Approval Sheet

The Major Project Phase-1 entitled "**Location detection in Multistory building with height determination**" submitted by **Amber Pandit** is approved as partial fulfillment of the **Bachelor of Engineering (Computer Science & Engineering)** degree by **Rajiv Gandhi Prodyogiki Vishwavidhyalaya, Bhopal.**

Internal Examiner **External Examiner**

Date: **Date:**

Director

School of Engineering
Sanghvi Institute of Management & Science, Indore

Recommendation

The Major Project Phase-1 entitled "**Location detection in Multistory building with height determination**" submitted by **Amber Pandit** is approved as partial fulfillment of the **Bachelor of Engineering (Computer Science & Engineering)** degree by **Rajiv Gandhi Prodyogiki Vishwavidhyalaya, Bhopal.**

Date:29-05-2015

Project Guide

Mr. Mayank Sharma

Asst. Professor

CSE Department

Recommended By

Prof Ritesh Shah

Department of Computer Sc. & Engg

Sanghvi Institute of Management & Science,

RGPV, Bhopal

Candidate Declaration

I hereby declare that the Major Project phase-1 work entitled **“Location detection in Multistory building with height determination”** is partial fulfillment of Bachelor of Engineering (Computer Science & Engineering) is an authentic record of my work carried out under the supervision of **Mr. Mayank Sharma, Asst. Professor, CSE Department.**

The matter embodied in this desertion work is authenticated and is genially done by me and has not been submitted whether to this university or any other university or institute.

Amber Pandit (0837CS111006)

Acknowledgement

This project is not person's solitary effort. Its successful completion is the result of many different people to whom we owe a debt beyond repayment. Unfortunately, we cannot acknowledge our indebtedness to all these people, so we must necessarily limit our thanks to those who have helped us directly in making our project work an incredibly pleasant task and giving to final shape.

I would also like to thank **Lt.col.P.Deogirikar,** Director, Sanghvi Institute of Management & Science, Indore (M.P.) for his continuous support in completion of my project.

We wish to acknowledge our gratitude to **Prof. R.P.Pandey,** Deputy Director, Sanghvi Institute of Management & Science, Indore (M.P.) for providing us the required resources and facility to undertake this project.

We wish to acknowledge our gratitude to **Prof. Ritesh Shah Sir,** HOD, Department of Computer Science, Sanghvi Institute of Management & Science, Indore (M.P.) for granting us the opportunity to undertake this project.

We offer our sincere thanks to **Mr. Mayank Sharma Sir,** Department of Computer Science, Sanghvi Institute of Management & Science, Indore (M.P.) without their guidance and support; this project would never been possible.

We also wish to convey our gratitude to the faculty members of SIMS for their continuous help and constructive criticism.

Abstract

***Location Detection System with height determination for multistory building** is useful to determine the height of Object from the ground surface whether there is obstacle between grounds or not as well as location. The well-established map services like Google maps, Bing maps, Nokia here Maps show s the longitude and latitude and as well as altitude. The altitude values determine the height from sea level for the given longitude (the angular distance of a place east or west of the Greenwich meridian) and latitude (the angular distance of a place north or south of the earth's equator).*

Now to get the accurate location of objects height from the ground surface is also a very important factor. The object's altitude value is same on the ground and top floor of the building. So to locate user or object in map there should be a parameter of measuring the height of user from the ground surface.

Vertical location finder helps to get a reference to measure the vertical distance from the ground surface of that location. User of the application is able to get the location as well as their height from the ground surface. The references are WI-FI signal strength, trigonometric approach and using the barometric sensor. The approach suggests the idea to get the vertical distance. The project helps in various domains like automated drones, location detection system inside a building, spying etc.

Table of Contents

1 INTRODUCTION

1.1 Background

Online mapping is the compilation and publication of Web sites that provide exhaustive graphical and text information in the form of maps and databases. Online mapping services are used for tasks such as planning trips, determining geographical positions, finding landmarks and businesses, obtaining addresses and phone numbers, and plotting storm tracks. The online mapping services use Global-Positioning-System to detect the user or objects.

GPS satellites circle the earth twice a day in a very precise orbit and transmit signal information to earth. GPS receivers take this information and use triangulation to calculate the user's exact location. Essentially, the GPS receiver compares the time a signal was transmitted by a satellite with the time it was received. The time difference tells the GPS receiver how far away the satellite is. Now, with distance measurements from a few more satellites, the receiver can determine the user's position and display it on the unit's electronic map.

A GPS receiver locked on to the signal of at least three satellites to calculate a 2D position (latitude and longitude) and track movement. With four or more satellites in view, the receiver can determine the user's 3D position (latitude, longitude and altitude). Every location on earth has a global address. Because the address is in numbers, people can communicate about location no matter what language they might speak. A global address is given as two numbers called coordinates. The two numbers are a location's latitude number and its longitude number ("Lat/Long").Some of these map services quietly added elevation data to bike routes in its Maps service, providing better intel for intrepid cyclists. The elevation value is defined by altitude, the height of an object or point in relation to sea level.

1.2 Needs

To design and construct the location detection system which incorporates with the height of object from the ground as well as the location of object Where GPS signals are not reachable.

1.3 Problem Definition:

The online map services provide the location of user on a two-dimensional plane and many of them provide altitude. The problem in existing map services is to detect device or object on the same value of longitude and latitude but different values of height from the relative ground surface, these map services show devices or object on the same location arise anomalies .So it becomes more difficult to detect object in such condition &The System which provide the 2 dimensional coordinates value , satellite positioning with GPS does not work under any environmental condition (e.g. in urban “canyons” with no satellite visibility and indoor)

Consider the example if the object is in the multi storey building on the nth floor and have to detect the current location via the existing system all that could be done is finding the latitude and longitude value of the object but still height factor is a mystery .

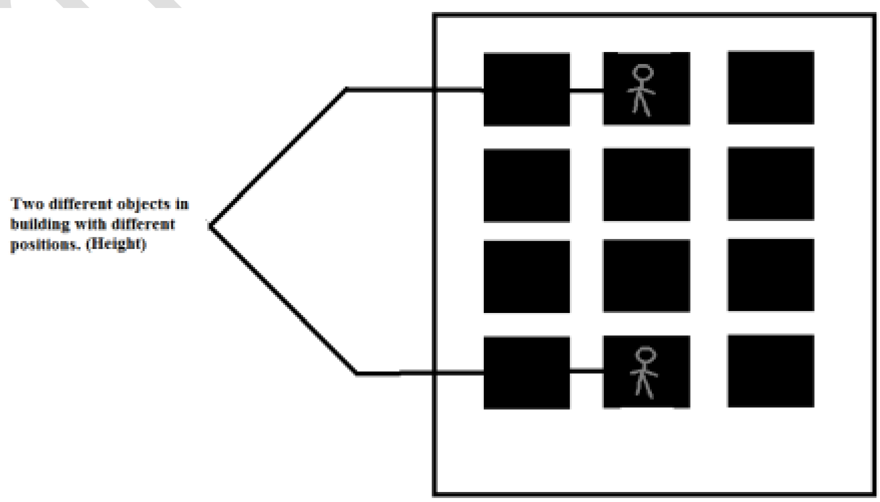
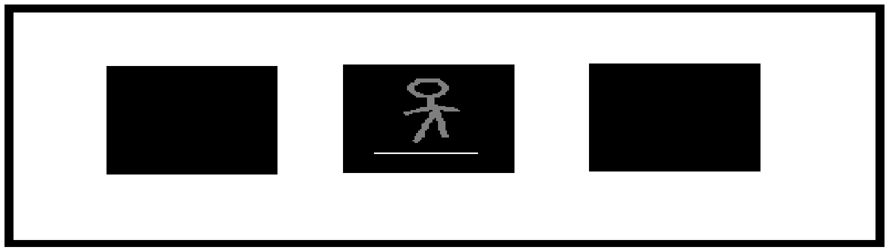


Fig.1.3.1:-front view of a building.



shows same location on map services (i.e
same lattitude and longitude)

Fig.1.3.2:Top view of building.

1.4 Aim and objective:

The work mainly relies on detection of object where the information provided by the existing system is not sufficient i.e. where the object has different height from ground and latitude and longitude are same.

- The work will use the reference factor of height with longitude and latitude where it can remove the anomalies of existing system. The height factor will differentiate between two locations where the existing system provides same information.
- The work will also detect user if there is no satellite positioning system is available.
- To assess the relevance of mobile applications focused on location features.
- To examine the current shortcomings with regard to infrastructure and technology.

1.5 Purpose

The work will provide a convenient way to detect object where existing system are not capable. With the changing infrastructure of the world today there are many multi storey building and the existing system is just capable of detecting the location provided by GPS in two dimensional

manner i.e. they are just capable of finding the latitude and longitude values of object and in some prior condition they are not reachable , but now we need to enhance our existing system in a manner that they could be capable finding the relative height of the object from the ground along with the latitude and longitudinal values or the 2 dimensional coordinates value.

The work will be useful when the object is concerned about the height of object from the relative ground level where the geo-location anomalies arise. The work finds its way when the height of the object to be traced is more important than just the geographical existence. The work is advancement of geo-location detection technique which can enrich the height of the object.

1.6 Scope

Location Based Service has been popular in the field of navigation, in a country like India, there are many innovative service options that can reach out, not just to the urban consumers, but also to the underprivileged in the cities and villages, giving them a stronger base for inclusion in the process of growth.

Location-based services with height reference will provide more accuracy and refer to applications that use knowledge of the geographical position of a mobile device to provide services based on that information. Such services include navigation assistance, identification in case of emergency or disaster relief, social networking through finding friends, map assistance through locating points of interest, spying ,pedestrian navigation system**,** floor detection & positioning system for multistory building,. Provision of LBS requires the correct positioning technology, a geographical information system that maps the areas.

1.7 Questionnaire

Q.1 How many location-based services like foursquare FourSquare, Gow Check-in ,do you use?

Q.2How do you access these location-based services?

Q.3Would you use more these type of services if there was an application that consolidates the services?

Q.4 What features of the location-based service(s) do you find could be improved?

Q.5 Have you ever faced location detection problem in multistory building?

2 RELATED WORK

2.1 A Brief Review of the work already done in the field.

The primary goal well established map services are to provide routes, direction to destination from the users the location behalf of two dimensional planes. Now offering services satellite imagery, street maps, and Street View perspectives, as well as functions such as a route planner for traveling by foot, car, bicycle (beta test), or with transportation.

The Wi-Fi-based positioning technology has a very promising application prospect mainly because of the ubiquitous and inexpensive nature of Wi-Fi infrastructure. Also, Wi-Fi is widely used and integrated in various electronic devices. Thus, the Wi-Fi based positioning systems can also reuse these mobile devices as tracking targets to locate users, which is a less intrusive way to provide location-aware services.

Barometers have recently started appearing on smart phones and tablets. Given the novelty of these sensors, we have found little prior work that has used them for indoor location.

Brush [1] unsuccessfully attempted to use a barometer to help users find where they parked their car. Lester [2] focused on activity recognition, including going up or down stairs and elevators. While their data collection included barometric pressure, it was not used to infer where a user is. Varshavsky [3] used a GSM fingerprinting based system to infer the current floor of a user, but with lower accuracy than our system and with higher fingerprinting overhead. Ojeda [4] deployed a dead reckoning system to capture floor level transitions. However, that system is limited to stairs and cannot capture floor changes via an elevator or escalator. Johannsson [5] used vertical accelerometer information over time to determine the number of floors traveled. While the paper reports high accuracy in classifying floor level transition it is unclear if the results hold for multiple buildings.

Similarly Ye [6] used a phone's accelerometer to record the time taken for traveling across different floors via an elevator and the step count in the case of stairs. The system was shown to achieve a high accuracy, but as we observe, the accelerometer sensor is susceptible to any sort of perturbation which can result in lower accuracies. A large body of literature proposes indoor location techniques which can be used to identify the physical location of users with varying levels of accuracy. This could in theory be translated to detection of floor changes.

Some techniques rely on custom hardware or radios [7,8,9,10]. Custom radios can also offer TOA (time of arrival), TDOA (time difference of arrival) or AOA (angle of arrival) information for location. These approaches have hardware adoption challenges.

2.2 Major Contribution:

While absolute pressure readings have significant time-of-day variations, the difference in pressure across different floor pairs is remarkably consistent and steady for any given building. As a consequence, we are able to use pressure difference as a useful fingerprint to detect the exact number of floors changed with almost 100% accuracy.[11]

The idea of fingerprint-based localization is to utilize the received signal strength (RSS) to estimate the location of a receiver. The main challenging issue for accurate fingerprint-based indoor localization is the design of robust and discriminative RSS signatures. The authors report a comparative survey of WLAN location fingerprinting methods. [12]

3 REQUIREMENT ANALYSIS

3.1 Requirement study

3.1.1. Functional requirement:-

- The system shall display the longitude and latitude or 2D location of the object through GPS/Wi-Fi.
- The system shall display the height of user from the relative ground surface.
- The system shall display the positions of the object on map.
- Browsing the spatial information on the city/area map.
- Users should be able to access the map and catalogue data based on their current location. If user face any system problem for acquiring location so that he/she must be able to enter his/her location manually. The system must be able to deal these requests through geo-coding procedure.
- Geographical information should come as textual description.
- Current status of the user must be represented as form of picture and text.

3.1.2. Non functional requirement:-

- Geospatial application should not use mobile network very intensively
- In the system, user interface must be user friendly, very simple and amount of shown information content should be limited and well described.
- All information related to location should be reliable. Incorrect or missed data can easily lead to create problem such as court examination.
- Software should be reliable for both side server and client sides.
- The user privacy should be considered as one of the main requirement.

- Locating method must provide accuracy subject to the requirements of the geospatial application and cost should be affordable.
- Try to locate mobile device current location as large as possible so that user can access more information. It should also be possible to find in advance where the device can be located.
- The location method should not create too much signals within the mobile network.
- Application should be able to locate a large group of users at the same time.
- Users should be able to control their location privacy.
- Application should be able to run on several types of terminals like PDA, Mobile Phone.

3.2 Technical Feasibility

The user or object's pinpointing on the map is feasible in one of the following ways:

- Using the GPS device that comes with the mobile
- Using the ID of the Cell that the user is currently served by.
- Using the Wi-Fi approach where no GPS signals available.
- The height factor can be obtained through the reference of barometer/altimeter.

3.3 Economical Feasibility

- Most of the existing smart phones don't have a barometer but barometer is significantly more robust than an accelerometer at detecting vertical activities. However, it consumes similar amounts of energy, even though on paper, the barometer is a very cheap sensor.

- If existing infrastructure such as Wi-Fi without additional hardware installation can be used for location determination, then the realization costs are small and the service can be offered under attractive conditions. Several systems are nowadays available for location determination using Wi-Fi signals.

3.4 System Requirement

3.4.1 Development requirement

Hardware

- RAM:-2 GB (4 GB recommended).
- Processor:-Intel dual core
- Storage:-50 GB

1. **Software**
    - Operating system:-Windows 7 and higher version, Linux, Mac Os.
    - Development tools:-
    - Android Developer Tools Plug in.
    - Java 7 or higher is required if you are targeting the L Developer Preview.
    - Java 1.6 or higher is required if you are targeting other releases.
    - Eclipse Indigo (Version 3.7.2) or higher is required.

Eclipse product includes:-

- JDT (JAVA development tool)
- CDT (C/C++ development tool)
- EMF (eclipse modeling framework)
- GEF (Graphical editing framework)

3.4.2 Deployment requirement

1. **Hardware**
    - Explicit Wi-Fi signal transmitter/Router (For hot-spot).
    - Device: - Smartphone, tablet, slate-book (GPS enabled)
    - RAM: - Min. 356 MB and higher
    - Processor: - 825MHz and higher
    - Storage: - Min. 5 MB
    - barometer

2. **Software**
    - Operating system:-Android gingerbread and higher.

4 SYSTEM ARCHITECTURE

4.1 Methodologies

The barometer provides the pressure values on each location. Starting from a given height the pressure changes can be converted in changes in height using the following equation:

$$\Delta H = H2 - H1 = 18464.(1 + 0{,}0037\, tm).(\lg B1 - \lg B2)$$

Where ΔH is the height difference between two stations 1 and 2, *B1* and *B2* are the pressure observations at station 1 and 2 and *tm* is the mean value of the temperature of both stations. It must be noted that this equation is an approximation formula that was valid for central Europe only (Kahmen, 1997). Tests showed that there is no significant difference between the
results using the approximation formula and an equation derived from Jordan which is also valid for other parts in the world and takes into account the geographic location of the two stations.[13]

Wi-Fi fingerprinting:-

The calculation of the location of a user takes place in two phases: an offline and an online
phase. During the offline phase, which has to be executed only once for each building, a so called radio map will be composed. This radio map can be considered to be a collection of
calibration points at different locations in the building, each with a list of radio signal strength indicator (RSSI) values for visible access points at that particular location. This process is also known as fingerprinting. During the online phase, the calibration points are
being used to calculate the most probable location of the user, whose actual location is unknown.

4.2 System architecture

The architecture of the Location Based System i.e. the work application system is a Client-server application based on five main elements:

- Mobile device (i.e., handset or PDA, etc.)
- Location server (Position-related database, General-geographic database)
- 3G/4G mobile communications Network (3G/4G, Wi-Fi)
- Internet - Remote server (map server, info server)
- Barometer
- Access Points

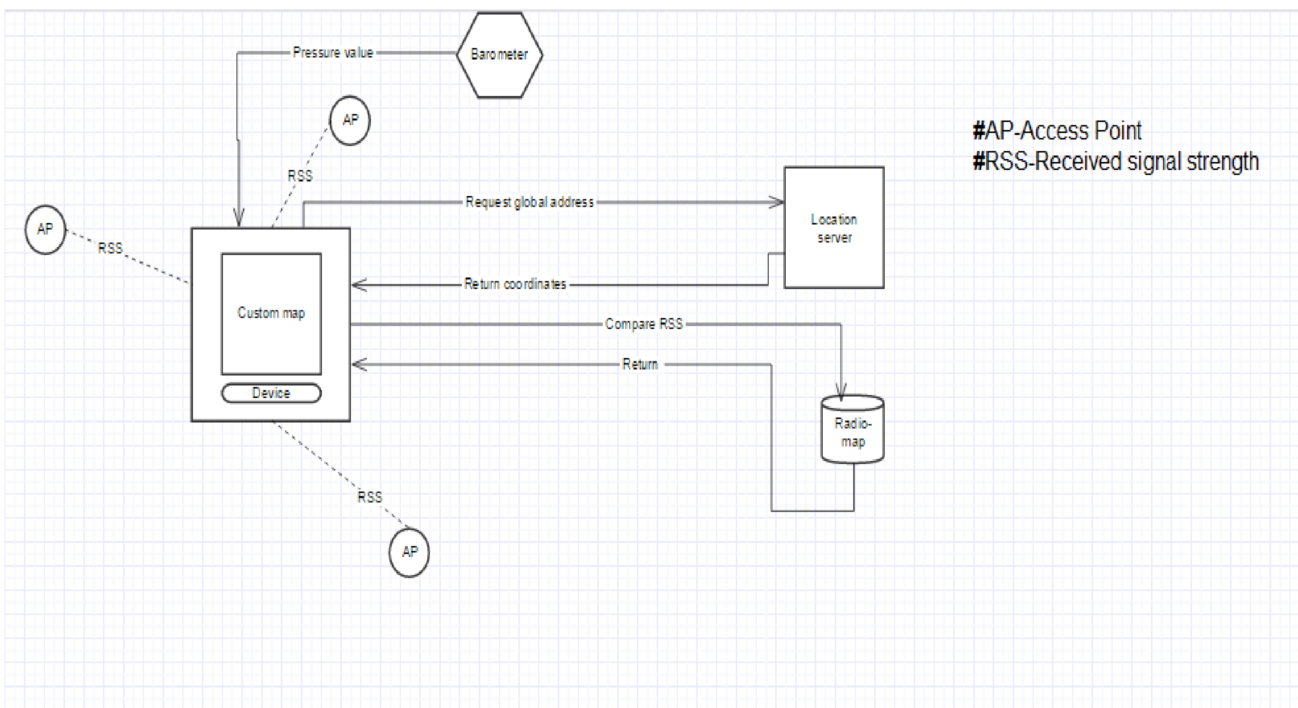


Fig.4.2:System Architecture.

4 SYSTEM ARCHITECTURE

When a user need some location-based service, firstly, the user send location request (It includes the symbol of location data type which the user need to output.). GPS receiver supplies geographic location data to 3G/4G mobile devices, and then these data transmit to location agent through 3G/4G mobile Communication network.

Through the symbol of location data type that the user sends, the location agent selects fit location data from the semantic position database and geographic position database which has been supplied, and do relevant disposals. Then it sends relevant

Information to the mobile user by LBS application interface in the form of text, voice or map etc. If the GPS are not available the system uses the Wi-Fi fingerprinting method on the basis of received signal strength the location approximation can be done. Finally, the location agent with calculation from barometer's data returns the relevant information about location through LBS application interface.

4.3 Use case diagram

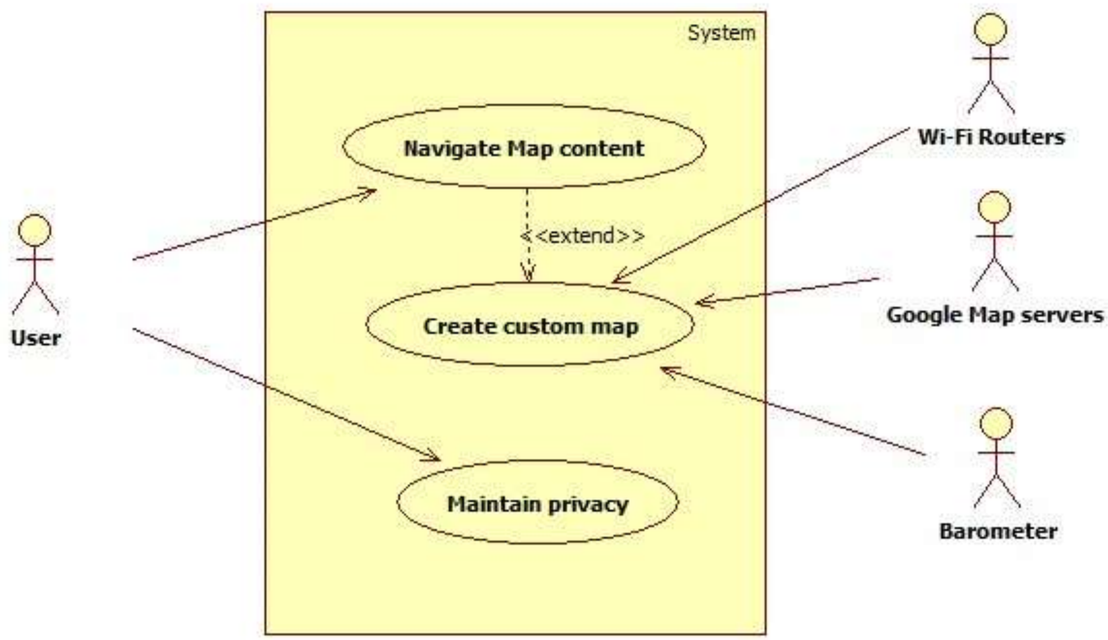


Fig.4.3: Use Case diagram

Description:-

User will be able to navigate himself by GPS or entering manual global address values. The navigation of user and map data are given by Google maps. If GPS is not available routers will help in the navigation. The barometer provides the pressure value which can be further changes

into height with appropriate conversation, thus user will be able to detect himself. The user will be able to maintain his privacy of location.

4.4Class diagram

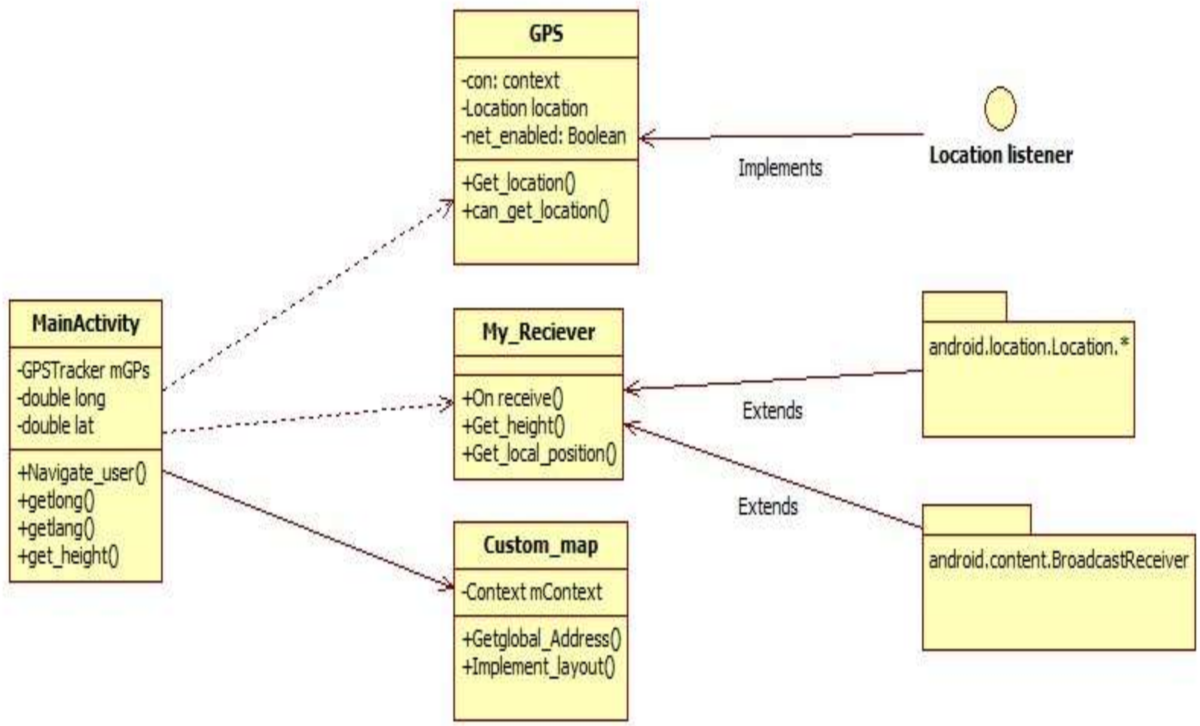


Fig.4.4: Class diagram

4.5 Sequence diagram

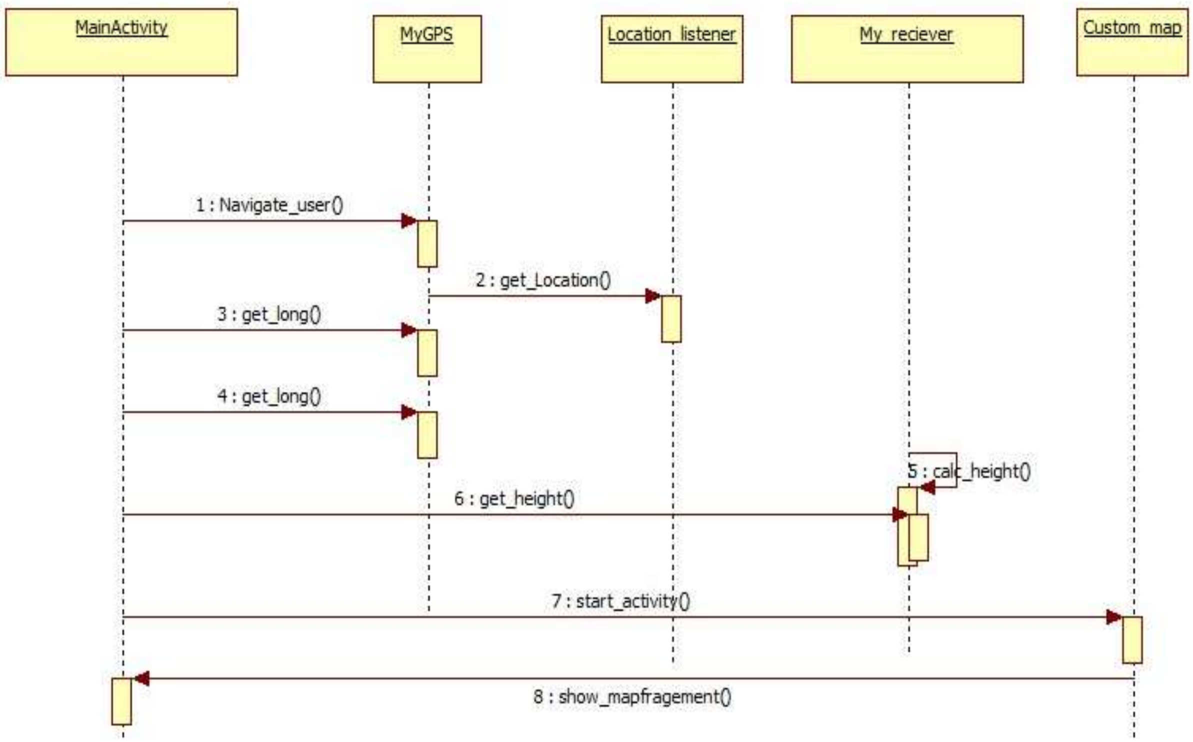


Fig.4.5: Sequence diagram

4.6Activity diagram

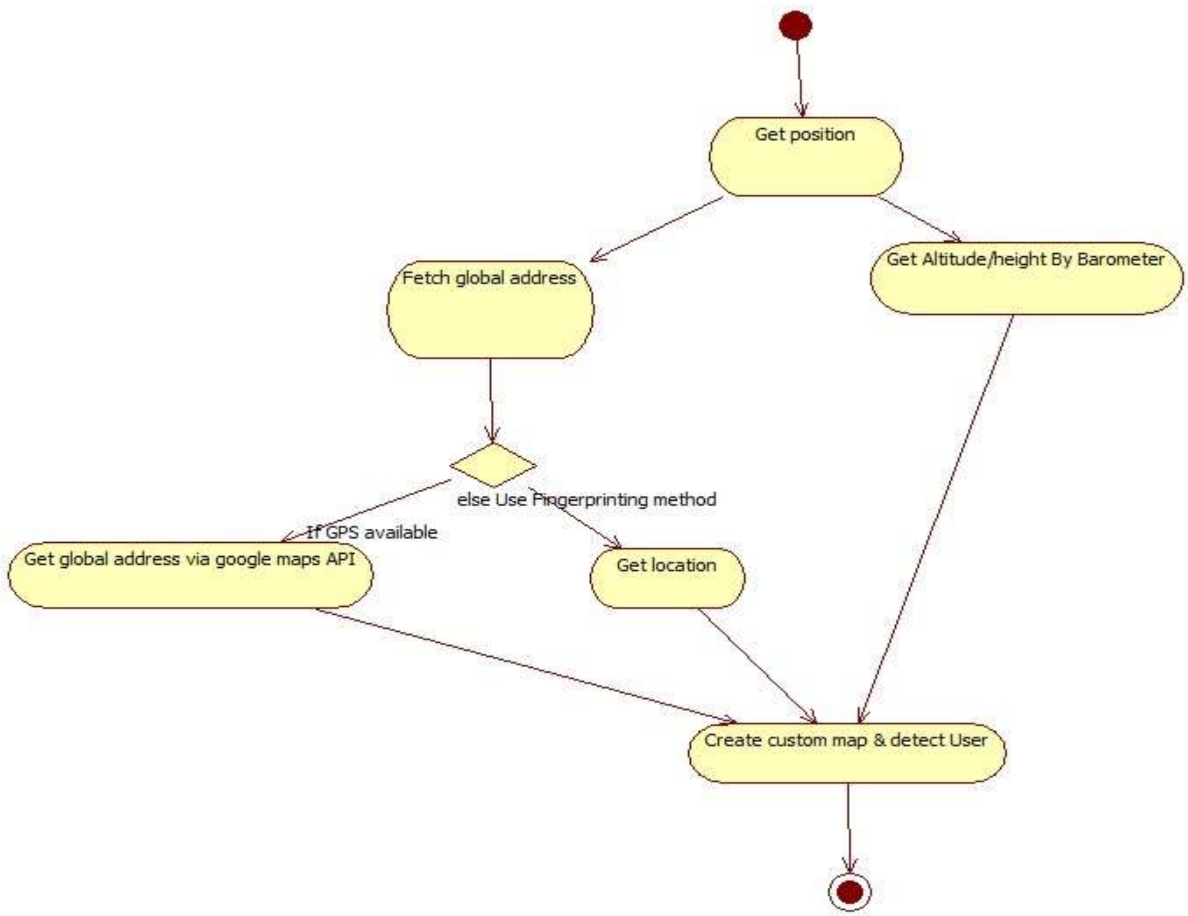


Fig 4.6: Activity diagram

4.7 Data flow diagram

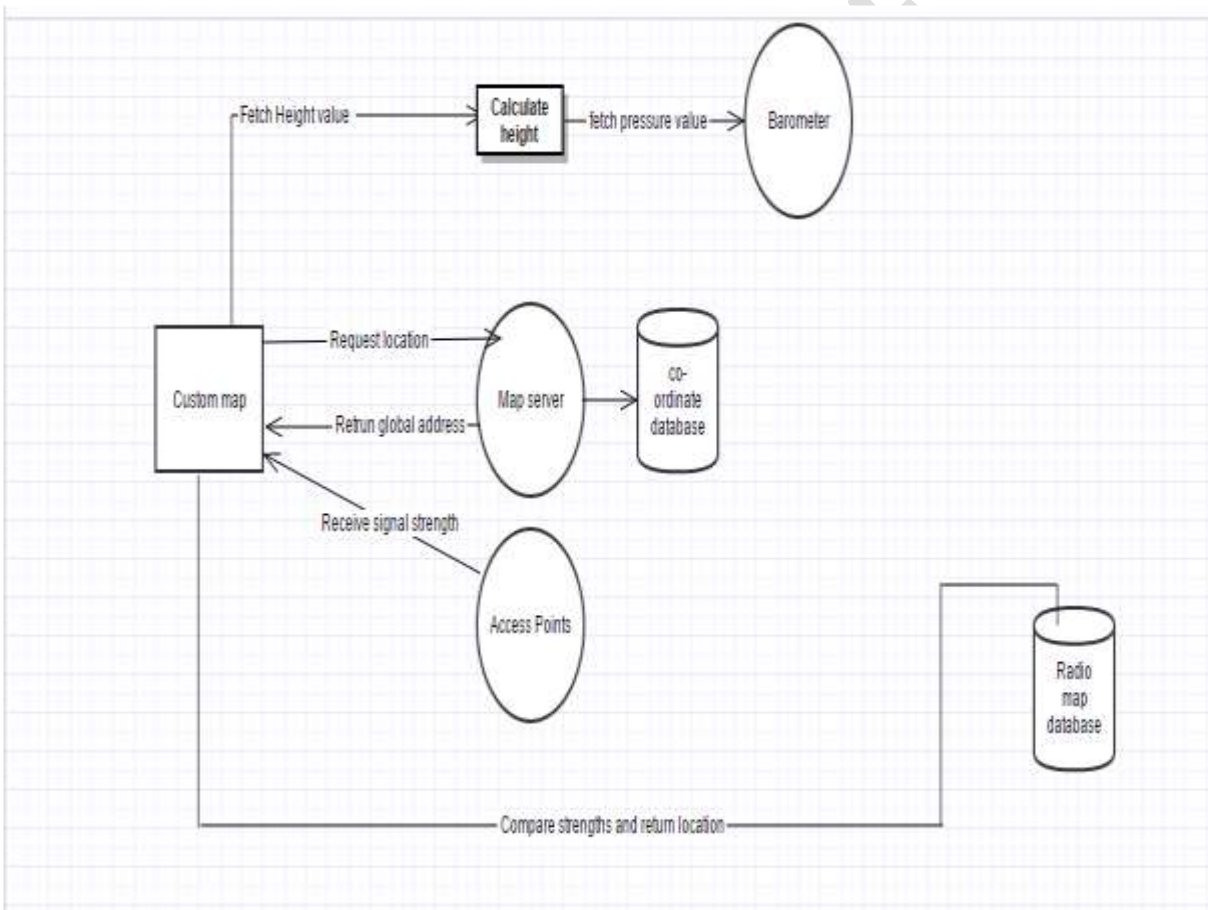


Fig.4.7: Data Flow Diagram

5 IMPLEMENTATION ALGORITHM

5.1 Related algorithm

Estimate the user's location using the K weighted nearest neighbour (KWNN) algorithm.

KWNN is a conventional algorithm used for fingerprint-based Wi-Fi positioning. Using this

algorithm, K ($K \geq 2$) nearest neighbours (those with the shortest signal distance) of a test

vector are chosen. The weighted average of the co-ordinates of K points can be used as the

estimate of the user's location. The inverse of the signal distance defines the weight. [14]

5.2 Proposed algorithm

Proposed algorithm does not contain any changes but few modifications are made to reduce the radio map data for the fingerprinting procedure.

Database Reduction Technique

As an AP of idijk could also be found at multiple FPjk, the RM database can be reduced according to the unique list of AP found in the entire RM database. The new database can be labeled according to each AP ID idm, where m is each ID of the unique AP found in RM database. The database refers to the unique AP ID's only contains its RSS at multiple locations, and the mutiple locations themselves where idm = {rssmjk, $1 \leq k \leq K$} and K is the total number of floor of the building. The new database is referred to as the ID database. In this work, the 'significant' APs from the RM database of a multi-floor building are chosen based on two conditions as follows:

AP located within the possible signal propagation distance:

In RM database, it is understood that multiple APs are found at only few fingerprint locations. As fingerprints are normally collected at very close distance e.g. at every 2 to 5 m in grid space, the APs that are frequently found at each fingerprint location are the 'significant' APs. To define how many fingerprint locations of an AP should present, the

number of location could be approximated based on longest distance the RF signal of an AP could propagate. RSS of an AP heard at multiple floor levels heuristically, the AP that locates within one floor level of the building can also be found at others. In the worst case, the RSS of an AP that locates within
the building is at least heard on two adjacent floors. Therefore, the AP that is heard at only one floor level is removed from the database.[15]

5.3 Implementation source code

5.3.1 Fetching the location object

```
public class MainActivity extends FragmentActivity implements LocationListener {

   GoogleMap googleMap;

   @Override

   protected void onCreate(Bundle savedInstanceState) {

      super.onCreate(savedInstanceState);

      setContentView(R.layout.activity_main);

      // Getting reference to the SupportMapFragment of activity_main.xml

      SupportMapFragment fm = (SupportMapFragment)
getSupportFragmentManager().findFragmentById(R.id.googleMap);

      // Getting GoogleMap object from the fragment

      googleMap = fm.getMap();

      // Enabling MyLocation Layer of Google Map

      googleMap.setMyLocationEnabled(true);
```

```java
        // Getting LocationManager object from System Service LOCATION_SERVICE
        LocationManager locationManager = (LocationManager) getSystemService(LOCATION_SERVICE);
        // Creating a criteria object to retrieve provider
        Criteria criteria = new Criteria();
        // Getting the name of the best provider
        String provider = locationManager.getBestProvider(criteria, true);
        // Getting Current Location
        Location location = locationManager.getLastKnownLocation(provider);
        if(location!=null){
            onLocationChanged(location);
        }
        locationManager.requestLocationUpdates(provider, 20000, 0, this);
    }
    @Override
    public void onLocationChanged(Location location) {
        TextView tvLocation = (TextView) findViewById(R.id.tvlocation);
        // Getting latitude of the current location
        double latitude = location.getLatitude();
        // Getting longitude of the current location
        double longitude = location.getLongitude();
```

```
        // Creating a LatLng object for the current location
        LatLng latLng = new LatLng(latitude, longitude);
        // Showing the current location in Google Map
        googleMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
        // Zoom in the Google Map
        googleMap.animateCamera(CameraUpdateFactory.zoomTo(15));
        // Setting latitude and longitude in the TextView tv_location
        tvLocation.setText("Latitude:" +  latitude  + ", Longitude:"+ longitude );
    }
}
```

5.3.2 Fetching JSON data from www.amberpandit.hostei.com

5.3.2.1 Class: MainActivity

```
public  class MainActivity extends ActionBarActivity {
  JSONParser jsonParser;
    TextView tv;
    Handler mhandle;
    String Url="http://amberpandit.hostei.com/";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```
        jsonParser=new JSONParser(MainActivity.this);

        //JSONArray values=jsonParser.getJSONFromUrl(Url);

        tv=(TextView)findViewById(R.id.textview);

        //tv.setText(values.toString());

        setContentView(R.layout.activity_main);

        jsonParser.execute(Url);

        JSONArray data= null;

        try {

            data = jsonParser.get();

        } catch (InterruptedException e) {

            e.printStackTrace();

        } catch (ExecutionException e) {

            e.printStackTrace();

        }

        if(data!=null)

        Toast.makeText(MainActivity.this,data.toString(),Toast.LENGTH_LONG).show();

        else

            Toast.makeText(MainActivity.this,"Something
wrong",Toast.LENGTH_LONG).show();

    }

}
```

5.3.2.2 class :JSONparser

```
class JSONParser extends AsyncTask<String, Void, JSONArray>

{

   InputStream is;

   ProgressDialog mdialog;

   JSONObject jsonobj;

   JSONArray jArray;

   Context mcontext;

   JSONParser(Context passed) {

      mcontext=passed;

   }

   public JSONArray getjArray() {

      return jArray;

   }

   @Override

   protected JSONArray doInBackground(String... params) {

      // TODO Auto-generated method stub

      String URL=params[0];

      jArray=getjsondata(URL);

      return jArray;

   }
```

```
@Override
protected void onPostExecute(JSONArray result) {
    // TODO Auto-generated method stub
    super.onPostExecute(result);
    mdialog.dismiss();//hide the progress bar after fetching the JSON data from amberpandit.hostei.com
    if(result!=null)
    {           //for(int i=0;i<result.length();i++)
            //{
              //Displaying mechanism
            //}
        //nothing to do here,'ll get json array in mainactivity.
    }
    else {
        Toast.makeText(mcontext, "Data not found on given Source", Toast.LENGTH_LONG).show();
    }
}
@Override
protected void onPreExecute() {
    // TODO Auto-generated method stub
    super.onPreExecute();
```

```
    mdialog=new ProgressDialog(mcontext);

    mdialog.setMessage("please wait...!!!");

    mdialog.show();

}

private JSONArray getjsondata(String server_url)

{

    String json=null;

    JSONArray jArray1=null;

    // Making HTTP request

    try {

        // defaultHttpClient

        DefaultHttpClient httpClient = new DefaultHttpClient();

        HttpPost httpPost = new HttpPost(server_url);

        HttpResponse httpResponse = httpClient.execute(httpPost);

        HttpEntity httpEntity = httpResponse.getEntity();

        is = httpEntity.getContent();

    } catch (UnsupportedEncodingException e) {

        e.printStackTrace();

    } catch (ClientProtocolException e) {

        e.printStackTrace();

    } catch (IOException e) {
```

```
  e.printStackTrace();
}
try {
   BufferedReader reader = new BufferedReader(new InputStreamReader(
         is, "iso-8859-1"), 8);
   StringBuilder sb = new StringBuilder();
   String line = null;
   while ((line = reader.readLine()) != null) {
      sb.append(line + "\n");
   }
   is.close();
 json = sb.toString();
} catch (Exception e) {
   Log.e("Buffer Error", "Error converting result " + e.toString());
}
// try parse the string to a JSON object
try {
   jArray1 = new JSONArray(json);
} catch (JSONException e) {
   Log.e("JSON Parser", "Error parsing data " + e.toString());
}
```

```
        // return JSON String
        return jArray1;
    }
}
```

5.3.3 Readings from Sensor type pressure

```
public class MainActivity extends Activity implements SensorEventListener
{//registered for sensor events
    private SensorManager mSensorManager;
    Sensor barometer;
    private float pressure_value = 0.0f;
    private float height = 0.0f;
    TextView heighttv,pressuretv;
    AlertDialog alert;
    Handler mhandle;
    JSONParser jsonParser;
    String Url="http://amberpandit.hostei.com/";
    Calendar calendar;
    int hour;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```
setContentView(R.layout.activity_main);
jsonParser= new JSONParser(MainActivity.this);
heighttv=(TextView)findViewById(R.id.textView);
pressuretv=(TextView)findViewById(R.id.textView2);
mSensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
if (mSensorManager.getDefaultSensor(Sensor.TYPE_PRESSURE) != null){
    // Success! There's a sensor.
barometer=mSensorManager.getDefaultSensor(Sensor.TYPE_PRESSURE);
    Toast.makeText(MainActivity.this, "You Got a Pressure Sensor", Toast.LENGTH_LONG).show();
    jsonParser.execute(Url);
    JSONArray data= null;
    try {
        data = jsonParser.get();
    } catch (InterruptedException e) {
        e.printStackTrace();
    } catch (ExecutionException e) {
        e.printStackTrace();
    }
    if(data!=null)
```

```
Toast.makeText(MainActivity.this,data.toString(),Toast.LENGTH_LONG).show();
            else
              Toast.makeText(MainActivity.this,"Something
wrong",Toast.LENGTH_LONG).show();
          }
          else {
            // Failure! No
            alert=new AlertDialog.Builder(MainActivity.this).create();
            alert.setButton("Close app",new DialogInterface.OnClickListener() {
              @Override
              public void onClick(DialogInterface dialog, int which) {
                finish();
              }
            });
            alert.setTitle("Oops!!");
            alert.setMessage("Your device doesn't have a pressure sensor");
            alert.show();
          }
        }
        @Override
        public final void onAccuracyChanged(Sensor sensor, int accuracy) {
```

```
    // Do something here if sensor accuracy changes.
  }
  @Override
  public final void onSensorChanged(SensorEvent event) {
    float millibars_of_pressure = event.values[0];
    // Do something with this sensor data.
    pressure_value = event.values[0];
    //or i can use the local airport data instead of the standard atmosphere //pressure.
    //http://avdata.geekpilot.net/airport/IDR.json
    // the link mentioned above is of indore airport's local data
    //or the data i manually created for a particular location
    Height= mSensorManager.getAltitude(SensorManager.PRESSURE_STANDARD_ATMOSPHERE, pressure_value);
    calendar=Calendar.getInstance();
    hour =calendar.get(Calendar.HOUR);
    //Get the pressure data for particular hour
    //Calculate the approximate height & and display on bar.
  }
  @Override
  protected void onResume() {
```

```
            // Register a listener for the sensor.

            super.onResume();

            mSensorManager.registerListener(this,                                   barometer,
SensorManager.SENSOR_DELAY_NORMAL);

        }

        @Override

        protected void onPause() {

            // Be sure to unregister the sensor when the activity pauses.

            super.onPause();

            mSensorManager.unregisterListener(this);

        }

    }
```

6 CONCLUSION

6.1 Conclusion: The outcome of the work will able provides a convenient way to trace object's location on the basis of barometer and Wi-Fi signal strength. The value of x co-ordinate and y co-ordinate for the map i.e. longitude and latitude is provided by GPS and located in Map services. The longitude, latitude and height reference are mapped in a frame layout to show the simulation of object's movement locating the position. The atmospheric pressure continuously fluctuates, the consistency is not preserved and requires a vast data sets gathered from training of system .The rate of change in different environment required and not suitable for global use.

6.2 Future Enhancement:

The code working as expected. The pressure changes with the weather you can get a big range even at sea level e.g. say -10% +5%..

For real altitude information work need to do some calibration which means you use a calibrated value instead of SensorManager.PRESSURE_STANDARD_ATMOSPHERE in your code above.

We need to calibration if at all really depends on what requires and trying to do. One method would be to work out the calibration value at a known altitude. e.g. at a known altitude get the pressure and then change the calibration value in the call in code till work get something close enough to the known altitude want out. Work need to be aware that the calibration value will change with the weather. How long it will be good enough for will depend on use case.

Unfortunately, barometers, as they are currently designed, are not well suited to miniaturization. The problem is that their sensitivity drops with decreasing size—by the time the barometer fits in cell phone.

REFERENCES

[1] Brush, A. B., Karlson, A. K., Scott, J., Sarin, R., Jacobs, A.,Bond, B., Murillo, O., Hunt, G., Sinclair, M., Hammil, K.,and Levi, S. User experiences with activity-based navigation
on mobile devices. MobileHCI, 2010.

[2] Lester, J., Choudhury, T., and Borriello, G. A practical approach to recognizing physical activities. Pervasive, 2006.

[03] Varshavsky, A., LaMarca, A., Hightower, J., and de Lara, E. The SkyLoc floor localization system. Percom, 2007.

[4] Ojeda, L. and Borenstein, J. Personal dead-reckoning system for GPS-denied environments. Safety, Security and Rescue Robotics, 2007. SSRR 2007. IEEE International Workshop
on, pages 1 –6, sept. 2007.

[5] Johannsson, H., Kaess, M., Fallon, M., and Leonard, J.Temporally scalable visual SLAM using a reduced pose graph. RSS Workshop on Long-term Operation of Autonomous Robotic Systems in Changing Environments,2012.

[6] Ye, H., Gu, T., Zhu, X., Xu, J., Tao, X., Lu, J., and Jin, N.FTrack: Infrastructure-free floor localization via mobile phone sensing. Percom, 2012.

[7] Chang, H., Tian, J., Lai, T., Chu, H., and Huang, P. Spinning beacons for precise indoor localization. SenSys, 2008.

[8] Kusy, B., Maroti, M., Balogh, G., Volgyesi, P., Sallai, J., Nadas, A., Ledeczi, A., and Meertens, L. Node density independent localization. IPSN/SPOTS, 2006.

[9] Muthukrishnan, K., Dulman, S., and Langendoen, K. Towards a rapidly deployable positioning system for emergency responders. International Conference on Ubiquitous,Positioning, Indoor Navigation and Location-Based Service, 2010.

[10] Priyantha, N. B., Chakraborty, A., and Balakrishnan, H. The Cricket Location-Support system. MobiCom, 2000.

[11] Kartik Muralidharan, Azeem Javed Khan, Archan Misra, Rajesh Krishna Balan, Sharad agarwal Singapore Management University, Oriental Institute of Management , Microsoft Research.

[12] V. Honkavirta, T. Perala, S. Ali-Loytty, and R. Piche, “A comparative survey of wlan location fingerprinting methods,” in Proc. WPNC ’09.

[13]An Intelligent Multi-sensor System for Pedestrian Navigation G. Retscher
Institute of Geodesy and Geophysics, Vienna University of Technology, Austria Gusshausstrasse 27-29, A – 1040 Wien, Austria

[14] Li, B. Terrestrial Mobile User Positioning Using TDOA and Fingerprinting Techniques.
Ph.D. Thesis, School of Surveying & Spatial Information Systems, University of New South Wales, Sydney, Australia, 2006.

[15] Floor Determination for Positioning in Multi-story Building Mohd Amiruddin Abd Rahman ∗†, Marzieh Dashti‡, Jie Zhang∗ ∗Department of Electronic and Electrical Engineering, University of Sheffield, UK †Department of Physics, Faculty of Science, Universiti Putra Malaysia, Malaysia