1440

# MU-Zulfi Campus Query

Kingdom of Saudi Arabia
Ministry of Higher Education
Majmaah University
College of Science in Al-Zulfi
Department of Computer Science and Information

المملكة العربية السعودية
وزارة التعليم العالى
جامعة المجمعة
كلية العلوم بالزلفى
قسم علوم الحاسب والمعلومات

**Department of Computer Science and Information**

Project Report for the 2st Semester of year 1439-1440

**Project Title:**

# MU-Zulfi Campus Query

Submitted by:

**Student: Hamad Ali Alname**

**ID : 342101728**

Supervisor:

**Dr. Abdullah Bajazer**

2019

# Abstract

Today, mobile applications are being used for different new fields. In this project, a mobile application is produced to help visitors and students to reach any places in the campus college of Science, at Al-Zulfi, in Majmaah university by providing them with either text instructions or path map with low time and effort.

In this project there are two choices for users to reach their target. Place The first choice is to display text instructions which describe the path that users can track. The second choice is to display a virtual map that will describe the path to destination with graph including directed lines from source place to target place they want to visit.

# Acknowledgement

I want to thank the supervisor of my project, **Dr. Abdullah Bajazer,** for his encouragement in order to complete this project and helping me in extracting the important things to achieve them in this work.

Also, I give my deep thanks to my college friends for their support and advices during working in this project.

Finally, I also thank my family for their continuous encouragement which increases my self-confidence to complete the project.

Thanks,

*Hamad Ali*

**Certificate by student**

# Table of Content

## 1. Contents

# List of Figures

# 1.1) Chapter 1: Introduction

The establishment of Majmaah University came as a result of the decree of the Custodian of the Two Holy Mosques King Abdullah Bin Abdul Aziz Al-Saud and the Prime Minister and Chairman of Higher Education on Ramadan 3rd, 1430 - 24th of August 2009 to establish Majmaah University along with three other universities in Dammam city, Kharj province and Shaqr'a province.


Figure 1. 1 Majmaah University

Majmaah University aimed to serve a wide area including Majmaah, Zulfi, Remah, Ghat and Hawtat Sudair. It also helps in achieving the Ministry of Higher Education's objective in expanding the university education across the country. Therefore, Majmaah University will meet the growing number of high school graduates in the region which will reduce the pressure on universities in big cities.

Another significant reason for the establishment of Majmaah University is the value it adds to the people of the region in various aspects including social, cultural and awareness service. Inevitably, this shall help in upgrading the level of performance appraisal of government sectors via providing advanced courses and consultations. With regard to scientific research, the University will provide programs of high quality that will be in compatible with the University strategic objectives.

The royal decree no: 194/A on Zul-Hejjah 30th, 1430 – 17th of October 2009 to appoint Dr. Khalid Sa'ad Al-Mugren as the Rector of Majmaah University with higher rank accelerated the development process at the University. Dr. Al-Mugren focused on developing the existence colleges as well as building new ones in order to increase the number of majors that will meet the market demands. The concern of Dr.khaled Al-Mugren is to make Majmaah University a beacon of knowledge and enlightenment that is capable of offering education of high quality.

The Department seeks to qualify the students as specialists in the field of computer science in terms of study and analysis of the computer system and methods of its construction. Students are equipped with experience in using software tools such as operating systems, various programming languages and computer networks. A comprehensive curriculum has been designed to provide necessary skills to the students for their proficiency in this field. The department staff is continuously working for updating the B.Sc. program and revising its curriculum in accordance with the latest technologies in computer science and the fast-changing needs of the society.

## 1.2) Problem definition

For anyone who reaches any place for the first time, he will spend a lot of time for exploring this place. Universities are considered from the most important places where any visitor can reach to study. Incoming students need to explore his/her University to reach any destination place.

For example, if any visitor /student/ wants to go to the library of the college, he/she has some choices, maybe ask someone to reach this place, or he/she can look at the manual maps at each floor in the building. Also he/she can look at all panels outside all rooms in all floors inside any building in the college.

Of course, any visitor consumes much effort and time reaching these places. Therefore, students will spend much time to reach location of their lecture halls or laboratories.


