Chapter 1

Introduction

1.1 Introduction:

The main objective of the Geofencing with Task Management is to reduce the human efforts without using any hardware module except the android phone. Our project aims at maintaining the schedule of teacher's attendance and timetable easily. Task Management is monitoring the teacher's attendance, viewing the timetable if required to change their schedule. If we want we can check the availability of classrooms so that it becomes easy to monitor the vacant classroom sitting in one place.

Geofencing is a technology that defines a virtual boundary around a real-world geographical area. In doing so, a radius of interest is established that can trigger an action in a geo-enabled phone or other portable electronic device. Geofencing uses GPS coordinates to encapsulate a geographic area and takes a mobile user's (who has opted in to receive push notifications) location data via GPS to determine his/her proximity to that particular region (whether they are inside or outside or if they just went in and came out of that particular area in a matter of seconds).

Geofencing allows automatic alerts to be generated based on the defined coordinates of a geographic area. A simple example might be an email or text message that is automatically triggered and sent to a user's cell phone when that user's child arrives home from school. In this example, the geofence would be a geographic virtual boundary surrounding the house.

1.2 Motivation:

1) The idea of making this app for college came up when the faculty is absent and we were waiting for teacher to come in classroom or finding her in whole college.
2) Also, to check whether the teacher is present requires human efforts while consumes more time and energy for students as well as for Head of Department.
3) The monitoring whether the teacher is in the college or not, conducting lectures/practical or not is a hectic task for Head Faculty.
4) Sometimes setting the time table requires a lot of changes which is very time consuming, also if the teacher is absent that timeslot of student is being wasted.

Considering all these points mentioned above motivate us to make an efficient, reliable and fast app for monitoring the teachers and task management in the cheapest and convenient way so that it saves the time of student or teacher thus utilizing the time and saving the efforts.

1.3 Scope of work:

- Geofencing with Task Management is an android app that monitors presence of teacher in college also that maintains or manipulates the schedule of teacher.
- Any personnel that may be Principal, Head of Department, Teacher or Class Representative can only access the functionality of the app until and unless he/she is their inside the geofenced area.
- The app involves a Login page & About us page.
- The app has four types of users :

    1. **Principal :** Principal is having the functionality of creating or deleting the Head of Department of various department. He/She can see the Present Head of Department List along with their details.

    2. **Head of Department :** Head of Department plays a very important role in this app. He/She can see the present teacher list along with their timestamp. Also he/she can see his today's schedule and schedule of other teachers of his department. He/She can edit the timetable of any teacher if required .He/She can set the Timetable by importing the master timetable in .csv file into the database. He/She can also check the availability of classroom by maintaining an online Google spreadsheet.

    3. **Teacher :** Teacher is able to view his/her today's schedule also he/she can see the other present teacher list. Teacher can scan the QR code for availability of classroom.

    4. **Class Representative :** Class Representative can just see the present teacher list.

- There is admin panel which requires human intervention in app while creating the principal and setting the geofencing co-ordinates.

Chapter 2

Literature Review

2.1 Existing System: -

Digitization has changed the way of thinking as well as the life style of people. Nowadays different measures for Real Time Location System (RTLS) are available. Among them the concepts mentioned in this project is Geofencing. The present invention generally relates to device tracking, and more particularly to method and associated systems for ensuring location adherence for a tracked device as well as ensuring route adherence by employing geofencing techniques. One of the first uses of geofencing was with cattle. Farmers would equip a herd of cattle with GPS units. When the herd of cattle would move out of the geofence set by the rancher, the rancher would receive an alert. These alerts can be in the form of text messages .Programs that incorporate geofencing allow an administrator to set up triggers so when a device enters (or exits) the boundaries defined by the administrator, a text message or email alert is sent. Applications define boundaries by longitude and latitude or through user-created and Web-based maps.

In the existing system, the app only provides the notification for entry and exit of the personnel from geofenced area. Also limitations are that:

1. Attendance of faculty : It is the main functionality of the proposed system but in the existing system, there is no facility like this, hence for the attendance they use the expensive hardware module for recording the attendance which requires more power supply and more human efforts. Also the cost for maintenance is high.
2. Monitoring the presence of faculty : For checking whether the teacher is present in the college or not requires manual searching which requires more efforts and consumes a lot of our precious time or else use of expensive sensor which is burdensome for faculty.
3. Utilization of time : If the faculty is absent and the student are waiting in classroom for teacher to come. This will lead to wastage of time since there is no functionality in the existing system.
4. Availability of classroom : If the Head of Department wants to know which classroom is vacant so that if required extra lectures could be conducted. For that purpose, manual searching of the classroom is the difficult task to do.
5. Information searching : If a faculty is absent ,Head of Department wants to know his/her schedule so that in that time slot another teacher can take the lecture which will utilize the time of student as well as teacher's time, but there is no such system for it instead manual searching is required.

2.2. Proposed System: -

Here, the implementation of app is done for college. Geofencing uses GPS coordinates to encapsulate a geographic area and takes a mobile user's (who has opted in to receive push notifications) location data via GPS to determine the proximity to that particular region (whether they are inside or outside or if they just went in and came out of that particular area in a matter of seconds). If the teacher or Head of Department is not within the geofence, then can just login into the account but can't access the functionality i.e. being idle like thin client.

As compared to current system our project provides an app which is simpler, easy and convenient for faculty as well as Head Staff. Our app uses internet for its connectivity. Geo-fence apps and tools monitors when employee enter or exit an established geo-fenced area and provide administrators with alerts and able to access with the rights to manage & manipulate the employee's schedule. Also the administrator can check the punctuality of employee, also he/she is able to keep a record of staff's attendance based on their entry and exit. Addition to it, he can check the availability of classroom which involves the concept of QR code scanning and cloud computing. The system is more efficient, reliable and fast as compared to the current system.

2.3. Features of proposed system: -

- **Efficient: -**
  - Geofencing with Task Management allows Head of Department to check the availability of teacher and classroom by sitting in his cabin which is more efficient.
  - The level of hierarchy is efficiently managed which provides data abstraction.
- **Reliable: -**
  - As our app provides continuous service which makes it reliable for Head of Department and our faculty to use.
  - There is no extra hardware module requirement for running the geofencing in background.
- **Time Management:-**
  - The app focuses the proper utilization time of the students and teachers. Also the Head of Department can also monitor punctuality of teacher.
  - Head of Department can utilize the vacant time of students instead of wasting without any manual efforts.
  - From faculty to students, everyone's time is saved.
- **Cost Effective:-**

  - Since this app doesn't require any extra hardware module which increase more complexity as we are using the inbuilt module of smartphone.

Chapter 3

Hardware & Software Requirements

3.1 Hardware requirement :

Android based Smartphone with following specifications :

- Android version 5.0 Lollipop or more
- 2 GB RAM
- 1.5 GHz or more processor

3.2 Software requirement :

FRONT END :

- HTML
- CSS
- Java Script
- PHP
- Android Studio
- Google Scripting

BACK END :

- MYSQL
- Google Spreadsheet

PLATFORM :

- Android Virtual Device Emulator or Android phone
- MAMP/XAMP/WAMP

Chapter 4

System Design

4.1 Control Flow Diagram: -

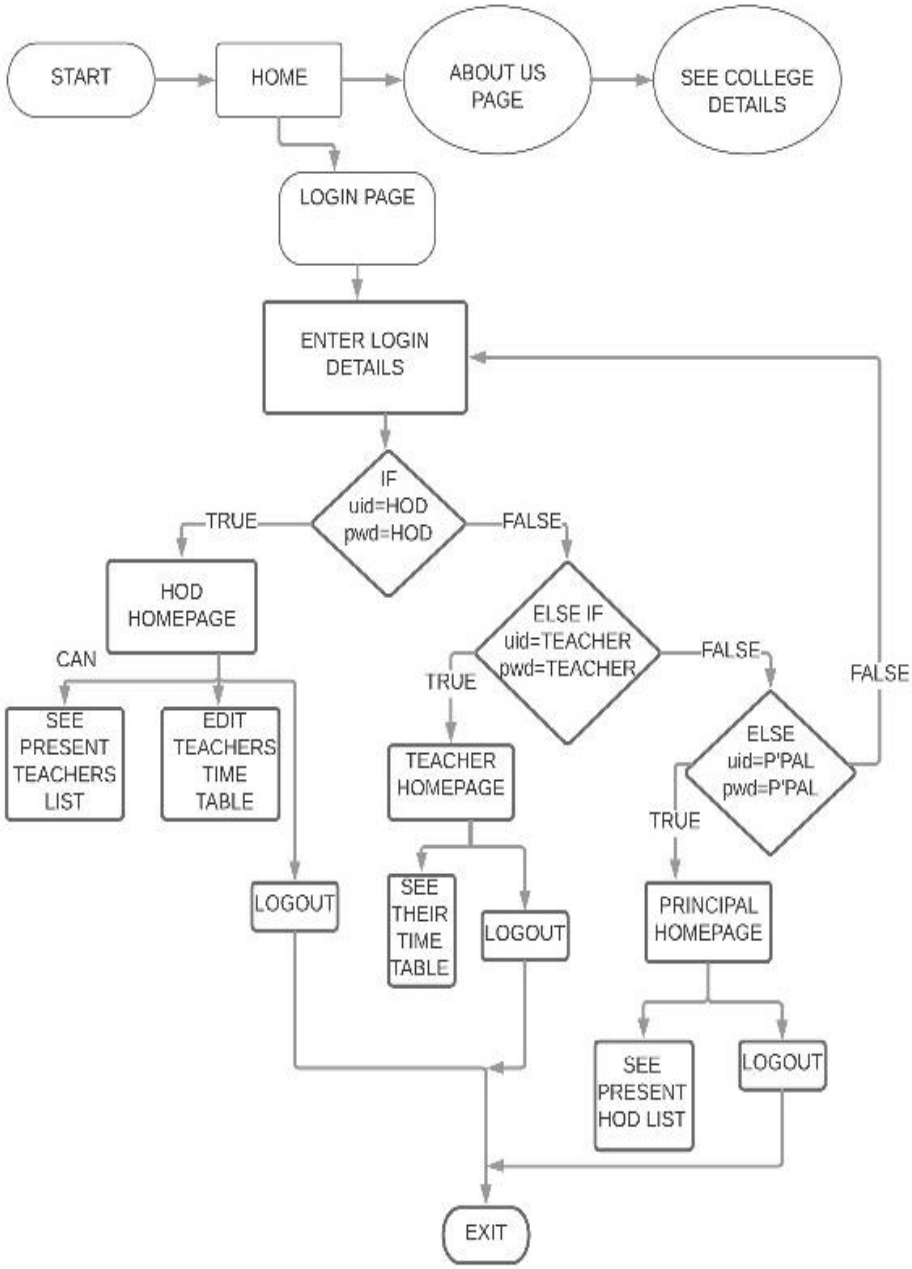


Fig: 4.1 Control Flow Diagram

Control Flow Diagram of Geofencing starts with Homepage. In that it contains two options – Login & About us page. Login page authenticates the user by checking the details from its input. If username = "HOD" or password = "HOD", then it will redirect to Head of Department's Homepage or else it will check whether username = "P'PAL" and password = "P'PAL". Then it will redirect to Principal's Homepage, otherwise if it is incorrect, error message will be displayed. If users Logout, it will redirect back to Login page.

