;

import android.location.LocationListener;

import android.location.LocationManager;

import android.os.Bundle;

import android.os.IBinder;

import android.provider.Settings;

import android.support.v4.app.ActivityCompat;

import android.util.Log;

public class GPSTracker extends Service implements LocationListener {

private final Context mContext;

// flag for GPS status

boolean isGPSEnabled = false;

// flag for network status

boolean isNetworkEnabled = false;

// flag for GPS status

boolean canGetLocation = false;

Location location; // location

double latitude; // latitude

double longitude; // longitude

// The minimum distance to change Updates in meters

private static final long MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES = 10; // 10 meters

// The minimum time between updates in milliseconds

private static final long MIN\_TIME\_BW\_UPDATES = 1000 \* 60 \* 1; // 1 minute

// Declaring a Location Manager

protected LocationManager locationManager;

public GPSTracker(Context context) {

this.mContext = context;

getLocation();

}

public Location getLocation() {

try {

locationManager = (LocationManager) mContext

.getSystemService(LOCATION\_SERVICE);

// getting GPS status

isGPSEnabled = locationManager

.isProviderEnabled(LocationManager.GPS\_PROVIDER);

// getting network status

isNetworkEnabled = locationManager

.isProviderEnabled(LocationManager.NETWORK\_PROVIDER);

if (!isGPSEnabled && !isNetworkEnabled) {

// no network provider is enabled

} else {

this.canGetLocation = true;

if (isNetworkEnabled) { locationManager.requestLocationUpdates(LocationManager.NETWORK\_PROVIDER, MIN\_TIME\_BW\_UPDATES, MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES, this);

Log.d("Network", "Network");

if (locationManager != null) {

location = locationManager.getLastKnownLocation(LocationManager.NETWORK\_PROVIDER);

if (location != null) {

latitude = location.getLatitude();

longitude = location.getLongitude();

}

}

}

// if GPS Enabled get lat/long using GPS Services

if (isGPSEnabled) {

if (location == null) {

locationManager.requestLocationUpdates(LocationManager.GPS\_PROVIDER, MIN\_TIME\_BW\_UPDATES, MIN\_DISTANCE\_CHANGE\_FOR\_UPDATES, this);

Log.d("GPS Enabled", "GPS Enabled");

if (locationManager != null) {

if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED && ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_COARSE\_LOCATION) != PackageManager.PERMISSION\_GRANTED) {

return null;

}

location = locationManager.getLastKnownLocation(LocationManager.GPS\_PROVIDER);

if (location != null) {

latitude = location.getLatitude();

longitude = location.getLongitude();

}

}

}

}

}

} catch (Exception e) {

e.printStackTrace();

}

return location;

}

public void stopUsingGPS() {

if (locationManager != null) {

if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED && ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_COARSE\_LOCATION) != PackageManager.PERMISSION\_GRANTED) {

// TODO: Consider calling

// ActivityCompat#requestPermissions

// here to request the missing permissions, and then overriding

// public void onRequestPermissionsResult(int requestCode, String[] permissions,

// int[] grantResults)

// to handle the case where the user grants the permission. See the documentation

// for ActivityCompat#requestPermissions for more details.

return;

}

locationManager.removeUpdates(GPSTracker.this);

}

}

public double getLatitude(){

if(location != null){

latitude = location.getLatitude();

}

// return latitude

return latitude;

}

public double getLongitude(){

if(location != null){

longitude = location.getLongitude();

}

// return longitude

return longitude;

}

public boolean canGetLocation() {

return this.canGetLocation;

}

public void showSettingsAlert(){

AlertDialog.Builder alertDialog = new AlertDialog.Builder(mContext);

// Setting Dialog Title

alertDialog.setTitle("GPS is settings");

// Setting Dialog Message

alertDialog.setMessage("GPS is not enabled. Do you want to go to settings menu?");

// On pressing Settings button

alertDialog.setPositiveButton("Settings", new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog,int which) {

Intent intent = new Intent(Settings.ACTION\_LOCATION\_SOURCE\_SETTINGS);

mContext.startActivity(intent);

}

});

// on pressing cancel button

alertDialog.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int which) {

dialog.cancel();

}

});

// Showing Alert Message

alertDialog.show();