Figure 1. 2 College of Science Al-Zulfi

### 1.1.1) Goals

This project aims to provides a system which will save time and effort for anyone especially students entering the university for the first time to reach any destination.

### 1.1.2) Objectives

In this project, an android mobile application is produced to help visitors and students to reach important places in the campus of college Science, Al-Zulfi, in Majmaah university. This application prevent time and effort consuming.

Figure 1. 3 Android Mobile App

This mobile application will provide a lot of facilities like:

(1) Display a list for all important places at each layer in the college building.

(2) When the visitor chooses any destination to reach, the application will display a window including the instruction he will follow to reach it.

(3) Also, it has another option (map) to display a path on visual map that he will follow to reach to his destination easily.

### 1.1.3) Critical success factors

Here, we will discuss the critical factors or activities required for ensuring the success of the project. The main of CSF is to determine what is central to the future of the project and achievement of that future; if the objectives associated with these

**1.1.4)** Organization chart and responsibilities

## **1.3.** General rules (assumptions)

A successful project manager always keeps an eye on his project's assumptions and constraints and understands them thoroughly as the risk management depends on these assumptions. If they are failed to be properly analyzed, it may affect the project's outcome.

- Concerning this project, we assumed that:
  - This project will serve to all visitors or new students to reach any destinations place in the college campus.
  - The feedback of all users using it will be positive.
  - It will save much time and effort than before.
  - All resources required for the project will be available.

# 2. Chapter 2: System Analysis and Specification

## 2.1. Introduction

The concept of the project is to easy navigate college campus to find any place easily without need for help. The users of the project maybe student of the college or visitor.

## 2.2. Description of Data Flow Diagram (DFD)

Data flow diagram (DFD) is a picture of the movement of data between external entities and the processes and data stores within a system.