4.2 UML Diagrams (Unified Modelling Language):-

4.2.1 Use Case Diagram: -

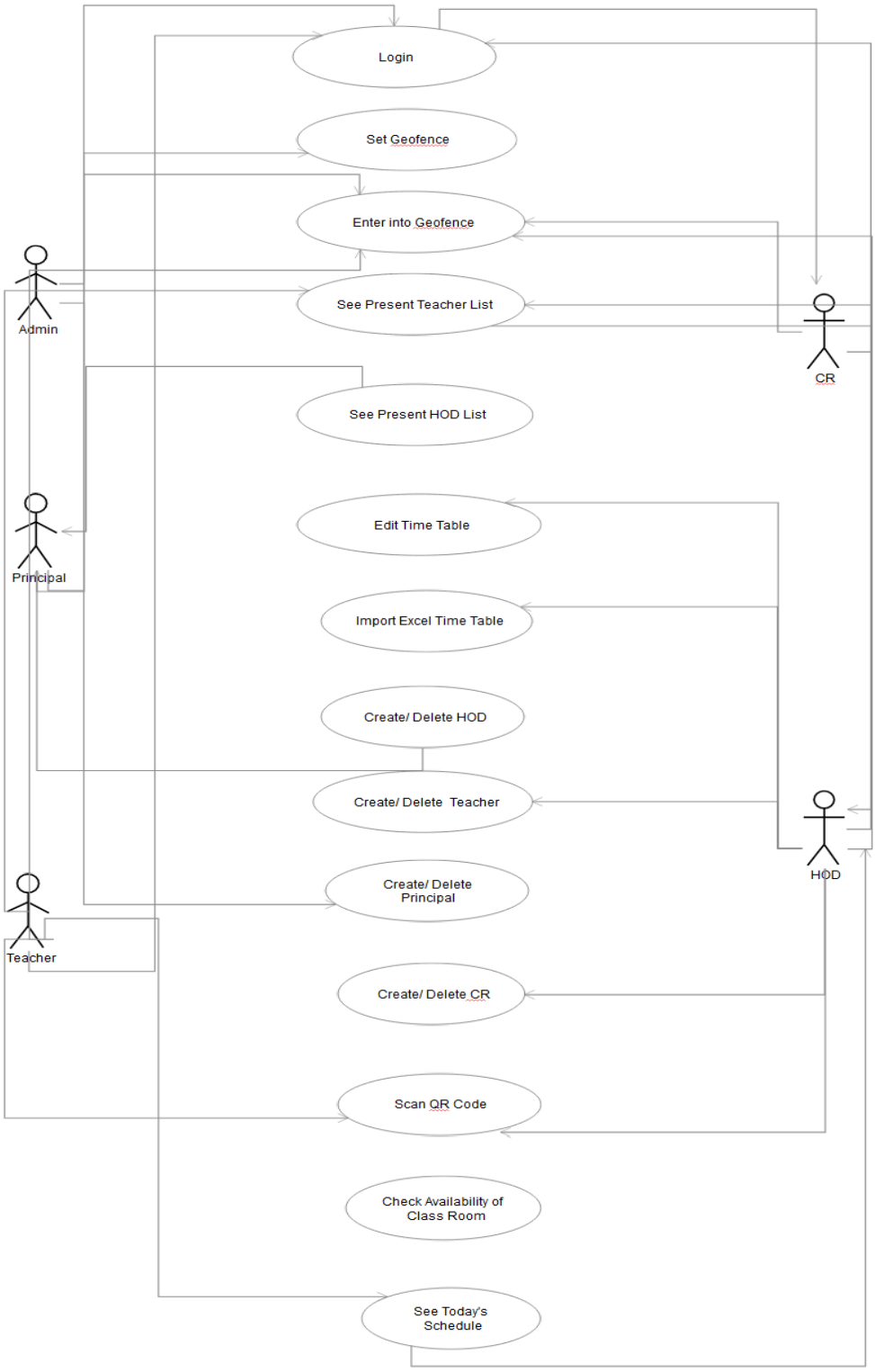


Fig: 4.2.1 Use Case diagram of the app

4.2.2. Class Diagram: -

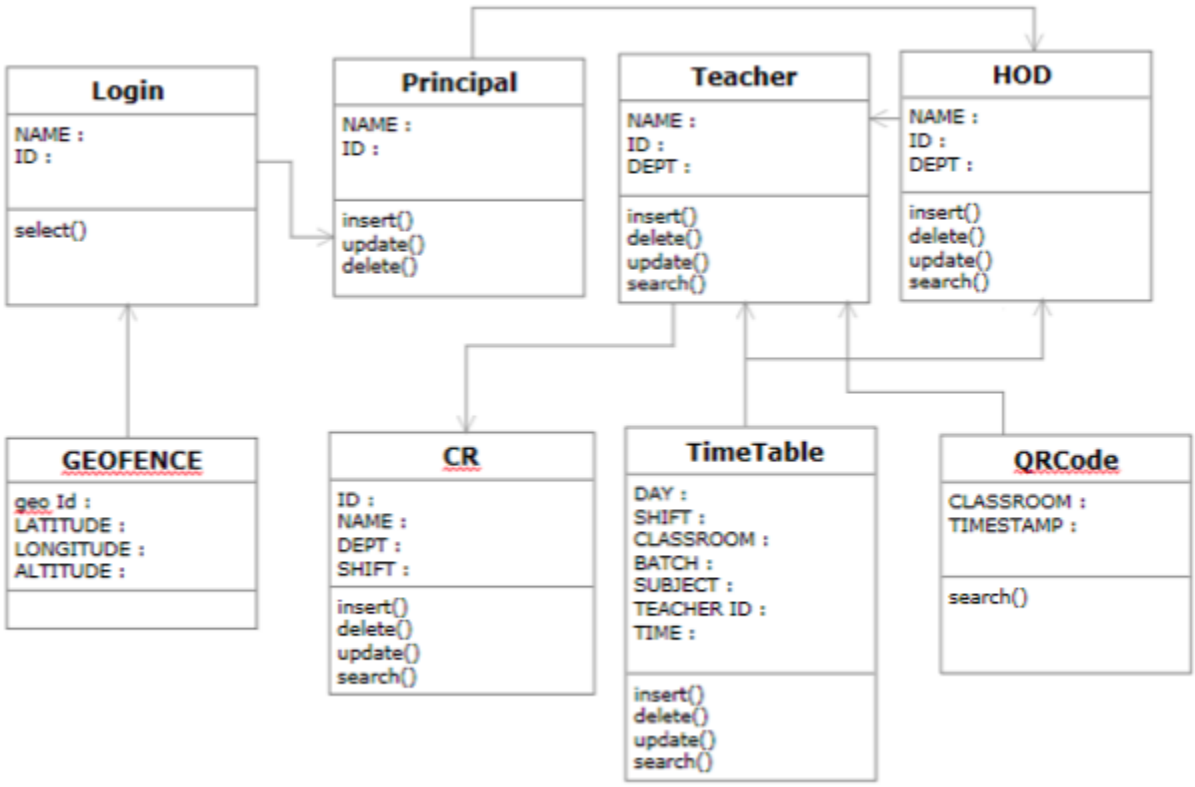


Fig: 4.2.2 Class Diagram

4.2.3 Sequence Diagram:-

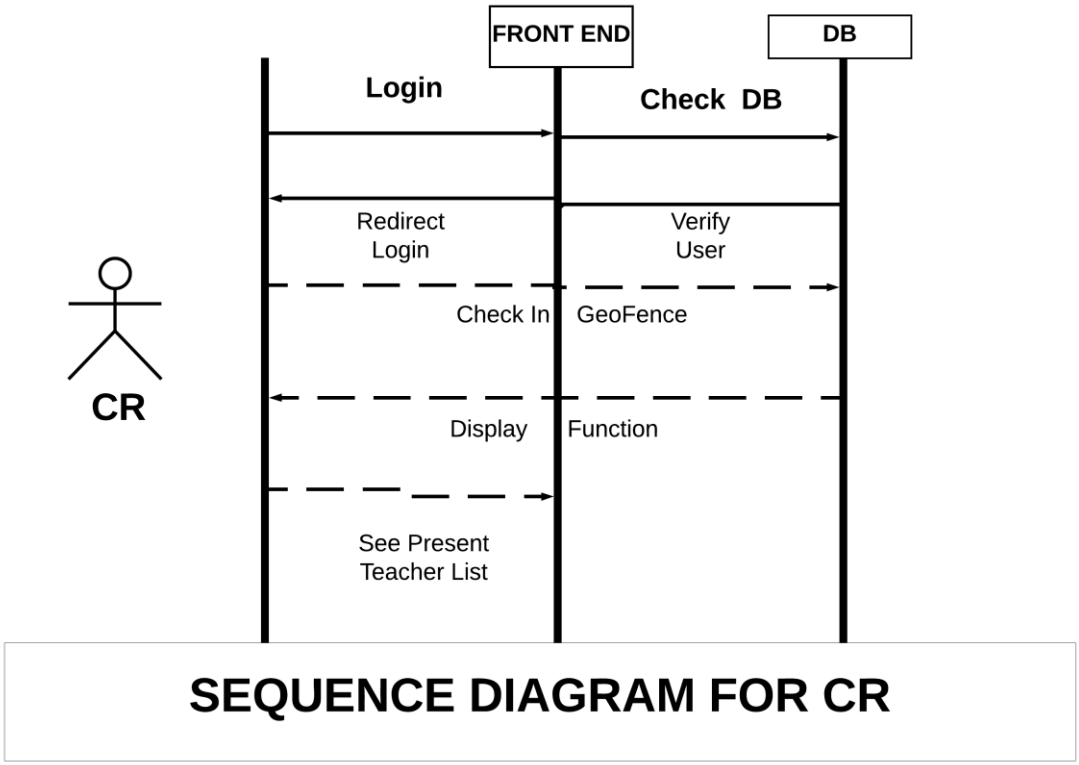


Fig: 4.2.3.1 Sequence Diagram for Class Representative

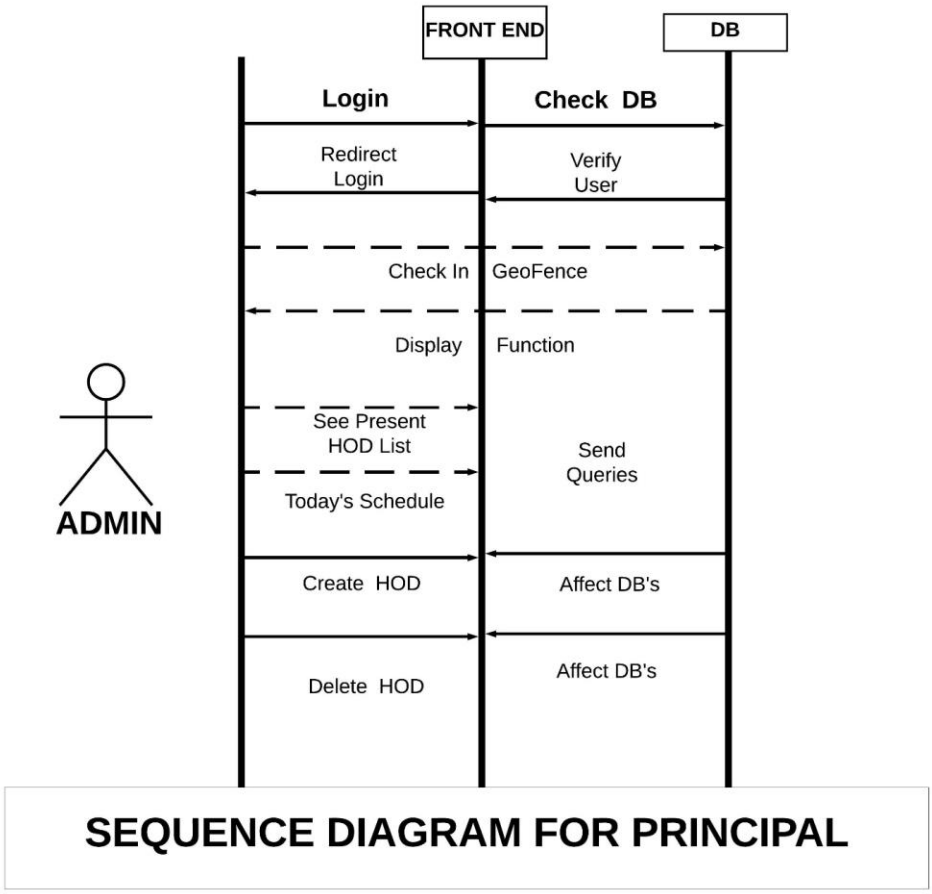


Fig: 4.2.3.2 Sequence Diagram for Principal

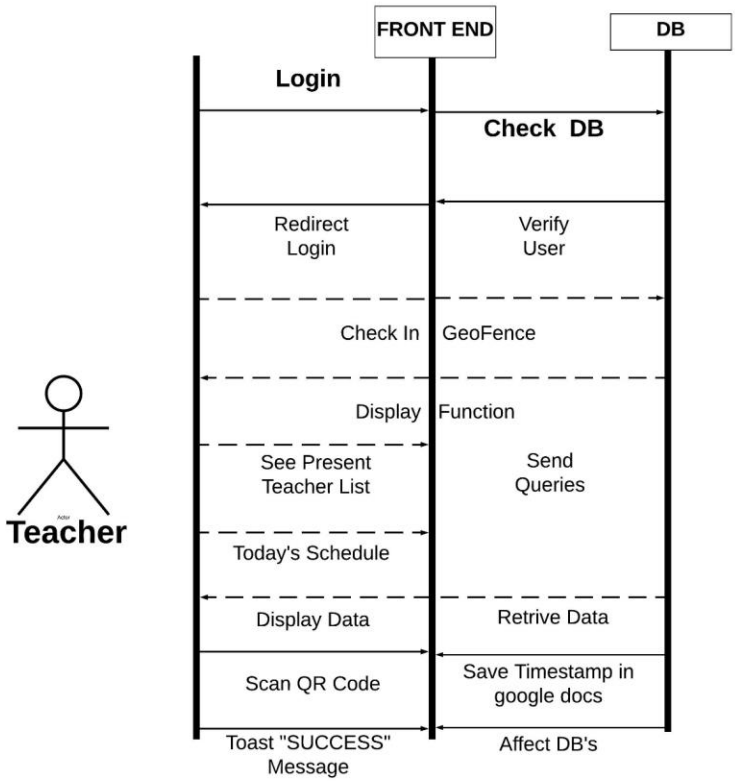


Fig: 4.2.3.3 Sequence Diagram for Teacher

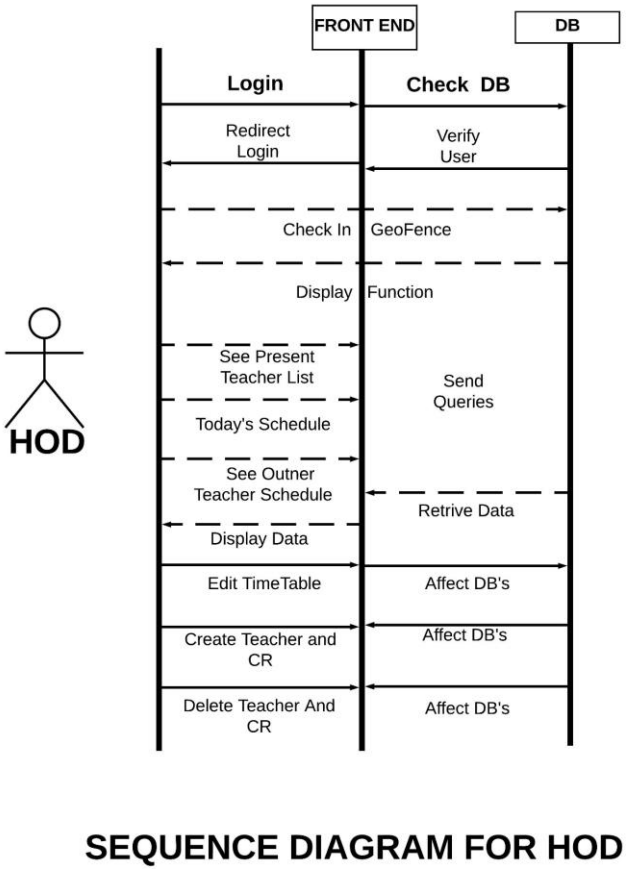


Fig: 4.2.3.4 Sequence Diagram for Head of Department

4.3 Life Cycle Model : -

▪ Spiral Model :-

- The baseline spiral, starting in the planning phase, requirements are gathered and risk is assessed.
- Each subsequent spirals builds on the baseline spiral.
- Spiral Model is a combination of a waterfall model and iterative model. Each phase in spiral model begins with a design goal and ends with the client reviewing the progress.
- The spiral model was first mentioned by Barry Boehm in his 1986 paper.
- The development team in Spiral-SDLC model starts with a small set of requirement and goes through each development phase for those set of requirements.
- The development team adds functionality for the additional requirement in every-increasing spirals until the application is ready for the production phase.
- The spiral model has four phases: Planning, Risk Analysis, Engineering and Evaluation.
- **Planning Phase:** Requirements are gathered during the planning phase. Requirements like 'BRS' that is 'Business Requirement Specifications' and 'SRS' that is 'System Requirement specifications'.
- **Risk Analysis:** In the risk analysis phase, a process is undertaken to identify risk and alternate solutions. A prototype is produced at the end of the risk analysis phase. If any risk is found during the risk analysis then alternate solutions are suggested and implemented.
- **Engineering Phase:** In this phase software is developed, along with testing at the end of the phase. Hence in this phase the development and testing is done.
- **Evaluation phase:** This phase allows the customer to evaluate the output of the project to date before the project continues to the next spiral.

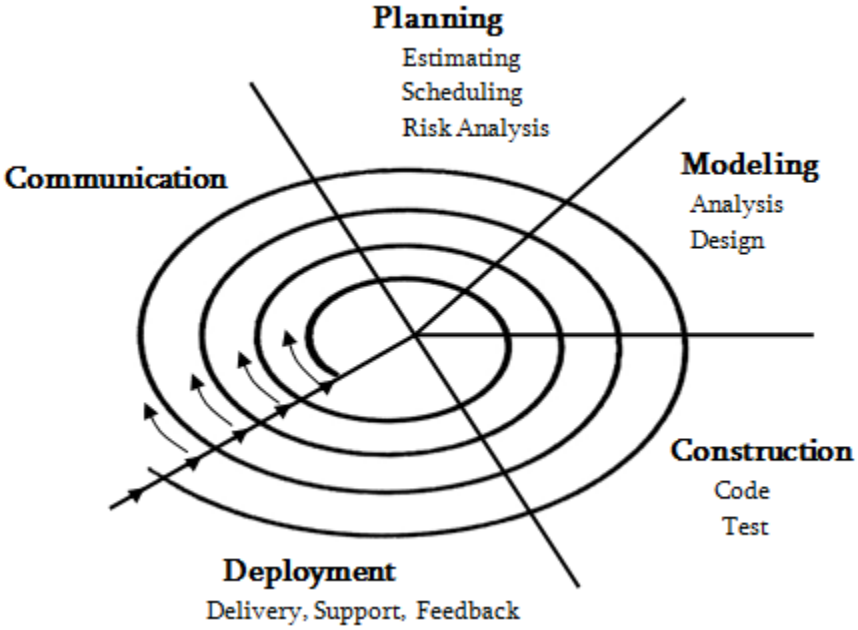


Fig 4.3 Spiral Model

Chapter 5

Project Implementation

5.1 Coding

GeoLocService.java

```
package e.gaura.geo;

import android.Manifest;
import android.app.PendingIntent;
import android.app.Service;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.location.Location;
import android.os.Bundle;
import android.os.IBinder;
import android.support.annotation.NonNull;
import android.support.annotation.Nullable;
import
android.support.v4.app.ActivityCompat;

import
com.google.android.gms.common.ConnectionResu
lt;
import
com.google.android.gms.common.api.GoogleApiC
lient;
import
com.google.android.gms.common.api.ResultCall
back;
import
com.google.android.gms.common.api.Status;
import
com.google.android.gms.location.Geofence;
import
com.google.android.gms.location.GeofencingRe
quest;
import
com.google.android.gms.location.LocationList
ener;
import
com.google.android.gms.location.LocationRequ
est;
import
com.google.android.gms.location.LocationServ
ices;

import java.util.ArrayList;

public class GeoLocService extends Service
implements
GoogleApiClient.ConnectionCallbacks,

GoogleApiClient.OnConnectionFailedListener,
LocationListener{

    GoogleApiClient googleApiClient;
    Place home;
    final int maxInterval = 5000,
minInterval = 2500;
    final String NAME_OF_THE_FENCE = "HOME";
    final int RADIUS =50;

    //gaurav home
    final double LATITUDE = 19.0237109;
    final double LONGITUDE = 73.0949885;

    @Override
    public void onCreate() {
        home = new Place(NAME_OF_THE_FENCE,
RADIUS, LATITUDE, LONGITUDE);
        googleApiClient = new
GoogleApiClient.Builder(getApplicationContex
t())

.addApi(LocationServices.API)

.addConnectionCallbacks(GeoLocService.this)

.addOnConnectionFailedListener(GeoLocService
.this)
                .build();
    }

    @Override
    public void onConnected(@Nullable Bundle
bundle) {
        X.log("Connected to Google API!");
        startLocationMonitoring();
    }

    @Override
    public void onConnectionSuspended(int i)
{
        X.log("Suspended connection to
Google API");
        googleApiClient.reconnect();
        X.log("Initiated reconnection . .
.");
    }

    @Override
    public void onConnectionFailed(@NonNull
ConnectionResult connectionResult) {
        X.log("Failed to connect to Google
API");
        googleApiClient.reconnect();
        X.log("Initiated reconnection . .
.");
    }

    @Override
    public void onLocationChanged(Location
location) {
        X.log("Latitude: " +
location.getLatitude() + " Longitude: " +
location.getLongitude());
        // X.toast(getApplicationContext(),
"Latitude: " + location.getLatitude() + "
Longitude: " + location.getLongitude());
    }

    @Override
    public IBinder onBind(Intent intent) {
        // TODO: Return the communication
channel to the service.
        throw new
UnsupportedOperationException("Not yet
implemented");
    }
```

```
    @Override
    public int onStartCommand(Intent intent,
int flags, int startId) {
        googleApiClient.connect();
        return Service.START_STICKY;
    }

    @Override
    public void onDestroy() {
        stopGeofenceMonitoring();
        if(googleApiClient.isConnected())
            googleApiClient.disconnect();
        Intent serviceGeo = new Intent(this,
GeofenceService.class);
        stopService(serviceGeo);
    }

    public void startLocationMonitoring() {
        LocationRequest locationRequest =
LocationRequest.create()
                .setInterval(maxInterval)

.setFastestInterval(minInterval)

.setPriority(LocationRequest.PRIORITY_HIGH_A
CCURACY);
        if (ActivityCompat
                .checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED
                &&
ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION)
!= PackageManager.PERMISSION_GRANTED) {
            return;
        }

LocationServices.FusedLocationApi.requestLoc
ationUpdates(googleApiClient,
                locationRequest,
GeoLocService.this);
        startGeofenceMonitoring();
    }

    public void startGeofenceMonitoring() {
        X.log("Starting Geofence
Monitor...");

        try{
            Geofence geofence = new
Geofence.Builder()

.setRequestId(home.GEOFENCE_ID)

.setCircularRegion(home.LATITUDE,
home.LONGITUDE, home.GEOFENCE_RADIUS)

.setExpirationDuration(Geofence.NEVER_EXPIRE
)

