PROJECT REPORT

ON

WiPAt:WIFI POSITIONING SYSTEM FOR ATTENDANCE

Submitted By

RISHI SHRIDHAR (B120388669)

OMKAR KHALIPE (B120388641)

SAURABH SHANBHAG (B120388684)

KOMAL GANJALE (B120388560)

in partial fulfillment for the award of the degree

of

Bachelor of Engineering

of

Savitribai Phule Pune University

IN

INFORMATION TECHNOLOGY

MITCOE COLLEGE OF ENGINEERING

2016-2017

PROJECT REPORT

ON

WiPAt:WIFI POSITIONING SYSTEM FOR ATTENDANCE

Submitted By

RISHI SHRIDHAR (B120388669)

OMKAR KHALIPE (B120388641)

SAURABH SHANBHAG (B120388684)

KOMAL GANJALE (B120388560)

Guided By

Prof. Dr.A.S.Hiwale

DEPARTMENT OF INFORMATION TECHNOLOGY

MITCOE COLLEGE OF ENGINEERING

PUNE

SAVITRIBAI PHULE PUNE UNIVERSITY

2016-2017

DEPARTMENT OF INFORMATION TECHNOLOGY

Certificate

This is to certify that,

RISHI SHRIDHAR (B120388669)

OMKAR KHALIPE (B120388641)

SAURABH SHANBHAG (B120388684)

KOMAL GANJALE (B120388560)

have successfully completed this project report entitled "**Wipat:Wifi Positioning System for Attendance**", under my guidance in partial fulfilment of the requirements for the degree of Bachelor of Engineering in Department of Information Technology of SavitibaiPhule Pune University, Pune during the academic year 2016-17.

Date –

Place-

Prof. Dr.A.S.Hiwale — Guide

Prof. Dr. A.S. Hiwale — Head Of Department

Acknowledgement

We take this opportunity to thank our project guide **Prof.(Dr.) A. S. Hiwale** and Head of the Department Prof.(Dr.) A. S. Hiwale for their valuable guidance and for providing all the necessary facilities, which were indispensable in the completion of this project report. We are also thankful to all the staff members of the Department of Information Technology of MIT College of Engineering, Savitibai Phule Pune University for their valuable time, support, comments, suggestions and persuasion. We would also like to thank the institute for providing the required facilities, Internet access and important books. And finally, we would like to thank **Mr. Indra Singh**, our project guide from the company **Persistent Systems Ltd.** for his help as a mentor as a part of the project sponsorship our group has received.

Rishi Shridhar

Omkar Khalipe

Saurabh Shanbhag

Komal Ganjale

ABSTRACT

An accurate and transparent attendance marking system is of utmost importance in educational institutions as well as businesses. In today's fast-moving world, the main drawback in traditional attendance systems is its tediousness, and manual effort that the teacher has to put in. The primary motivation for this project is the need for automation that would reduce the time and efforts required for this monotonous job. An automated environment is attractive, reliable, advantageous, and desired by every person nowadays. This project will enable us to automate the process of marking attendance of students in a college, using Wi-Fi Positioning System, which works by using routers and mobile devices, both of which are pretty commonly used today, hence making the project cost-effective.

TABLE OF CONTENTS

CHP.NO.		CHAPTER NAME	PAGE NO.
1.		Introduction	1.
	a.	Need	2
	b.	Basic Concepts	2
	c.	Application	2
2.		Literature Survey	3
	a.	Related Work Done	4
	b.	Existing System	7
3.		Project Statement	8
	a.	Problem Statement	9
	b.	Goals and Objectives	9
	c.	Technologies Used	9
4.		System Requirement and Specification	13
	a.	Software Requirements	14
	b.	Hardware Requirements	14
5.		System Design	15
	a.	Concept	16
	b.	Flowchart	19
	c.	Data Flow Diagrams	20
	d.	UML Diagrams	22
6.		Working Code And Application	27
	a.	Locate.java	28
	b.	ScanBackground,java	38
7.		Planning And Sheduling	46
8.		References	49

LIST OF FIGURES

Fig. No.	Description	Page No.
1	Position estimator algorithm	9
2	Trilateration	11
3	System Architecture	12
4	Signals Emanating from Routers	16
5	8 Points and Calibrate Reading	16
6	Euclidean Distance from unknown point	17
7	Euclidean Distance from x and y	17
8	Nearest K Points are Chosen	18
9	Weighted Avg of Nearest 2 points	18
10	Flowchart	19
11	DFD – 0	20
12	DFD – 1	21
13	Use Case Diagram	22
14	Activity Diagram	23
15	Sequence Diagram	24
16	Collaboration Diagram	25
17	Class Diagram	26
18	Gantt Chart of Project	48

LIST OF TABLES

Table No.	Description	Page No.
1	Project Plan Table	48

Chapter 1

INTRODUCTION

1. INTRODUCTION

1.1 NEED

- Automation of attendance marking system
- Accuracy and transparency in attendance marking system
- Making efficient use of resources that are commonly available in today's world like Wi-Fi routers and mobile devices.

1.2 BASIC CONCEPTS

In recent years, many approaches have been used for running attendance systems such as biometric sensors, face recognitions, etc. This endeavor aims to increase efficiency and use minimal human intervention. The system will find and compare the signal strengths from different routers at a given position to determine the relative position of the device. The concept used for getting accurate signal strength is RSSI (Received Signal Strength Indicator) and for comparing and determining the position, weighted K-NN algorithm is implemented. Following this, it can automatically mark the attendance by checking if the device is inside the classroom during the specified period.

1.3 APPLICATION

- Attendance system in educational institutions
- Attendance system in offices

Chapter 2

LITERATURE SURVEY

a. RELATED WORK DONE

1. ***Peerapong Torteeka, Xiu Chundi:***

"Hybrid technique for indoor positioning system based on Wi-Fi received signal strength indication"- 2014, DOI-10.1109/IPIN.2014.7275467:

This IEEE paper is about an indoor positioning system based on Receive Signal Strength Indication (RSSI) from wireless access equipment. While outdoor environment using Global Navigation Satellite System (GNSS) and cellular network work well and are widespread used for navigation. However, there is a problem with signal propagation from satellites or cell site. They cannot be used effectively inside complex building areas or even in an urban environment. In this paper, the hybrid algorithm is proposed which is the combination of trilateration and location fingerprint (dataset mining) systems, which is able to improve the accuracy, stability and robustness. The performance of this algorithm is evaluated by the experimental results, which shows that the proposed scheme can achieve a certain level of positioning system accuracy.

Pros: The drawbacks of Satellite based positioning systems are effectively removed using this approach. Also, the hybrid approach reduces fluctuations caused by either of the two approaches.

2. An Analysis of Wi-Fi Based Indoor Positioning Accuracy

Gints Jekabsons, Vadim Kairish, Vadim Zuravlyov,(Riga Technical University)

The aim of this study is to examine several aspects of location fingerprinting based indoor positioning that affect positioning accuracy. Overall, the positioning accuracy achieved in the performed experiments is 2.0 to 2.5 meters. The increasing demand for location based services inside buildings has made indoor positioning a significant research topic. Although the Global Positioning System is the most popular outdoor positioning system, its signals are easily blocked by most construction materials making it useless for indoor positioning. This study deals with indoor positioning using the Wireless Ethernet IEEE 802.11 (Wi-Fi) standard that has distinct advantage of low cost over other indoor wireless technologies – it has relatively cheap equipment and in many areas usually a Wi-Fi network already exists as a part of the communication infrastructure avoiding expensive and time consuming infrastructure deployment. Although Wi-Fi has not been designed for positioning, its radio signals can be used for location estimation by exploiting the Received Signal Strength (RSS)

values measured in any off-the-shelf mobile device equipped with Wi-Fi facilities – and no additional special-purpose hardware is required. Such a positioning system can be relatively easily implemented for notebook computers, personal digital assistants (PDAs), smartphones, and other Wi-Fi enabled mobile devices.

Pros: The use of Wireless networks makes it very user friendly as it is cheap and present everywhere nowadays.

Cons: The small scale usage opens new problems such as power fluctuations, propagation obstacles, etc.

3. Optimization Wifi Indoor Positioning KNN Algorithm Location-based Fingerprint Xingbin Ge, Zhiyi Qu, School of Tnformation Science & Engineering, Lanzhou University

Software Engineering and Service Science (ICSESS), 2016 7th IEEE International Conference on 26-28 Aug. 2016

DOI: 10.1109/ICSESS.2016.7883033

Location-based services have been deep into all aspects of life and it provides a convenient and efficient service experience for people. Currently, technology is relatively mature and widely used in the outdoor positioning. By contrast, for indoor positioning, although there are a lot of hot technologies, but they are mostly insufficient lead to it is hard to popularize. So how to improve the popularity of indoor positioning in the case of improves the positioning accuracy has become a hot research topic. This paper analyzes and studies several typical fingerprint localization algorithms, including NN, KNN and WKNN, and then proposes an algorithmic improvement program.

Pros: The KNN approach helps the system to achieve accuracy by analyzing multiple values and then determining the best possible position. Hence, degree of accuracy is raised.

Cons: By this method, we can compare and decide, but the exact position cannot be mapped and hence, needs more optimization.

4. ***Soumaya Zirari, Philippe Canalda , François Spies , "WiFi GPS based combined positioning algorithm", 2010, DOI: 10.1109/WCINS.2010.5544653:***

This IEEE paper proposes a GPS-WiFi combined positioning algorithm, based on trilateration technique. Real experiments and other simulation are conduced and demonstrate accuracy gains, even where various criteria dilution of precision (GPS dop's criteria, or WiFi geometrical and signal attenuation's dop proposal, or hybrid dop ones) indicate all the disruption of positioning service. A testbed scenario issued from a real urban campus environment validates not only the GPS-WiFi combined positioning algorithm but also an implementation of pertinent positioning techniques and dop's criteria. This work constitutes a further step to better position everywhere and to ensure continuity of a positioning service.

Pros: The technique of trilateration is a very conceptually ideal method for finding precise location of a device. This is precisely the technique used by GPS using Satellites for triangulation.

Cons: Through intensive experiments, we found that this technique is not practically feasible. The fluctuations in RSSI and dependencies on device antenna strength, router capacity, power supply, obstacles, exponential distance function, etc. make it impossible to give a single accurate position.

5. ***Beom-Ju Shin, Kwang-Won Lee, Sun-Ho Choi, Joo-Yeon Kim, Woo Jin Lee, Hyung Seok Kim: "Indoor WiFi positioning system for Android-based smartphone", 2010, DOI: 10.1109/ICTC.2010.5674691:***

WiFi positioning system has been studied in many fields since the past. Recently, a lot of mobile companies are competing for smartphones. Accordingly, this IEEE paper proposes an indoor WiFi positioning system using Android-based smartphones. This emphasizes on how a device (Android in this case) interacts with routers, even when it isn't connected to the router or the internet. The MAC id of the device is used for unique identification and positioning.

Pros: Scalability increases and effective use of concept is possible.

b. EXISTING SYSTEMS:

Over recent years, the potential of Location-based Services has been hotly debated, with several methods being put forward to provide location information. GPS has emerged as the leading method of providing position information but it has **limited use indoors**. There has been a lot of research into methods which are capable of providing indoor location information, however most of these require **specialized equipment** or have **lengthy set-up procedures**. We are therefore developing a system which has neither of these drawbacks. We are utilizing the standard Received Signal Strength (RSS) measurements available on Wi-Fi equipment because Wi-Fi networks are becoming increasingly available, meaning that extra specialized equipment is not required. The currently present models for indoor positioning systems use the technique of trilateration with 3 routers. But the drawbacks of this approach include **signal fluctuations** and **obstacles** in signal propagation. We have devised a method to take care of all these problems.

Chapter 3

PROJECT STATEMENT

a. PROBLEM STATEMENT:

To create a system that can **automate** the process of marking attendance by using the concept of **Wi-Fi Positioning System**.

Goals And Objectives:

- Automate the procedure of marking attendance.
- Increase the accuracy and efficiency of marking attendance
- Reducing the chances of fraudulent activity i.e. increase reliability.

b. TECHNOLOGIES USED:

To build this project, the following technologies have been used:

1. Location Fingerprinting

• Received Signal Strength (RSS) values from multiple routers act as a fingerprint for a location. Different locations are most likely to have unique fingerprints

• Fingerprint of ith location is denoted by ri

ri = {ri1, ri2, ri3,.....,rim }

rij = RSS value of jth router from ith location

• Several locations are chosen and RSS values from them are recorded and form a radio map.

• The recording at the ith location is of the form : (qi,ri)

qi - the geometric coordinates (xi,yi)

ri - the location fingerprint

• A position estimator algorithm is used to find the coordinates of unknown location.

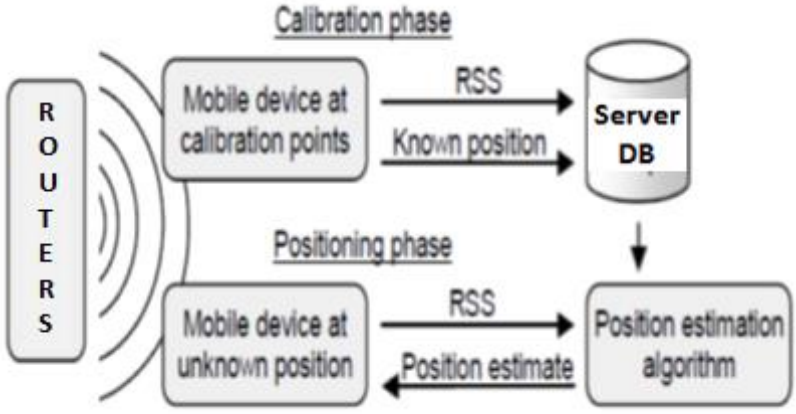


Fig:1 Basic concept (calibrate + locate)

2. Weighted k Nearest Neighbour Algorithm :

• The position estimator algorithm used is the Weighted k Nearest Neighbour (WkNN).

• Finds the k nearest chosen locations from unknown location based on Euclidean Distance.

• Calculates coordinates of unknown location as the weighted average of the nearest k points. Weight is the inverse of the Euclidean distance.

• k can be considered as a tuning parameter in the algorithm.

• When k=1, algorithm acts as a simple look up table

$$F(x_q) = \frac{\sum_{i=1}^{k} w_i f(x_i))}{\sum_{i=1}^{k} w_i}$$

3. Received signal strength indicator (RSSI):

Received signal strength indicator (RSSI) is a measurement of the power present in a received radio signal. [5]. RSSI is usually invisible to a user of a receiving device. However, because signal strength can vary greatly and affect functionality in wireless networking, IEEE 802.11 devices often make the measurement available to users. The RSSI is used to calculate the distance of the mobile device from the router.

Unit of Signal Strength - DBm:

dBm (sometimes dBmW or decibel-milliwatts) is an abbreviation for the power ratio in decibels (dB) of the measured power referenced to one milliwatt (mW). It is used in radio, microwave and fiber optic networks as a convenient measure of absolute power because of its capability to express both very large and very small values in a short form. dBm is used to measure the beacon strength of the router signal.

4. Trilateration (Alternative Method):

Trilateration is the process of determining absolute or relative locations of points by measurement of distances, using the geometry of circles, spheres or triangles. [4] This method has been used to locate the position of the mobile device. We initially tried to make a working model of trilateration using 3 routers. But owing to the huge number of uncertainties and fluctuations, we decided to change the approach and look for a more accurate approach in KNN technique.

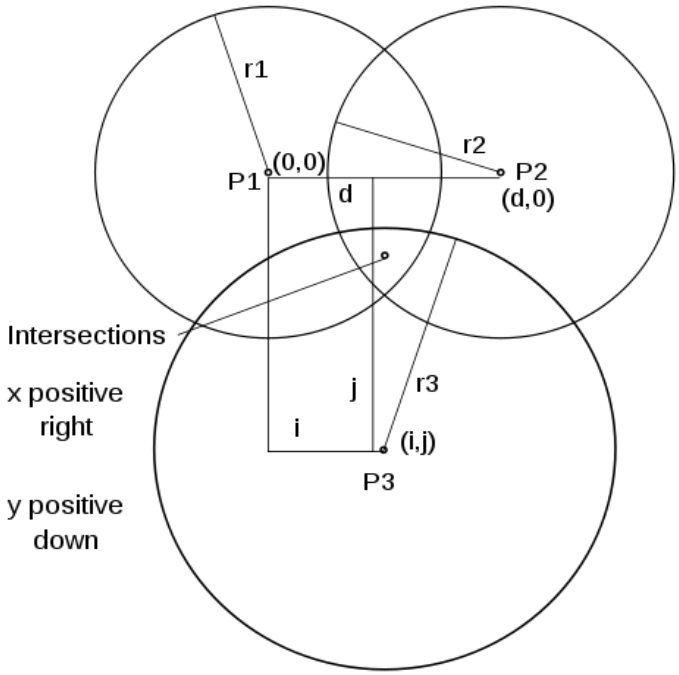


Fig 2. Trilateration

SYSTEM ARCHITECTURE

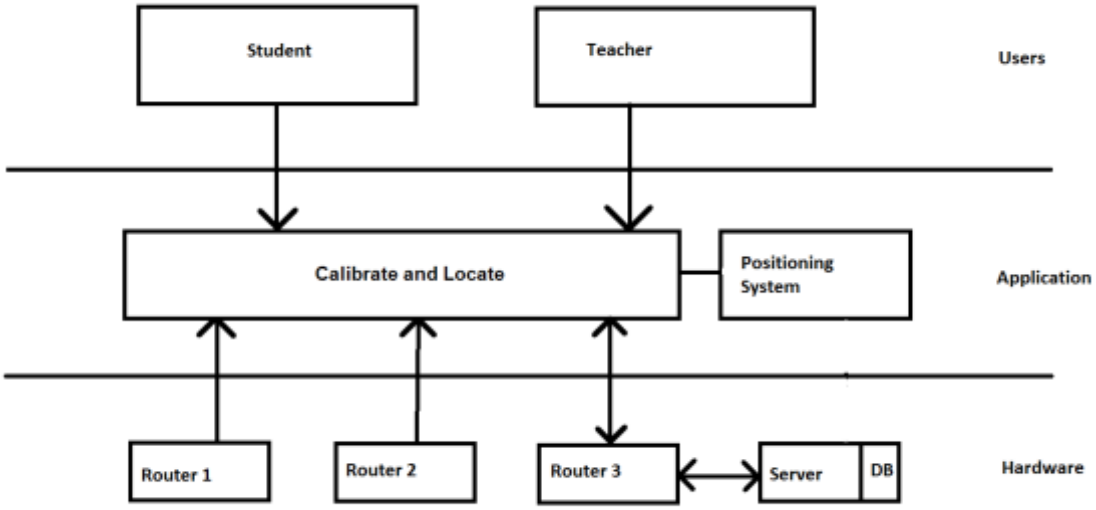


Fig.3. System Architecture

Chapter 4

SYSTEM REQUIREMENTS AND SPECIFICATIONS

4. SYSTEM REQUIREMENTS AND SPECIFICATIONS

4.1 SOFTWARE REQUIREMENTS:

A: Developer

Operating System	: Windows 7/ 8/ 10
Programming Language	: JAVA (Android)
Java Version	: JDK 1.6 & above.
IDE	: Android Studio
Database	: MySQL 5.5.
Web Server	: Apache

B: User

Operating System	: Android

4.2 HARDWARE REQUIREMENTS:

A: Developer

Wireless Routers	: Quantity: 3 or more, Signal Strength: 150m

B: User

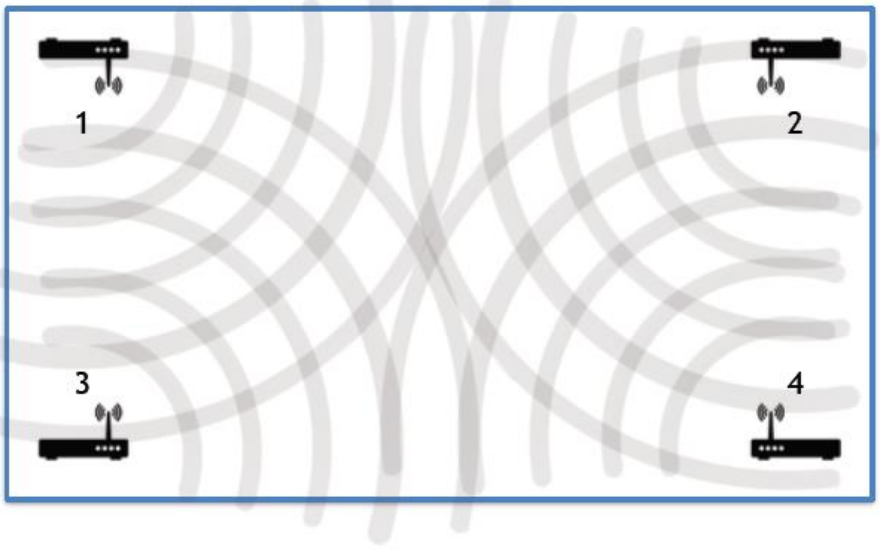
Mobile phone	: Android Smartphone

Chapter 5

SYSTEM DESIGN

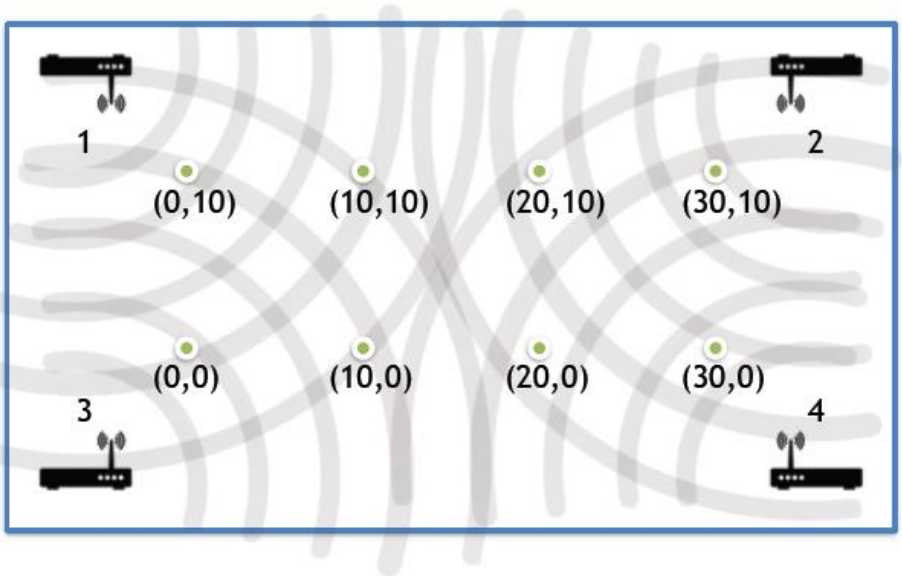
5. System Design:

a. CONCEPT:



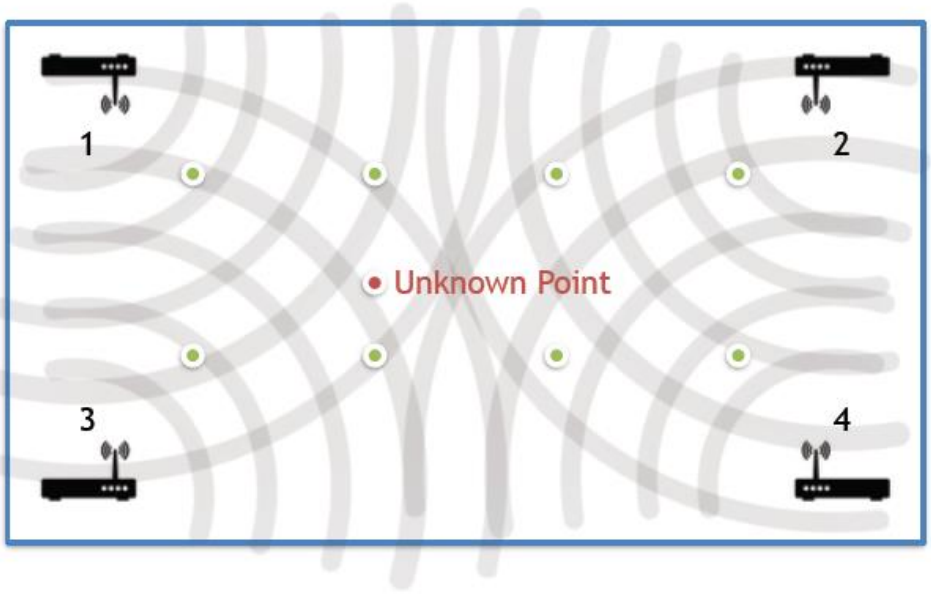
The signals emanating from routers leave a unique fingerprint at each location

Fig:4



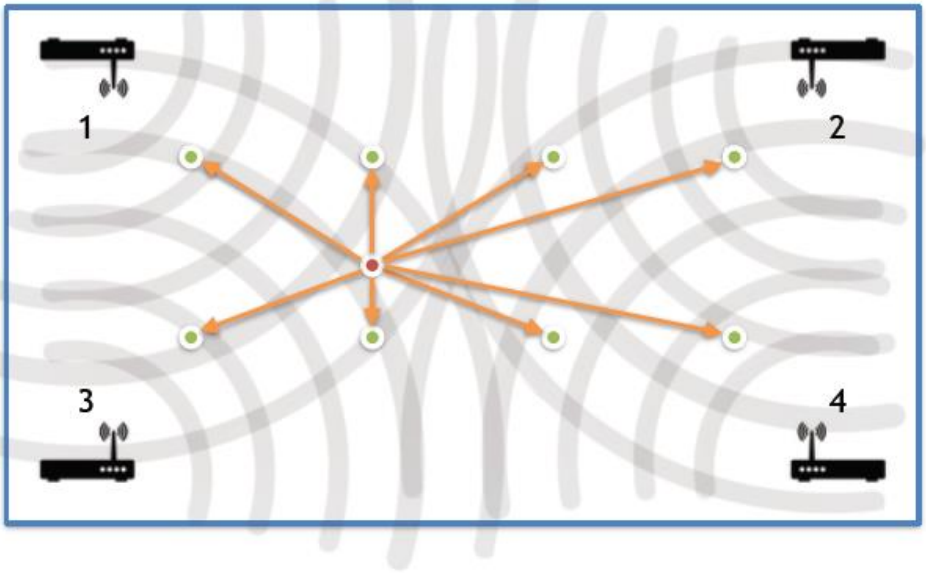
Choose 8 points on the floor and calibrate readings

Fig:5



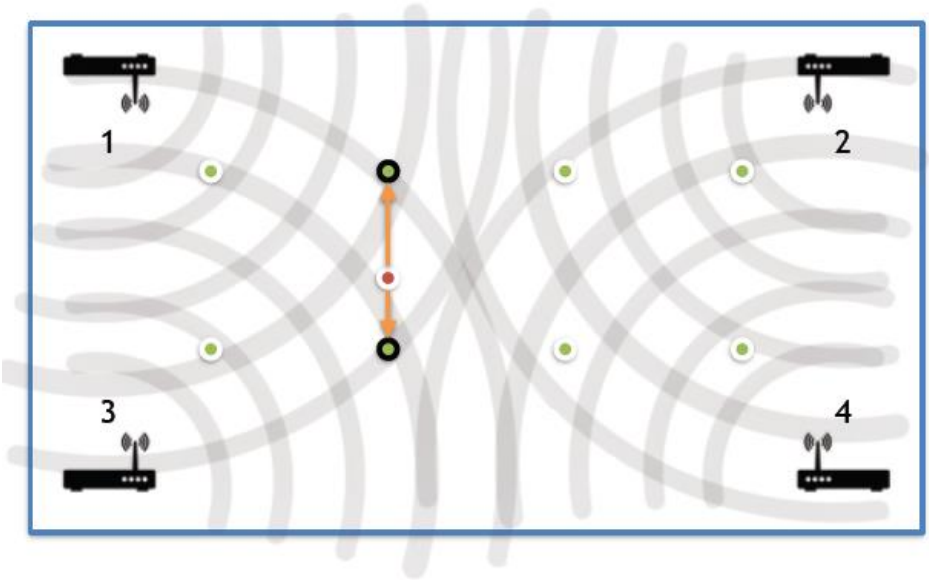
Calculate Euclidean distance from unknown point to all chosen points

Fig:6



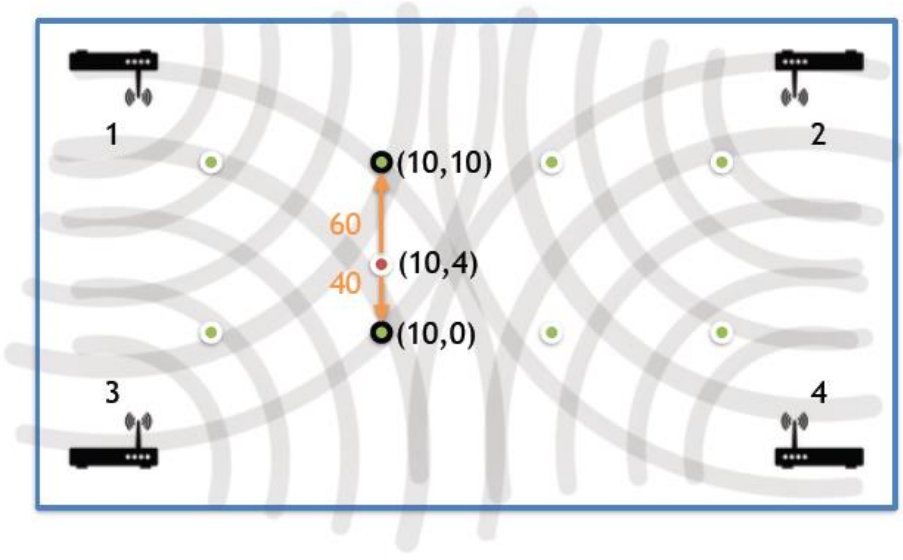
Euclidean distance between points x and y is calculated by
$\text{sqrt}((r_{x1}-r_{y1})^2+(r_{x2}-r_{y2})^2+(r_{x3}-r_{y3})^2+(r_{x4}-r_{y4})^2)$

Fig:7



Nearest k points are chosen. Here k = 2.

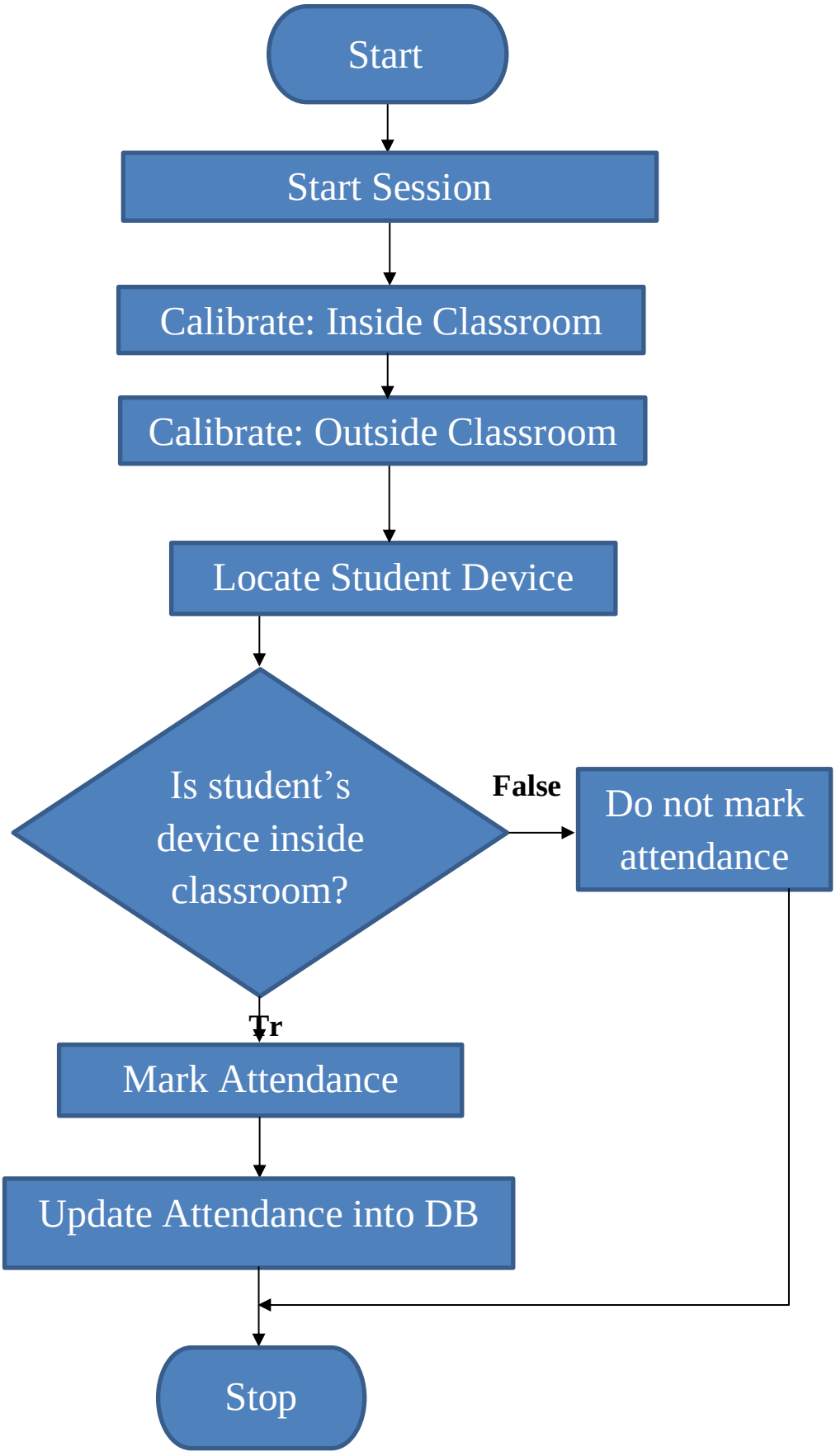
Fig:8



Coordinate of unknown location is the weighted average of the 2 nearest points

Fig:9

b.FLOWCHART


Start
Start Session
Calibrate: Inside Classroom
Calibrate: Outside Classroom
Locate Student Device
Is student's device inside classroom?
False
Do not mark attendance
Tr
Mark Attendance
Update Attendance into DB
Stop

c. DATA FLOW DIAGRAMS:

- **DFD 0:**

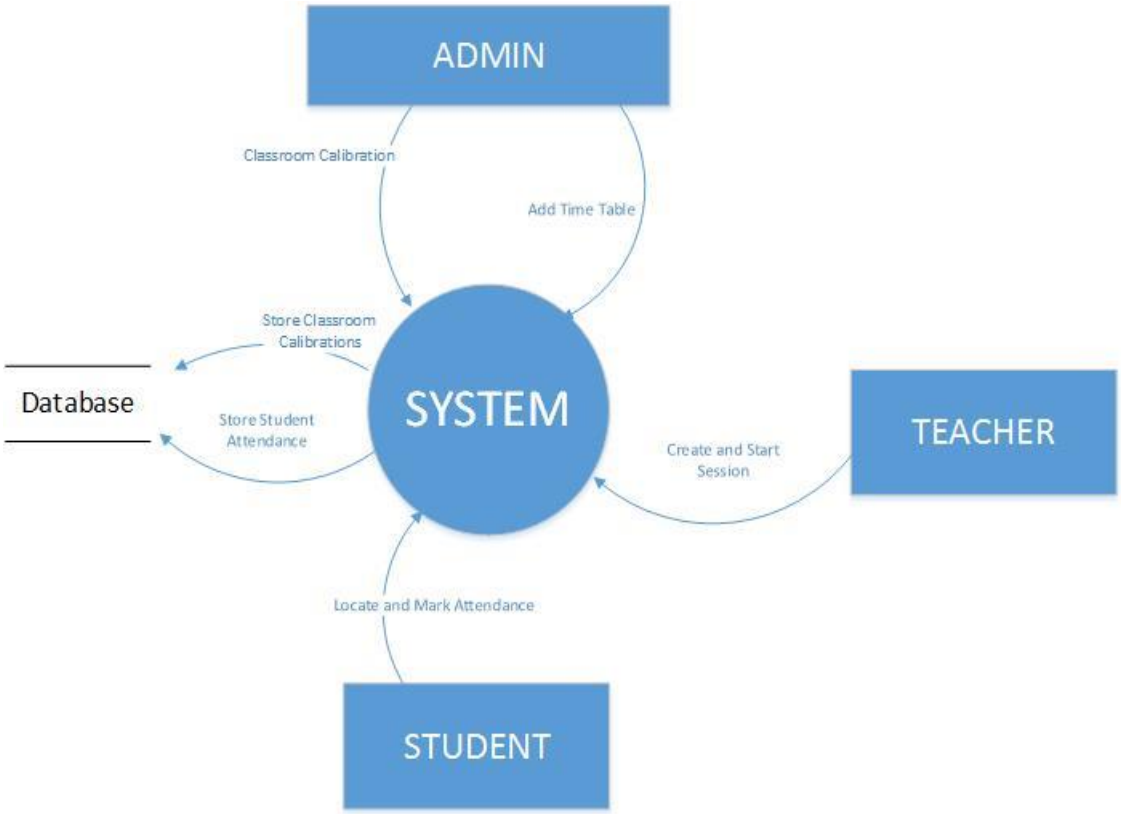


Fig 11: DFD - 0

- **DFD 1:**

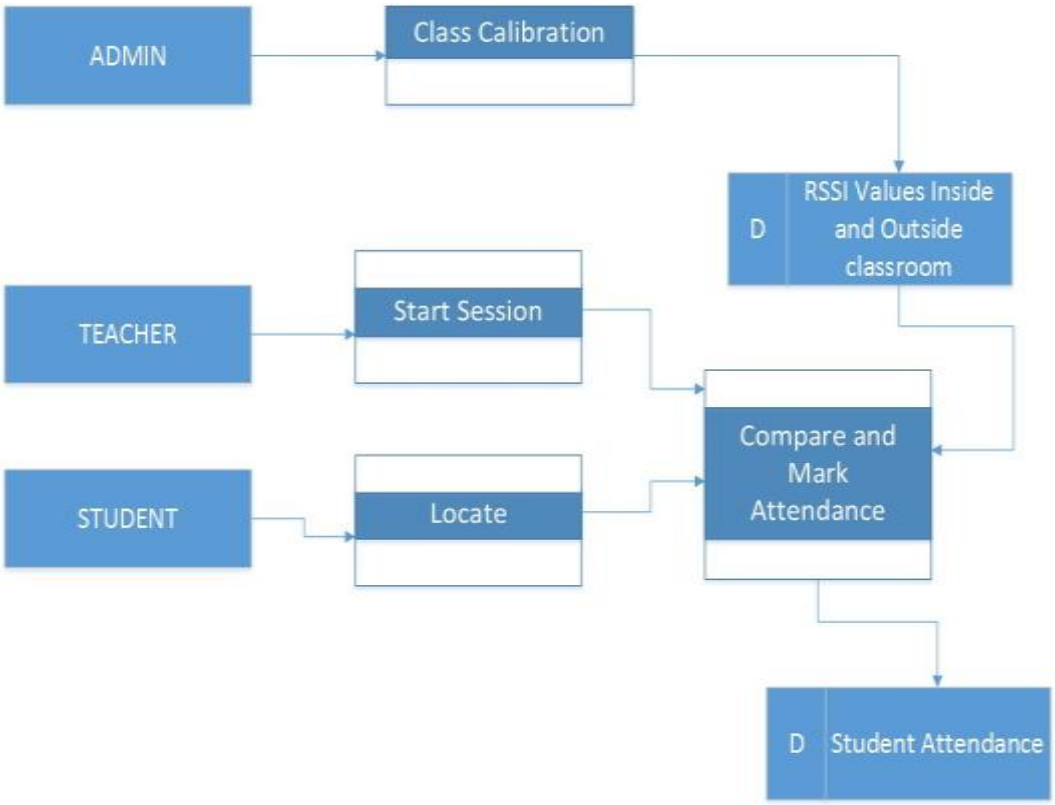


Fig 12: DFD - 1

d.UML DIAGRAMS

- **Use Case diagram**

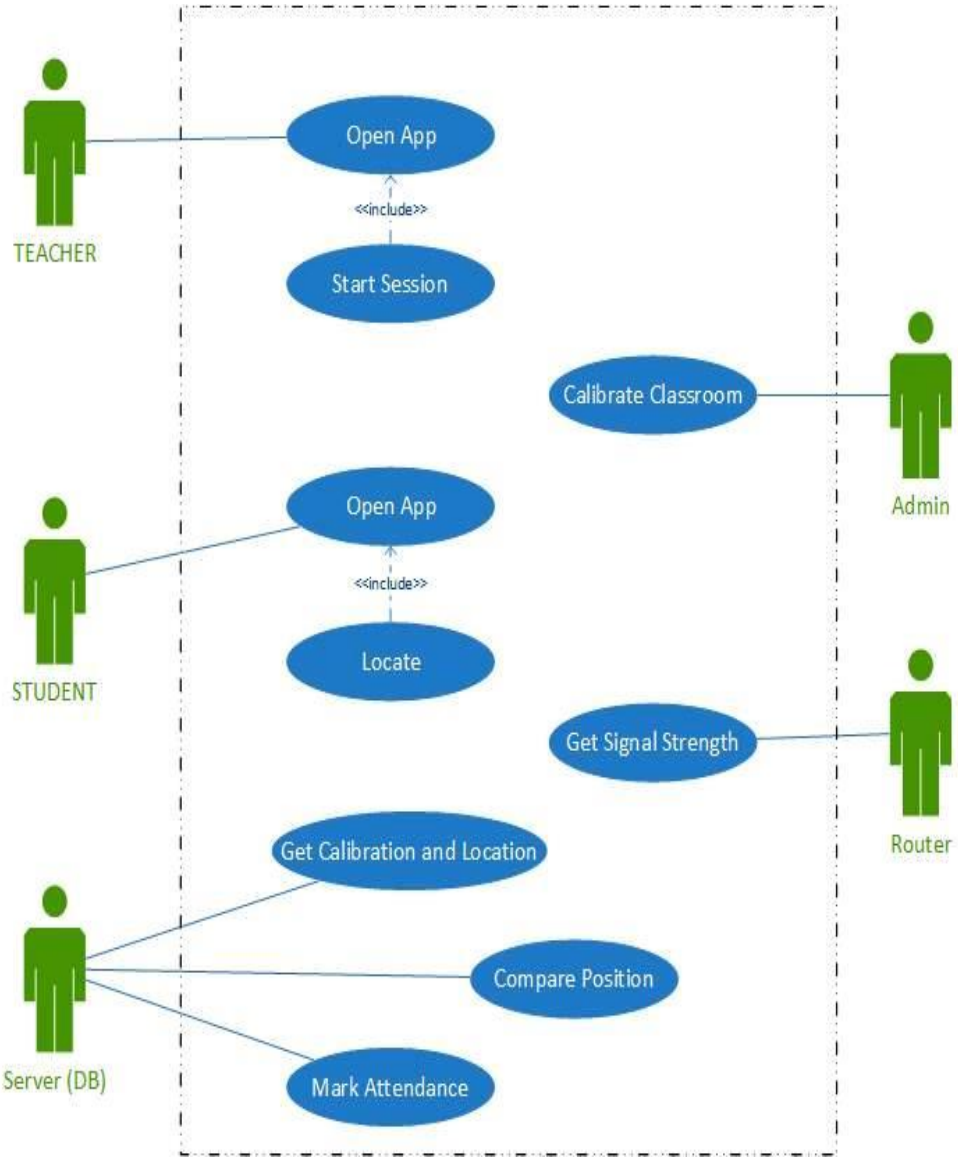


Fig 13. Use case diagram

- **Activity diagram**

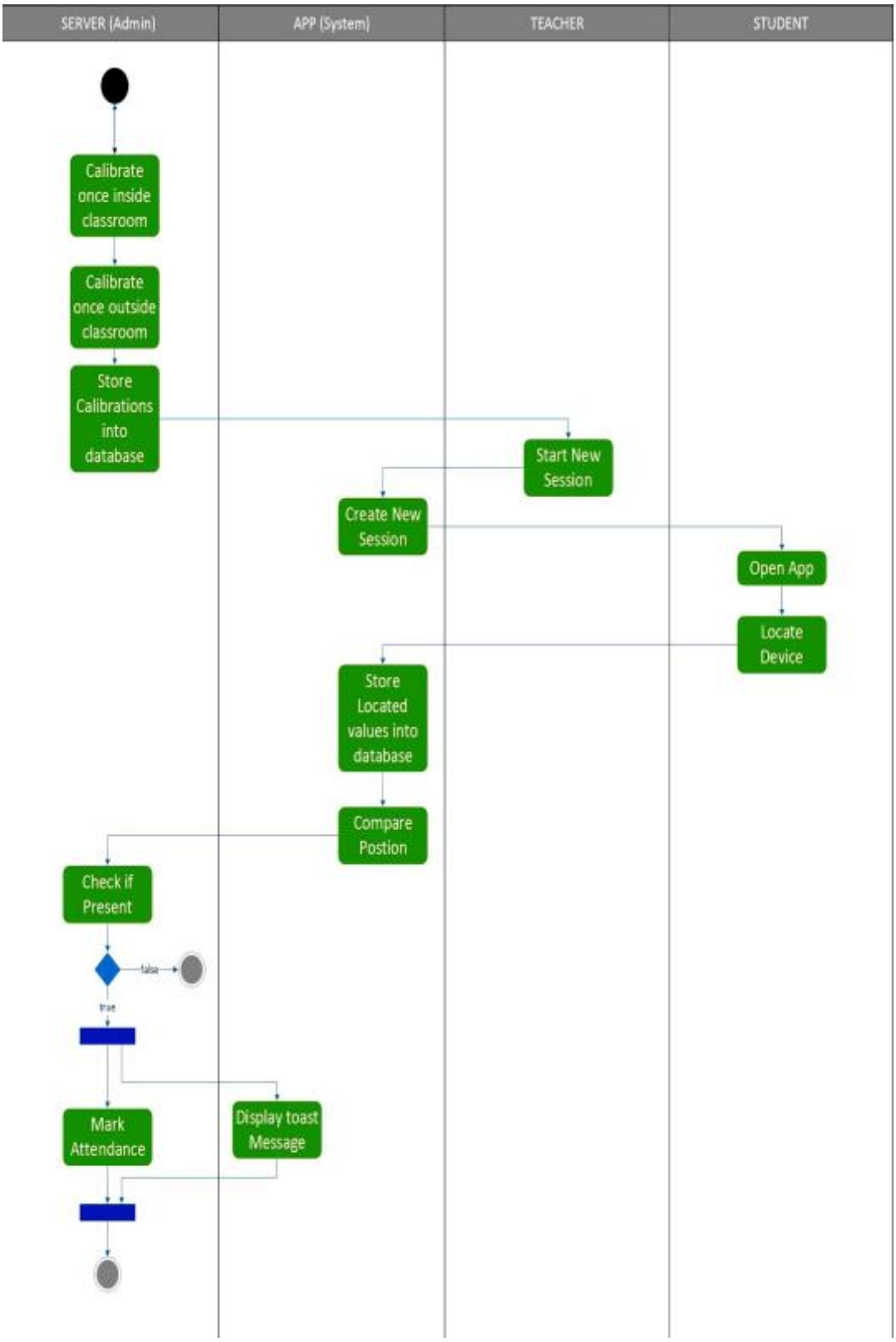


Fig 14: Activity diagram

- **Sequence (Interaction) diagram**

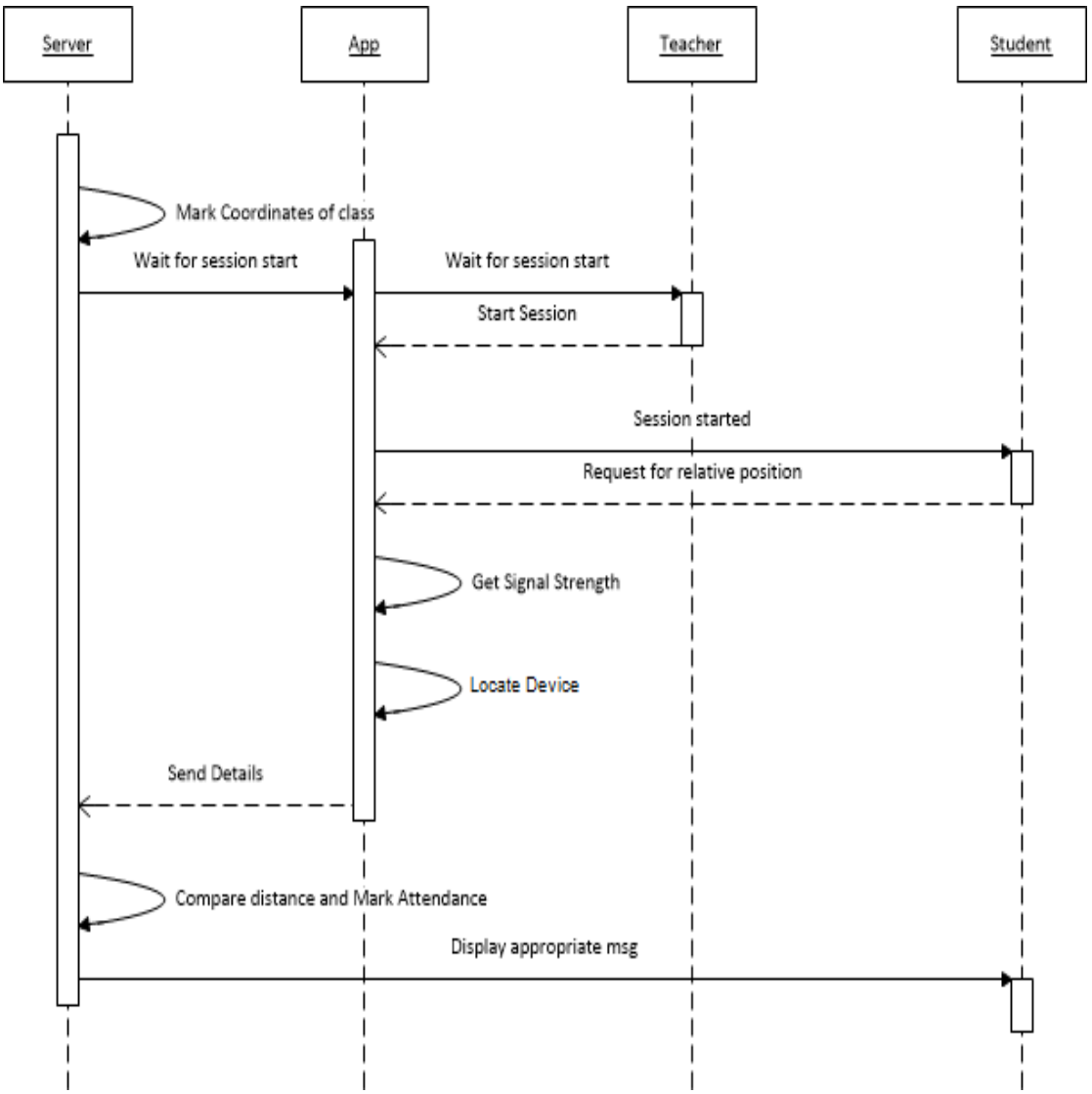


Fig 15: Sequence diagram

- **Collaboration diagram**

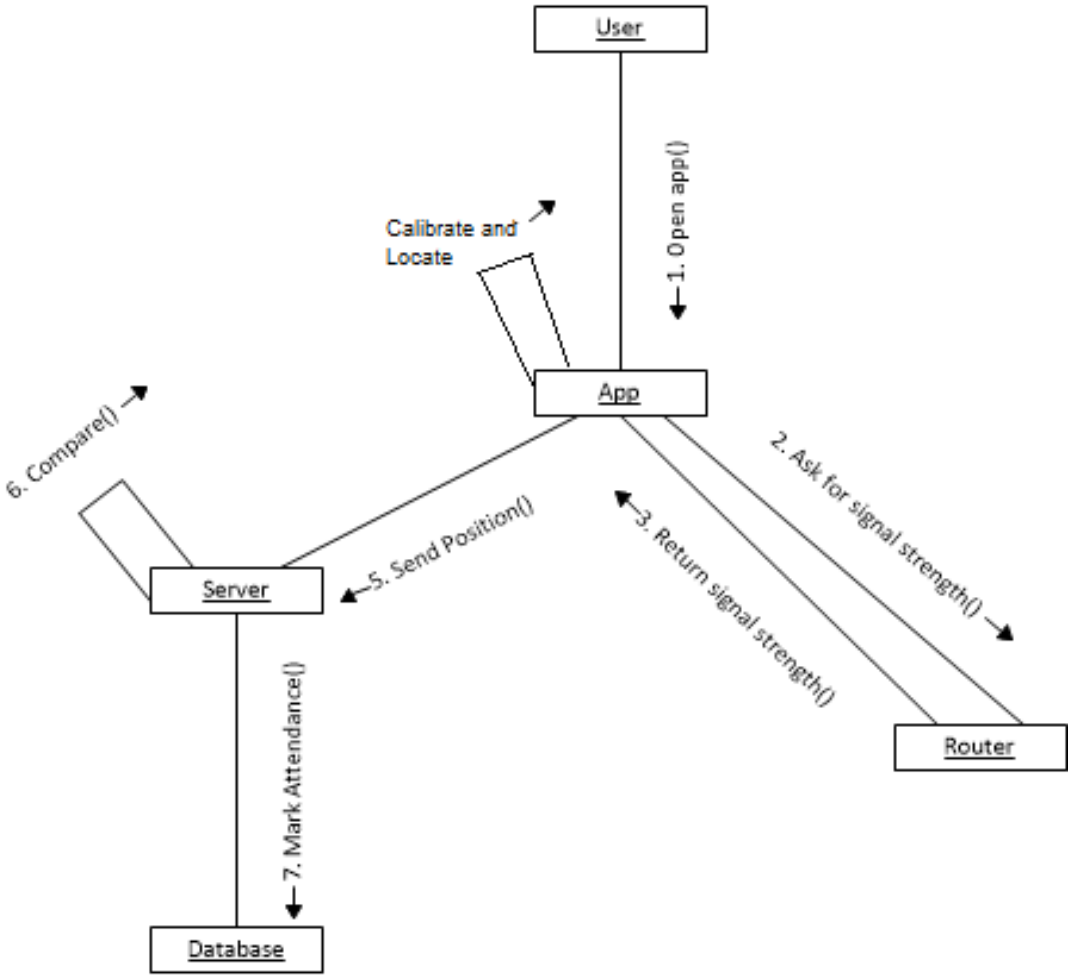


Fig 16: Collaboration Diagram

Class diagram

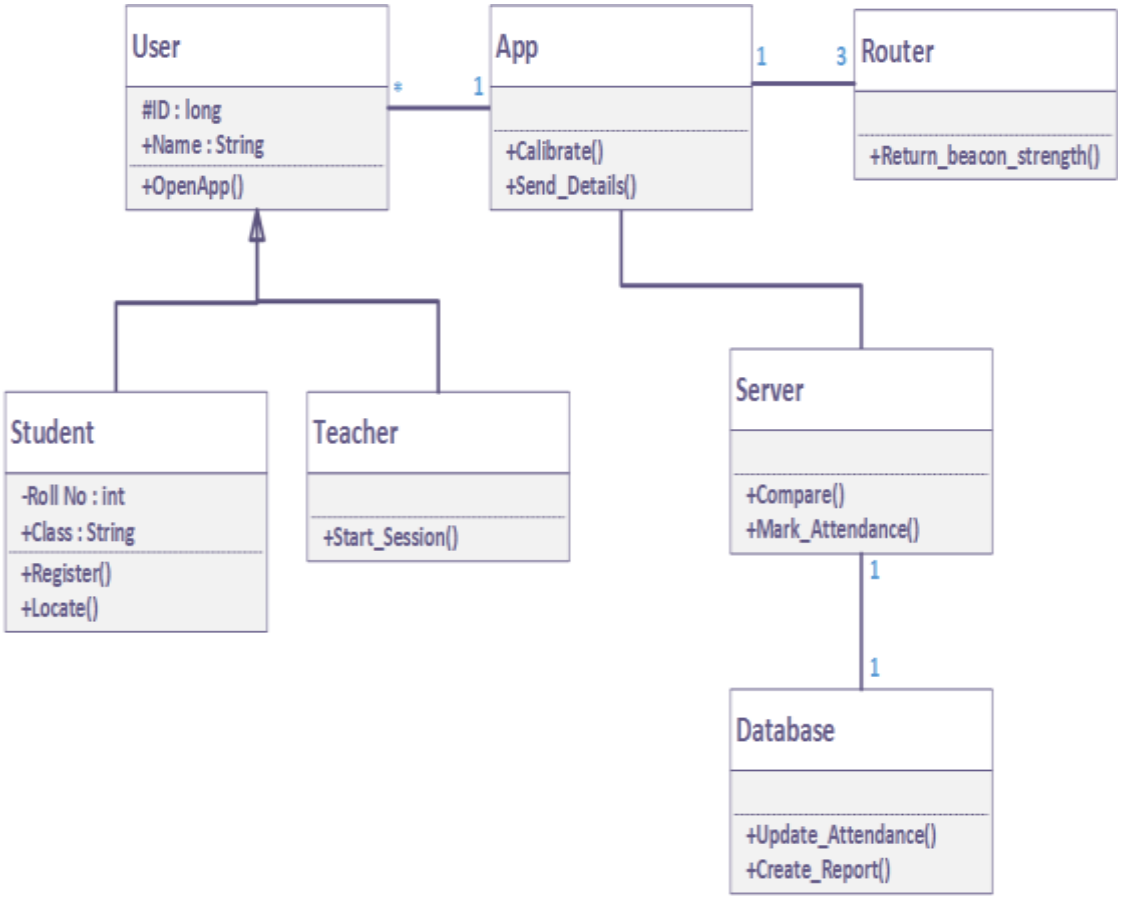


Fig 17: Class diagram

Chapter 6

WORKING CODE AND APPLICATION

a. Locate.java –

```
package com.example.wipat;

import java.net.NetworkInterface;
import java.util.ArrayList;
import java.util.Collections;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import com.android.volley.Request;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
/*
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.HttpClient;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.message.BasicNameValuePair;
import org.apache.http.util.EntityUtils;*/
import org.json.JSONArray;
import org.json.JSONException;

public class Locate extends Activity {

    String res;
    int posflag=0;
    ArrayList<String> buildings;
    DatabaseHelper db;
    ArrayAdapter<String> arrayAdapter;
    ArrayList<PositionData> positionsData;
    String b;
    TextView result1;
    Button locate;
    TextView result2;
    JSONArray jsonObject;
    int flag;

    String classid,lno,sub,topic,rollno;
    StringBuilder current_mac;
    String mac = "0";


    public void btn(View view) {

        Intent intent = new Intent(getApplicationContext(), Scan.class);

        intent.putExtra("isLearning", false);
        startActivityForResult(intent, 0);
    }

    public void onCreate(Bundle saveInstanceState) {
        super.onCreate(saveInstanceState);
```

```
        setContentView(R.layout.locate);
        db = new DatabaseHelper();

        try{
            List<NetworkInterface> all =
Collections.list(NetworkInterface.getNetworkInterfaces());
            for (NetworkInterface networkInterface : all) {
                if (!networkInterface.getName().equalsIgnoreCase("wlan0"))
continue;

                byte[] macBytes = networkInterface.getHardwareAddress();
                if (macBytes == null) {
                    current_mac.append("00:00:00:00:00:00");
                }

                current_mac = new StringBuilder();
                for (byte b : macBytes) {
                    current_mac.append(String.format("%02X", b));
                }

                if (current_mac.length() > 0) {
                    current_mac.deleteCharAt(current_mac.length() - 1);
                }
            }

            /**************finish finding current mac*************/

        } catch (Exception e) {
            Toast.makeText(getApplicationContext(),"Exception:
"+e,Toast.LENGTH_LONG).show();
        }

        result1 = (TextView) findViewById(R.id.result);
        result2 = (TextView) findViewById(R.id.result2);
        arrayAdapter = new ArrayAdapter<String>(this,
                android.R.layout.simple_spinner_item, buildings);


        arrayAdapter = new ArrayAdapter<String>(this,
                android.R.layout.simple_list_item_1, buildings);
        // Set The Adapter


        /*********start volley***************/
        String ur="http://192.168.0.101/wipat/api.php";
        StringRequest post = new StringRequest(Request.Method.POST, ur,
                new Response.Listener<String>() {


                    @Override
                    public void onResponse(String response) {
                        try {
                            JSONArray jsonObject = new JSONArray(response);

                            mac =
jsonObject.getJSONObject(0).getString("mac");

                        } catch (JSONException e) {
                            Toast.makeText(getApplicationContext(),
response, Toast.LENGTH_LONG).show();
                            /*    Intent i = new
Intent(Locate.this,StudentRegister.class);
                            startActivity(i);*/
                            //e.printStackTrace();
                        }
```

```
                        if(mac.equals(current_mac.toString()))
                            flag=1;
                        else
                            flag=0;

//Toast.makeText(getApplicationContext(),"Successfully
registered",Toast.LENGTH_LONG).show();

                        Button btnPos=(Button)findViewById(R.id.btnPos);
                        if(flag==0)
                            btnPos.setEnabled(false);
                        else
                            btnPos.setEnabled(true);
                    }
                },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {
                        // error
                        Toast.makeText(getApplicationContext(), "req 13.
"+error.toString(), Toast.LENGTH_LONG).show();
                        Log.d("Error.Response", error.toString());
                    }
                }

        ) {

            @Override
            protected Map<String, String> getParams() {


                Map<String, String> params = new HashMap<String, String>();
                params.put("req","req13");
                params.put("current_mac",current_mac.toString());


                return params;


            }


        };

MySingleton.getInstance(getApplicationContext()).addToRequestQueue(post);

        /************end volley***************/



        /*********start volley***************/
        String u="http://192.168.0.101/wipat/api.php";
        StringRequest pos = new StringRequest(Request.Method.POST, u,
                new Response.Listener<String>() {


                    @Override
                    public void onResponse(String response) {
                        try {
                            JSONArray jsonObject = new JSONArray(response);
                            classid =
jsonObject.getJSONObject(0).getString("classid");
                            rollno=
jsonObject.getJSONObject(0).getString("rollno");
                        } catch (JSONException e) {
                            e.printStackTrace();
                        } Log.d("response",response);
```

```
                    }
                },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {
                        // error
                        Toast.makeText(getApplicationContext(),
error.toString(), Toast.LENGTH_LONG).show();
                        Log.d("Error.Response", error.toString());
                    }
                }

        ) {

            @Override
            protected Map<String, String> getParams() {


                Map<String, String> params = new HashMap<String, String>();
                params.put("req","req14");
                params.put("current_mac",current_mac.toString());


                return params;


            }


        };

MySingleton.getInstance(getApplicationContext()).addToRequestQueue(pos);

        /************end volley***************/


    }


    @Override
    protected void onActivityResult(final int requestCode, final int
resultCode,
                                    final Intent intent) {
        // TODO Auto-generated method stub
        if (resultCode == RESULT_OK) {


            final PositionData positionData = (PositionData) intent
                    .getSerializableExtra("PositionData");

            DatabaseHelper db = new DatabaseHelper();
            db.deleteReading(positionData.getName(),
getApplicationContext());

            if (flag == 1) {


                /*********start volley***************/
                String u = "http://192.168.0.101/wipat/api.php";
                StringRequest pos = new StringRequest(Request.Method.POST,
u,
                        new Response.Listener<String>() {


                            @Override
                            public void onResponse(String response) {
                                try {
                                    JSONArray jsonObject = new
```

```
JSONArray(response);
                                    lno =
jsonObject.getJSONObject(0).getString("lno");
                                    sub =
jsonObject.getJSONObject(0).getString("sub");

                                } catch (JSONException e) {

Toast.makeText(getApplicationContext(),response,Toast.LENGTH_LONG).show();
                                    Intent i = new
Intent(Locate.this,StartingScreen.class);
                                    startActivity(i);
                                }
                            }
                        },
                        new Response.ErrorListener() {
                            @Override
                            public void onErrorResponse(VolleyError error) {
                                // error
                                Log.d("Error.Response", error.toString());
                            }
                        }

                ) {

                    @Override
                    protected Map<String, String> getParams() {


                        Map<String, String> params = new HashMap<String,
String>();
                        params.put("req", "req12");
                        params.put("classid", classid);


                        return params;


                    }


                };

MySingleton.getInstance(getApplicationContext()).addToRequestQueue(pos);

                /************end volley***************/


                String closestPosition = null;
                final ArrayList<Router> result = new ArrayList<Router>();
                int a = 5;
                String url = "http://192.168.0.101/wipat/api.php";
                StringRequest postRequest = new
StringRequest(Request.Method.POST, url,
                        new Response.Listener<String>() {


                            @Override
                            public void onResponse(String response) {
                                try {
                                    int db1[] = new int[10];
                                    int db2[] = new int[10];
                                    HashMap<String, PositionData> positions
= new HashMap<String, PositionData>();

                                    jsonObject = new JSONArray(response);
                                    for (int i = 0; i < jsonObject.length();
i++) {
```

```
                                        result.add(new
Router(jsonObject.getJSONObject(i).getString("ssid"),
jsonObject.getJSONObject(i).getString("mac_id")));

                                    }
                                    int x = 0;
                                    String routt[] = new String[40];
                                    int mnn[] = new int[3];
                                    mnn[0] = 100;
                                    mnn[1] = 100;
                                    mnn[2] = 100;
                                    for (int i = 0; i < jsonObject.length();
i++) {
                                        int myval =
Integer.parseInt(jsonObject.getJSONObject(i).getString("rssi"));
                                        String mymac =
jsonObject.getJSONObject(i).getString("mac_id");


                                        int min = 100;

                                        for (final Map.Entry<String,
Integer> e : positionData.getValues().entrySet()) {
                                            final String s1 = e.getKey();//
current mac
                                            final int s2 = e.getValue();
//current rssi

                                            if (mymac.equals(s1)) {

                                                int sub = Math.abs(s2 -
myval);

                                                if (sub <= mnn[i / 2]) {
                                                    mnn[i / 2] = sub;

                                                    routt[i / 2] =
jsonObject.getJSONObject(i).getString("position_id");

                                                }
                                            }
                                            //
db1[i]=Integer.parseInt(jsonObject.getJSONObject(i).getString("rssi"));

                                        }
                                        x++;

                                    }


                                    int f = 0, s = 0;

                                    //int k=0;
                                    for (int i = 0; i < 3; i++) {
                                        if
(jsonObject.getJSONObject(0).getString("position_id").length() == 0)

Toast.makeText(getApplicationContext(), "One or more friendly wifis is out
of range. Please check.", Toast.LENGTH_LONG).show();
                                        else if
(routt[i].equals(jsonObject.getJSONObject(0).getString("position_id"))) {
                                            f++;
                                        }

Log.d("jsonob2nd",""+jsonObject.getJSONObject(1).getString("position_id"));
                                        if
(jsonObject.getJSONObject(1).getString("position_id").length() == 0)
```

```
Toast.makeText(getApplicationContext(), "One or more friendly wifis is out
of range. Please check.", Toast.LENGTH_LONG).show();
                                        else if
(routt[i].equals(jsonObject.getJSONObject(1).getString("position_id"))) {
                                            s++;
                                        }

                                    }

                                    if (f > s) {
                                        res =
jsonObject.getJSONObject(0).getString("position_id");
                                        posflag=1;
                                    }
                                    else {
                                        res =
jsonObject.getJSONObject(1).getString("position_id");
                                        posflag=0;
                                    }

                                    /*************nested volley
request*****************************/

                                    if (posflag==1) {

                                        /*********start
volley***************/
                                        String uuu =
"http://192.168.0.101/wipat/api.php";
                                        StringRequest posss = new
StringRequest(Request.Method.POST, uuu,
                                                new
Response.Listener<String>() {


                                                    @Override
                                                    public void
onResponse(String response) {

//Toast.makeText(getApplicationContext(),response,Toast.LENGTH_LONG).show();
                                                    }
                                                },
                                                new Response.ErrorListener()
{
                                                    @Override
                                                    public void
onErrorResponse(VolleyError error) {
                                                        // error
                                                    //
Toast.makeText(getApplicationContext(), "req 8 Your mobile device isn't
registered. Please register via student registration. " +error.toString(),
Toast.LENGTH_LONG).show();

Log.d("Error.Response", error.toString());
                                                    }
                                                }

                                        ) {

                                            @Override
                                            protected Map<String, String>
getParams() {


                                                Map<String, String> params =
new HashMap<String, String>();

                                                params.put("req", "req8");
```

```
                                                params.put("lecture_no",
lno);
                                                ;
                                                params.put("Class_id",
classid);
                                                params.put("subj_id", sub);


                                                return params;


                                            }


                                        };
MySingleton.getInstance(getApplicationContext()).addToRequestQueue(posss);

                                        /************end
volley***************/


                                        /*********start
volley***************/
                                        String uu =
"http://192.168.0.101/wipat/api.php";
                                        StringRequest poss = new
StringRequest(Request.Method.POST, uu,
                                                new
Response.Listener<String>() {

                                                    @Override
                                                    public void
onResponse(String response) {

Toast.makeText(getApplicationContext(), response, Toast.LENGTH_LONG).show();
                                                    }
                                                },
                                                new Response.ErrorListener()
{
                                                    @Override
                                                    public void
onErrorResponse(VolleyError error) {
                                                        // error
                                                        //
Toast.makeText(getApplicationContext(), "req 7 Your mobile device isn't
registered. Please register via student registration. "+error.toString(),
Toast.LENGTH_LONG).show();

Log.d("Error.Response", error.toString());
                                                    }
                                                }

                                        ) {

                                            @Override
                                            protected Map<String, String>
getParams() {


                                                Map<String, String> params =
new HashMap<String, String>();

                                                params.put("req", "req7");
                                                params.put("student_id",
rollno);
                                                params.put("lecture_no",
lno);
                                                params.put("topic", topic);
```

```
                                                params.put("attendance",
"1");
                                                params.put("Class_id",
classid);
                                                params.put("subj_id", sub);

                                                return params;

                                            }

                                        };
MySingleton.getInstance(getApplicationContext()).addToRequestQueue(poss);
                                        /************end
volley***************/



                                    }
                                    else
                                        Toast.makeText(Locate.this,"You are
outside the class or too far away from the centre. Attendance not
marked.",Toast.LENGTH_LONG).show();

                                    /***************end nested volley
request*****************/



                                    Log.d("Final result is: ", res);
                                    result1.setText("Your position is :  " +
res);

                                    System.out.println("Reading done");
                                    positionsData = new
ArrayList<PositionData>();
                                    for (Map.Entry<String, PositionData> e :
positions.entrySet())
                                        positionsData.add(e.getValue());


                                } catch (JSONException e) {
                                    Toast.makeText(getApplicationContext(),
"One or more friendly wifis is out of range. Please check.",
Toast.LENGTH_LONG).show();

                                }

                            }
                        },
                        new Response.ErrorListener() {
                            @Override
                            public void onErrorResponse(VolleyError error) {
                                // error
                                // Toast.makeText(getApplicationContext(),
"Enter correct IP address of server via admin login!!!",
Toast.LENGTH_LONG).show();
                                Log.d("Error.Response", error.toString());
                            }
```

```
                        }

                ) {

                    @Override
                    protected Map<String, String> getParams() {


                        Map<String, String> params = new HashMap<String,
String>();
                        params.put("req", "req6"); //make r1 global
                        // params.put("p", position_id);


                        return params;

                    }


                };

MySingleton.getInstance(getApplicationContext()).addToRequestQueue(postReque
st);


            }
            else
                Toast.makeText(getApplicationContext(), "Error.",
Toast.LENGTH_LONG).show();

        }
        super.onActivityResult(requestCode, resultCode, intent);
    }

}
```

b. ScanBackground.java

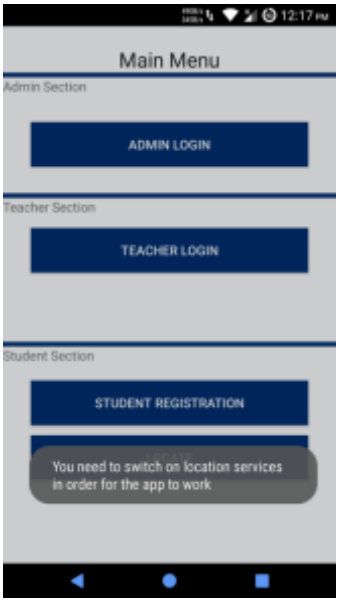
```
package com.example.wipat;

import android.content.Context;
import android.net.wifi.ScanResult;
import android.net.wifi.WifiManager;
import android.os.AsyncTask;
import android.widget.Toast;

import java.util.ArrayList;
import java.util.List;
import java.util.Set;
import java.util.Timer;
import java.util.TimerTask;
import android.util.Log;


public class ScanBackground extends AsyncTask<Void, Void, PositionData> {

    PositionData pos;
    private int readingCount = 30;
    private int currentCount;
    String currentPositionName;
    WifiManager wifi;
    Timer timer;
    TimerTask myTimerTask;

    private Context context;
    public ScanBackground(Context context){
        this.context=context;
    }


    protected void onPreExecute(String status) {

        Log.i("aaki", "reached pre");
        Toast.makeText(context, "Scanning started!",
Toast.LENGTH_LONG).show();

    }

    @Override
    protected void onPostExecute(PositionData posi) {

        Toast.makeText(context, "Done scanning !",
Toast.LENGTH_LONG).show();
        this.pos=posi;
    }
    public class ResultData {
        private Router router;

        public Router getRouter() {
            return this.router;
        }

        public List<Integer> values;

        public ResultData(Router router) {
            // TODO Auto-generated constructor stub
            this.router = router;
            values = new ArrayList<Integer>();
        }
    }

    private List<ResultData> resultsData;
```

```
    private List<PositionData> positionsData;
    private PositionData positionData;

    @Override
    protected PositionData doInBackground(Void... params) {
        wifi = (WifiManager) context.getSystemService(Context.WIFI_SERVICE);
        currentPositionName =null;

        resultsData = new ArrayList<ResultData>();
        currentCount = 0;
        timer = new Timer();
        myTimerTask = new TimerTask() {

            @Override
            public void run() {
                // TODO Auto-generated method stub
                refresh();
            }
        };
        timer.schedule(myTimerTask, 0, 1000);
        return null;
    }

    private void refresh() {
        // TODO Auto-generated method stub
        if (currentCount >= readingCount) {
            if (myTimerTask != null) {
                myTimerTask.cancel();
                returnResults();
            }
        }
        currentCount++;
        wifi.startScan();
        List<ScanResult> results = wifi.getScanResults();
        for (int i = 0; i < results.size(); i++) {
            // System.out.println("test2");
            String ssid0 = results.get(i).SSID;
            String bssid = results.get(i).BSSID;

            int rssi0 = results.get(i).level;
            boolean found = false;
            for (int pos = 0; pos < resultsData.size(); pos++) {
                if
(resultsData.get(pos).getRouter().getBSSID().equals(bssid)) {
                    found = true;
                    resultsData.get(pos).values.add(rssi0);
                    break;
                }
            }
            if (!found) {

                ResultData data = new ResultData(new Router(ssid0, bssid));
                data.values.add(rssi0);
                resultsData.add(data);
            }

        }

    }

    private void returnResults() {
        // TODO Auto-generated method stub

        positionData = new PositionData(currentPositionName);
        for (int length = 0; length < resultsData.size(); length++) {

            int sum = 0;
            for (int l = 0; l < resultsData.get(length).values.size(); l++)
```

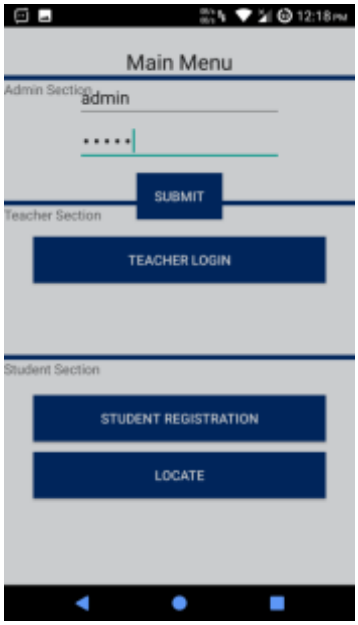
```
{
                sum += resultsData.get(length).values.get(l);

            }
            int average = sum / resultsData.get(length).values.size();

            positionData.addValue(resultsData.get(length).getRouter(),
average);
        }
        Set<String> keys=positionData.values.keySet();
        for(String i: keys)
            Log.i("a", Integer.toString(positionData.values.get(i)));
    }

}
```

- **Application**

1



2

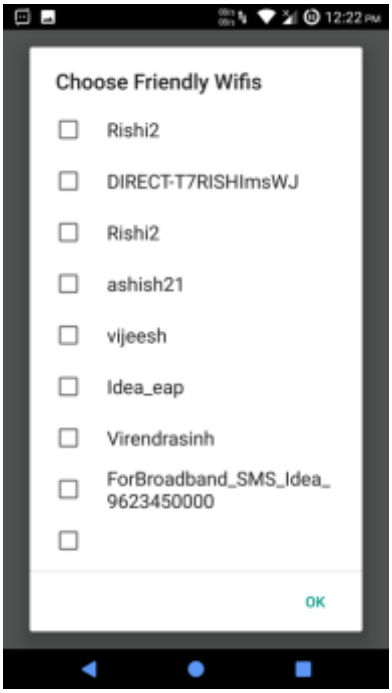


3

4


12:21 PM
ADD MORE WIFIS
SAVE

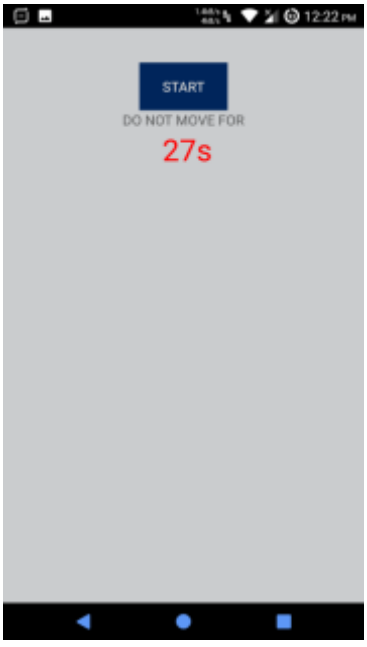
5


12:22 PM
Choose Friendly Wifis
Rishi2
DIRECT-T7RISHImsWJ
Rishi2
ashish21
vijeesh
Idea_eap
Virendrasinh
ForBroadband_SMS_Idea_9623450000
OK

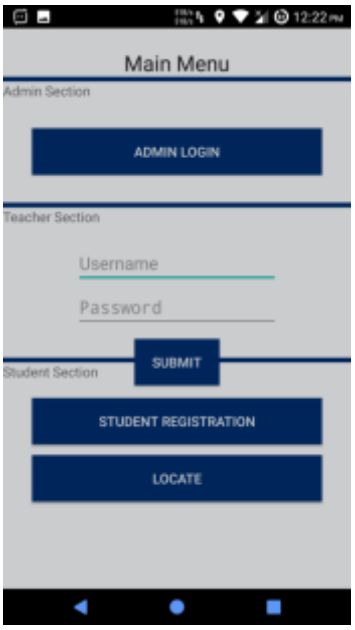
6


12:22 PM
CHOOSE FRIENDLY WIFIS
inside
CALIBRATE
UPDATE

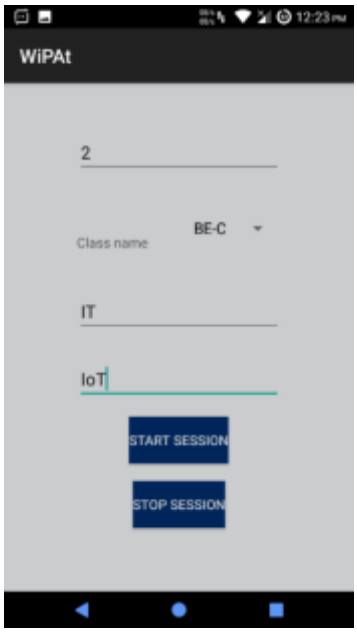
7


12:22 PM
START
DO NOT MOVE FOR
27s

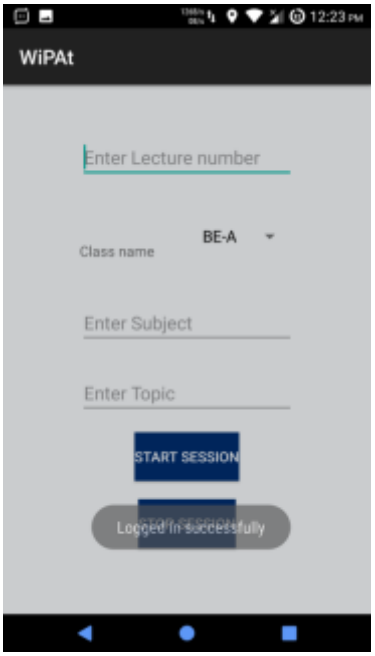
8


12:22
Main Menu
Admin Section
ADMIN LOGIN
Teacher Section
Username
Password
SUBMIT
Student Section
STUDENT REGISTRATION
LOCATE

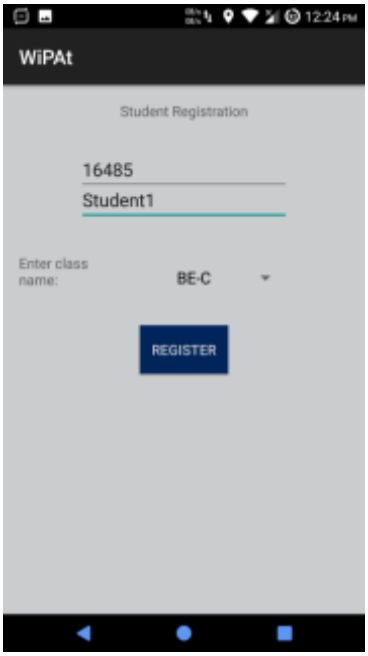
9


12:23
WiPAt
2
Class name
BE-C
IT
IoT
START SESSION
STOP SESSION

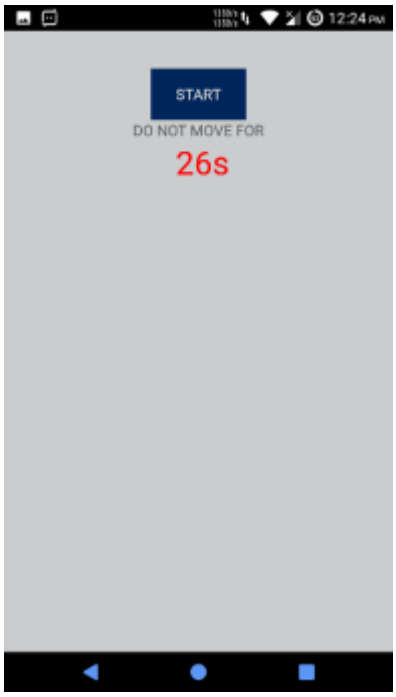
10


12:23
WiPAt
Enter Lecture number
Class name
BE-A
Enter Subject
Enter Topic
START SESSION
Logged in successfully

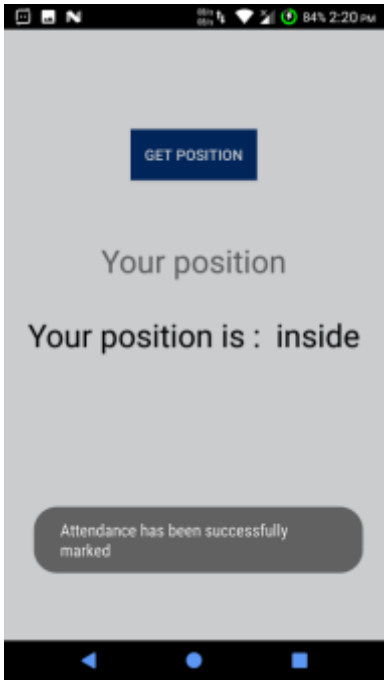
11


12:24
WiPAt
Student Registration
16485
Student1
Enter class name:
BE-C
REGISTER

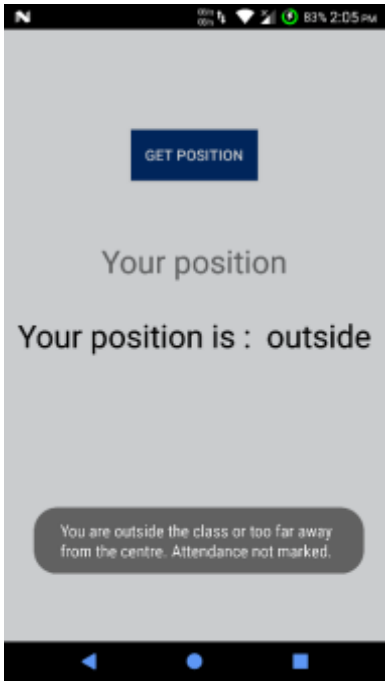
12


START
DO NOT MOVE FOR
26s

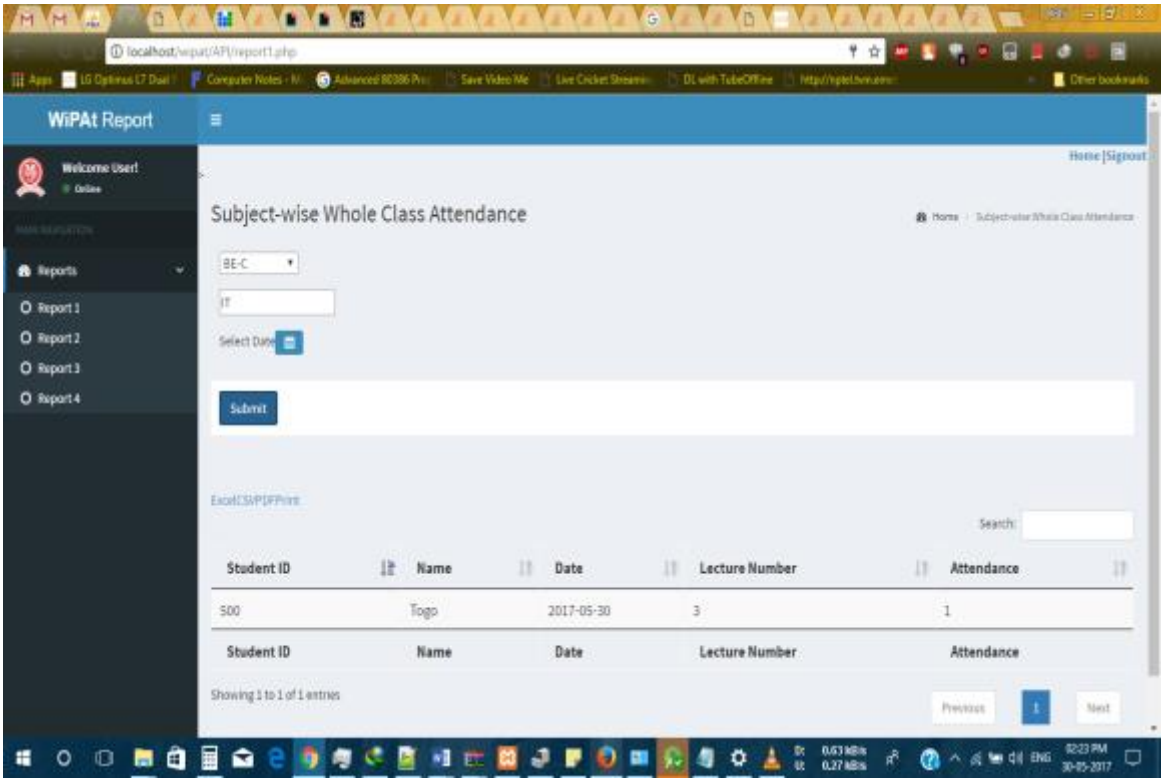
13


GET POSITION
Your position
Your position is : inside
Attendance has been successfully marked

14


GET POSITION
Your position
Your position is : outside
You are outside the class or too far away from the centre. Attendance not marked.

15



Attendance Report

- **Test Cases**

1. ADMIN LOGIN

S.No	Test Case	Excepted Result	Test Result
1	Login : Enter valid admin username and password & click on login button	Software should display next window	Successful
2	Invalid login credentials	Software should show toast message saying invalid details.	Successful

2. ADMIN CALIBRATION

SL.No	Test Case	Excepted Result	Test Result
1	On clicking Choose Friendly Wi-Fi's Button	A window with checkbox options of all routers in range should appear and user can select routers from that list	Successful
2.	On the Click of Save Button	Selected Wi-Fi's will be saved to the database and would appear on the screen.	Successful
3.	On the Click of Update Button	Change the selected Wi-Fi's with a new choice of routers.	Successful
4.	On the Click of Calibrate Button	Displays a textbox where we can name the position that we are calibrating which will be used for further reference. Admin will do this twice: Inside and outside class.	Successful
5.	On the Click of Start Button	A timer for 30 seconds starts which calculates RSSI values over those 30 seconds for that position id.	Successful

3. TEACHER LOGIN

S.No	Test Case	Excepted Result	Test Result
1	Login : Enter valid teacher username and password & click on login button	Software should display next window	Successful
2	Invalid login credentials	Software should show toast message saying invalid details.	Successful

4. TEACHER SESSION DETAILS

S.No	Test Case	Excepted Result	Test Result
1	On clicking Start Session button	The lecture number, subject and topic that the teacher entered will be stored in the database. Only after this, students can mark their attendance.	Successful
2.	On clicking Stop Session button	This deletes the details of the class from the temporary database. Students cannot mark attendance after session is stopped.	Successful

5. STUDENT

SL.No	Test Case	Excepted Result	Test Result
1	On clicking Student Registration button	The next window should appear where student needs to enter details. This is a one-time procedure and multiple registrations are not allowed from one device.	Successful
2.	On clicking Locate Button	Once Locate function is started, a 30 second timer appears. The device position is calculated. After locating, the positions are compared with calibrated positions and the nearest position id is determined.	Successful

6. ATTENDANCE DATABASE

SL.No	Test Case	Excepted Result	Test Result
1	After Locate Button	The nearest position id is compared with new device location. If the resulting location is Inside the classroom, add the attendance in the database system. If it is outside, do not update the attendance.	Successful

Chapter 7

PLANNING AND SCHEDULING

PROJECT PLANNING AND SCHEDULING

a. Project plan

Sr No	Task	Description	Duration (Days)
1	Problem Analysis	Analyzing a real-world problem	15
2	Problem Statement Definition	Defining the problem statement intended to be solved.	5
3	Research	Gaining deeper knowledge about the problem statement and looking for feasible solutions.	20
4	Literature survey	Analysis of technical publications related to the project.	10
5	Sponsorship meetings	Regular meetings with the sponsors and discussing the feasibility and requirements of the project.	5
6	Feasibility Study	Finalizing the scope and possibilities to be implemented.	10
7	Existing system analysis	Studying methodologies and draw-backs of existing systems like ERP	15
8	Proposed system	Proposal of an efficient and improved system	10
9	Design	Creation of algorithms and various design models like DFD, UML etc.	30

10	Development	Implementation of algorithms	45
11	Testing	Testing all aspects of the project	80
12	Debugging	Debugging the errors found	20
13	Documentation	Documenting the project	140

Table 1. Project plan table

6.1.1 PERT chart/ Gantt chart

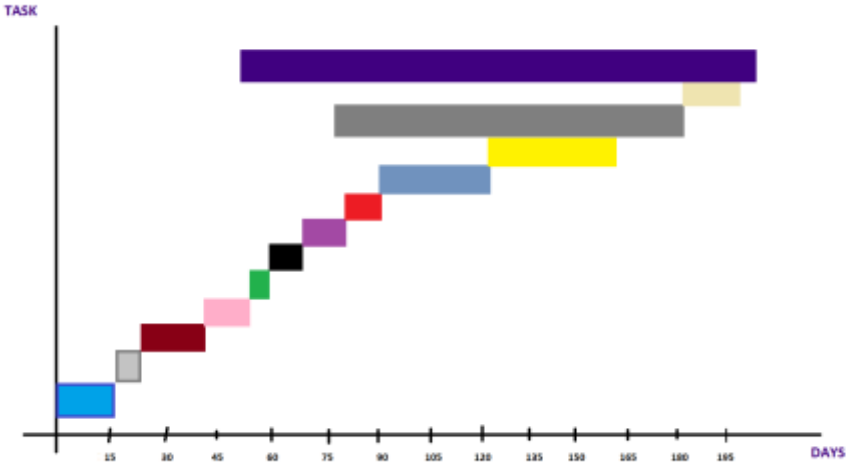


Fig 18: Gantt chart of project

`

Chapter 8

CONCLUSION AND FUTURE SCOPE

a. Future Scope:

Wi-Fi Positioning System is a technology with tremendous scope for the future. The concepts, approach, tools and algorithms used by us enable efficient positioning with minimum error. The attendance system can be put to use in various educational institutions and offices with some more optimization. Basic optimizations include techniques like face recognition/fingerprint to increase security. There are many other areas where the Wi-Fi Positioning Systems can be used effectively. Some of them are:-

1. Device/Person/Vehicle Tracking
2. Improved GPS accuracy with 3D Positioning
3. Crowd Handling
4. Augmented Reality
5. Security and Theft Detection

Since Wi-Fi based systems remove the drawbacks of satellite based systems, this concept can prove to be a revolutionary one in the world of positioning systems in the future. Also, in future, a separate sensor/device dedicated to this process of Wi-Fi based positioning can be made and all students can carry this device.

b. Conclusion:

Thus, we have achieved a high degree of accuracy in Wi-Fi Positioning and used it in our application, WiPAt. It is a very human friendly attendance system with highly efficient technology and algorithms. Using this, human efforts can be reduced to dramatic proportions and the time and resources that were wasted for this monotonous process can be used for better causes. This system, after a few feasibility tests, can be officially used in colleges and offices where traditional unautomated systems are still in practice.

Chapter 8

REFERENCES

8. REFERENCES

[1] *Peerapong Torteeka, Xiu Chundi*: "Hybrid technique for indoor positioning system based on Wi-Fi received signal strength indication"- 2014, *DOI-10.1109/IPIN.2014.7275467*

[2] *Beom-Ju Shin, Kwang-Won Lee, Sun-Ho Choi, Joo-Yeon Kim, Woo Jin Lee, Hyung Seok Kim:* "Indoor WiFi positioning system for Android-based smartphone*", 2010, DOI: 10.1109/ICTC.2010.5674691*

[3] *Rui Zhang, A. Bannoura, F. Hoinger, L.M. Reindl, and C. Schindel-hauer.* "Indoor localization using a smart phone"- In Sensors Applications Symposium (SAS), 2013 IEEE, pages 38-42, Feb 2013.

[4] An Analysis of Wi-Fi Based Indoor Positioning Accuracy
Gints Jekabsons[1], Vadim Kairish[2], Vadim Zuravlyov[3], [1-3]Riga Technical University

[5] *Xingbin Ge; Zhiyi Qu :* "Optimization WIFI indoor positioning KNN algorithm location-based fingerprint"- in Software Engineering and Service Science (ICSESS), 2016 7th IEEE International Conference on 26-28 Aug. 2016
DOI: 10.1109/ICSESS.2016.7883033

[6] *Jun Ma ; Xuansong Li ; Xianping Tao ; Jian Lu* : Cluster filtered KNN: A WLAN-based indoor positioning scheme.
IEEE International Conference on 23-26 June 2008
DOI: 10.1109/WOWMOM.2008.4594840

[7]*http://scholarworks.wmich.edu/cgi/viewcontent.cgi?article=3411&context=honors_theses*