### 2.2.1) Context Diagram

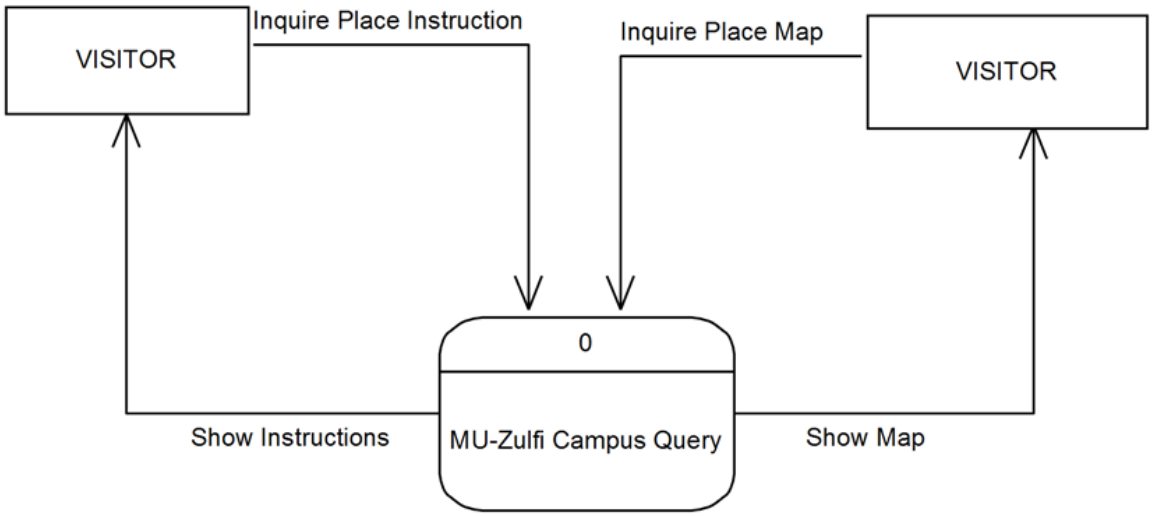



Figure 2. 1 Context Diagram for the system

**2.2.2)** Overview diagram (level 0)

**2.2.3)** Detailed DFDs

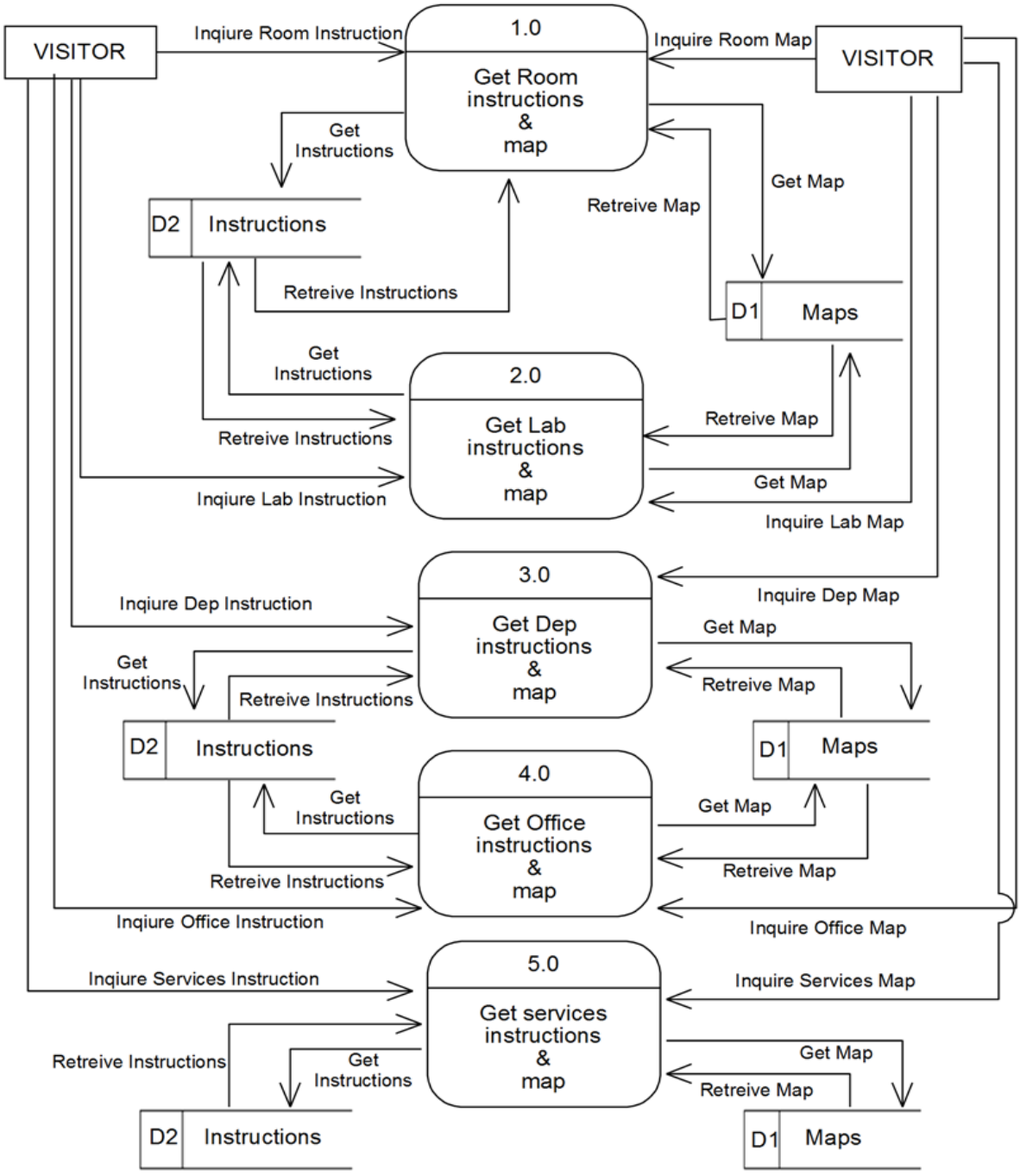



Figure 2. 2 Data Flow Diagram for the system

## 2.3. Data Modeling using (Entity Relationship Diagram (ERD) / Class diagram , Use case diagram / activity diagram)

In the above class diagram it's a disgram that illustrate the classes of the project and the functions and attributes that the the very class contain. As well as it's illustrate the relation between classes of the whole project for example in the office class the attributes of this class are

- **Id** : it's identifier that count the nuber of office that is required like office id 1,2,3 and so
- **Name** : it's contain the office name that belongs to specified doctor for example.
- **Description**: contains the path that can be viewed to the visitor to illustrate the path to him.
- **Map**: it's containd the map the illustrate the rout to the specified office .

