# **Appendix**

***Appendix A***

* mobile application Screen
  + Start Page Interface

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

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="#ffffff" >

<Button

android:id="@+id/button1"

android:layout\_width="250dp"

android:layout\_height="50dp"

android:background="@drawable/button\_sample1"

android:layout\_alignParentBottom="true"

android:layout\_centerHorizontal="true"

android:text="@string/ok"

android:clickable="true"

android:onClick="okay"

android:layout\_marginBottom="10dp" />

<ImageView

android:src="@drawable/ic\_launcher"

android:layout\_height="300dp"

android:layout\_width="200dp"

android:layout\_below="@+id/app\_name"

android:layout\_centerHorizontal="true" />

<TextView

android:id="@+id/app\_name"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentTop="true"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="40dp"

android:text="@string/friend\_locator"

android:fontFamily="sans-serif-condensed"

android:textAppearance="?android:attr/textAppearanceLarge"

android:textColor="#ff8f06"

android:textSize="40sp"

android:textStyle="bold"

android:typeface="monospace" />

</RelativeLayout>

* + Login Interface

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 tools:context=".Login\_page" >  
  
 <EditText  
 android:id="@+id/password"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerHorizontal="true"  
 android:layout\_centerVertical="true"  
 android:ems="10"  
 android:inputType="textPassword"  
 android:hint="@string/password" >  
  
   
 </EditText>  
  
 <EditText  
 android:id="@+id/username"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_above="@+id/password"  
 android:layout\_alignLeft="@+id/password"  
 android:layout\_alignRight="@+id/password"  
 android:ems="10"  
 android:layout\_marginBottom="5dp"  
 android:hint="@string/username" >  
  
 <requestFocus />  
 </EditText>  
  
 <Button  
 android:id="@+id/login"  
 android:layout\_width="100dp"  
 android:layout\_height="40dp"  
 android:layout\_alignLeft="@+id/password"  
 android:layout\_below="@+id/password"  
 android:layout\_marginTop="20dp"  
 android:text="@string/login"  
 android:background="#ff8f06"  
 android:clickable="true"  
 android:onClick="login" />  
  
 <Button  
 android:id="@+id/button2"  
 android:layout\_width="100dp"  
 android:layout\_height="40dp"  
 android:layout\_alignBaseline="@+id/login"  
 android:layout\_alignBottom="@+id/login"  
 android:layout\_alignRight="@+id/password"  
 android:text="@string/signup"  
 android:background="#ff8f06"  
 android:clickable="true"  
 android:onClick="signup" />  
  
</RelativeLayout>

* + Signup Interface

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent" >  
  
 <EditText  
 android:id="@+id/password"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerHorizontal="true"  
 android:layout\_centerVertical="true"  
 android:ems="10"  
 android:hint="@string/password"  
 android:inputType="textPassword" >  
  
   
 </EditText>  
  
 <EditText  
 android:id="@+id/username"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_above="@+id/password"  
 android:layout\_alignLeft="@+id/password"  
 android:layout\_alignRight="@+id/password"  
 android:ems="10"  
 android:layout\_marginBottom="5dp"  
 android:hint="@string/username" >  
  
 <requestFocus />  
 </EditText>  
  
 <EditText  
 android:id="@+id/email"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignLeft="@+id/password"  
 android:layout\_alignRight="@+id/password"  
 android:layout\_below="@+id/password"  
 android:ems="10"  
 android:hint="@string/email"  
 android:inputType="textEmailAddress"  
 android:layout\_marginTop="5dp" />  
  
 <Button  
 android:id="@+id/sign\_up"  
 android:layout\_width="100dp"  
 android:layout\_height="40dp"  
 android:layout\_below="@+id/email"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="20dp"  
 android:text="@string/signup"  
 android:background="#ff8f06"  
 android:clickable="true"  
 android:onClick="sign\_up" />  
  
</RelativeLayout>

* + Main Activity Interface

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:paddingBottom="@dimen/activity\_vertical\_margin"  
 android:paddingLeft="@dimen/activity\_horizontal\_margin"  
 android:paddingRight="@dimen/activity\_horizontal\_margin"  
 android:paddingTop="@dimen/activity\_vertical\_margin"  
 tools:context=".G\_Maps" >  
  
   
  
 <fragment  
 android:id="@+id/map"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 class="com.google.android.gms.maps.MapFragment" />  
  
</RelativeLayout>

* + Friends Page Interface

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent" >  
  
 <ListView  
 android:id="@+id/friends\_list"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentTop="true"  
 android:layout\_centerHorizontal="true"  
 android:cacheColorHint="@android:color/transparent"  
 android:background="#ff8f06" >  
 </ListView>  
  
</RelativeLayout>

* + Profile Page Interface

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="#ffffff" >  
  
 <TextView  
 android:id="@+id/user"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentTop="true"  
 android:layout\_centerHorizontal="true"  
 android:textIsSelectable="true"  
 android:layout\_marginTop="5dp"  
 android:textAppearance="?android:attr/textAppearanceLarge" />  
  
 <TextView  
 android:id="@+id/email"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@+id/user"  
 android:layout\_centerHorizontal="true"  
 android:textIsSelectable="true"  
 android:layout\_marginTop="1dp"  
 android:textAppearance="?android:attr/textAppearanceSmall" />  
  
 <EditText  
 android:id="@+id/search"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentLeft="true"  
 android:layout\_below="@+id/email"  
 android:layout\_marginLeft="10dp"  
 android:layout\_marginTop="5dp"  
 android:layout\_marginRight="5dp"  
 android:hint="@string/search"  
 android:layout\_toLeftOf="@+id/search\_button"  
 android:ems="10"  
 android:textColorHint="@android:color/black" />  
  
 <Button  
 android:id="@+id/search\_button"  
 android:layout\_width="40dp"  
 android:layout\_height="40dp"  
 android:layout\_alignBottom="@+id/search"  
 android:layout\_alignParentRight="true"  
 android:layout\_alignTop="@+id/search"  
 android:layout\_marginRight="10dp"  
 android:background="@drawable/search\_new"  
 android:clickable="true"  
 android:onClick="search\_method" />  
  
 <ListView  
 android:id="@+id/search\_list"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@+id/search"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="5dp"  
 android:background="#ff8f06"  
 android:cacheColorHint="@android:color/transparent" >  
  
 </ListView>  
  
</RelativeLayout>

* + Main Page Interface

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical" >  
  
 <TabHost  
 android:id="@android:id/tabhost"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent" >  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical" >  
  
 <TabWidget  
 android:id="@android:id/tabs"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content" >  
 </TabWidget>  
  
 <FrameLayout  
 android:id="@android:id/tabcontent"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent" >  
  
 <LinearLayout  
 android:id="@+id/tab1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
  
 <TabWidget  
 android:id="@android:id/tabs"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content" >  
 </TabWidget>  
  
 </LinearLayout>  
  
 <LinearLayout  
 android:id="@+id/tab2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent" >  
 </LinearLayout>  
  
 <LinearLayout  
 android:id="@+id/tab3"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent" >  
 </LinearLayout>  
 </FrameLayout>  
 </LinearLayout>  
 </TabHost>  
  
</LinearLayout>

* + Notification Page Interface

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent" >  
  
 <TextView  
 android:id="@+id/Noti\_text"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentTop="true"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="5dp"  
 android:text="@string/notification"  
 android:textAppearance="?android:attr/textAppearanceMedium" />  
  
 <ListView  
 android:id="@+id/list\_notification"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@+id/Noti\_text"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginTop="10dp"  
 android:cacheColorHint="@android:color/transparent"  
 android:background="#ff8f06" >  
 </ListView>  
  
</RelativeLayout>

***Appendix B***

* + Database Activity Handler

**package** com.amira.googlemaps;  
  
**import** java.util.ArrayList;  
**import** java.util.List;  
  
**import** android.content.ContentValues;  
**import** android.content.Context;  
**import** android.database.Cursor;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.database.sqlite.SQLiteOpenHelper;  
**import** android.util.Log;  
  
**public class** DatabaseHandlerActivity **extends** SQLiteOpenHelper  
{  
  
 *// All Static variables  
 // Database Version* **private static final int *DATABASE\_VERSION*** = 1;  
  
   
 *// Database Name* **private static final** String ***DATABASE\_NAME*** =**"friendslocator"**;  
  
 *// Contacts table name* **private static final** String ***TABLE\_LOCATIONS*** = **"flocation"**;  
  
 *// Contacts Table Columns names* **private static final** String ***KEY\_ID*** = **"id"**;  
 **private static final** String ***KEY\_LAT*** = **"latitude"**;  
 **private static final** String ***KEY\_LON*** = **"longitude"**;  
 **private static final** String ***KEY\_TIME*** = **"time"**;  
  
  
 **public** DatabaseHandlerActivity(Context context)  
 {  
 **super**(context, ***DATABASE\_NAME***, **null**,***DATABASE\_VERSION***);  
 *//* ***TODO Auto-generated constructor stub*** }  
  
  
 @Override  
 **public void** onCreate(SQLiteDatabase db)  
 {  
 *//* ***TODO Auto-generated method stub*** *//creating table values* String CREATE\_CONTACTS\_TABLE = **"CREATE TABLE "** + ***TABLE\_LOCATIONS*** + **"("** + ***KEY\_ID*** + **" INTEGER PRIMARY KEY,"** + ***KEY\_LAT*** + **" TEXT,"** + ***KEY\_LON*** + **" TEXT,"** + ***KEY\_TIME*** + **" TEXT"** + **")"**;  
 db.execSQL(CREATE\_CONTACTS\_TABLE);  
 }  
  
 *//Upgrading table* @Override  
 **public void** onUpgrade(SQLiteDatabase db, **int** oldVersion, **int** newVersion) {  
 *//* ***TODO Auto-generated method stub*** *// Drop older table if existed* db.execSQL(**"DROP TABLE IF EXISTS "** + ***TABLE\_LOCATIONS***);  
  
 *// Create tables again* onCreate(db);  
  
 }  
  
  
 *// Adding new contact* **public void** addLocation(Locations\_viewer location)  
 {  
 SQLiteDatabase db = **this**.getWritableDatabase();*// making writable database* ContentValues values = **new** ContentValues();  
 values.put(***KEY\_LAT***, location.getLat()); *// latitude* values.put(***KEY\_LON***, location.getLong()); *// longitude* values.put(***KEY\_TIME***, location.getTime());  
  
 *// Inserting Row* db.insert(***TABLE\_LOCATIONS***, **null**, values);  
 db.close(); *// Closing database connection* }  
  
 *// Getting single contact* **public** Locations\_viewer getLocation(**int** id)   
 {  
 SQLiteDatabase db = **this**.getReadableDatabase();  
  
 Cursor cursor = db.query(***TABLE\_LOCATIONS***, **new** String[] { ***KEY\_ID***,***KEY\_LAT***, ***KEY\_LON***,***KEY\_TIME*** }, ***KEY\_ID*** + **"=?"**,  
 **new** String[] { String.*valueOf*(id) }, **null**, **null**, **null**, **null**);  
 **if** (cursor != **null**)  
 cursor.moveToFirst();  
  
 Locations\_viewer contact = **new** Locations\_viewer(Integer.*parseInt*(cursor.getString(0)),cursor.getString(1), cursor.getString(2),cursor.getString(3));  
 *// return contact* **return** contact;  
 }  
  
 *// Getting All Contacts* **public** List<Locations\_viewer> getAllLocations() {  
 List<Locations\_viewer> contactList = **new** ArrayList<Locations\_viewer>();  
  
 *// Select All Query* String selectQuery = **"SELECT \* FROM "** + ***TABLE\_LOCATIONS***;  
  
 SQLiteDatabase db = **this**.getWritableDatabase();  
 Cursor cursor = db.rawQuery(selectQuery, **null**);  
  
 *// looping through all rows and adding to list* **if** (cursor.moveToFirst()) {  
 **do** {  
 Locations\_viewer location = **new** Locations\_viewer();  
 location.setID(Integer.*parseInt*(cursor.getString(0)));  
 location.setLAt(cursor.getString(1));  
 location.setLongi(cursor.getString(2));  
 location.setTime(cursor.getString(3));  
  
 *// Adding contact to list* contactList.add(location);  
 } **while** (cursor.moveToNext());  
 }  
  
 *// return contact list* **return** contactList;  
 }  
  
  
 *// Getting contacts Count* **public int** getLocationsCount()   
 {  
 String countQuery = **"SELECT \* FROM "** + ***TABLE\_LOCATIONS***;  
  
 SQLiteDatabase db = **this**.getReadableDatabase();*//database get readable* Cursor cursor = db.rawQuery(countQuery, **null**);  
 cursor.close();  
  
 *// return count* **return** cursor.getCount();  
 }  
  
  
 *// Updating single contact* **public int** updateLocation(Locations\_viewer locations\_viewer)  
 {  
 SQLiteDatabase db = **this**.getWritableDatabase();  
  
 ContentValues values = **new** ContentValues();  
 values.put(***KEY\_LAT***, locations\_viewer.getLat());  
 values.put(***KEY\_LON***, locations\_viewer.getLong());  
 values.put(***KEY\_TIME***, locations\_viewer.getTime());  
 Log.*d*(**"Latitude"**,locations\_viewer.getLat());  
 Log.*d*(**"Longitude"**,locations\_viewer.getLong());  
 Log.*d*(**"Time"**, locations\_viewer.getTime());  
  
 *// updating row* **return** db.update(***TABLE\_LOCATIONS***, values, ***KEY\_ID*** + **" = ?"**,**new** String[] { String.*valueOf*(locations\_viewer.getID()) });  
 }  
  
 *// Deleting single contact* **public void** deleteLocation(Locations\_viewer locations\_viewer) {SQLiteDatabase db = **this**.getWritableDatabase();  
 db.delete(***TABLE\_LOCATIONS***, ***KEY\_ID*** + **" = ?"**,**new** String[] { String.*valueOf*(locations\_viewer.getID()) });  
 db.close();  
  
  
 }  
 **public void** deleteAll()  
 {  
 SQLiteDatabase db=**this**.getWritableDatabase();  
 db.delete(***TABLE\_LOCATIONS***, **null**, **null**);  
 }  
 **public void** delete\_byID(**int** id)  
 {  
 SQLiteDatabase db=**this**.getWritableDatabase();  
 db.delete(***TABLE\_LOCATIONS***, ***KEY\_ID***+**"="**+id, **null**);  
 }  
  
  
  
}

* + First\_Page

**package** com.amira.googlemaps;  
  
**import** android.app.Activity;  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.view.View;  
  
**public class** First\_Page **extends** Activity  
{  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState)  
 {  
 *//* ***TODO Auto-generated method stub* super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***first\_page***);  
 }  
 **public void** okay(View v)  
 {  
 Intent i = **new** Intent(getApplicationContext(),Login\_page.**class**);  
 startActivity(i);  
 }  
}

* + Friends\_Page

**package** com.amira.googlemaps;  
  
**import** java.util.List;  
  
**import** com.parse.FindCallback;  
**import** com.parse.ParseException;  
**import** com.parse.ParseObject;  
**import** com.parse.ParseQuery;  
**import** com.parse.ParseUser;  
  
**import** android.app.Activity;  
**import** android.app.AlertDialog;  
**import** android.app.ProgressDialog;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.AdapterView;  
**import** android.widget.AdapterView.OnItemClickListener;  
**import** android.widget.ArrayAdapter;  
**import** android.widget.ListView;  
**import** android.widget.Toast;  
  
**public class** Friends\_page **extends** Activity  
{  
  
 ProgressDialog **myPd\_ring**;  
 ListView **friends\_list**;  
 List<ParseObject> **loc\_friend**;  
 String **lat**, **lon**;  
 String **Time**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 *//* ***TODO Auto-generated method stub* super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***friends\_page***);  
 **friends\_list** = (ListView)findViewById(R.id.***friends\_list***);  
 **final** ArrayAdapter<String> my\_adapter = **new** ArrayAdapter<String>(getApplicationContext(), android.R.layout.***simple\_list\_item\_1***);  
   
   
   
 **myPd\_ring**=ProgressDialog.*show*(Friends\_page.**this**, **"Please wait"**, **"Loading please wait.."**, **true**);  
 **myPd\_ring**.setCancelable(**true**);  
 **new** Thread(**new** Runnable() {   
 @Override  
 **public void** run() {  
 *//* ***TODO Auto-generated method stub* try** {  
 String our\_Id=ParseUser.*getCurrentUser*().getUsername();  
 ParseQuery friends =**new** ParseQuery(**"Friends"**);  
 friends.whereEqualTo(**"UserId"**, our\_Id);  
 friends.findInBackground(**new** FindCallback() {  
   
 @Override  
 **public void** done(List<ParseObject> arg0, ParseException arg1) {  
 my\_adapter.clear();  
 *//* ***TODO Auto-generated method stub* for**(ParseObject obj:arg0)  
 {  
 my\_adapter.add(obj.getString(**"FriendId"**));  
 }  
   
 my\_adapter.notifyDataSetChanged();  
 **friends\_list**.setAdapter(my\_adapter);  
 **myPd\_ring**.cancel();  
 }  
 });  
   
  
 Thread.*sleep*(5000);  
 }**catch**(Exception e){}  
  
 *// myPd\_ring.cancel();* }  
 }).start();  
   
   
 **friends\_list**.setOnItemClickListener(**new** OnItemClickListener() {  
  
 @Override  
 **public void** onItemClick(AdapterView<?> arg0, View arg1, **int** position,  
 **long** arg3) {  
 *//* ***TODO Auto-generated method stub*** String friend = **friends\_list**.getItemAtPosition(position).toString();  
 ParseQuery location = ParseUser.*getQuery*();  
 location.whereEqualTo(**"username"**, friend);  
 **try** {  
 **loc\_friend**=location.find();  
 } **catch** (ParseException e) {  
 *//* ***TODO Auto-generated catch block*** e.printStackTrace();  
 }  
   
 **for**(ParseObject obj:**loc\_friend**)  
 {  
 **lat**=obj.getString(**"Latitude"**);  
 **lon**=obj.getString(**"Longitude"**);  
 **Time** = obj.getString(**"Time"**);  
   
 }  
*// DateFormat df = new SimpleDateFormat("MM/dd/yyyy HH:mm");  
// String new\_Time = df.format(Time);* Toast.*makeText*(getApplicationContext(), **"Time ="**+**Time**, Toast.***LENGTH\_SHORT***).show();  
   
 Intent i = **new** Intent(Friends\_page.**this**,G\_Maps\_Friends.**class**);  
 i.putExtra(**"lat"**, **lat**);  
 i.putExtra(**"lon"**, **lon**);  
 i.putExtra(**"time"**, **Time**);  
 startActivity(i);  
   
   
 }  
 });  
 }  
   
 @Override  
 **public void** onBackPressed() {  
 *//* ***TODO Auto-generated method stub*** *// super.onBackPressed();* **new** AlertDialog.Builder(Friends\_page.**this**)  
 .setTitle(**"Are you sure,"**)  
 .setMessage(**"Want to exit?"**)  
 .setNegativeButton(**"No"**, **new** DialogInterface.OnClickListener() {  
  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 *//* ***TODO Auto-generated method stub*** dialog.dismiss();  
 }  
 })  
 .setPositiveButton(**"Yes"**, **new** DialogInterface.OnClickListener() {  
  
 @Override  
 **public void** onClick(DialogInterface dialog\_main, **int** which) {  
 *//* ***TODO Auto-generated method stub*** dialog\_main.dismiss();  
 finish();  
 }  
 }).show();  
 }  
}

* + GMaps

**package** com.amira.googlemaps;  
  
**import** java.util.List;  
  
**import** android.os.Build;  
**import** android.os.Bundle;  
  
**import** com.google.android.gms.maps.CameraUpdateFactory;  
**import** com.google.android.gms.maps.GoogleMap;  
**import** com.google.android.gms.maps.MapFragment;  
**import** com.google.android.gms.maps.model.LatLng;  
**import** com.google.android.gms.maps.model.MarkerOptions;  
**import** android.annotation.TargetApi;  
**import** android.app.Activity;  
**import** android.view.Menu;  
  
@TargetApi(Build.VERSION\_CODES.***HONEYCOMB***)  
**public class** G\_Maps **extends** Activity {  
  
 **private** GoogleMap **mMap**;  
 Double **latitude**, **longitude**;  
 LatLng **CIU**;  
 List<Locations\_viewer> **location\_list**;  
   
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
   
   
 MapFragment mpFragment= ((MapFragment) getFragmentManager().findFragmentById(R.id.***map***));  
 **mMap**=mpFragment.getMap();  
 **mMap**.setMapType(GoogleMap.***MAP\_TYPE\_HYBRID***);  
   
 **final** DatabaseHandlerActivity db = **new** DatabaseHandlerActivity(getApplicationContext());  
 **location\_list** = db.getAllLocations();  
 db.close();  
   
 **if**(**location\_list**.isEmpty())  
 {  
*// Toast.makeText(getApplicationContext(), "Pre-Location", Toast.LENGTH\_SHORT).show();  
// Calendar c = Calendar.getInstance();  
// SimpleDateFormat df = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss");  
// String formattedDate = df.format(c.getTime());* **CIU** = **new** LatLng(10.523067500000000000, 76.222210599999920000);  
 MapFragment mpFragment2= ((MapFragment) getFragmentManager().findFragmentById(R.id.***map***));  
 **mMap**=mpFragment2.getMap();  
 **mMap**.setMapType(GoogleMap.***MAP\_TYPE\_HYBRID***);  
 **mMap**.addMarker(**new** MarkerOptions()  
 .position(**CIU**).title(**"Location"**).snippet(**"Latitude=10.52306750, Longitude=76.22221059999992"**));  
   
   
 }  
 **else**{  
   
 **for**(Locations\_viewer locations\_viewer:**location\_list**)  
 {  
 **latitude** = Double.*parseDouble*(locations\_viewer.getLat());  
 **longitude**= Double.*parseDouble*(locations\_viewer.getLong());  
 **CIU** = **new** LatLng(**latitude**, **longitude**);  
 **mMap**.addMarker(**new** MarkerOptions()  
 .position(**CIU**).title(**"Location"**).snippet(**"Latitude="**+locations\_viewer.getLat()+**", Longitude="**+locations\_viewer.getLong()));  
 }  
 }  
 **mMap**.moveCamera(CameraUpdateFactory.*newLatLngZoom*(**CIU**, 10));  
 **mMap**.animateCamera(CameraUpdateFactory.*zoomTo*(10),2000,**null**);  
 }  
  
 @Override  
 **public boolean** onCreateOptionsMenu(Menu menu) {  
 *// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.***main***, menu);  
 **return true**;  
 }  
  
}

* + GMaps Friends

**package** com.amira.googlemaps;  
  
**import** android.os.Build;  
**import** android.os.Bundle;  
  
**import** com.google.android.gms.maps.CameraUpdateFactory;  
**import** com.google.android.gms.maps.GoogleMap;  
**import** com.google.android.gms.maps.MapFragment;  
**import** com.google.android.gms.maps.model.LatLng;  
**import** com.google.android.gms.maps.model.MarkerOptions;  
**import** android.annotation.TargetApi;  
**import** android.app.Activity;  
  
@TargetApi(Build.VERSION\_CODES.***HONEYCOMB***)  
**public class** G\_Maps\_Friends **extends** Activity {  
  
 **private** GoogleMap **mMap**;  
 Double **latitude**, **longitude**;  
 String **Time**;  
 LatLng **CIU**;  
   
   
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
   
 String lat = getIntent().getStringExtra(**"lat"**);  
 String lon = getIntent().getStringExtra(**"lon"**);  
 **Time** = getIntent().getStringExtra(**"time"**);  
 **latitude** = Double.*parseDouble*(lat);  
 **longitude** = Double.*parseDouble*(lon);  
   
   
   
 MapFragment mpFragment= ((MapFragment) getFragmentManager().findFragmentById(R.id.***map***));  
 **mMap**=mpFragment.getMap();  
 **mMap**.setMapType(GoogleMap.***MAP\_TYPE\_HYBRID***);  
 **CIU** = **new** LatLng(**latitude**, **longitude**);  
 MapFragment mpFragment2= ((MapFragment) getFragmentManager().findFragmentById(R.id.***map***));  
 **mMap**=mpFragment2.getMap();  
 **mMap**.setMapType(GoogleMap.***MAP\_TYPE\_HYBRID***);  
   
 **mMap**.addMarker(**new** MarkerOptions()  
 .position(**CIU**).title(**"Location"**).snippet(**"Latitude="**+lat+**" Longitude="**+lon));  
   
   
  
   
 **mMap**.moveCamera(CameraUpdateFactory.*newLatLngZoom*(**CIU**, 10));  
 **mMap**.animateCamera(CameraUpdateFactory.*zoomTo*(10),2000,**null**);  
 }  
  
  
  
}

* + Location Viewer

**package** com.amira.googlemaps;  
  
**public class** Locations\_viewer   
{  
  
 **int id**;  
 String **Latitude**;  
 String **Longitude**;  
 String **Time**;  
   
 *//empty constructor* **public** Locations\_viewer()  
 {  
   
 }  
   
 **public** Locations\_viewer(**int** \_id,String lat,String longi,String time)  
 {  
 **this**.**id**=\_id;  
 **this**.**Latitude**=lat;  
 **this**.**Longitude**=longi;  
 **this**.**Time**=time;  
 }  
   
 **public** Locations\_viewer(String lat,String longi,String time)  
 {  
 **this**.**Latitude**=lat;  
 **this**.**Longitude**=longi;  
 **this**.**Time**=time;  
 }  
   
 **public int** getID(){  
 **return this**.**id**;  
 }  
  
 *// setting id* **public void** setID(**int** id){  
 **this**.**id** = id;  
 }  
  
 *// getting name* **public** String getLat(){  
 **return this**.**Latitude**;  
 }  
  
 *// setting lat* **public void** setLAt(String lat)  
 {  
 **this**.**Latitude** = lat;  
 }  
  
 *// getting long* **public** String getLong(){  
 **return this**.**Longitude**;  
 }  
  
 *// setting phone number* **public void** setLongi(String Longi){  
 **this**.**Longitude** = Longi;  
 }  
 *// getting time* **public** String getTime(){  
 **return this**.**Time**;  
 }  
  
 *// setting time* **public void** setTime(String time){  
 **this**.**Time** = time;  
 }  
}

* + Login\_Page

**package** com.amira.googlemaps;  
  
**import** com.parse.LogInCallback;  
**import** com.parse.ParseException;  
**import** com.parse.ParseUser;  
  
**import** android.os.Bundle;  
**import** android.app.Activity;  
**import** android.app.AlertDialog;  
**import** android.app.ProgressDialog;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.view.View;  
**import** android.widget.EditText;  
**import** android.widget.Toast;  
  
**public class** Login\_page **extends** Activity  
{  
  
 EditText **userField**, **passField**;  
 String **username**, **password**;  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState)  
 {  
 **super**.onCreate(savedInstanceState);  
  
 ParseUser currentUser = ParseUser.*getCurrentUser*();  
 **if** (currentUser != **null**) {  
 *// do stuff with the user* Intent i=**new** Intent(getApplicationContext(),MainPage.**class**);  
 startActivity(i);  
 }  
 **else** {  
 setContentView(R.layout.***login\_page***);  
 **userField** = (EditText)findViewById(R.id.***username***);  
 **passField** = (EditText)findViewById(R.id.***password***);  
 }  
  
  
 }  
  
  
   
 **public void** login(View v)  
 {  
  
  
  
 **username** = **userField**.getText().toString().trim();  
 **password** = **passField**.getText().toString().trim();  
  
 **final** ProgressDialog mDialog = **new** ProgressDialog(**this**);  
 mDialog.setMessage(**"Login Progress.."**);  
  
 mDialog.show();  
  
 ParseUser.*logInInBackground*(**username**, **password**,  
 **new** LogInCallback() {  
  
 @Override  
 **public void** done(ParseUser user, ParseException e) {  
 **if** (user != **null**) {  
 *// Hooray! The user is logged in.* mDialog.cancel();  
 Toast toast = Toast.*makeText*(  
 getApplicationContext(),  
 **"Successfull Login! "**,  
 Toast.***LENGTH\_LONG***);  
 toast.show();  
  
 Intent signupScreen = **new** Intent(getApplicationContext(),  
 MainPage.**class**);  
 startActivity(signupScreen);  
  
  
  
 } **else** {  
  
 mDialog.cancel();  
 *// Signup failed. Look at the ParseException  
 // to see what happened.* AlertDialog.Builder builder = **new** AlertDialog.Builder(Login\_page.**this**);  
 builder.setMessage(e.getMessage())  
 .setCancelable(**false**)  
  
 .setNegativeButton(**"Ok!"**, **new** DialogInterface.OnClickListener() {  
 **public void** onClick(DialogInterface dialog, **int** id) {  
 dialog.cancel();  
 }  
 });  
 AlertDialog alert = builder.create();  
 alert.show();  
  
 }  
  
 }  
  
  
 });  
  
  
  
 }  
 **public void** signup(View v)  
 {  
 Intent i = **new** Intent(getApplicationContext(),SignUp.**class**);  
 startActivity(i);  
 }  
}

* + Main\_Page

**package** com.amira.googlemaps;  
  
**import** java.text.DateFormat;  
**import** java.text.SimpleDateFormat;  
**import** java.util.Calendar;  
**import** java.util.Date;  
  
**import** com.parse.ParseUser;  
  
**import** android.app.TabActivity;  
**import** android.content.Context;  
**import** android.content.Intent;  
**import** android.content.res.Resources;  
**import** android.location.Location;  
**import** android.location.LocationListener;  
**import** android.location.LocationManager;  
**import** android.os.Bundle;  
**import** android.view.Menu;  
**import** android.view.MenuItem;  
**import** android.widget.TabHost;  
**import** android.widget.Toast;  
  
@SuppressWarnings(**"deprecation"**)  
**public class** MainPage **extends** TabActivity   
{  
  
 LocationManager **locmngr**;  
 Location **our\_location**;  
 Double **L1** = 10.000886500000000000; *// setting an initial latitude.* Double **L2** = 76.299572799999960000;  
 LocationListener **loclis**;  
   
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 *//* ***TODO Auto-generated method stub* super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***mainpage***);  
  
 Resources re=getResources();  
 TabHost th=getTabHost();  
 TabHost.TabSpec spec;  
 Intent i;  
  
   
   
 **locmngr**=(LocationManager)getSystemService(Context.***LOCATION\_SERVICE***);  
 **loclis** =**new** MyLocationListner();  
 **our\_location**=**locmngr**.getLastKnownLocation(LocationManager.***NETWORK\_PROVIDER***);  
 **locmngr**.requestLocationUpdates( LocationManager.***NETWORK\_PROVIDER***, 0, 0,**loclis**);  
   
   
   
  
 i=**new** Intent(**this**,Profile.**class**).addFlags(Intent.***FLAG\_ACTIVITY\_CLEAR\_TOP***);  
  
 spec = th.newTabSpec(**"tab1"**).setIndicator(**""**,re.getDrawable(R.drawable.***users\_newone***)).setContent(i);  
 th.addTab(spec);  
  
 i=**new** Intent(**this**,Friends\_page.**class**).addFlags(Intent.***FLAG\_ACTIVITY\_CLEAR\_TOP***);  
  
 spec = th.newTabSpec(**"tab2"**).setIndicator(**""**,re.getDrawable(R.drawable.***friend\_list\_new***)).setContent(i);  
 th.addTab(spec);  
   
 i=**new** Intent(**this**,Notification\_new.**class**).addFlags(Intent.***FLAG\_ACTIVITY\_CLEAR\_TOP***);  
  
 spec = th.newTabSpec(**"tab3"**).setIndicator(**""**,re.getDrawable(R.drawable.***notification*** )).setContent(i);  
 th.addTab(spec);  
  
 th.setCurrentTab(0);  
  
 }  
  
  
  
  
 @Override  
 **public boolean** onCreateOptionsMenu(Menu menu)   
 {  
 *// Inflate the menu; this adds items to the action bar if it is present.* getMenuInflater().inflate(R.menu.***new\_menu***, menu);  
 **return true**;  
 }  
 @Override  
 **public boolean** onOptionsItemSelected(MenuItem item) {  
 *//* ***TODO Auto-generated method stub* switch**(item.getItemId())  
 {  
 **case** R.id.***logout\_menu***:  
 {  
 ParseUser.*logOut*();  
 Toast.*makeText*(**this**, **"Please Log In, to continue"**, Toast.***LENGTH\_SHORT***).show();  
 Intent i =**new** Intent(getApplicationContext(),Login\_page.**class**);  
 startActivity(i);  
 MainPage.**this**.finish();  
 **break**;  
 }  
  
 **case** R.id.***history\_menu***:  
 Intent i = **new** Intent(**this**,G\_Maps.**class**);  
 startActivity(i);  
 **break**;  
  
 }  
 **return true**;  
 }  
   
   
 **public class** MyLocationListner **implements** LocationListener{  
  
 @Override  
 **public void** onLocationChanged(Location location)  
 {  
 *//* ***TODO Auto-generated method stub* L1**=location.getLatitude();  
 **L2**=location.getLongitude();  
 String currentDateTimeString = DateFormat.*getDateTimeInstance*().format(**new** Date());  
 *// Toast.makeText(getApplicationContext(), currentDateTimeString, Toast.LENGTH\_SHORT).show();* DatabaseHandlerActivity db = **new** DatabaseHandlerActivity(getApplicationContext());  
 db.addLocation(**new** Locations\_viewer(**L1**.toString(), **L2**.toString(), currentDateTimeString));  
  
 Toast.*makeText*(getApplicationContext(), **"New Location "**+**"Latitude ="**+**L1**+  
 **" Longitude ="**+**L2**, Toast.***LENGTH\_LONG***).show();  
  
   
 Calendar c = Calendar.*getInstance*();  
 SimpleDateFormat df = **new** SimpleDateFormat(**"yyyy-MM-dd HH:mm:ss"**);  
 String formattedDate = df.format(c.getTime());  
   
   
 ParseUser.*getCurrentUser*().put(**"Latitude"**, **L1**.toString());  
 ParseUser.*getCurrentUser*().put(**"Longitude"**, **L2**.toString());  
   
 ParseUser.*getCurrentUser*().put(**"Time"**, formattedDate);  
   
   
 ParseUser.*getCurrentUser*().saveInBackground();  
  
 *// Toast.makeText(getApplicationContext(), "Latitude ="+L1, Toast.LENGTH\_LONG).show();  
 // Toast.makeText(getApplicationContext(), "Longitude ="+L2, Toast.LENGTH\_LONG).show();* }  
  
 @Override  
 **public void** onProviderDisabled(String provider) {  
 *//* ***TODO Auto-generated method stub*** }  
  
 @Override  
 **public void** onProviderEnabled(String provider) {  
 *//* ***TODO Auto-generated method stub*** }  
  
 @Override  
 **public void** onStatusChanged(String provider, **int** status, Bundle extras) {  
 *//* ***TODO Auto-generated method stub*** }  
 }  
   
  
}

* + Notification

**package** com.amira.googlemaps;  
  
**import** java.util.ArrayList;  
**import** java.util.List;  
  
**import** com.parse.FindCallback;  
**import** com.parse.ParseException;  
**import** com.parse.ParseObject;  
**import** com.parse.ParseQuery;  
**import** com.parse.ParseUser;  
  
**import** android.app.Activity;  
**import** android.app.AlertDialog;  
**import** android.app.ProgressDialog;  
**import** android.content.DialogInterface;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.AdapterView;  
**import** android.widget.AdapterView.OnItemClickListener;  
**import** android.widget.ArrayAdapter;  
**import** android.widget.ListView;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
  
**public class** Notification\_new **extends** Activity {  
  
 ArrayAdapter<String> **noti\_adapter**;  
 ProgressDialog **myPd\_ring**;  
 ListView **list\_notification**;  
 TextView **Noti\_text**;  
 ArrayList<ParseObject> **obj\_notification** = **new** ArrayList<ParseObject>();  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 *//* ***TODO Auto-generated method stub* super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***notification***);  
 **noti\_adapter** = **new** ArrayAdapter<String>(getApplicationContext(), android.R.layout.***simple\_list\_item\_1***);  
 **list\_notification** = (ListView)findViewById(R.id.***list\_notification***);  
 **Noti\_text** = (TextView)findViewById(R.id.***Noti\_text***);  
  
 **myPd\_ring**=ProgressDialog.*show*(Notification\_new.**this**, **"Please wait"**, **"Loading please wait.."**, **true**);  
 **myPd\_ring**.setCancelable(**true**);  
 **new** Thread(**new** Runnable() {  
  
 @Override  
 **public void** run() {  
 *//* ***TODO Auto-generated method stub* try** {  
 ParseQuery myquery = **new** ParseQuery(**"Request"**);  
 myquery.whereEqualTo(**"UserId"**, ParseUser.*getCurrentUser*().getUsername());  
 myquery.findInBackground(**new** FindCallback() {  
  
 @Override  
 **public void** done(List<ParseObject> arg0, ParseException arg1) {  
 *//* ***TODO Auto-generated method stub* for**(ParseObject obj:arg0)  
 {  
 **noti\_adapter**.add(obj.getString(**"RequestingId"**));  
 **obj\_notification**.add(obj);  
 }  
  
 **noti\_adapter**.notifyDataSetChanged();  
  
 **if**(**noti\_adapter**.isEmpty())  
 {  
 **Noti\_text**.setText(**"No Notifications!"**);  
 }  
  
 **list\_notification**.setAdapter(**noti\_adapter**);  
  
  
 **myPd\_ring**.dismiss();  
  
 }  
 });  
  
 Thread.*sleep*(5000);  
 } **catch** (InterruptedException e) {  
 *//* ***TODO Auto-generated catch block*** e.printStackTrace();  
 }  
  
 }  
  
 }).start();  
  
  
  
  
  
 **list\_notification**.setOnItemClickListener(**new** OnItemClickListener() {  
  
 @Override  
 **public void** onItemClick(AdapterView<?> arg0, View arg1, **final int** position, **long** arg3)  
 {  
 *//* ***TODO Auto-generated method stub*** AlertDialog.Builder builder = **new** AlertDialog.Builder(Notification\_new.**this**);  
 builder.setMessage(**"Would you like to add "**+**noti\_adapter**.getItem(position)+**" as your friend?"**)  
 .setCancelable(**false**)  
  
 .setPositiveButton(**"Yes"**, **new** DialogInterface.OnClickListener() {  
   
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 *//* ***TODO Auto-generated method stub***String our\_Name = ParseUser.*getCurrentUser*().getUsername();  
 ParseObject Request = **new** ParseObject(**"Friends"**);  
 Request.put(**"UserId"**,our\_Name);  
 Request.put(**"FriendId"**, **noti\_adapter**.getItem(position));  
 Request.saveEventually();  
 ParseObject Request\_relation = **new** ParseObject(**"Friends"**);  
 Request\_relation.put(**"UserId"**, **noti\_adapter**.getItem(position));  
 Request\_relation.put(**"FriendId"**, our\_Name);  
 Request\_relation.saveInBackground();  
 Toast.*makeText*(getApplicationContext(), **"You are now friends with "**+**noti\_adapter**.getItem(position), Toast.***LENGTH\_SHORT***).show();  
   
   
 }  
 })  
 .setNegativeButton(**"No"**, **new** DialogInterface.OnClickListener() {  
 **public void** onClick(DialogInterface dialog, **int** id) {  
 dialog.cancel();  
 }  
   
 });  
 AlertDialog alert = builder.create();  
 alert.show();  
   
 **obj\_notification**.get(position).deleteEventually();  
   
 }  
 });  
  
  
  
 }  
  
}

* + Parse Aplication

**package** com.amira.googlemaps;  
  
**import** com.parse.Parse;  
**import** android.app.Application;  
  
**public class** ParseApplication **extends** Application {  
 @Override  
 **public void** onCreate() {  
 *//* ***TODO Auto-generated method stub* super**.onCreate();  
 Parse.*initialize*(**this**, **"cOF2YCY3vHAJeG0Y2fkVDxW4TXXcqjpkZfXCKnU5"**,  
 **"We0vjt4xAh2MSGy3Xxofq0q3CpnqXpF8QNSDpCtr"**);  
 }  
  
}

* + Profile

**package** com.amira.googlemaps;  
  
**import** java.util.ArrayList;  
**import** java.util.List;  
  
**import** com.parse.FindCallback;  
**import** com.parse.ParseException;  
**import** com.parse.ParseObject;  
**import** com.parse.ParseQuery;  
**import** com.parse.ParseUser;  
  
**import** android.app.Activity;  
**import** android.app.AlertDialog;  
**import** android.app.ProgressDialog;  
**import** android.content.DialogInterface;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.AdapterView;  
**import** android.widget.AdapterView.OnItemClickListener;  
**import** android.widget.ArrayAdapter;  
**import** android.widget.EditText;  
**import** android.widget.ListView;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**public class** Profile **extends** Activity   
{  
  
 Boolean **req\_status**=**false**;  
 Boolean **noti\_status**=**false**;  
 String username, **email**;  
 TextView **name**, **mail**;  
 EditText **search**;  
 ListView **SearchList**;  
 ArrayAdapter<String> **Adapter**;  
 *// ArrayList<String> friends\_Id = new ArrayList<String>();* ProgressDialog **myPd\_ring**;  
**volatile** Boolean **check\_delay**=**true**;  
  
  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 *//* ***TODO Auto-generated method stub* super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***profile\_page***);  
  
 **search** = (EditText)findViewById(R.id.***search***);  
 **SearchList**=(ListView)findViewById(R.id.***search\_list***);  
  
 **name** = (TextView)findViewById(R.id.***user***);  
 **mail** = (TextView)findViewById(R.id.***email***);  
  
 username = ParseUser.*getCurrentUser*().getUsername();  
 **email** = ParseUser.*getCurrentUser*().getEmail();  
  
  
  
 **name**.setText(username);  
 **mail**.setText(**email**);}  
 **public void** search\_method(View v)  
 {  
  
  
 **Adapter**=**new** ArrayAdapter<String>(getApplicationContext(), android.R.layout.***simple\_list\_item\_1***);  
 **if**(**Adapter**.isEmpty()==**false**)  
 **Adapter**.clear();  
 **final** String search\_item=**search**.getText().toString().trim();  
  
  
  
 **myPd\_ring**=ProgressDialog.*show*(Profile.**this**, **"Please wait"**, **"Loading please wait.."**, **true**);  
 **myPd\_ring**.setCancelable(**true**);  
 **new** Thread(**new** Runnable() {   
 @Override  
 **public void** run() {  
 *//* ***TODO Auto-generated method stub* try** {  
 ParseQuery match =ParseUser.*getQuery*();  
 match.whereEqualTo(**"username"**, search\_item);  
 match.findInBackground(**new** FindCallback() {  
 **public void** done(List<ParseObject> objects, ParseException e) {  
 **if** (e == **null**) {  
 *// The query was successful.* **for**(ParseObject i: objects){  
  
 **Adapter**.add(i.getString(**"username"**));  
 }  
  
 set\_search\_list();  
 } **else** {  
 *// Something went wrong.* }  
 }  
  
  
 });   
  
 Thread.*sleep*(5000);  
 }**catch**(Exception e){}  
  
 *// myPd\_ring.cancel();* }  
 }).start();  
  
  
 }  
  
 **public void** set\_search\_list()  
 {  
  
  
  
  
 **if**(**Adapter**.isEmpty()){  
  
 **myPd\_ring**.cancel();  
 AlertDialog.Builder builder = **new** AlertDialog.Builder(Profile.**this**);  
 builder.setMessage(**"No Results Found"**)  
 .setCancelable(**false**)  
  
 .setNegativeButton(**"Ok!"**, **new** DialogInterface.OnClickListener() {  
 **public void** onClick(DialogInterface dialog, **int** id) {  
 dialog.cancel();  
 }  
 });  
 AlertDialog alert = builder.create();  
 alert.show();  
  
 }  
  
 **myPd\_ring**.cancel();  
 **Adapter**.notifyDataSetChanged();  
  
  
 **SearchList**.setAdapter(**Adapter**);  
  
 **SearchList**.setOnItemClickListener(**new** OnItemClickListener() {  
  
 @Override  
 **public void** onItemClick(AdapterView<?> arg0, View arg1, **int** position,  
 **long** arg3) {  
 *//* ***TODO Auto-generated method stub*** String our\_name = ParseUser.*getCurrentUser*().getUsername();  
  
  
  
 *//Checking whether friend is already added* Check\_friend(position,our\_name);}   
  
  
 });  
 }  
  
  
  
 **public** Boolean Check\_notification(**int** position,String our\_name)  
 {  
  
 ParseQuery noti\_1 = **new** ParseQuery(**"Request"**);  
 noti\_1.whereEqualTo(**"UserId"**, **Adapter**.getItem(position));  
 noti\_1.whereEqualTo(**"RequstingId"**, our\_name);  
  
  
 ParseQuery noti\_2 = **new** ParseQuery(**"Request"**);  
 noti\_2.whereEqualTo(**"UserId"**, our\_name);  
 noti\_2.whereEqualTo(**"RequstingId"**, **Adapter**.getItem(position));  
  
 List<ParseQuery> queries = **new** ArrayList<ParseQuery>();  
 queries.add(noti\_1);  
 queries.add(noti\_2);  
  
 ParseQuery mainQuery = ParseQuery.*or*(queries);  
 mainQuery.findInBackground(**new** FindCallback() {  
 **public void** done(List<ParseObject> results, ParseException e) {  
  
  
  
 **noti\_status**=results.isEmpty();*//true if empty* }  
  
 });  
  
  
 **return noti\_status**;  
 }  
  
  
 **public void** Check\_friend(**final int** position, **final** String our\_name)  
 {  
 **final** ProgressDialog obj\_dlg=**new** ProgressDialog(Profile.**this**);  
 obj\_dlg.setMessage(**"Please wait"**);  
 obj\_dlg.setCancelable(**true**);  
 obj\_dlg.show();  
 **new** Thread(**new** Runnable() {  
  
 @Override  
 **public void** run() {  
 *//* ***TODO Auto-generated method stub*** ParseQuery friend\_query = **new** ParseQuery(**"Friends"**);  
  
 friend\_query.whereEqualTo(**"UserId"**, our\_name);  
 friend\_query.whereEqualTo(**"FriendId"**, **Adapter**.getItem(position));  
 friend\_query.findInBackground(**new** FindCallback() {  
  
 @Override  
 **public void** done(List<ParseObject> friends, ParseException arg1) {  
 *//* ***TODO Auto-generated method stub* req\_status**=friends.isEmpty(); *//true if empty  
// Toast.makeText(getApplicationContext(), "friends status= "+req\_status.toString(), Toast.LENGTH\_LONG).show();* **if**(**req\_status**==**false**)  
 {  
 obj\_dlg.cancel();  
 AlertDialog.Builder build = **new** AlertDialog.Builder(Profile.**this**);  
 build.setMessage(**"You are already friends with "**+**Adapter**.getItem(position)+**"!"**)  
 .setCancelable(**false**)  
 .setPositiveButton(**"Ok"**, **new** DialogInterface.OnClickListener() {  
   
 @Override  
 **public void** onClick(DialogInterface dia, **int** which) {  
 *//* ***TODO Auto-generated method stub*** dia.dismiss();  
 }  
 }).show();  
 }  
   
 **else**{  
 obj\_dlg.cancel();  
 send\_request(position);  
 }  
   
   
  
 }  
   
   
   
   
   
   
 });  
  
  
 }  
 }).start();  
  
  
  
  
 }  
  
  
 **public void** send\_request(**final int** position)  
 {  
 AlertDialog.Builder builder = **new** AlertDialog.Builder(Profile.**this**);  
 builder.setMessage(**"Do you want to add "**+**Adapter**.getItem(position)+**" as your friend?"**)  
 .setCancelable(**false**)  
  
 .setPositiveButton(**"Yes"**, **new** DialogInterface.OnClickListener() {  
  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 *//* ***TODO Auto-generated method stub*** String our\_Name = ParseUser.*getCurrentUser*().getUsername();   
 ParseObject Request = **new** ParseObject(**"Request"**);  
  
  
 Request.put(**"UserId"**,**Adapter**.getItem(position));  
 Request.put(**"RequestingId"**, our\_Name);  
 Request.saveInBackground();  
 Toast.*makeText*(getApplicationContext(), **"Request sent"**, Toast.***LENGTH\_SHORT***).show();  
  
  
 }  
 })  
 .setNegativeButton(**"No"**, **new** DialogInterface.OnClickListener() {  
 **public void** onClick(DialogInterface dialog, **int** id) {  
 dialog.cancel();  
 }  
  
 });  
 AlertDialog alert = builder.create();  
 alert.show();  
 }  
  
  
 @Override  
 **public void** onBackPressed() {  
 *//* ***TODO Auto-generated method stub*** *// super.onBackPressed();* **new** AlertDialog.Builder(Profile.**this**)  
 .setTitle(**"Are you sure,"**)  
 .setMessage(**"Want to exit?"**)  
 .setNegativeButton(**"No"**, **new** DialogInterface.OnClickListener() {  
  
 @Override  
 **public void** onClick(DialogInterface dialog, **int** which) {  
 *//* ***TODO Auto-generated method stub*** dialog.dismiss();  
 }  
 })  
 .setPositiveButton(**"Yes"**, **new** DialogInterface.OnClickListener() {  
  
 @Override  
 **public void** onClick(DialogInterface dialog\_main, **int** which) {  
 *//* ***TODO Auto-generated method stub*** dialog\_main.dismiss();  
 finish();  
 }  
 }).show();  
 }}

* + Sign\_up

**package** com.amira.googlemaps;  
  
**import** java.text.SimpleDateFormat;  
**import** java.util.Calendar;  
  
**import** com.parse.ParseException;  
**import** com.parse.ParseUser;  
**import** com.parse.SignUpCallback;  
  
**import** android.app.Activity;  
**import** android.app.AlertDialog;  
**import** android.app.ProgressDialog;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.view.View;  
**import** android.widget.EditText;  
**import** android.widget.Toast;  
  
**public class** SignUp **extends** Activity  
{  
 EditText **usernameField**, **passwordField**, **emailField**;  
 String **username**, **password**, **email**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState)  
 {  
 *//* ***TODO Auto-generated method stub* super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***signup\_page***);  
 **usernameField** = (EditText)findViewById(R.id.***username***);  
 **passwordField** = (EditText)findViewById(R.id.***password***);  
 **emailField** = (EditText)findViewById(R.id.***email***);  
 }  
 **public void** sign\_up(View v)  
 {  
**if** (**usernameField**.getText().length()>0) {  
 **if** (**passwordField**.getText().length()>0) {  
 **if** (**emailField**.getText().length()>0) {  
   
   
 **username** = **usernameField**.getText().toString().trim();  
 **password** = **passwordField**.getText().toString().trim();  
 **email** = **emailField**.getText().toString().trim();  
   
   
 ParseUser user = **new** ParseUser();  
 user.setUsername(**username**);  
 user.setPassword(**password**);  
 user.setEmail(**email**);  
 user.put(**"Latitude"**, **"10.523067500000000000"**);  
 user.put(**"Longitude"**, **"76.222210599999920000"**);  
   
   
 Calendar c = Calendar.*getInstance*();  
 SimpleDateFormat df = **new** SimpleDateFormat(**"yyyy-MM-dd HH:mm:ss"**);  
 String formattedDate = df.format(c.getTime());  
   
   
 user.put(**"Time"**, formattedDate);  
   
*// user.put("SimSerialNumber", simSerialNumber);  
// user.put("UniqueNumber", uniqueNumber);* **final** ProgressDialog mDialog = **new** ProgressDialog(  
 SignUp.**this**);  
 mDialog.setMessage(**"Signing Up.."**);  
  
 mDialog.show();  
  
 user.signUpInBackground(**new** SignUpCallback() {  
 **public void** done(ParseException e) {  
  
 mDialog.cancel();  
  
 **if** (e == **null**) {  
 Toast toast = Toast.*makeText*(  
 getApplicationContext(),  
 **"Successful signUp! "**,  
 Toast.***LENGTH\_LONG***);  
 toast.show();  
 Intent signupScreen = **new** Intent(  
 getApplicationContext(),  
 Login\_page.**class**);  
 startActivity(signupScreen);  
 *// Hooray! Let them use the app now.* } **else** {  
  
 AlertDialog.Builder builder = **new** AlertDialog.Builder(  
 SignUp.**this**);  
 builder.setMessage(e.getMessage())  
 .setCancelable(**false**)  
  
 .setNegativeButton(  
 **"Ok!"**,  
 **new** DialogInterface.OnClickListener() {  
 **public void** onClick(  
 DialogInterface dialog,  
 **int** id) {  
 dialog.cancel();  
 }  
 });  
 AlertDialog alert = builder.create();  
 alert.show();  
  
 *// Sign up didn't succeed. Look at the  
 // ParseException  
 // to figure out what went wrong* }  
 }  
 });  
 }   
 **else** {  
 Toast toast = Toast.*makeText*(getApplicationContext(),  
 **"Please enter an email! "**, Toast.***LENGTH\_LONG***);  
 toast.show();  
 }  
   
 }  
 **else** {  
 Toast toast = Toast.*makeText*(getApplicationContext(),  
 **"Please enter a password! "**, Toast.***LENGTH\_LONG***);  
 toast.show();  
  
 }  
  
 } **else** {  
 Toast toast = Toast.*makeText*(getApplicationContext(),  
 **"Please enter a valid username! "**,  
 Toast.***LENGTH\_LONG***);  
 toast.show();  
 }  
  
  
 }  
}

**Refrences:**

[1] [Mahmood, F.M.](http://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=p_Authors:.QT.Mahmood,%20F.M..QT.&newsearch=true) ; [Bin Abdul Salam, Z.A](http://ieeexplore.ieee.org/search/searchresult.jsp?searchWithin=p_Authors:.QT.Bin%20Abdul%20Salam,%20Z.A..QT.&newsearch=true) “A conceptual framework for personalized location-based Services (LBS) tourism mobile application leveraging semantic web to enhance tourism experience,” Advance Computing Conference (IACC), 2013 IEEE 3rd International Page(s):287 - 291 Print ISBN: 978-1-4673-4527-9

[2] Federal Communication Commission 911 services website, accessed August 24, 2006, http://www.fcc.gov/911/enhanced/

[3] The Parlay Group, [http://www.parlay.org](http://www.parlay.org/)

[4] The Open Mobile Alliance (OMA), “The Mobile Location Protocol,” http://www.openmobilealliance.org

[5] Sean J. Barbeau, Miguel A. Labrador, Philip L. Winters, Rafael Pérez, and Nevine Labib Georggi, A General Architecture in Support of Interactive, Multimedia, Location-Based Mobile Applications, IEEE Communications Magazine • November 2006

[6] J. Hightower and G. Borriello, “A Survey and Taxonomy of Location Systems for Ubiquitous Computing,” IEEE Computer, vol. 34, no. 8, 2001, pp. 57–66.

[7] A. Kupper, Location-Based Services: Fundamentals and Operation, New York: Wiley, 2005.

[8] H. Karl and A. Willig, Protocols and Architectures for Wireless Sensor Networks, John Wiley, 2005.

[9] Sun Microsystems, “Java Remote Method Invocation Specification,” 1998.

[10] M. Spanoudakis et al., “Extensible Platform for Location Based Services Provisioning,” Proc. 4th Int’l. Conference Web Information Systems Engineering Workshop. (WISEW), 2004.

[11] A. Krevl, M. Ciglaric, "A framework for developing distributed location based applications", IPDPS, 2006, Parallel and Distributed Processing Symposium, International, Parallel and Distributed Processing Symposium, International 2006, pp. 294, doi:10.1109/IPDPS.2006.1639551

[12] Ch.ChakradharaRao et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 4 (6) , 2013, 979-982 “GPS Based Vehicle Navigation System Using Google Maps”.

[13] <https://parse.com/docs/android/guide>

[14] http://www.tutorialspoint.com/android/