.setNotificationResponsiveness(2500)

.setTransitionTypes(Geofence.GEOFENCE_TRANSI
TION_ENTER |
Geofence.GEOFENCE_TRANSITION_EXIT)
                    .build();

            GeofencingRequest
geofencingRequest = new
GeofencingRequest.Builder()

.setInitialTrigger(GeofencingRequest.INITIAL
_TRIGGER_ENTER)
                    .addGeofence(geofence)
                    .build();

            X.log("Initial trigger set to
"+geofencingRequest.getInitialTrigger());

            Intent serviceGeo = new
Intent(this, GeofenceService.class);
            PendingIntent pendingIntent =
PendingIntent.getService(this, 0,
serviceGeo,
PendingIntent.FLAG_UPDATE_CURRENT);

            if
(!googleApiClient.isConnected()) {
                X.log("Google services not
connected");
            } else {
                if
(ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
                    return;
                }

LocationServices.GeofencingApi.addGeofences(
googleApiClient, geofencingRequest,
pendingIntent)

.setResultCallback(new
ResultCallback<Status>() {
                            @Override
                            public void
onResult(@NonNull Status status) {
                                if
(status.isSuccess()) {

X.log("Added geofence successfully!");
                                } else {

X.log("Failed to add geofence " +
status.getStatus());
                                }
                            }

                        });
            }
        } catch(SecurityException e){
            String secErr = "Error: " +
e.toString();
            X.log(secErr);
        }
    }

    public void stopGeofenceMonitoring(){
        X.log("Stopped monitoring");
        ArrayList<String> geofenceIds = new
ArrayList<>();
        geofenceIds.add(home.GEOFENCE_ID);
```

```
LocationServices.GeofencingApi.removeGeofences(googleApiClient, geofenceIds);
    }
}
```

GeofenceService.java

```
package e.gaura.geo;

import android.app.IntentService;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.support.annotation.Nullable;
import
android.support.v4.app.NotificationCompat;
import android.util.Log;

import
com.google.android.gms.location.Geofence;
import
com.google.android.gms.location.GeofencingEvent;

import java.io.IOException;
import java.util.List;

public class GeofenceService extends
IntentService {

    public static final String TAG = "PTM";
    public static final int NOTIFICATION_ID
= 1;
    SharedPreferences sharedPref;


    public GeofenceService(){
        super("GeofenceService");
    }

    @Override
    protected void onHandleIntent(@Nullable
final Intent intent) {
        sharedPref =
this.getSharedPreferences(getString(R.string
.preference_file_key),
Context.MODE_PRIVATE);


        GeofencingEvent event =
GeofencingEvent.fromIntent(intent);
        if(event != null) {
            if (event.hasError()) {
                Log.e(TAG,
String.valueOf(event.getErrorCode()));
            } else {
                int transition =
event.getGeofenceTransition();
                List<Geofence> geofences =
event.getTriggeringGeofences();
                Geofence geofence =
geofences.get(0);
                final String requestID =
geofence.getRequestId();

                if (transition ==
Geofence.GEOFENCE_TRANSITION_ENTER) {
                    final String enterText =
"You have entered geofence";
                    {

SharedPreferences.Editor editor =
sharedPref.edit();

editor.putBoolean("InFence", true);
                        editor.commit();

                        try {
                            String url =
"http://" + sharedPref.getString("IP", null)
+

"/GeoFence/MARK_ATT.php?ID="+sharedPref.getS
tring("ID", null);

StaticFunction.httpRequest(url);
                        } catch (IOException
e) {

e.printStackTrace();
                        }

                    }

                    X.log(enterText +
requestID);

                    // Firing a notification

sendNotification(enterText, requestID);
                    // Change shared
preference
                    // sharedPrefMetHead of
Department();

                } else if (transition ==
Geofence.GEOFENCE_TRANSITION_EXIT) {
                    final String exitText =
"You have exited geofence";

                    SharedPreferences.Editor
editor = sharedPref.edit();

editor.putBoolean("InFence", false);
                    editor.commit();


                    try {
                        String url =
"http://" + sharedPref.getString("IP", null)
+

"/GeoFence/MARK_OUT.php?ID="+sharedPref.getS
tring("ID", null);

StaticFunction.httpRequest(url);
                    } catch (IOException e)
{
```

```
                        e.printStackTrace();
                    }
                    X.log(exitText +
requestID);

                    // Firing a notification

sendNotification(exitText, requestID);
                    // Change shared
preference
                    // sharedPrefMetHead of
Department();
                }else if (transition ==
Geofence.GEOFENCE_TRANSITION_DWELL) {
                    final String enterText =
"You have entered geofence";
                    {

SharedPreferences.Editor editor =
sharedPref.edit();

editor.putBoolean("InFence", true);
                        editor.commit();

                        try {
                            String url =
"http://" + sharedPref.getString("IP", null)
+

"/GeoFence/MARK_ATT.php?ID="+sharedPref.getS
tring("ID", null);

StaticFunction.httpRequest(url);
                        } catch (IOException
e) {

e.printStackTrace();
                        }


                    }

                    X.log(enterText +
requestID);

                    // Firing a notification

sendNotification(enterText, requestID);
                    // Change shared
preference
                    // sharedPrefMetHead of
Department();

                }
            }
        }
    }

    private void sendNotification(String
title, String content) {
        NotificationManager
mNotificationManager = (NotificationManager)

this.getSystemService(Context.NOTIFICATION_S
ERVICE);

        PendingIntent contentIntent =
PendingIntent.getActivity(this, 0,
                new Intent(this,
HomeScreen.class), 0);

        NotificationCompat.Builder mBuilder
= new NotificationCompat.Builder(this)
                .setContentTitle(title)
                .setStyle(new
NotificationCompat.BigTextStyle().bigText(co
ntent))
                .setContentText(content)

.setSmallIcon(R.mipmap.ic_launcher_round);

mBuilder.setContentIntent(contentIntent);

mNotificationManager.notify(NOTIFICATION_ID,
mBuilder.build());
    }
}
```

QRcode.java

```
package e.gaura.geo;

import android.app.Activity;
import android.content.Intent;
import android.os.AsyncTask;
import
android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import org.json.JSONObject;

import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.InputStreamReader;
import java.io.OutputStream;
import java.io.OutputStreamWriter;
import java.net.HttpURLConnection;
import java.net.URL;
import java.net.URLEncoder;
import java.util.Iterator;
import
com.google.zxing.integration.android.IntentI
ntegrator;
import
com.google.zxing.integration.android.IntentR
esult;

import javax.net.ssl.HttpsURLConnection;

public class qr extends AppCompatActivity {

    String scannedData;
    Button scanBtn;
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```
setContentView(R.layout.activity_qr);
        final Activity activity =this;
        scanBtn =
(Button)findViewById(R.id.scan_btn);

        scanBtn.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View view) {
                IntentIntegrator integrator
= new IntentIntegrator(activity);

integrator.setDesiredBarcodeFormats(IntentIn
tegrator.QR_CODE_TYPES);

integrator.setPrompt("Scan");

integrator.setBeepEnabled(false);
                integrator.setCameraId(0);

integrator.setBarcodeImageEnabled(false);
                integrator.initiateScan();
            }
        });
    }

    @Override
    protected void onActivityResult(int
requestCode, int resultCode, Intent data) {
        IntentResult result =
IntentIntegrator.parseActivityResult(request
Code,resultCode,data);
        if(result!=null) {
            scannedData =
result.getContents();
            if (scannedData != null) {
                // Here we need to handle
scanned data...
                new SendRequest().execute();


            }else {
            }
        }
        super.onActivityResult(requestCode,
resultCode, data);
    }
    public class SendRequest extends
AsyncTask<String, Void, String> {


        protected void onPreExecute(){}

        protected String
doInBackground(String... arg0) {

            try{

                //Enter script URL Here
                URL url = new
URL("https://script.google.com/macros/s/AKfy
cbxmN0n8c6zTSnJDVN3Y15oK-
yA1ESqYQGxFSe2JpiegVpKSf42m/exec");

                JSONObject postDataParams =
new JSONObject();

                //int i;
                //for(i=1;i<=70;i++)


                //    String usn =
Integer.toString(i);

                //Passing scanned code as
parameter


postDataParams.put("sdata",scannedData);



Log.e("params",postDataParams.toString());

                HttpURLConnection conn =
(HttpURLConnection) url.openConnection();
                conn.setReadTimeout(15000 /*
milliseconds */);
                conn.setConnectTimeout(15000
/* milliseconds */);
                conn.setRequestMetHead of
Department("GET");
                conn.setDoInput(true);
                conn.setDoOutput(true);

                OutputStream os =
conn.getOutputStream();
                BufferedWriter writer = new
BufferedWriter(
                        new
OutputStreamWriter(os, "UTF-8"));

writer.write(getPostDataString(postDataParam
s));

                writer.flush();
                writer.close();
                os.close();

                int
responseCode=conn.getResponseCode();

                if (responseCode ==
HttpsURLConnection.HTTP_OK) {

                    BufferedReader in=new
BufferedReader(new
InputStreamReader(conn.getInputStream()));
                    StringBuffer sb = new
StringBuffer("");
                    String line="";

                    while((line =
in.readLine()) != null) {

                        sb.append(line);
                        break;
                    }

                    in.close();
                    return sb.toString();
```

```
                }
                else {
                    return new String("false
: "+responseCode);
                }
            }
            catch(Exception e){
                return new
String("Exception: " + e.getMessage());
            }
        }

        @Override
        protected void onPostExecute(String
result) {

Toast.makeText(getApplicationContext(),
result,

Toast.LENGTH_LONG).show();

        }
    }
    public String
getPostDataString(JSONObject params) throws
Exception {

        StringBuilder result = new
StringBuilder();
        boolean first = true;

        Iterator<String> itr =
params.keys();

        while(itr.hasNext()){

            String key= itr.next();
            Object value = params.get(key);

            if (first)
                first = false;
            else
                result.append("&");


result.append(URLEncoder.encode(key, "UTF-
8"));
            result.append("=");

result.append(URLEncoder.encode(value.toStri
ng(), "UTF-8"));

        }
        return result.toString();
    }
}
```

GoogleScripting

```
function doGet(e){

 var ss =
SpreadsheetApp.openByUrl("https://docs.google.com/sp
readsheets/d/1CTKPaYE2sMqbUkqEgG6Y0VP8byZYH3Yld
SsYgXkXDxg/edit#gid=0");

 var sheet = ss.getSheetByName("Sheet1");

 return insert(e,sheet);

}

function doPost(e){

 var ss =
SpreadsheetApp.openByUrl("https://docs.google.com/sp
readsheets/d/1CTKPaYE2sMqbUkqEgG6Y0VP8byZYH3Yld
SsYgXkXDxg/edit#gid=0");

 var sheet = ss.getSheetByName("Sheet1");

 return insert(e,sheet);

}

function insert(e,sheet) {

 var scannedData  = e.parameter.sdata;

 var d = new Date();

 var ctime =  d.toLocaleString();

 var rowData=sheet.appendRow([scanned Data,ctime]);

 return ContentService

 .createTextOutput("Success")

 .setMimeType(ContentService.MimeType.JAVASCRIPT);

}
```

ExcelTTIndex.php

```
<!DOCTYPE  html>

<?php

        include 'DB.php';

?>

<html lang="en">

        <head>

                <meta charset="utf-8">

                <title>Import Excel sheet of
TimeTable</title>

                <meta name="viewport"
content="width=device-width,  initial-scale=1.0">

                <meta name="description"
```

```
content="Import Excel File To Database">

        <link rel="stylesheet"
href="css/bootstrap.min.css">

        <link rel="stylesheet"
href="css/bootstrap-responsive.min.css">

        <link rel="stylesheet"
href="css/bootstrap-custom.css">

    </head>

    <body>

    <!—Navbar
================================================
== -->

    <div class="navbar  navbar-inverse navbar-fixed-
top">

        <div class="navbar-inner">

            <div class="container">

                <a class="btn btn-
navbar" data-toggle="collapse"  data-target=".nav-
collapse">

                    <span
class="icon-bar"></span>

                    <span
class="icon-bar"></span>

                    <span
class="icon-bar"></span>

                </a>

            </di            v>

        </div>

    </div>

    <div id="wrap">

    <div class="container">

        <div class="row">

            <div class="span3  hidden-
phone"></div>

            <div class="span6"  id="form-
login">

                <form class="form-
horizontal well" action="import.php" metHead of
Department="post" name="upload_excel"
enctype="multipart/form-data">

                    <fieldset>

    <legend>Import CSV/Excel file</legend>

                    <div
class="control-group">

                        <div class="control-label">

                <label>CSV/Excel  Error! Hyperlink
reference not valid.
                        </div>
                            <div
class="controls"<input  type="file" name="file" id="file"
class="input-large">
                        </div>
                        </div>

                    <div
class="control-group">

                        <div class="controls">
        <button type="submit" id="submit"
name="Import" class="btn btn-primary button-loading"
data-loading-text="Loading...">Upload</button>
        </div>

            </div>

                    </fieldset>

                </form>

            </div>

            <div class="span3  hidden-
phone"></div>

        </div>

        <table class="table  table-bordered">

            <thead>

                    <tr>
    <th>DAY</th>

    <th>SUBJECT</th>
    <th>TEACHER_ID</th>
    <th>TIME_SLOT</th>
    <th>CLASSROOM</th>
    <th>YEAR</th>
    <th>DIVISION</th>
    <th>BATCH</th>

                    </tr>
```

```
                                    </thead>

                            <?php

                                    $SQLSELECT = "SELECT
* FROM TIME_TABLE";

                                    $result_set =
mysql_query($SQLSELECT,  $conn);

                                    while($row =
mysql_fetch_array($result_set))

                                    {

                                    ?>

                                            <tr>


        <td><?php echo $row['DAY']; ?></td>
        <td><?php echo $row['SUBJECT']; ?></td>
        <td><?php echo $row['TEACHER_ID']; ?></td>

        <td><?php echo $row['TIME_SLOT']; ?></td>

        <td><?php echo $row['CLASSROOM']; ?></td>

        <td><?php echo $row['YEAR']; ?></td>

        <td><?php echo $row['DIVISION']; ?></td>

        <td><?php echo $row['BATCH']; ?></td>

                                            </tr>

                                    <?php

                                    }

                            ?>

                    </table>

        </div>

        </div>

        </body>

</html>
```

ImportExcelTT.php

```
<?php

include 'db.php';

if(isset($_POST["Import"])){

                    echo
$filename=$_FILES["file"]["tmp_name"];

                    if($_FILES["file"]["size"] > 0)

                    {

                            $file = fopen($filename, "r");

                while (($emapData = fgetcsv($file, 10000,
",")) !== FALSE)

                {

                 //It wiil insert a row to our subject table
from our csv file`

                 $sql = "INSERT into TIME_TABLE(`DAY`,
`SUBJECT`, `TEACHER_ID`, `TIME_SLOT`,`CLASSROOM`,
`YEAR`, `DIVISION`,`BATCH`)


values('$emapData[0]','$emapData[1]','$emapData[2]','$
emapData[3]','$emapData[4]','$emapData[5]','$emapDat
a[6]','$emapData[7]')";

                //we are using mysql_query function. it
returns a resource on true else False on error

                $result = mysql_query( $sql, $conn );

                                    if(! $result )

                                    {

                                            echo "<script
type=\"text/javascript\">

alert(\"Invalid File:Please Upload CSV File.\");

window.location = \"exxceltt_index.php\"

            </script>";

                                    }

                }

                fclose($file);

                //throws a message if data successfully
imported to mysql database from excel file

                echo "<script type=\"text/javascript\">

alert(\"CSV File has been successfully Imported.\");

window.location = \"exceltt_index.php\"

                                            </script>";

                            //close of connection

                            mysql_close($conn);

                    }       }       >?
```

5.2 Snapshots of Geofencing with Task Management:

Home Page: -

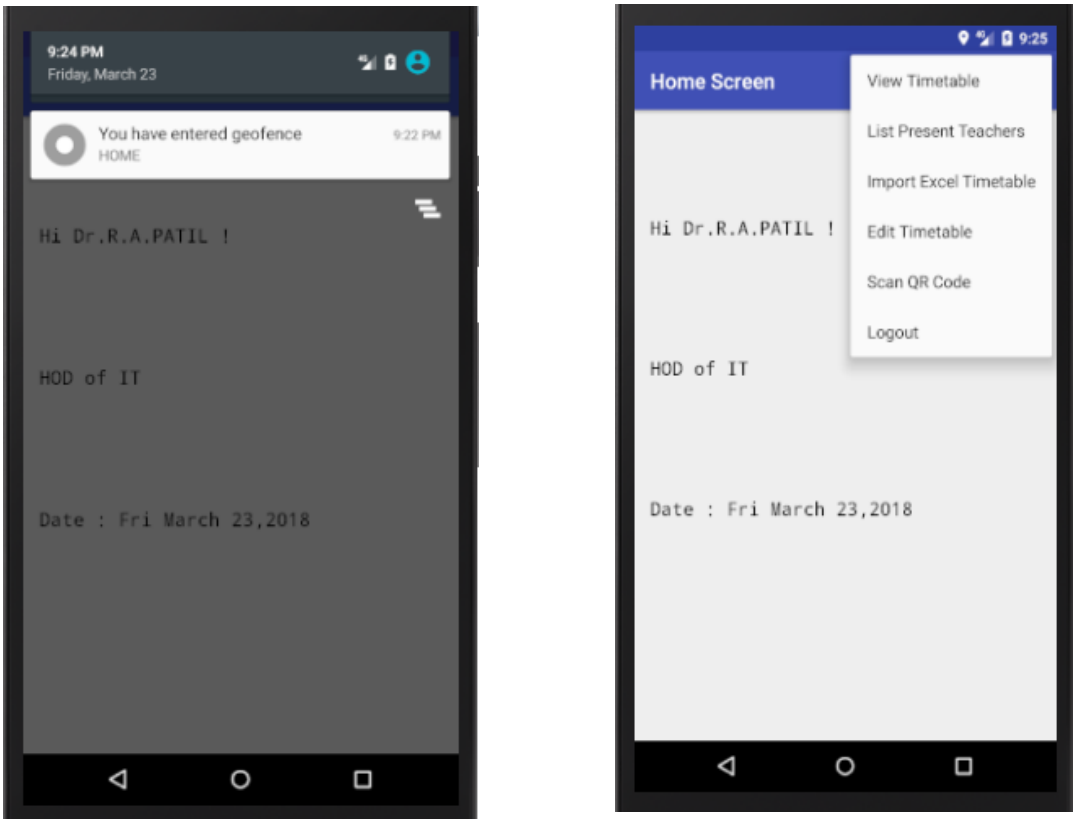


Fig: 5.2.1 Home Page of Geofencing with Task Management

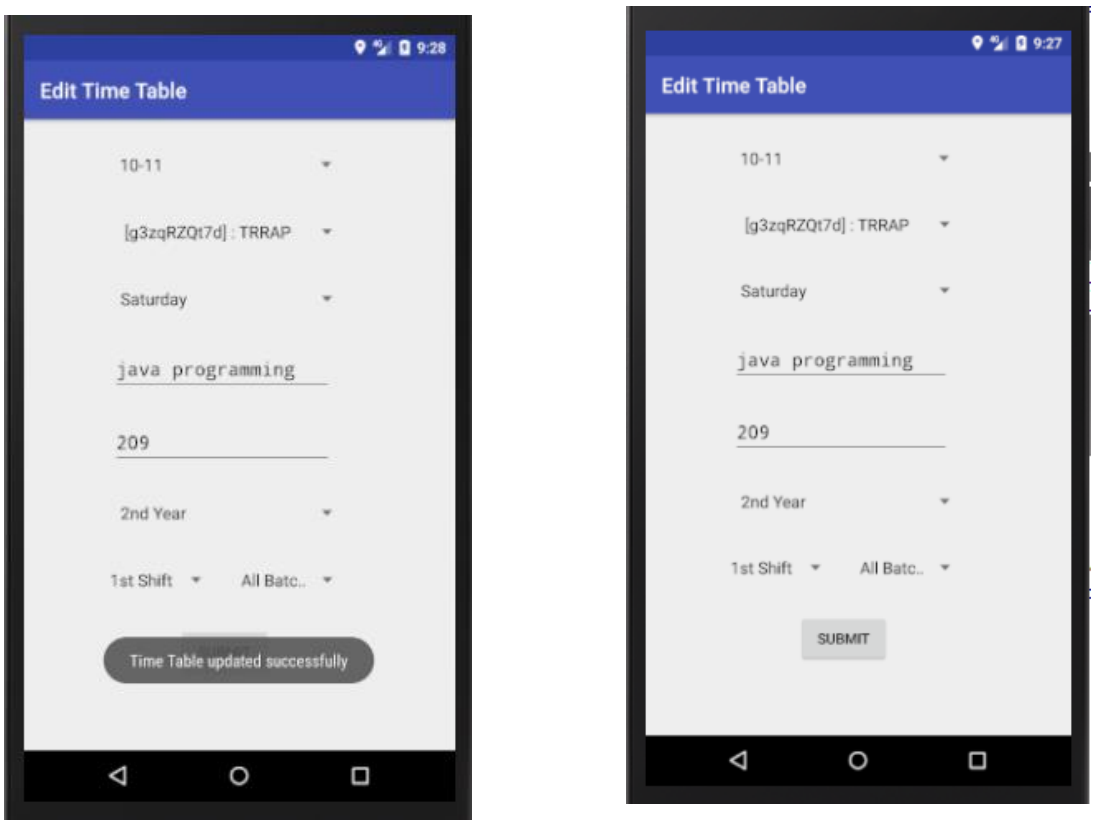


Fig: 5.2.2 Edit TimeTable page of Geofencing Task Management

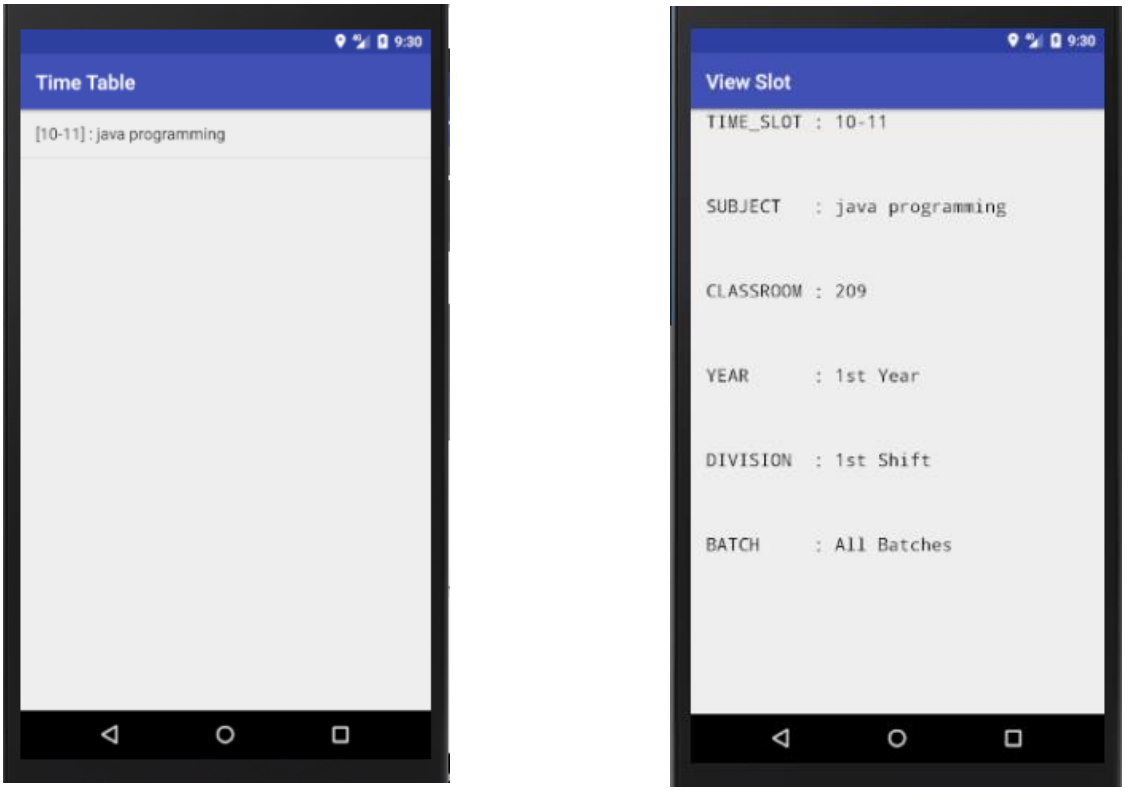


Fig: 5.2.3 View TimeTable in Geofencing with Task Management

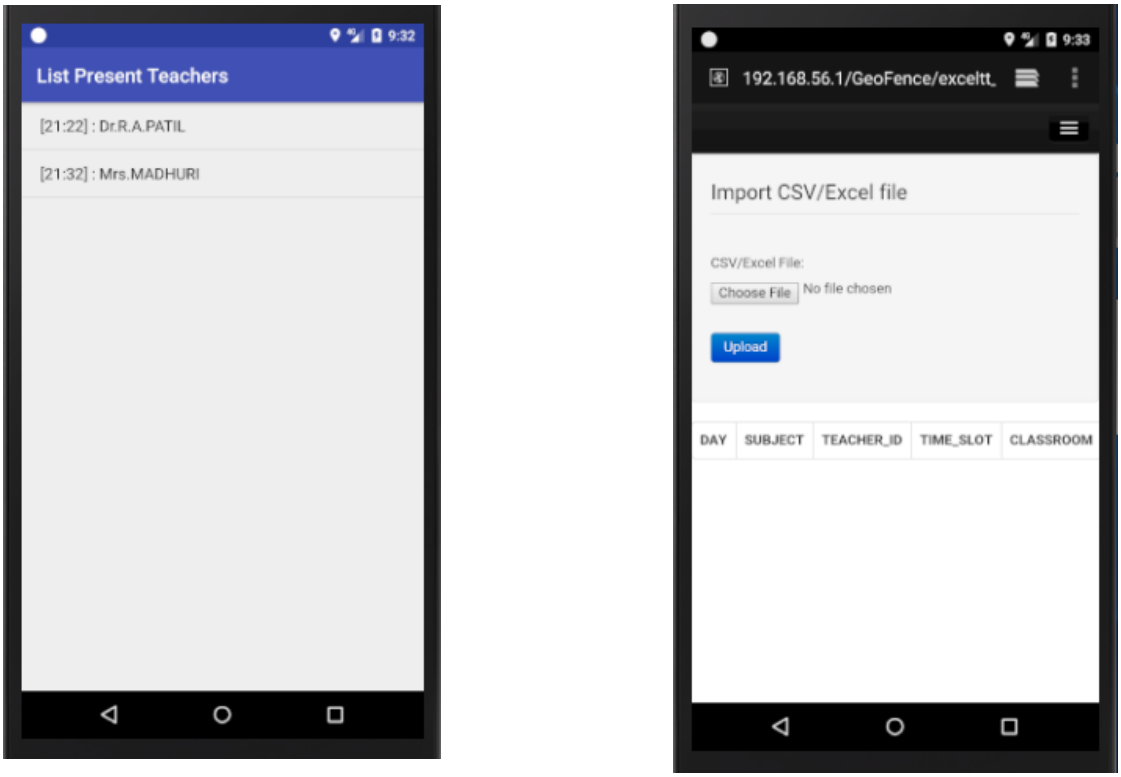


Fig: 5.2.4 Import Excel Sheet page in Geofencing with Task Management

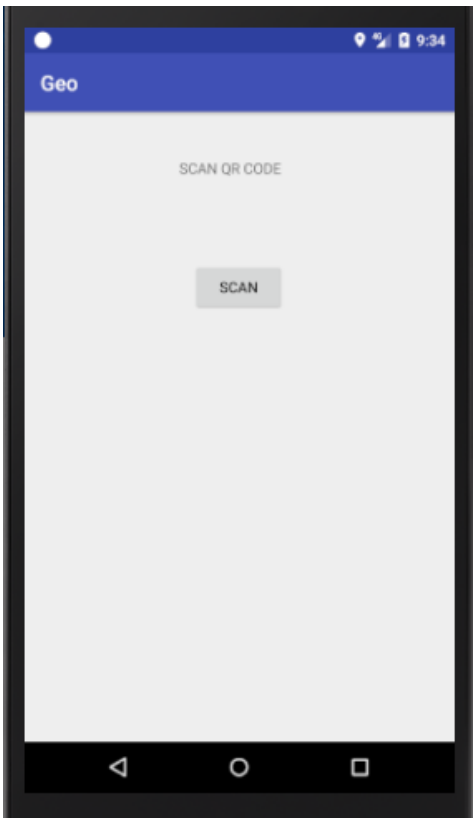
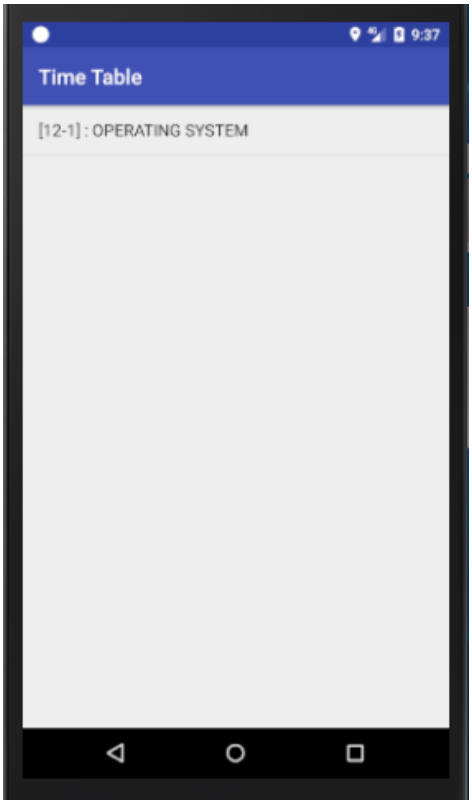


Fig: 5.2.5 Scan QRcode page of Geofencing with Task Management



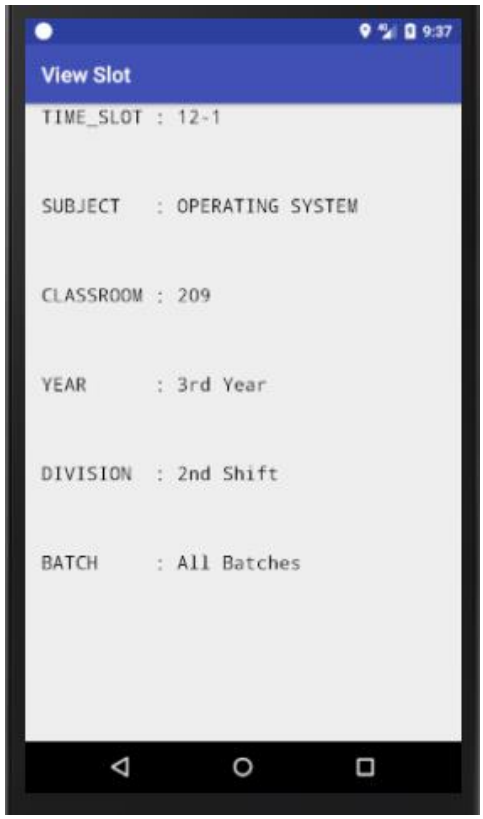
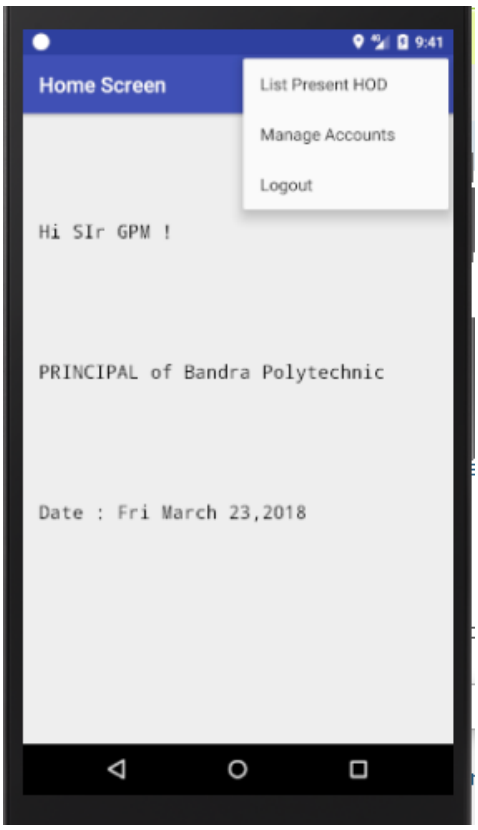
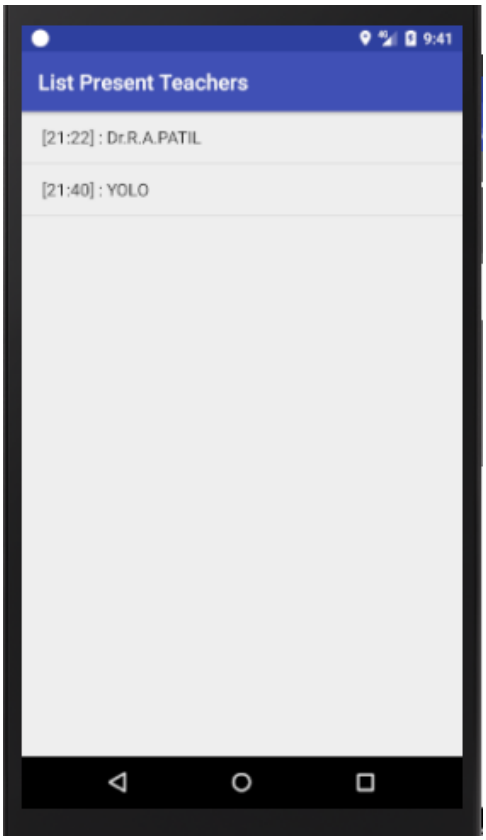


Fig: 5.2.6 View TimeTable page in Geofencing with Task Management





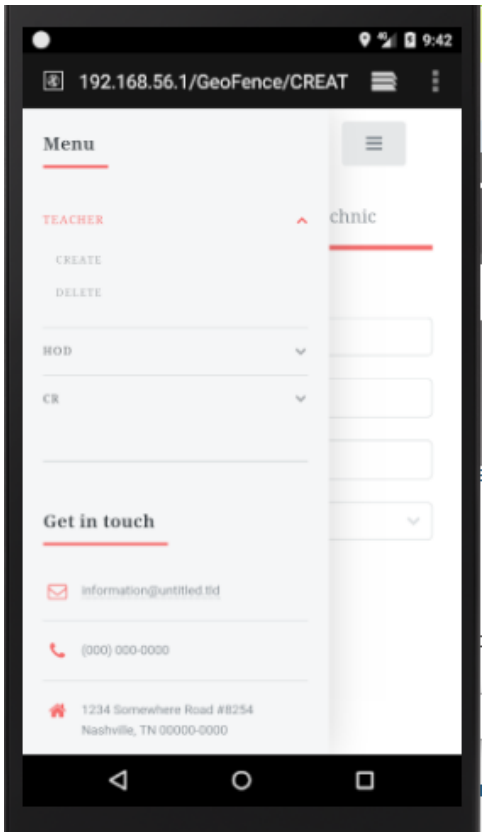


Fig: 5.2.7 Principal Page of Geofencing with Task Management

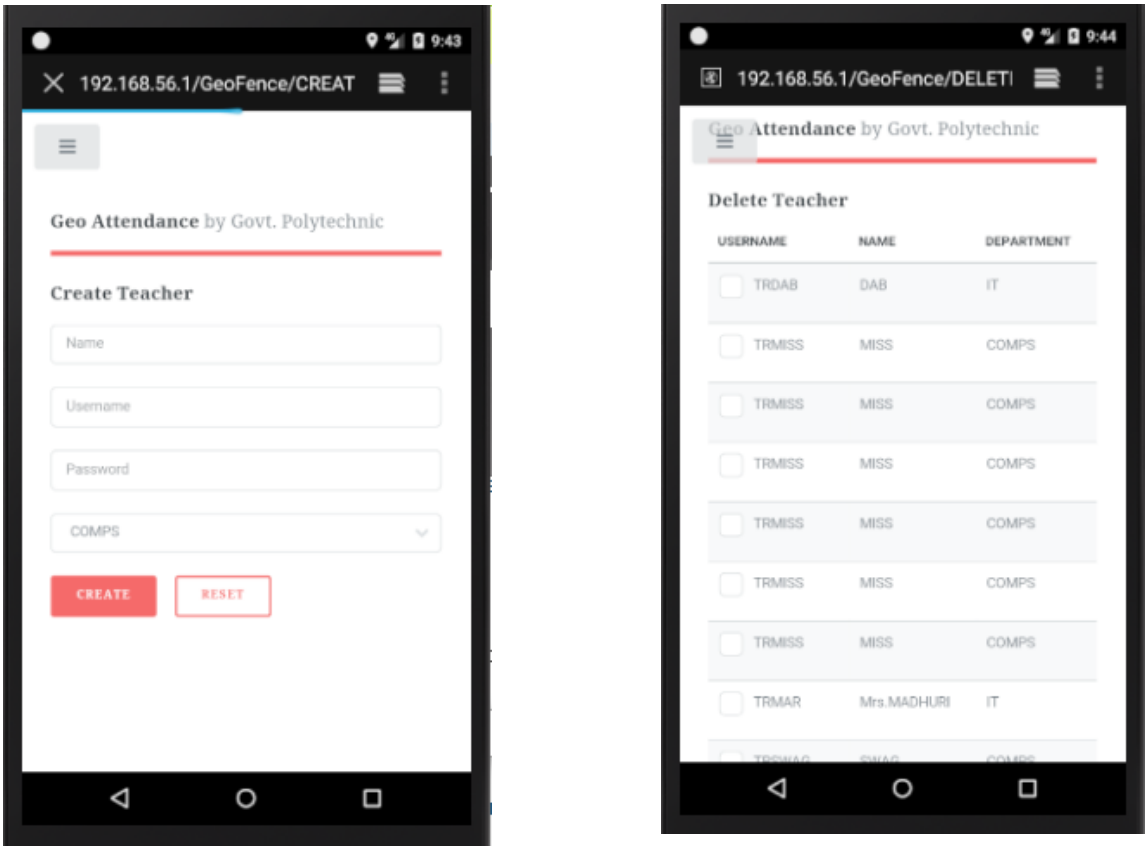
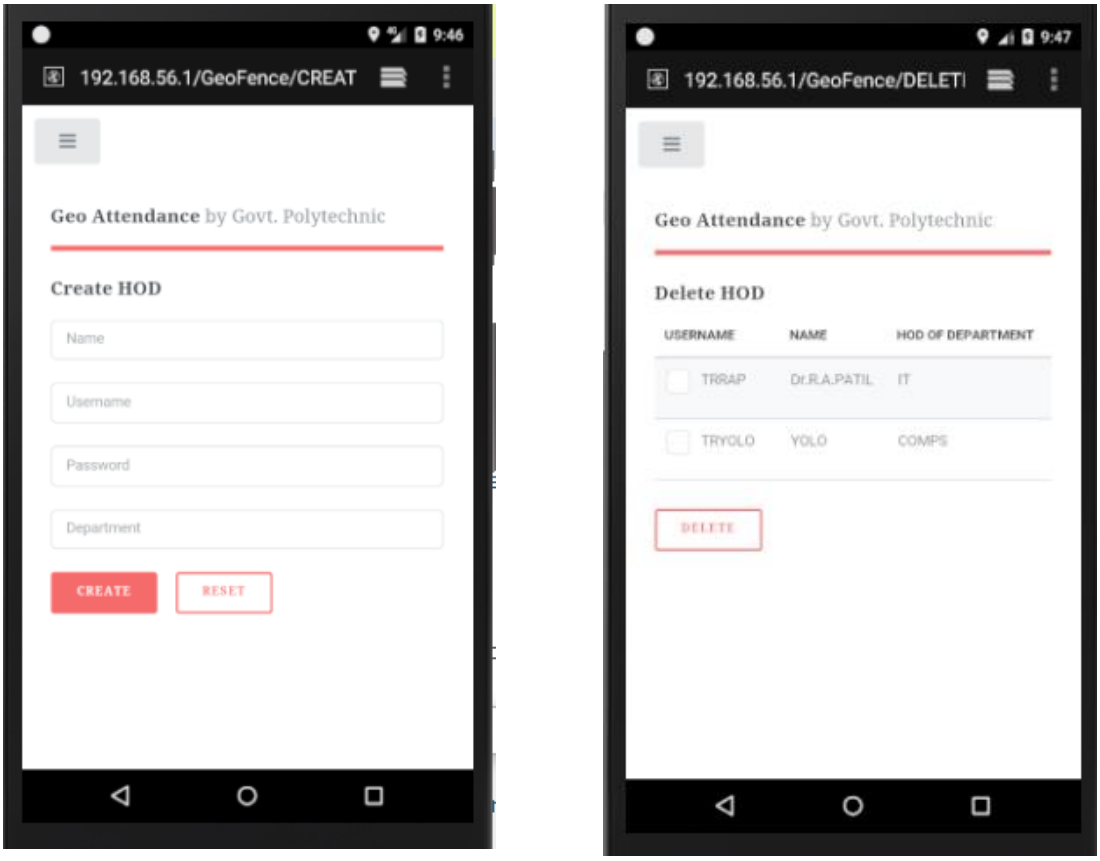


Fig: 5.2.8 Creation & Deletion page of Teacher in Geofencing with Task Management



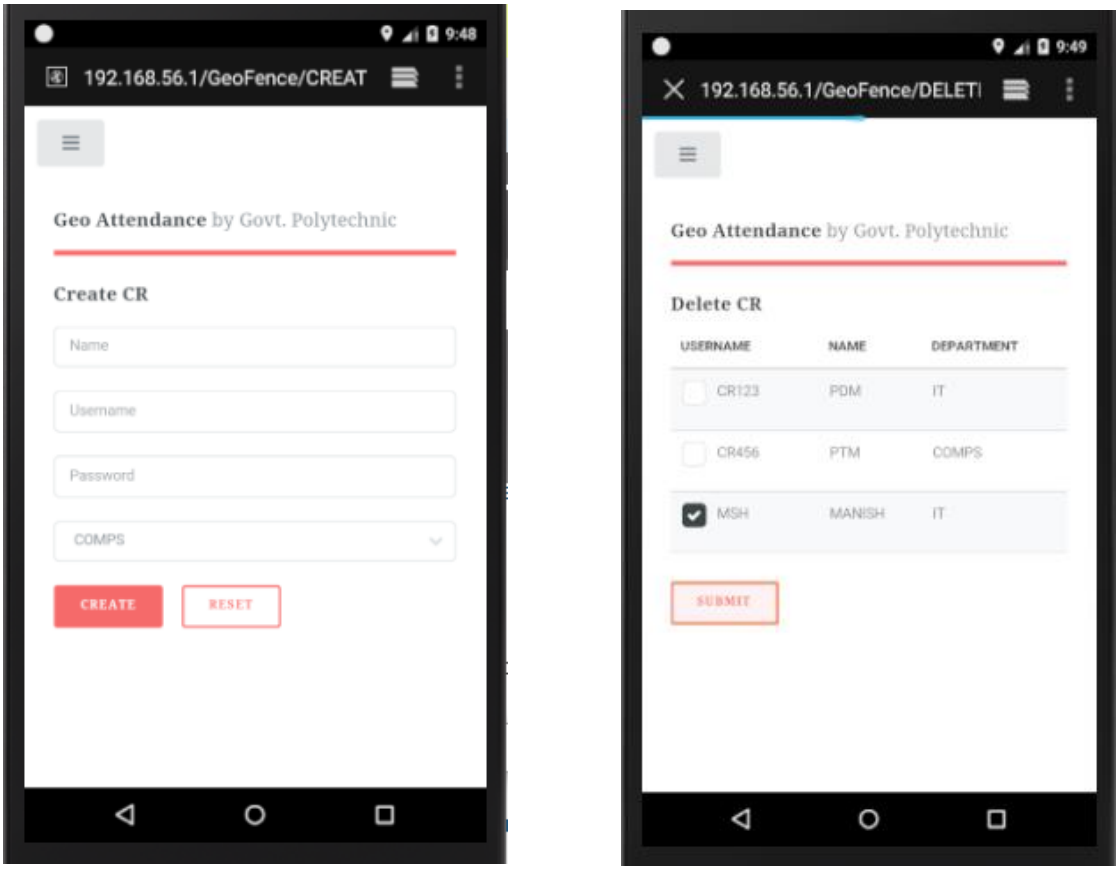


Fig: 5.2.9 Creation & Deletion page of Class Representative in Geofencing with Task Management

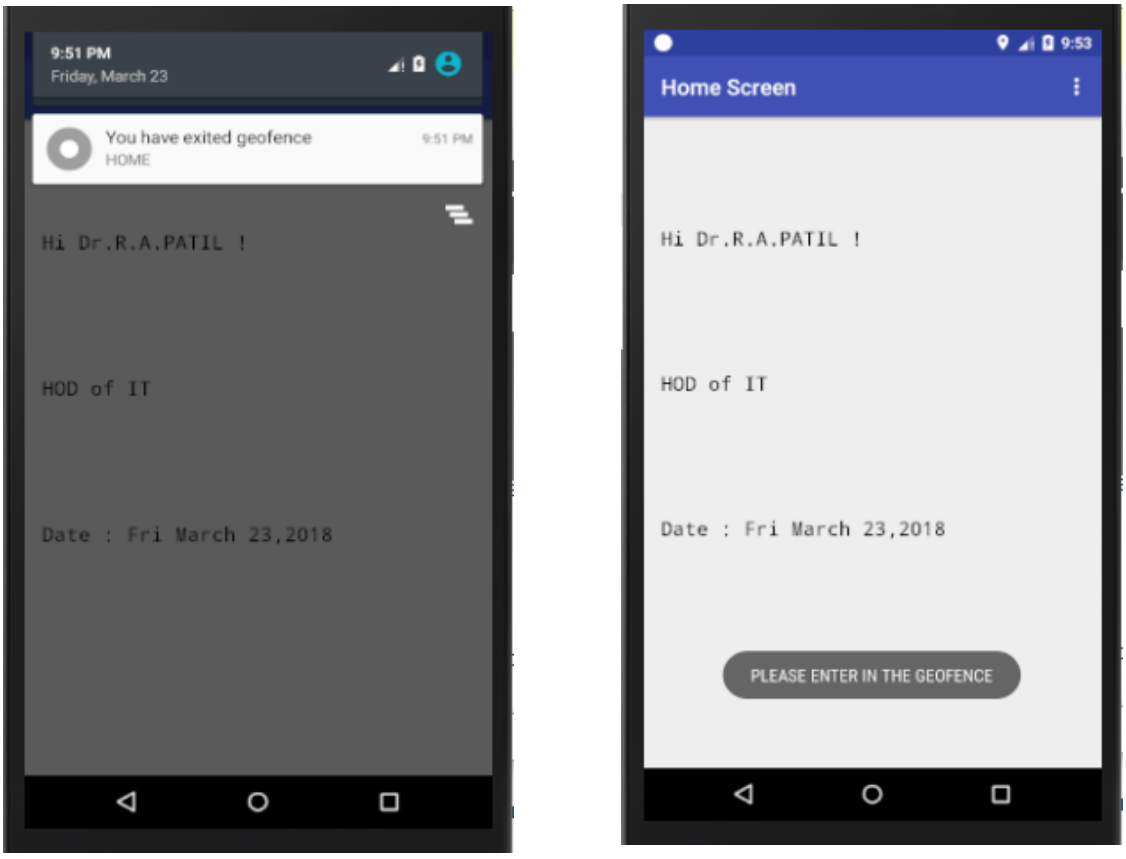
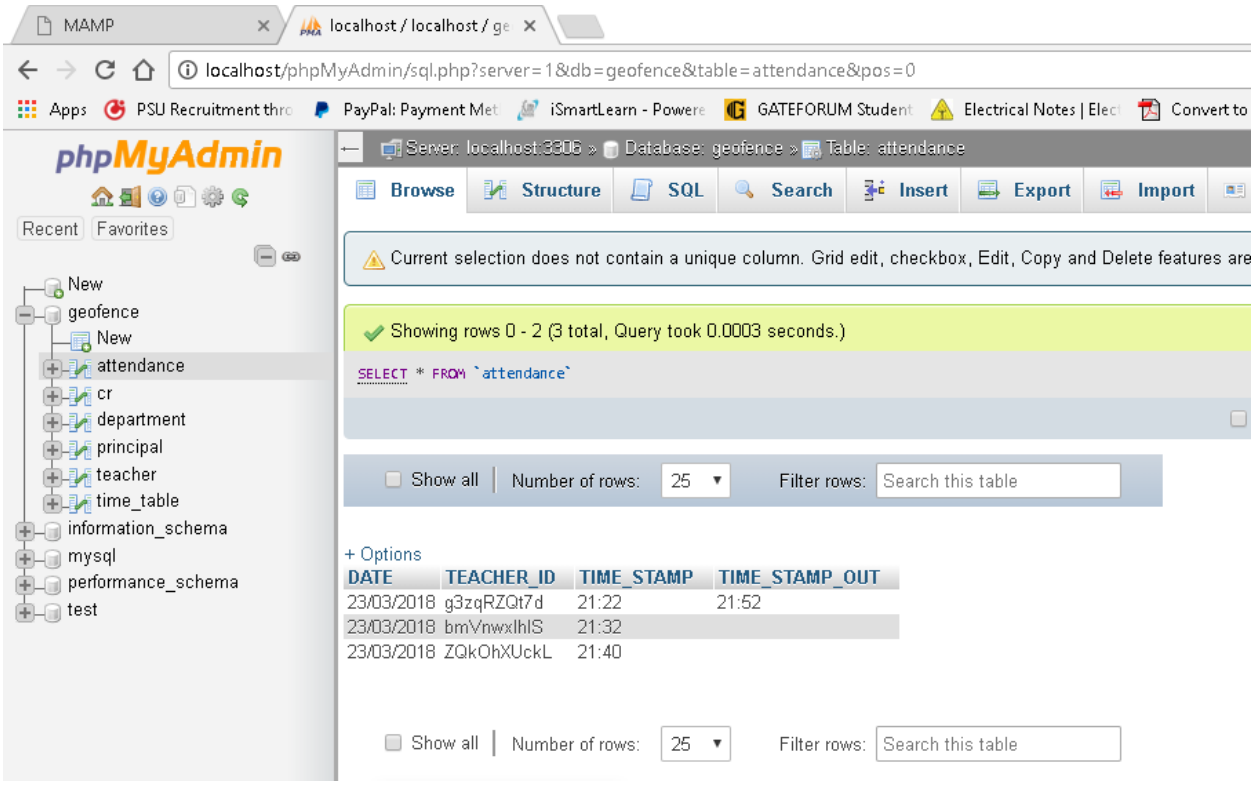
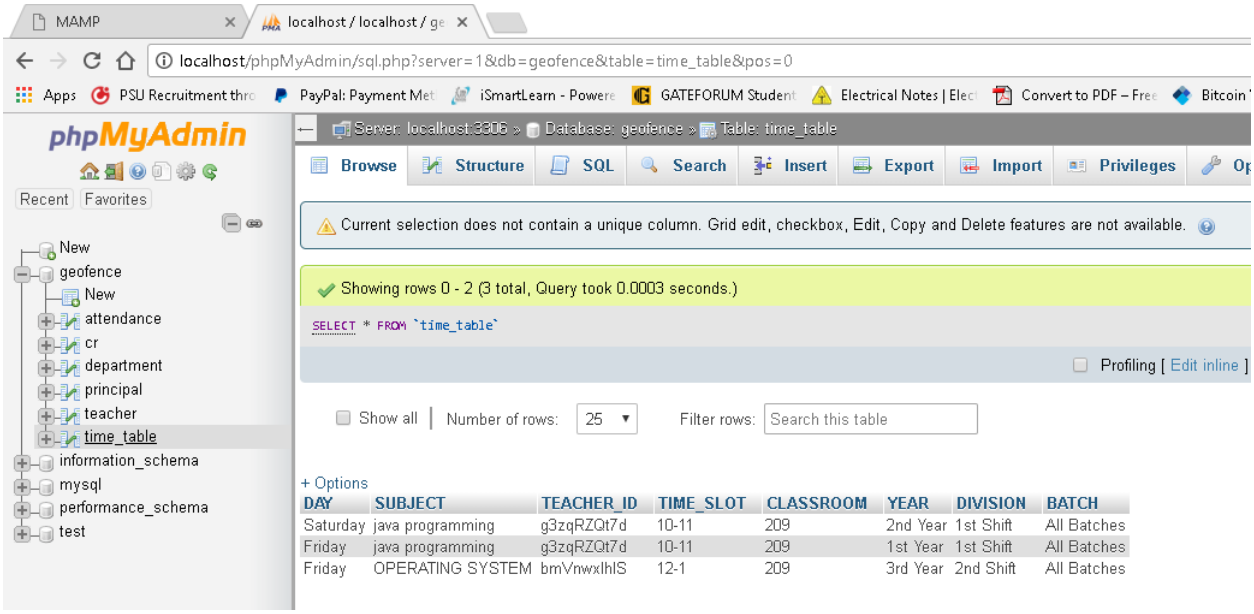


Fig: 5.2.10 Exiting from Geofencing


MAMP
localhost / localhost / ge
localhost/phpMyAdmin/sql.php?server=1&db=geofence&table=attendance&pos=0
phpMyAdmin
Recent Favorites
New
geofence
New
attendance
cr
department
principal
teacher
time_table
information_schema
mysql
performance_schema
test
Server: localhost:3306 » Database: geofence » Table: attendance
Browse Structure SQL Search Insert Export Import
Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are
Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)
SELECT * FROM `attendance`
Show all | Number of rows: 25 Filter rows: Search this table
+ Options
DATE TEACHER_ID TIME_STAMP TIME_STAMP_OUT
23/03/2018 g3zqRZQt7d 21:22 21:52
23/03/2018 bmVnwxlhlS 21:32
23/03/2018 ZQkOhXUckL 21:40
Show all | Number of rows: 25 Filter rows: Search this table


MAMP
localhost / localhost / ge
localhost/phpMyAdmin/sql.php?server=1&db=geofence&table=time_table&pos=0
phpMyAdmin
Recent Favorites
New
geofence
New
attendance
cr
department
principal
teacher
time_table
information_schema
mysql
performance_schema
test
Server: localhost:3306 » Database: geofence » Table: time_table
Browse Structure SQL Search Insert Export Import Privileges
Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.
Showing rows 0 - 2 (3 total, Query took 0.0003 seconds.)
SELECT * FROM `time_table`
Profiling [Edit inline]
Show all | Number of rows: 25 Filter rows: Search this table
+ Options
DAY SUBJECT TEACHER_ID TIME_SLOT CLASSROOM YEAR DIVISION BATCH
Saturday java programming g3zqRZQt7d 10-11 209 2nd Year 1st Shift All Batches
Friday java programming g3zqRZQt7d 10-11 209 1st Year 1st Shift All Batches
Friday OPERATING SYSTEM bmVnwxlhlS 12-1 209 3rd Year 2nd Shift All Batches

Chapter 6
Testing

6.1 Testing:

Module 1: Login Page

TEST CASE	USERNAME	PASSWORD	EXCEPTED OUTPUT	ACTUAL OUTPUT
1	TRRAP	TRRAP	Redirect to Head of Department Logged in page	Redirect to Head of Department Logged in page
2	TRRAP		Error: Not able to Login. Enter the password	Error: Not able to Login. Enter the password
3	RAP	RAP	Error: Invalid username	Error: Invalid Username
4	TRMAD	TRMAD	Redirect to Teacher Logged in page	Redirect to Teacher Logged in page
5	PRINCI	PRINCI	Redirect to Principal Logged in page	Redirect to Principal Logged in page

Module 2: Geofencing Verification / Access Panel

TEST ID	TEST SCENARIO	TEST CASE	PRE-CONDITION	TEST DATA	EXPECTED OUTPUT	ACTUAL OUTPUT	POST-CONDITION
1	Head of Department (Logged in page)	Must entered in geofenced area	Login should be successful. The user must login through Head of Department. Head of Department logged in page should be loaded successfully.	(Not in geofence)	On clicking any of the functionalities, Head of Department would not be able to access it.	Toast message "Enter into Geofence"	The user must be in geofenced area.
2	Head of Department (Logged in page)	Must entered in geofenced area	Login should be successful. The user must login through Head of Department. Head of Department logged in page should be loaded successfully.	(Inside the geofence)	On clicking the Head of Department functionalities, he must be able to access it.	On clicking the Head of Department functionalities, he must be able to access it.	
3	Head of Department (Logged in page)	Accessing the functionalities such as View Timetable.	Login should be successful and he/she should access it using Head of Department Login. Must be geofence.		On selecting the "View Timetable", Head of Department must be able to see his today's timetable.	Your today's Timetable :- Time : Batch : Class : Year : Division : Subject :	
4	Head of Department (Logged in	Accessing the functionalities such as List	Login should be successful and he/she		On selecting the "List Present Teacher", Head	List of Present Teachers :- Teacher1[time]	

	page)	Present Teachers.	should access it using Head of Department Login. Must be geofence.		of Department must be able to see a list of present teachers of their department and if Head of Department clicks on teacher name, details of their schedule should be viewed.	Teacher2[time]	
5	Head of Department (Logged in page)	Accessing the functionalities such as Import Excel Timetable.	Login should be successful and he/she should access it using Head of Department Login. Must be geofence.		On selecting the "Import Excel Timetable", the import excel.php should open successfully.	In Import Excel Timetable:- Choose the CSV file and Upload.	
6	Head of Department (Logged in page)	Accessing the functionalities such as Importing Excel Timetable.	Login should be successful and he/she should access it using Head of Department Login. Must be in geofence.	Import Excel Timetable webpage should loaded successfully.	After that webpage has loaded, select the ".CSV" file of your excel by clicking on upload file and import it.	Toast message "File Uploaded Successfully."	
7	Head of Department (Logged in page)	Accessing the functionalities such as Edit Timetable.	Login should be successful and he/she should access it using Head of Department Login. Must be in geofence.		On clicking the "Edit Timetable", the Edit Timetable should get loaded successfully.	On clicking the "Edit Timetable", the Edit Timetable should get loaded successfully.	
8	Head of Department (Logged in page)	Editing the Timetable by inputting details.	Login should be successful and he/she should access it using Head of Department Login. Must be in geofence. Edit Timetable page should be loaded successfully.	Time : 10-11am Day : Wednesday Subject : Java Programming Classroom No. : 209 Year : 2nd Year Shift : 1st Shift Batches : All	Head of Department should enter the details for changing the schedule of teachers which should be valid and click the Submit button toast message "Timetable Edited Successfully".	On clicking the Submit button toast a message "Timetable Edited Successfully".	
9	Head of Department (Logged in page)	Accessing the functionalities such as Scan QR Code.	Login should be successful and he/she should access it using Head of Department Login. Must be in		On clicking "Scan QR Code", the QR Code page should get loaded successfully.	On clicking "Scan QR Code", the QR Code page should get loaded successfully.	

			geofence.				
10	Head of Department (Logged in page) and Scan QR Code	Scanning the QR Code.	Login should be successful and he/she should access it using Head of Department Login. Must be in geofence. Scan QR Code page should be loaded.		On Clicking Scan Button, (Camera with red line) Scanner should get opened. Scan the QR Code and toast "Success" message.	On Clicking Scan Button, (Camera with red line) Scanner should get opened. Scan the QR Code and toast "Success" message.	
11	Head of Department (Logged in page)	Accessing the functionalities such as Logout.	Login should be successful and he/she should access it using Head of Department Login. Must be in geofence.		On clicking the Logout Button, basic Login page should be loaded.	On clicking the Logout Button, basic Login page should be loaded.	
12	Head of Department (Logged in page)	Accessing the functionalities such as Manage Accounts.	Login should be successful and he/she should access it using Head of Department Login. Must be in geofence.		On clicking the Manage Accounts, the "Manage Accounts" page should be loaded successfully. Head of Department has the ability to create teacher and CR.	On clicking the Manage Accounts, the "Manage Accounts" page should be loaded successfully. Head of Department has the ability to create teacher and CR.	
13	Head of Department (Logged in page)	Accessing the functionalities such as creating Teacher.	Login should be successful and he/she should access it using Head of Department Login. Must be in geofence. Manage Account page should be successfully loaded.	Name : Madhuri Arade Username : TRMAD Password : TRMAD Department : IT	On clicking the Sidebar, the Head of Department can create Teacher, fill details and Submit it. Toast "Teacher created Successfully" message.	On clicking the Sidebar, the Head of Department can create Teacher, fill details and Submit it. Toast "Teacher created Successfully" message.	
14	Head of Department (Logged in page)	Accessing the functionalities such as Creating CR.	Login should be successful and he/she should access it using Head of Department Login. Must be in geofence. Manage Account page should be	Name : Dhanashree Bhandare Username : CRDB Password : CRDB Department : IT	On clicking the Sidebar, the Head of Department can create CR, fill details and Submit it. Toast "Teacher created Successfully" message.	On clicking the Sidebar, the Head of Department can create Teacher, fill details and Submit it. Toast "Teacher created Successfully" message.	

			successfully loaded.				
15	Head of Department (Logged in page)	Accessing the functionalities such as Deleting Teacher.	Login should be successful and he/she should access it using Head of Department Login. Must be in geofence. Manage Account page should be successfully loaded.		On clicking the Sidebar, the Head of Department can Delete the Teacher by clicking it, checked the Teacher name and delete it. Toast "Teacher Deleted Successfully" message.	On clicking the Sidebar, the Head of Department can Delete the Teacher by clicking it, checked the Teacher name and delete it. Toast "Teacher Deleted Successfully" message.	
16	Head of Department (Logged in page)	Accessing the functionalities such as Deleting CR.	Login should be successful and he/she should access it using Head of Department Login. Must be in geofence. Manage Account page should be successfully loaded.		On clicking the Sidebar, the Head of Department can Delete the CR by clicking it, checked the CR name and delete it. Toast "CR Deleted Successfully" message.	On clicking the Sidebar, the Head of Department can Delete the CR by clicking it, checked the CR name and delete it. Toast "CR Deleted Successfully" message.	
17	Teacher (Logged in page)	Must entered in geofenced area	Login should be successful. The user must login through Teacher. Teacher logged in page should be loaded successfully.	(Not in geofence)	On clicking any of the functionalities, Teacher would not be able to access it.	Toast message "Enter into Geofence".	The user must be in geofenced area.
18	Teacher (Logged in page)	Must entered in geofenced area	Login should be successful. The user must login through Teacher. Teacher logged in page should be loaded successfully.	(Inside the geofence)	On clicking the Teacher functionalities, he must be able to access it.	On clicking the Teacher functionalities, he must be able to access it.	
19	Teacher (Logged in page)	Accessing the functionalities such as View Timetable.	Login should be successful and he/she should access it using Teacher Login. Must be geofence.		On selecting the "View Timetable", Teacher must be able to see his today's timetable.	Your today's Timetable :- Time : Batch : Class : Year : Division : Subject :	
20	Teacher (Logged in page)	Accessing the functionalities such as List Present Teachers.	Login should be successful and he/she should access it using Teacher		On selecting the "List Present Teacher", Teacher must be able to see a list of present	List of Present Teachers :- Teacher1[time] Teacher2[time]	

			Login. Must be geofence.		teachers of their department and if Teacher clicks on Teacher's name, details of their schedule should be viewed.		
21	Teacher (Logged in page)	Accessing the functionalities such as Import Excel Timetable.	Login should be successful and he/she should access it using Teacher Login. Must be geofence.		On selecting the "Import Excel Timetable", the import excel.php should open successfully.	In Import Excel Timetable:- Choose the CSV file and Upload.	
22	Teacher (Logged in page)	Accessing the functionalities such as Importing Excel Timetable.	Login should be successful and he/she should access it using Teacher Login. Must be in geofence.	Import Excel Timetable webpage should loaded successfully.	After that webpage has loaded, select the ".CSV" file of your excel by clicking on upload file and import it.	Toast message "File Uploaded Successfully."	
23	Teacher (Logged in page)	Accessing the functionalities such as Scan QR Code.	Login should be successful and he/she should access it using Teacher Login. Must be in geofence.		On clicking "Scan QR Code", the QR Code page should get loaded successfully.	On clicking "Scan QR Code", the QR Code page should get loaded successfully.	
24	Teacher (Logged in page) and Scan QR Code	Scanning the QR Code.	Login should be successful and he/she should access it using Teacher Login. Must be in geofence. Scan QR Code page should be loaded.		On Clicking Scan Button, (Camera with red line) Scanner should get opened. Scan the QR Code and toast "Success" message.	On Clicking Scan Button, (Camera with red line) Scanner should get opened. Scan the QR Code and toast "Success" message.	
25	Teacher (Logged in page)	Accessing the functionalities such as Logout.	Login should be successful and he/she should access it using Teacher Login. Must be in geofence.		On clicking the Logout Button, basic Login page should be loaded.	On clicking the Logout Button, basic Login page should be loaded.	
26	CR (Logged in page)	Must entered in geofenced area	Login should be successful. The user must login through CR. CR logged in page should be loaded successfully.	(Not in geofence)	On clicking any of the functionalities, CR would not be able to access it.	Toast message "Enter into Geofence".	The user must be in geofenced area.

27	CR (Logged in page)	Must entered in geofenced area	Login should be successful. The user must login through CR. CR logged in page should be loaded successfully.	(Inside the geofence)	On clicking the CR functionalities, he must be able to access it.	On clicking the CR functionalities, he must be able to access it.	
28	CR (Logged in page)	Accessing the functionalities such as View Timetable.	Login should be successful and he/she should access it using CR Login. Must be geofence.		On selecting the "View Timetable", CR must be able to see today's timetable.	Your today's Timetable :- Time : Batch : Class : Year : Division : Subject :	
29	CR (Logged in page)	Accessing the functionalities such as List Present Teachers.	Login should be successful and he/she should access it using CR Login. Must be geofence.		On selecting the "List Present Teacher", CR must be able to see a list of present teachers of their department and if CR clicks on Teacher's name, details of their schedule should be viewed.	List of Present Teachers :- Teacher1[time] Teacher2[time]	
30	CR (Logged in page)	Accessing the functionalities such as Logout.	Login should be successful and he/she should access it using CR Login. Must be in geofence.		On clicking the Logout Button, basic Login page should be loaded.	On clicking the Logout Button, basic Login page should be loaded.	

Chapter 7

Future Scope

7.1 Future Scope:

- The security can be more enhanced using biometrics like Iris scan, Retina scan or Finger print scan.
- Functionality of Student Attendance-another module can be added for teachers for taking the attendance of student in the same app thus reducing the paper work.
- Different Graphical User Interface (GUI) to be integrated with different schools and colleges.
- Teachers can be monitored not only within the campus but also in classroom whether they are actually conducting the lectures in classroom or not.

Chapter 8

Conclusion

8.1 Conclusion:

Geofencing with Task Management is an android app that is built without using any hardware module except the android smartphone for colleges, school or university so that the staff's burden load can be decreased a little. The proposed system will make the procedure of task management of teacher so much easier which may otherwise is required to be done manually, which takes lots of time and human efforts. So proposed system helps to overcome this disadvantage. This system collapses all errors of the previous generation system and provides user friendly GUI. Proposed system helps to differentiates Principal, Head of Department and teacher functionalities hence provides security. This also helps to monitor their schedule instead of doing paper work daily. A number of hours which we are spending to manage the schedule of teacher and their attendance record can be reduced through our app known as GEOFENCING WITH TASK MANAGEMENT.

Chapter 9

References

9.1 References:

1. Ijeh, Anthony C., et al. "Geofencing Security Engineering." *Proceedings of the International MultiConference of Engineers and Computer Scientists*. Vol. 2. 2010.
2. https://blog.beaconstac.com/2015/09/beacons-vs-Geofencing-which-location-aware-technology-should-your-business-use/
3. Krify.https://krify.co/tag/Geofencing-and-rfid/
4. https://www.techopedia.com/definition/14937/geofencing
5. Dube, Sibusisiwe, et al. "QR code based patient medical health records transmission: Zimbabwean case." *Proceedings of informing science & IT education conference (InSITE)*. 2015.
6. Tseng, Ming-Hseng, and Hui-Ching Wu. "A cloud medication safety support system using QR code and Web services for elderly outpatients." *Technology and Health Care* 22.1 (2014): 99-113. Rahate, S.W. and Shaikh, M.Z., 2016. Geo-fencing Infrastructure: Location Based Service.
7. Sergio Tenreiro de Magalhaes, Hamid Jahankhani, Ali G. Hessami,EdGlobal Security, Safety, and Sustainability: 5th International Conference, ICGS3 2009, London, UK, September 1-2, 2009, Proceedings