}

@Override

public void onLocationChanged(Location location) {

}

@Override

public void onProviderDisabled(String provider) {

}

@Override

public void onProviderEnabled(String provider) {

}

@Override

public void onStatusChanged(String provider, int status, Bundle extras) {

}

@Override

public IBinder onBind(Intent arg0) {

return null;

}

}

**location\_finder.java**

package com.example.win8.myapplication;

import android.Manifest;

import android.content.Context;

import android.content.Intent;

import android.content.pm.PackageManager;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.location.Address;

import android.location.Criteria;

import android.location.Geocoder;

import android.location.Location;

import android.location.LocationListener;

import android.net.ConnectivityManager;

import android.net.NetworkInfo;

import android.net.Uri;

import android.os.Bundle;

import android.support.v4.app.ActivityCompat;

import android.support.v7.app.AppCompatActivity;

import android.telephony.SmsManager;

import android.util.Log;

import android.view.View;

import android.widget.Toast;

import com.google.android.gms.maps.CameraUpdateFactory;

import com.google.android.gms.maps.GoogleMap;

import java.io.IOException;

import java.net.HttpURLConnection;

import java.net.URL;

import java.util.List;

import java.util.Locale;

public class location\_finder extends AppCompatActivity implements LocationListener {

private static final String SELECT\_SQL = "SELECT \* FROM persons";

private SQLiteDatabase db;

private Cursor cur;

private GoogleMap mMap;

// GPSTracker class

GPSTracker gps;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

// Obtain the SupportMapFragment and get notified when the map is ready to be used.

}

public boolean isInternetWorking() {

ConnectivityManager cm =

(ConnectivityManager) getSystemService(Context.CONNECTIVITY\_SERVICE);

NetworkInfo activeNetwork = cm.getActiveNetworkInfo();

boolean isConnected = activeNetwork != null && activeNetwork.isConnectedOrConnecting();

return isConnected;

}

public void callhelp(View v) throws IOException {

/\* Intent in = new Intent(MapsActivity.this,help.class);

// Starts TargetActivity

startActivity(in);

\*/

gps = new GPSTracker(location\_finder.this);

// check if GPS enabled

double latitude = 0;

double longitude = 0;

String add = null;

if (gps.canGetLocation()) {

latitude = gps.getLatitude();

longitude = gps.getLongitude();

if (isInternetWorking()) {

Geocoder geocoder = new Geocoder(this, Locale.ENGLISH);

//Place your latitude and longitude

List<Address> addresses = geocoder.getFromLocation(latitude, longitude, 1);

if (addresses != null) {

Address fetchedAddress = addresses.get(0);

StringBuilder strAddress = new StringBuilder();

for (int i = 0; i < fetchedAddress.getMaxAddressLineIndex(); i++) {

strAddress.append(fetchedAddress.getAddressLine(i)).append("\n");

}

add = strAddress.toString();

}

} else {

add = "";

}

// Toast.makeText(getApplicationContext(),"I am at"+add+""+"lat "+latitude+"long"+longitude, Toast.LENGTH\_LONG).show();

String sms = "I am at location\n:" + add + "lat:" + latitude + " long:" + longitude;

Log.i("Send SMS", "");

openDatabase();

cur = db.rawQuery(SELECT\_SQL, null);

cur.moveToFirst();

while (cur.isAfterLast() == false) {

String phoneNo = cur.getString(2);

try {

SmsManager smsManager = SmsManager.getDefault();

smsManager.sendTextMessage(phoneNo, null, sms, null, null);

Toast.makeText(getApplicationContext(), "SMS sent.", Toast.LENGTH\_LONG).show();

} catch (Exception e) {

Toast.makeText(getApplicationContext(), "SMS failed, please try again.", Toast.LENGTH\_LONG).show();

e.printStackTrace();

}

cur.moveToNext();

}

cur.close();

// \n is for new line

// Toast.makeText(getApplicationContext(), "Your Location is - \nLat: " + latitude + "\nLong: " + longitude, Toast.LENGTH\_LONG).show();

} else {

// can't get location

// GPS or Network is not enabled

// Ask user to enable GPS/network in settings

gps.showSettingsAlert();

}

}

protected void openDatabase() {

db = openOrCreateDatabase("PersonDB", Context.MODE\_PRIVATE, null);

}

public void adduser(View paramView) {

Intent i = new Intent(location\_finder.this, addUser.class);

// Starts TargetActivity

startActivity(i);

}

public void call\_emergency() {

String uri = "tel:" + "100";

Uri callUri = Uri.parse(uri);

Intent callIntent = new Intent(Intent.ACTION\_CALL, callUri);

callIntent.setFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK | Intent.FLAG\_ACTIVITY\_NO\_USER\_ACTION);

if (ActivityCompat.checkSelfPermission(this, Manifest.permission.CALL\_PHONE) != PackageManager.PERMISSION\_GRANTED) {

// TODO: Consider calling

// ActivityCompat#requestPermissions

// here to request the missing permissions, and then overriding

// public void onRequestPermissionsResult(int requestCode, String[] permissions,

// int[] grantResults)

// to handle the case where the user grants the permission. See the documentation

// for ActivityCompat#requestPermissions for more details.

return;

}

startActivity(callIntent);

}

public void call\_delete(View v) {

Intent i = new Intent(location\_finder.this, ViewPeople.class);

// Starts TargetActivity

startActivity(i);

// finish();

}

public void callambulance(View v) {

Intent callintent = new Intent(Intent.ACTION\_CALL);

callintent.setData(Uri.parse("tel:102"));

if (ActivityCompat.checkSelfPermission(this, Manifest.permission.CALL\_PHONE) != PackageManager.PERMISSION\_GRANTED) {

// TODO: Consider calling

// ActivityCompat#requestPermissions

// here to request the missing permissions, and then overriding

// public void onRequestPermissionsResult(int requestCode, String[] permissions,

// int[] grantResults)

// to handle the case where the user grants the permission. See the documentation

// for ActivityCompat#requestPermissions for more details.

return;

}

startActivity(callintent);

}

public void callpolice(View v) {

Intent callintent = new Intent(Intent.ACTION\_CALL);

callintent.setData(Uri.parse("tel:100"));

if (ActivityCompat.checkSelfPermission(this, Manifest.permission.CALL\_PHONE) != PackageManager.PERMISSION\_GRANTED) {

// TODO: Consider calling

// ActivityCompat#requestPermissions

// here to request the missing permissions, and then overriding

// public void onRequestPermissionsResult(int requestCode, String[] permissions,

// int[] grantResults)

// to handle the case where the user grants the permission. See the documentation

// for ActivityCompat#requestPermissions for more details.

return;

}

startActivity(callintent);

}

public void callwomanhelp(View v) {

Intent callintent = new Intent(Intent.ACTION\_CALL);

callintent.setData(Uri.parse("tel:181"));

if (ActivityCompat.checkSelfPermission(this, Manifest.permission.CALL\_PHONE) != PackageManager.PERMISSION\_GRANTED) {

// TODO: Consider calling

// ActivityCompat#requestPermissions

// here to request the missing permissions, and then overriding

// public void onRequestPermissionsResult(int requestCode, String[] permissions,

// int[] grantResults)

// to handle the case where the user grants the permission. See the documentation

// for ActivityCompat#requestPermissions for more details.

return;

}

startActivity(callintent);

}

public void mapstart(View v) {

Intent i = new Intent(location\_finder.this, Maps.class);

// Starts TargetActivity

startActivity(i);

// finish();

}

@Override

public void onLocationChanged(Location location) {

}

@Override

public void onStatusChanged(String provider, int status, Bundle extras) {

}

@Override

public void onProviderEnabled(String provider) {

}

@Override

public void onProviderDisabled(String provider) {

}

}

**Maps.java**

package com.example.win8.myapplication;

import android.Manifest;

import android.content.pm.PackageManager;

import android.location.Criteria;

import android.location.Location;

import android.location.LocationListener;

import android.location.LocationManager;

import android.support.v4.app.ActivityCompat;

import android.support.v4.app.FragmentActivity;

import android.os.Bundle;

import com.google.android.gms.maps.CameraUpdateFactory;

import com.google.android.gms.maps.GoogleMap;

import com.google.android.gms.maps.OnMapReadyCallback;

import com.google.android.gms.maps.SupportMapFragment;

import com.google.android.gms.maps.model.LatLng;

import com.google.android.gms.maps.model.MarkerOptions;

public class Maps extends FragmentActivity implements OnMapReadyCallback ,LocationListener{

private GoogleMap mMap;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_maps);

// Obtain the SupportMapFragment and get notified when the map is ready to be used.

SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()

.findFragmentById(R.id.map);

mapFragment.getMapAsync(this);

}

@Override

public void onMapReady(GoogleMap googleMap) {

mMap = googleMap;

if (ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_FINE\_LOCATION) != PackageManager.PERMISSION\_GRANTED && ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS\_COARSE\_LOCATION) != PackageManager.PERMISSION\_GRANTED) {

// TODO: Consider calling

// ActivityCompat#requestPermissions

// here to request the missing permissions, and then overriding

// public void onRequestPermissionsResult(int requestCode, String[] permissions,

// int[] grantResults)

// to handle the case where the user grants the permission. See the documentation

// for ActivityCompat#requestPermissions for more details.

return;

}

mMap.setMyLocationEnabled(true);

LocationManager locationManager = (LocationManager) getSystemService(LOCATION\_SERVICE);

Criteria criteria = new Criteria();

String bestProvider = locationManager.getBestProvider(criteria, true);

Location location = locationManager.getLastKnownLocation(bestProvider);

if (location != null) {

onLocationChanged(location);

}

locationManager.requestLocationUpdates(bestProvider, 20000, 0, this);

}

@Override

public void onLocationChanged(Location location) {

double latitude = location.getLatitude();

double longitude = location.getLongitude();

LatLng latLng = new LatLng(latitude, longitude);

mMap.addMarker(new MarkerOptions().position(latLng).title("I am here!"));

mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));

mMap.animateCamera(CameraUpdateFactory.zoomTo(15));

}

@Override

public void onStatusChanged(String provider, int status, Bundle extras) {

}

@Override

public void onProviderEnabled(String provider) {

}

@Override

public void onProviderDisabled(String provider) {

}

}

**ViewPeople.java**

package com.example.win8.myapplication;

import android.os.Bundle;

import android.support.v7.app.AppCompatActivity;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.app.AlertDialog;

import android.content.Context;

import android.content.DialogInterface;

import android.content.Intent;

import android.database.Cursor;

import android.database.sqlite.SQLiteDatabase;

import android.view.Menu;

import android.view.MenuItem;

import android.widget.Toast;

public class ViewPeople extends AppCompatActivity{

private EditText editTextName;

private EditText editTextAdd;

private EditText editTextId;

private Button btnPrev;

private Button btnNext;

private Button btnSave;

private Button btnDelete;

private static final String SELECT\_SQL = "SELECT \* FROM persons";

private SQLiteDatabase db;

private Cursor c;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_view\_people);

openDatabase();

editTextId = (EditText) findViewById(R.id.editTextId);

editTextName = (EditText) findViewById(R.id.editTextName);

c = db.rawQuery(SELECT\_SQL,null);

c.moveToFirst();

showRecords();

}

protected void openDatabase() {

db = openOrCreateDatabase("PersonDB", Context.MODE\_PRIVATE, null);

// db.execSQL("CREATE TABLE IF NOT EXISTS persons(id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL, name VARCHAR,phn\_no VARCHAR);");

}

protected void showRecords() {

String id = c.getString(0);

String name = c.getString(1);

String add = c.getString(2);

editTextId.setText(id);

editTextName.setText(name);

editTextAdd.setText(add);

}

protected void moveNext() {

if (!c.isLast())

c.moveToNext();

showRecords();

}

protected void movePrev() {

if (!c.isFirst())

c.moveToPrevious();

showRecords();

}

protected void saveRecord() {

String name = editTextName.getText().toString().trim();

String add = editTextAdd.getText().toString().trim();

String id = editTextId.getText().toString().trim();

String sql = "UPDATE persons SET name='" + name + "', phn\_no='" + add + "' WHERE id=" + id + ";";

if (name.equals("") || add.equals("")) {

Toast.makeText(getApplicationContext(), "You cannot save blank values", Toast.LENGTH\_LONG).show();

return;

}

db.execSQL(sql);

Toast.makeText(getApplicationContext(), "Records Saved Successfully", Toast.LENGTH\_LONG).show();

c = db.rawQuery(SELECT\_SQL, null);

c.moveToPosition(Integer.parseInt(id));

}

private void deleteRecord() {

AlertDialog.Builder alertDialogBuilder = new AlertDialog.Builder(this);

alertDialogBuilder.setMessage("Are you sure you want delete this person?");

alertDialogBuilder.setPositiveButton("Yes",

new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface arg0, int arg1) {

String id = editTextId.getText().toString().trim();

String sql = "DELETE FROM persons WHERE id=" + id + ";";

db.execSQL(sql);

Toast.makeText(getApplicationContext(), "Record Deleted", Toast.LENGTH\_LONG).show();

c = db.rawQuery(SELECT\_SQL,null);

}

});

alertDialogBuilder.setNegativeButton("No",

new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface arg0, int arg1) {

}

});

AlertDialog alertDialog = alertDialogBuilder.create();

alertDialog.show();

}

@Override

public boolean onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.menu\_view\_people, menu);

return true;

}

@Override

public boolean onOptionsItemSelected(MenuItem item) {

int id = item.getItemId();

if (id == R.id.action\_settings) {

return true;

}

return super.onOptionsItemSelected(item);

}

public void next(View v) {

moveNext();

}

public void prev(View v) {

movePrev();

}

public void save(View v) {

// if (v == btnAdd) {

saveRecord();

// }

}

public void del(View v) {

// if (v == btnAdd) {

deleteRecord();

// }

}

}

**AndroidManifest.xml**

<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

package="com.example.win8.myapplication">

<!--

The ACCESS\_COARSE/FINE\_LOCATION permissions are not required to use

Google Maps Android API v2, but you must specify either coarse or fine

location permissions for the 'MyLocation' functionality.

-->

<uses-permission android:name="android.permission.ACCESS\_FINE\_LOCATION" />

<!-- <uses-permission android:name="android.permission.ACCESS\_COARSE\_LOCATION" /> -->

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

<uses-permission android:name="android.permission.SEND\_SMS" />

<uses-permission android:name="com.javapapers.currentlocationinmap.permission.MAPS\_RECEIVE" />

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />

<uses-permission android:name="com.google.android.providers.gsf.permission.READ\_GSERVICES" />

<!-- Network State Permissions -->

<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />

<!-- <uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" /> -->

<uses-permission android:name="android.permission.CALL\_PHONE" />

<application

android:allowBackup="true"

android:icon="@mipmap/app"

android:label="@string/app\_name"

android:supportsRtl="true"

android:theme="@style/AppTheme">

<!--

The API key for Google Maps-based APIs is defined as a string resource.

(See the file "res/values/google\_maps\_api.xml").

Note that the API key is linked to the encryption key used to sign the APK.

You need a different API key for each encryption key, including the release key that is used to

sign the APK for publishing.

You can define the keys for the debug and release targets in src/debug/ and src/release/.

-->

<meta-data

android:name="com.google.android.geo.API\_KEY"

android:value="@string/google\_maps\_key" />

<activity

android:name=".location\_finder"

android:label="@string/title\_activity\_maps">

<intent-filter>

<action android:name="android.intent.action.MAIN" />

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

</activity>

<activity

android:name=".addUser"

android:label="@string/title\_activity\_add\_user"

android:theme="@style/AppTheme.NoActionBar" />

<activity

android:name=".ViewPeople"

android:label="@string/title\_activity\_view\_people"

android:theme="@style/AppTheme.NoActionBar" />

<activity

android:name=".Maps"

android:label="@string/title\_activity\_maps"></activity>

</application>

</manifest>

**activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:orientation="vertical" android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@drawable/bckgrnd"

android:weightSum="1">

<!-- <Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="HELP!"

android:id="@+id/help"

android:layout\_gravity="center\_horizontal"

android:onClick="callhelp" /> -->

<ImageButton android:id="@+id/myButton"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:background="@null"

android:onClick="callhelp"

android:layout\_marginTop="20dp"

android:layout\_marginLeft="92dp"

android:layout\_marginBottom="30dp"

android:layout\_centerVertical="true"

android:layout\_centerHorizontal="true"

android:src="@drawable/help\_button" />

<!--

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Delete"

android:id="@+id/button"

android:layout\_gravity="center\_horizontal"

android:onClick="call\_delete" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="ADD"

android:id="@+id/add"

android:layout\_gravity="center\_horizontal"

android:onClick="adduser" /> -->

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="30dp">

<ImageButton

android:layout\_width="85dp"

android:layout\_height="85dp"

android:background="@null"

android:id="@+id/imageButton"

android:onClick="callambulance"

android:layout\_marginLeft="10dp"

android:src="@drawable/download" />

<ImageButton android:id="@+id/girl"

android:layout\_width="85dp"

android:layout\_height="85dp"

android:onClick="callwomanhelp"

android:background="@null"

android:layout\_marginLeft="63dp"

android:src="@drawable/girl" />

<ImageButton android:id="@+id/police"

android:layout\_width="85dp"

android:layout\_height="85dp"

android:onClick="callpolice"

android:background="@null"

android:layout\_marginLeft="40dp"

android:src="@drawable/police" />

<RatingBar

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:id="@+id/ratingBar" />

</LinearLayout>

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Show your Location"

android:id="@+id/yourlocation"

android:onClick="mapstart"

android:layout\_marginBottom="10dp"

android:layout\_gravity="center\_horizontal"

android:nestedScrollingEnabled="false" />

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="0dp"

android:weightSum="1">

<ImageButton android:id="@+id/button"

android:layout\_width="wrap\_content"

android:layout\_height="match\_parent"

android:onClick="call\_delete"

android:background="@null"

android:src="@drawable/delimag" />

<ImageButton android:id="@+id/add"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:onClick="adduser"

android:background="@null"

android:layout\_marginLeft="199dp"

android:src="@drawable/addnew" /></LinearLayout>

</LinearLayout>

**activity\_add\_user.xml**

<?xml version="1.0" encoding="utf-8"?>

<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:fitsSystemWindows="true"

tools:context="com.example.win8.myapplication.addUser">

<android.support.design.widget.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:theme="@style/AppTheme.AppBarOverlay">

<android.support.v7.widget.AppCompatEditText

android:id="@+id/toolbar"

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

android:background="?attr/colorPrimary"

app:popupTheme="@style/AppTheme.PopupOverlay"

android:text="FEEL SAFE EVERYWHERE!!"

android:textColor="#e5f8f409"

android:textSize="@dimen/abc\_dialog\_padding\_material" />

</android.support.design.widget.AppBarLayout>

<include layout="@layout/content\_add\_user" />

<!-- <android.support.design.widget.FloatingActionButton

android:id="@+id/fab"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="bottom|end"

android:layout\_margin="@dimen/fab\_margin"

android:src="@android:drawable/ic\_dialog\_email" />-->

</android.support.design.widget.CoordinatorLayout>

**activity\_maps.xml**

<fragment xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:map="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:id="@+id/map"

android:name="com.google.android.gms.maps.SupportMapFragment"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context="com.example.win8.myapplication.Maps" />

**activity\_view\_people.xml**

<?xml version="1.0" encoding="utf-8"?>

<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:fitsSystemWindows="true"

tools:context="com.example.win8.myapplication.ViewPeople">

<android.support.design.widget.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:theme="@style/AppTheme.AppBarOverlay">

<android.support.v7.widget.Toolbar

android:id="@+id/toolbar"

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

android:background="?attr/colorPrimary"

app:popupTheme="@style/AppTheme.PopupOverlay" />

</android.support.design.widget.AppBarLayout>

<include layout="@layout/content\_view\_people" />

</android.support.design.widget.CoordinatorLayout>

**content\_add\_user.xml**

<?xml version="1.0" encoding="utf-8"?>

<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:fitsSystemWindows="true"

tools:context="com.example.win8.myapplication.ViewPeople">

<android.support.design.widget.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:theme="@style/AppTheme.AppBarOverlay">

<android.support.v7.widget.Toolbar

android:id="@+id/toolbar"

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

android:background="?attr/colorPrimary"

app:popupTheme="@style/AppTheme.PopupOverlay" />

</android.support.design.widget.AppBarLayout>

<include layout="@layout/content\_view\_people" />

</android.support.design.widget.CoordinatorLayout>

**content\_view.xml**

<?xml version="1.0" encoding="utf-8"?>

<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:fitsSystemWindows="true"

tools:context="com.example.win8.myapplication.ViewPeople">

<android.support.design.widget.AppBarLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:theme="@style/AppTheme.AppBarOverlay">

<android.support.v7.widget.Toolbar

android:id="@+id/toolbar"

android:layout\_width="match\_parent"

android:layout\_height="?attr/actionBarSize"

android:background="?attr/colorPrimary"

app:popupTheme="@style/AppTheme.PopupOverlay" />

</android.support.design.widget.AppBarLayout>

<include layout="@layout/content\_view\_people" />

</android.support.design.widget.CoordinatorLayout>