The only function in this class is the **View()** method that is used for viewing the instructions and map of the desired office from the visitor.

Classes of Labs, Rooms, Departments, Services and adminstrations are contain the same attriutes and finction of the office class.

Map class contains the map id for specified room description for every map and the name of map that belong to a specified place.

**Campus_query** class is the class that containf object form every class in the project so it's contain the whole attributes and function of the all classes.

Class Diagram**:**

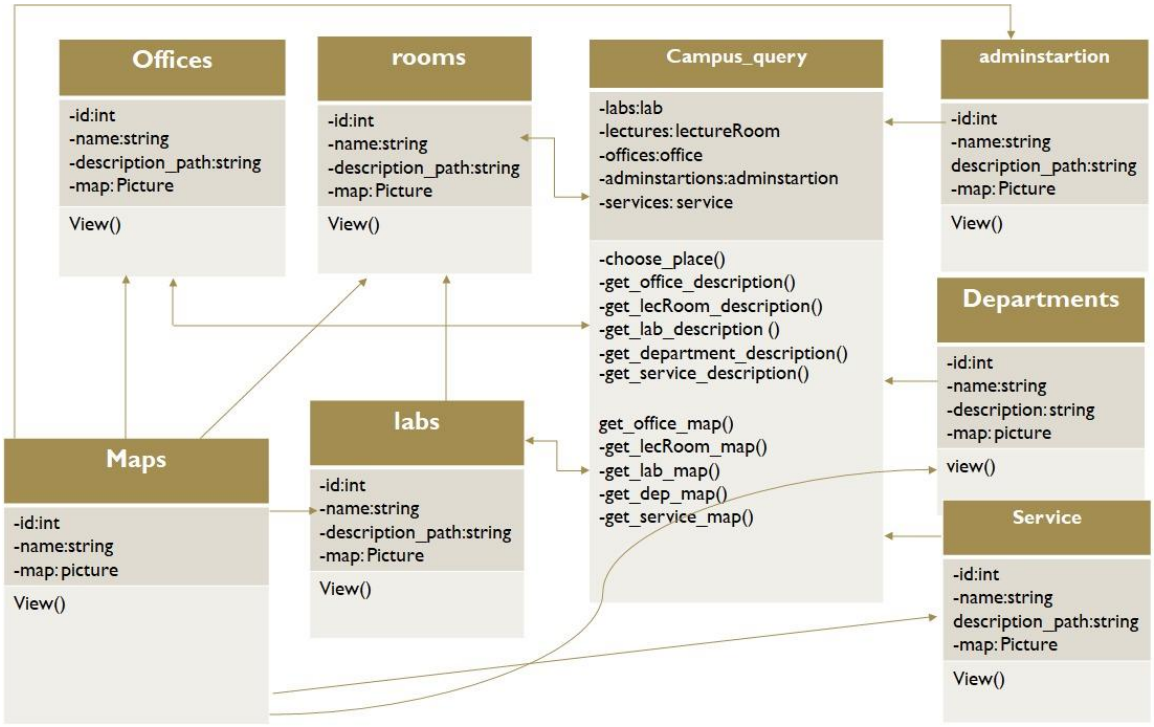



Figure 2. 3 Class Diagram for the system

Use Case Diagram:

- Use case that shows how the visitor acts to the application.

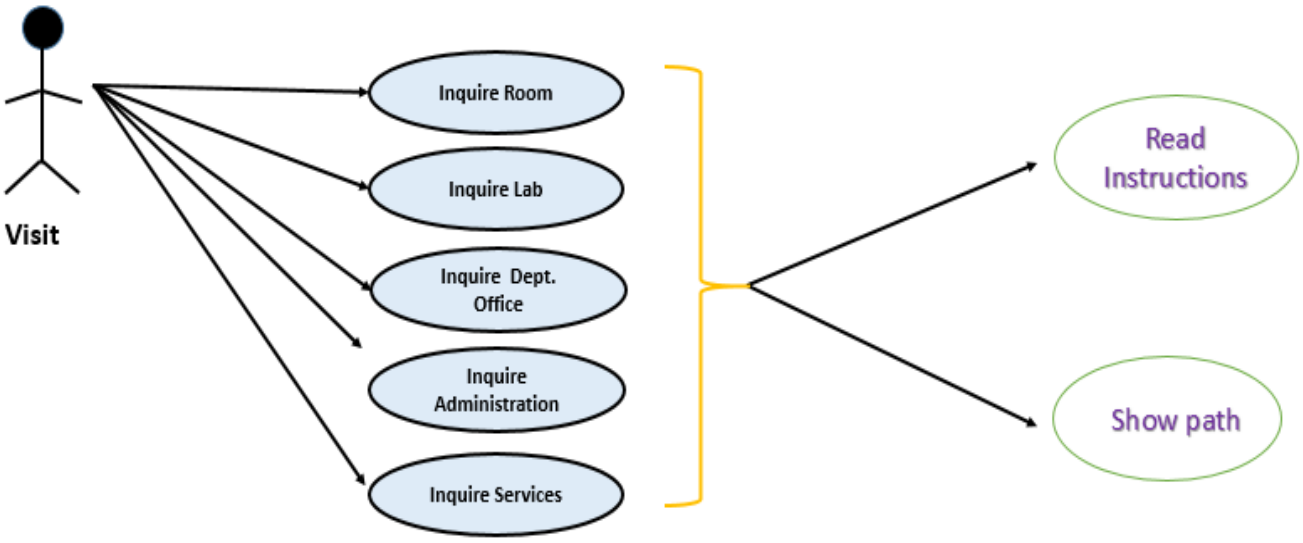



Figure 2. 4 Use Case diagram for the visitors

Sequence Diagram:

- Sequnce diagram of the Visitor in the system and how he interacts with the system via select source, select destination, find path, view instructions and view graphic map.

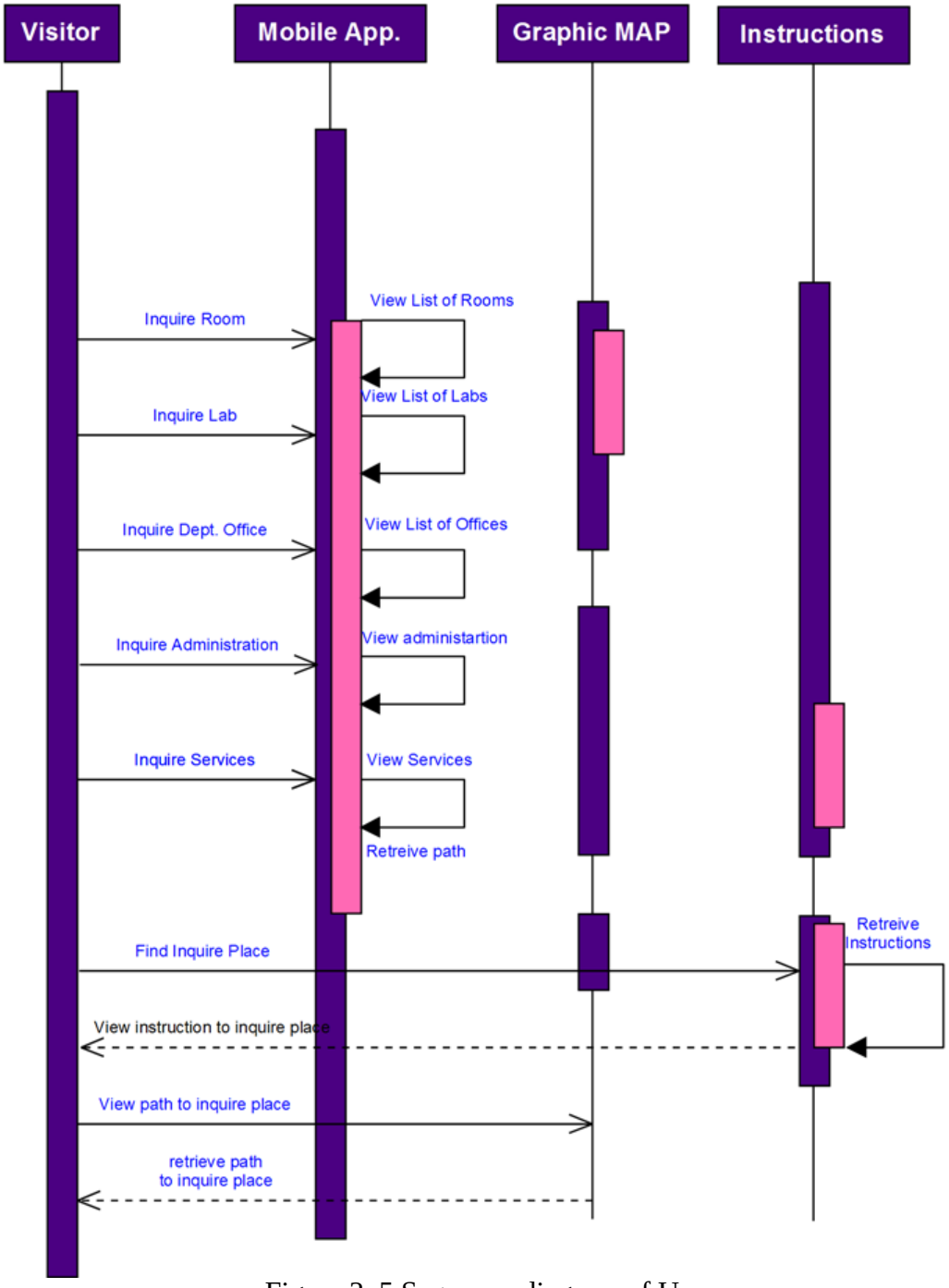



Figure 2. 5 Sequence diagram of User

## Activity Diagram

The basic purposes of activity diagrams. It captures the dynamic behavior of the system. Other diagrams are used to show the message flow from one object to another, but activity diagram is used to show message flow from one activity to another.

Activity is an operation of the system. Activity diagrams are not only used for visualizing dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in activity diagram is the message part.

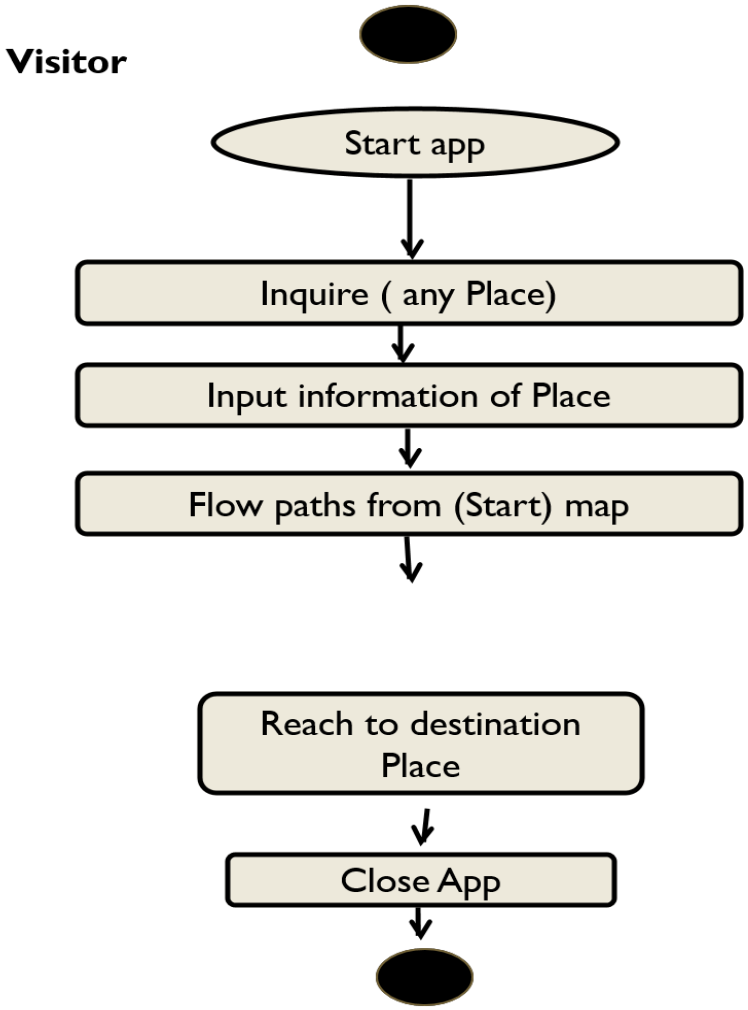



Figure 2. 6 Activity diagram

State Diagram

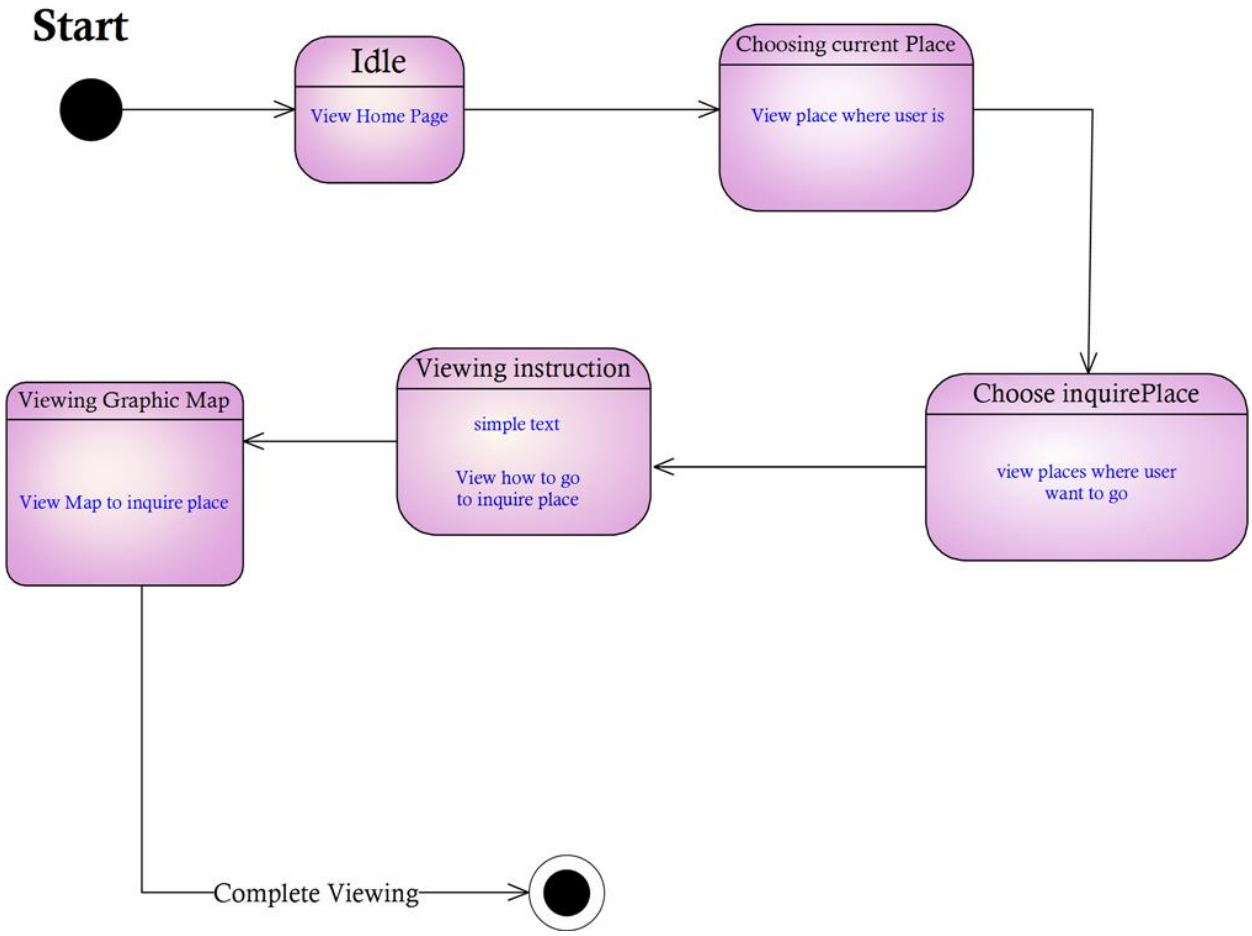



Figure 2. 7 State diagram

## Mapping Diagram

Mapping is the process of converting the entity relationship diagram (ERD) to database tables which we use it to create the database in any framework such as MY SQL workbench,. In this diagram, we join between tables by foreign key.

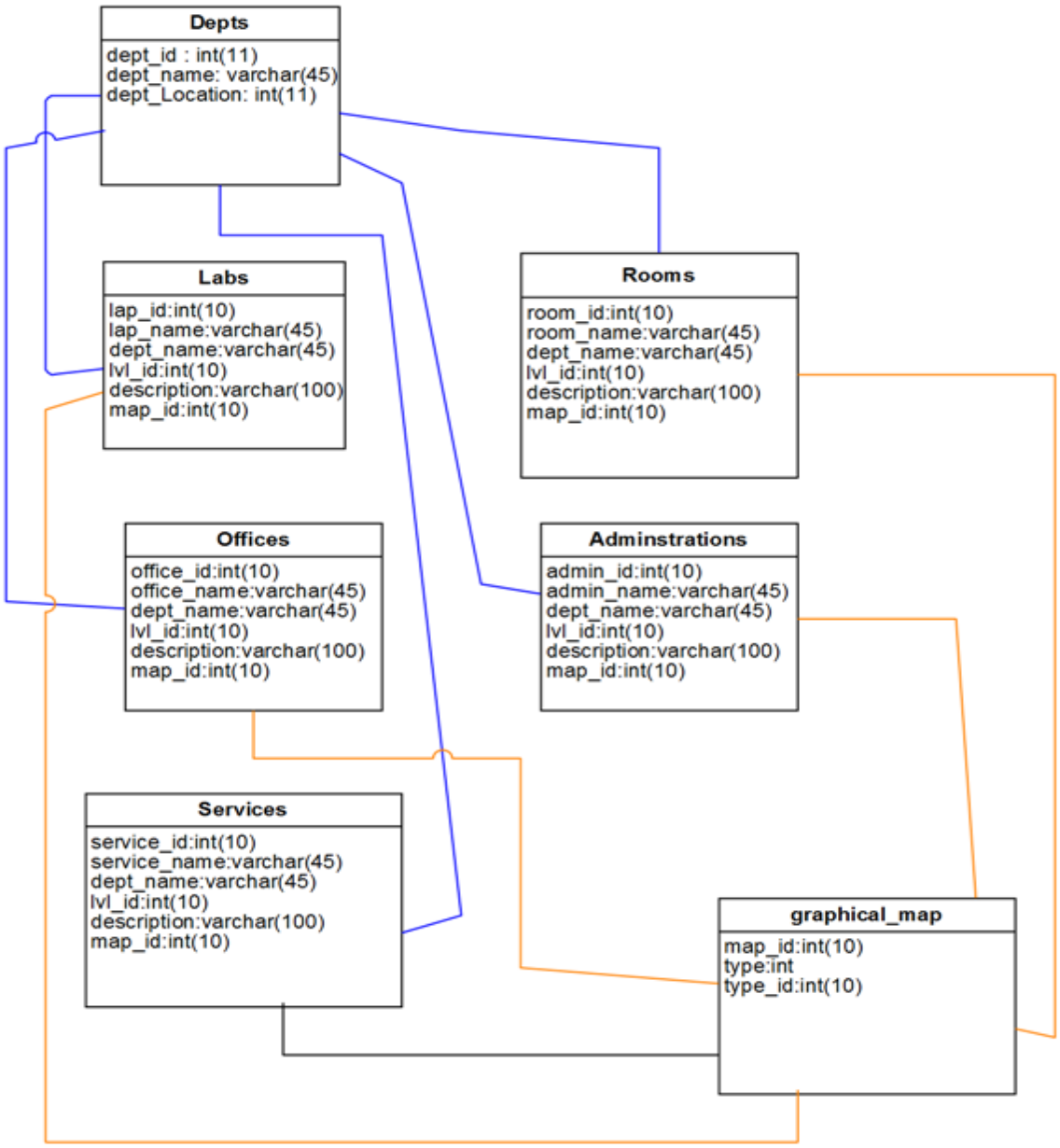



Figure 2. 8 Entity Relationship Diagram

.

### **2.3.1)** