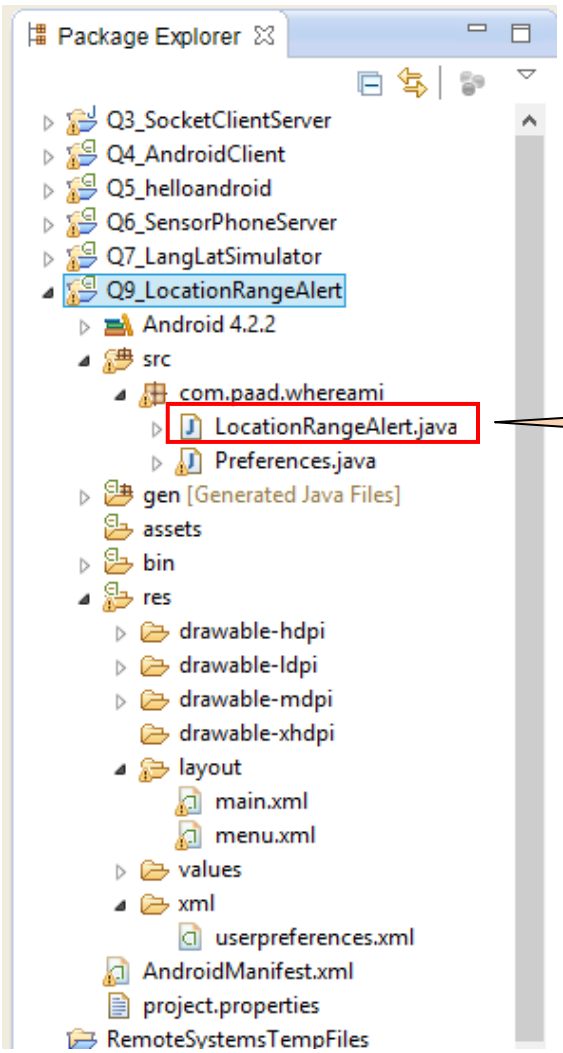
Manisha Vyas

Program :

Q8_LocationRangeAlert::

1. Advanced Geo-location applications
   - Using Proximity Alerts
   - Proximity alerts let your applications set triggers that are fired when a user moves within or beyond a set distance from a geographic location.
   - integrate sensor simulator with your code so that the GPS data is obtained from sensor simulator.

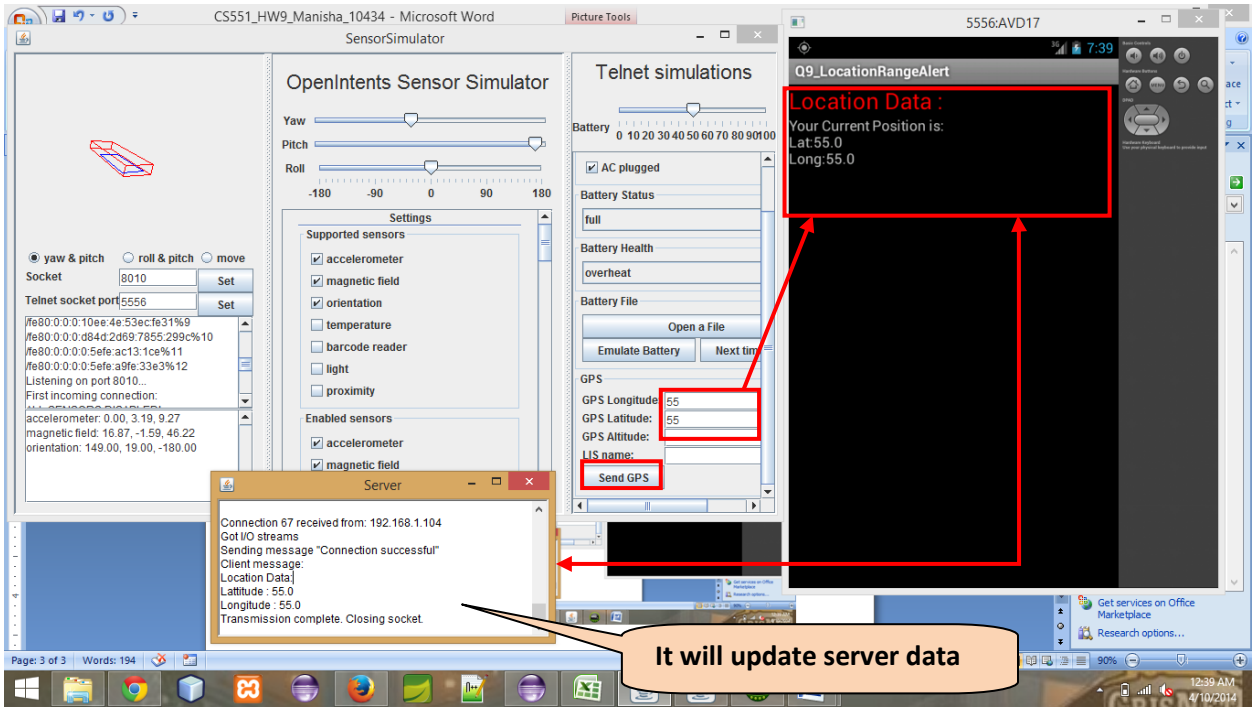
- **In this program i have used. Sensor Simulator to Simulate GPS coordinates.**
- **We can set Range for MAX and MIN LONGITUDE/LATITUDE using preference screen.**
- **Any time when person with Device changes his location Server receives data .**
- **If he goes outside of Preferred range of Latitude/Longitude he is given an Alert with location coordinates are sent to Server.**



I have mainly used only 1 class to implement this functionality

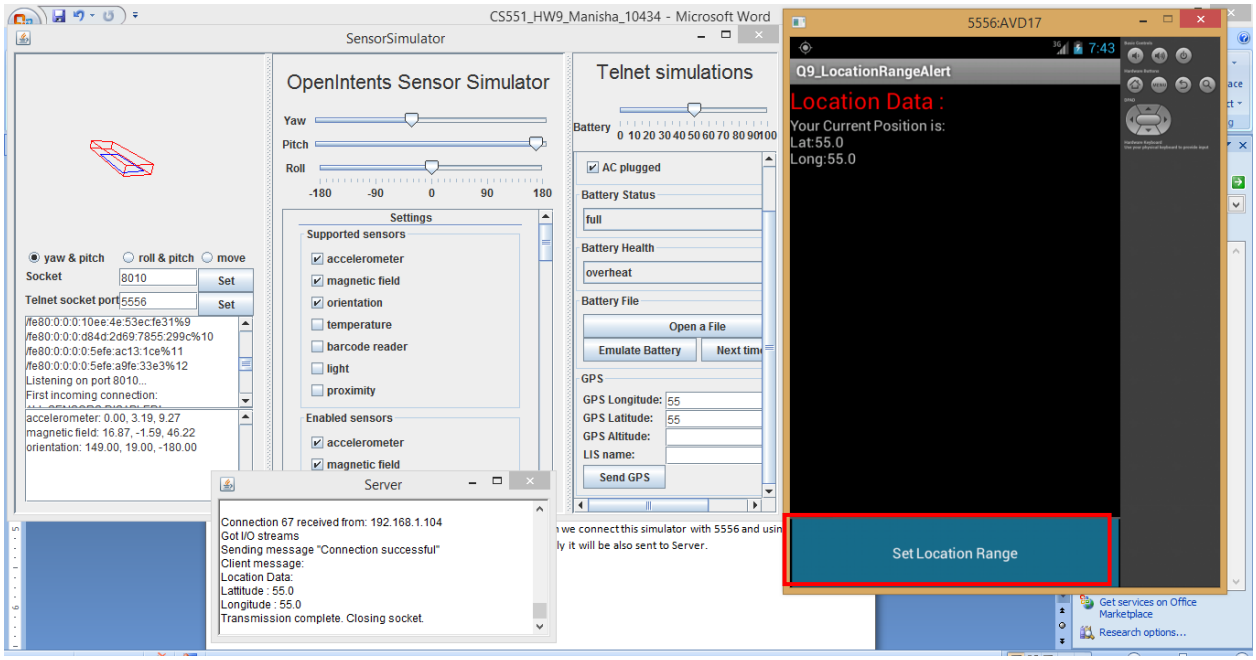
Other class is for Preference Screen

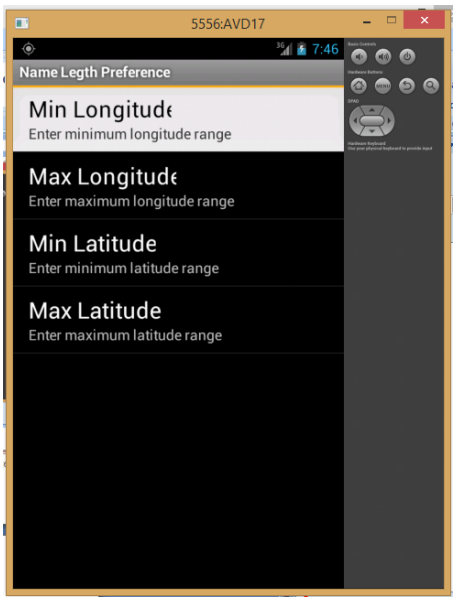
ScreenShot:

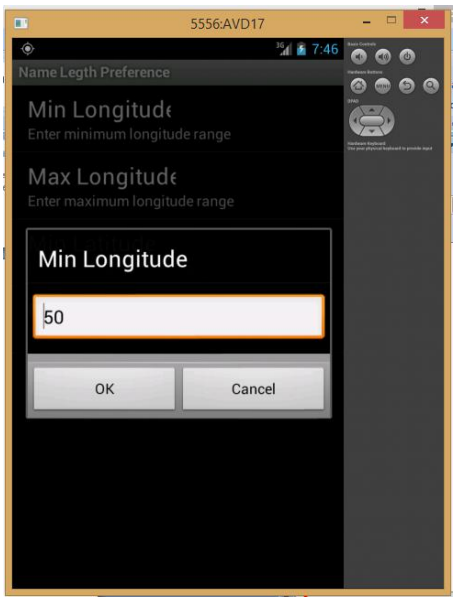


This program is connected with Sensor Simulator. When we connect this simulator with 5556 and using socket 8010. it will send data to device. and automatically it will be also sent to Server.

We can set Preferred screen by pressing on MENU > SET LOCATION RANGE (PREFERENCE SCREEN)

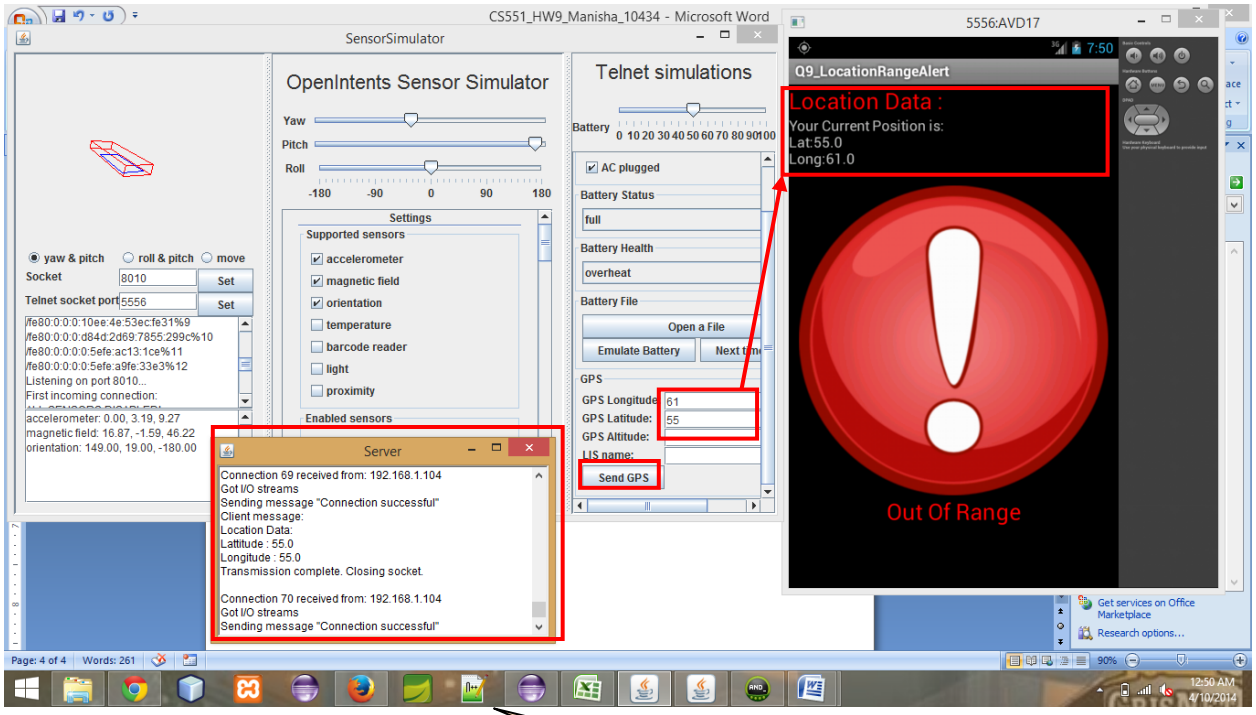


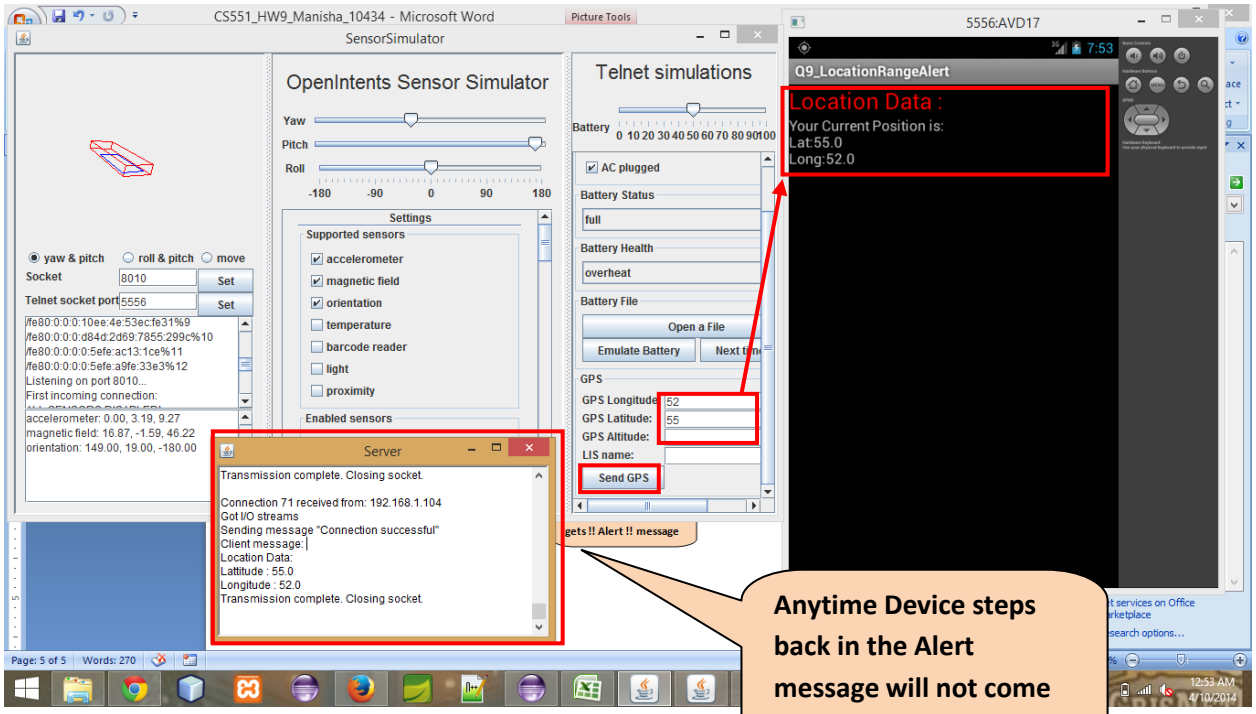


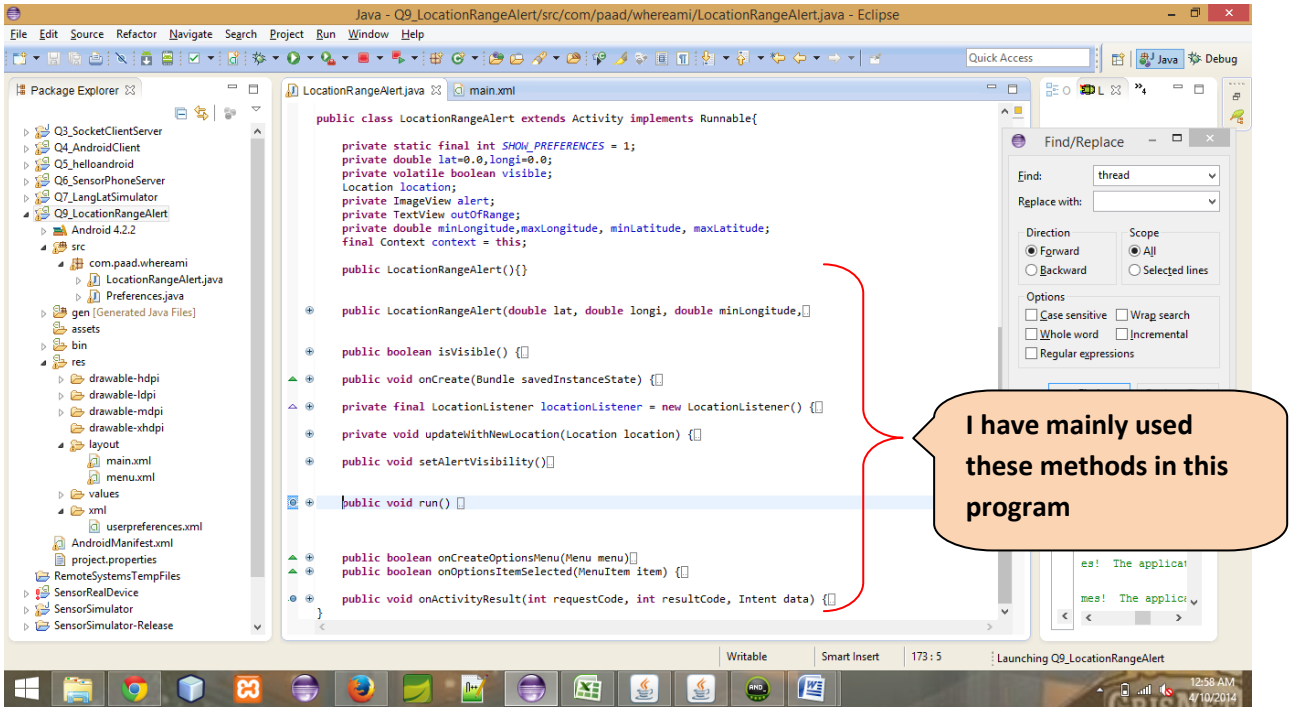


We can Manually enter each range for MAX/MIN LONGITUDE/LATTITUDE

Here I have Values set


CS551_HW9_Manisha_10434 - Microsoft Word
SensorSimulator
OpenIntents Sensor Simulator
Telnet simulations
Server
Connection 69 received from: 192.168.1.104
Got I/O streams
Sending message "Connection successful"
Client message:
Location Data:
Lattitude : 55.0
Longitude : 55.0
Transmission complete. Closing socket.
Connection 70 received from: 192.168.1.104
Got I/O streams
Sending message "Connection successful"
5556:AVD17
Q9_LocationRangeAlert
Location Data :
Your Current Position is:
Lat:55.0
Long:61.0
Out Of Range
Client and Server Both gets !! Alert !! message


CS551_HW9_Manisha_10434 - Microsoft Word
SensorSimulator
OpenIntents Sensor Simulator
Telnet simulations
Server
Transmission complete. Closing socket.
Connection 71 received from: 192.168.1.104
Got I/O streams
Sending message "Connection successful"
Client message:
Location Data:
Lattitude : 55.0
Longitude : 52.0
Transmission complete. Closing socket.
5556:AVD17
Q9_LocationRangeAlert
Location Data :
Your Current Position is:
Lat:55.0
Long:52.0
Anytime Device steps back in the Alert message will not come nither device nor Server



LocationRangeAlert.java

```
package com.paad.whereami;

import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.IOException;
import java.net.Socket;
import java.net.UnknownHostException;
import java.util.List;
import java.util.Locale;
import android.app.Activity;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.location.Address;
import android.location.Criteria;
import android.location.Geocoder;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.preference.PreferenceManager;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;
```

```
import android.widget.Toast;

public class LocationRangeAlert extends Activity implements Runnable{

	private static final int SHOW_PREFERENCES = 1;
	private double lat=0.0,longi=0.0;
	private volatile boolean visible;
	Location location;
	public boolean isVisible() {
		return visible;
	}

	private ImageView alert;
	private TextView outOfRange;
	private double minLongitude,maxLongitude, minLatitude, maxLatitude;
	final Context context = this;

	public LocationRangeAlert(){}


public LocationRangeAlert(double lat, double longi, double minLongitude,
		double maxLongitude, double minLatitude, double
maxLatitude,boolean visible) {
		super();
		this.lat = lat;
		this.longi = longi;
		this.minLongitude = minLongitude;
		this.maxLongitude = maxLongitude;
		this.minLatitude = minLatitude;
		this.maxLatitude = maxLatitude;
		this.visible = visible;
	}


@Override
  public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.main);

    alert = (ImageView)findViewById(R.id.alert);
    outOfRange = (TextView)findViewById(R.id.outOfRange);

    alert.setVisibility(View.GONE);
    outOfRange.setVisibility(View.GONE);

    LocationManager locationManager;
    String context = Context.LOCATION_SERVICE;
    locationManager = (LocationManager)getSystemService(context);


    Criteria criteria = new Criteria();
    criteria.setAccuracy(Criteria.ACCURACY_FINE);
    criteria.setAltitudeRequired(false);
    criteria.setBearingRequired(false);
    criteria.setCostAllowed(true);
```

```
    criteria.setPowerRequirement(Criteria.POWER_LOW);
    String provider = locationManager.getBestProvider(criteria, true);
    //String provider = "gps";
         location = locationManager.getLastKnownLocation(provider);
    updateWithNewLocation(location);

    locationManager.requestLocationUpdates(provider, 2000, 10,
                                           locationListener);
  }

  private final LocationListener locationListener = new LocationListener() {
    public void onLocationChanged(Location location) {
      updateWithNewLocation(location);
    }

    public void onProviderDisabled(String provider){
      updateWithNewLocation(null);
    }

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

  private void updateWithNewLocation(Location location) {
       double tempLat=0.0,tempLng=0.0;
    String latLongString;
    TextView myLocationText;
    myLocationText = (TextView)findViewById(R.id.myLocationText);

    String addressString = "No address found";

   // double lat=0.0,lng=0.0;
    if (location != null) {
      double lat = location.getLatitude();
      double lng = location.getLongitude();
      tempLat = lat;
      tempLng = lng;
      latLongString = "Lat:" + lat + "\nLong:" + lng;

      double latitude = location.getLatitude();
      double longitude = location.getLongitude();
      Geocoder gc = new Geocoder(this, Locale.getDefault());
      try {
        List<Address> addresses = gc.getFromLocation(latitude, longitude, 1);
        StringBuilder sb = new StringBuilder();
        if (addresses.size() > 0) {
          Address address = addresses.get(0);

          for (int i = 0; i < address.getMaxAddressLineIndex(); i++)
            sb.append(address.getAddressLine(i)).append("\n");

            sb.append(address.getLocality()).append("\n");
            sb.append(address.getPostalCode()).append("\n");
            sb.append(address.getCountryName());
```

```
        }
        addressString = sb.toString();

      } catch (IOException e) {}
    } else {
      latLongString = "No location found";
    }

    myLocationText.setText("Your Current Position is:\n" +
                           latLongString + "\n" + addressString);
    LocationRangeAlert la =  new LocationRangeAlert(tempLat,tempLng,
             minLatitude,maxLatitude,minLongitude,maxLongitude,visible);

    Thread aThread = new Thread(la);//tempLat,tempLng
    aThread.run();

    visible = la.isVisible();
    setAlertVisibility();
  }

  public void setAlertVisibility()
  {
        if(visible)
          {
             alert.setVisibility(View.VISIBLE);
              outOfRange.setVisibility(View.VISIBLE);
          }
          else
          {
             alert.setVisibility(View.GONE);
              outOfRange.setVisibility(View.GONE);
          }
  }


@Override
public void run()
{
       Socket socket = null;
        DataOutputStream dataOutputStream = null;
        DataInputStream dataInputStream = null;

        try
        {
                socket = new Socket("192.168.1.104", 5000);
                dataOutputStream = new DataOutputStream(socket.getOutputStream());
                dataInputStream = new DataInputStream(socket.getInputStream());

                String alertStr="";
                if(lat <= minLatitude || lat>=maxLatitude || longi <= minLongitude ||
longi>=maxLongitude)
                {
                       visible = true;
                       alertStr= "!! Alert !!\nDevice out side of Range !!\n";
                }
```

```
                else
                {
                        visible = false;
                }
                dataOutputStream.writeUTF("\nLocation Data:"+"\n"
                                +alertStr
                                +"Lattitude : "+lat+"\n"
                                +"Longitude : "+longi);
        }
        catch (UnknownHostException e)
        {
                e.printStackTrace();
        }
        catch (IOException e)
        {
                e.printStackTrace();
        }
        finally
        {
                    if (socket != null)
                    {
                            try
                            {
                                    socket.close();
                            }
                            catch (IOException e)
                            {
                                    e.printStackTrace();
                            }
                    }
                    if (dataOutputStream != null)
                    {
                             try
                             {
                               dataOutputStream.close();
                             }
                             catch (IOException e)
                             {
                                     e.printStackTrace();
                             }
                    }
                    if (dataInputStream != null)
                    {
                            try
                            {
                                    dataInputStream.close();
                            }
                            catch (IOException e)
                            {
                                    e.printStackTrace();
                            }
                    }
                }
}
```

```
	@Override
	public boolean onCreateOptionsMenu(Menu menu)
	{
	    MenuInflater menuInflater = getMenuInflater();
	    menuInflater.inflate(R.layout.menu, menu);
	    return true;
	}
	public boolean onOptionsItemSelected(MenuItem item) {
	  super.onOptionsItemSelected(item);

	  switch (item.getItemId())
	  {
	  case R.id.filter_name:
	      Intent i = new Intent(this, Preferences.class);
	      startActivityForResult(i, SHOW_PREFERENCES);
	      return true;

	  default:
	      return super.onOptionsItemSelected(item);
	  }
	}

	@Override
	public void onActivityResult(int requestCode, int resultCode, Intent data) {
	  super.onActivityResult(requestCode, resultCode, data);

	    switch (requestCode)
	    {
	              case(SHOW_PREFERENCES) :
	              {
	                    SharedPreferences prefs =
PreferenceManager.getDefaultSharedPreferences(context);
	                    minLongitude =
Double.parseDouble(prefs.getString(Preferences.MIN_LONGITUDE, "0"));
	                    maxLongitude =
Double.parseDouble(prefs.getString(Preferences.MAX_LONGITUDE, "90"));
	                    minLatitude =
Double.parseDouble(prefs.getString(Preferences.MIN_LATITIDE, "0"));
	                    maxLatitude =
Double.parseDouble(prefs.getString(Preferences.MAX_LATITIDE, "90"));
	                    //Toast.makeText(this, " Values are :\n" +
minLongitude+"\n" +maxLongitude, Toast.LENGTH_LONG).show();//updateFromPreferences();
	                    updateWithNewLocation(location);
	                   break;
	              }
	     }
	    }
}
```

please find remaining classes in exported android program.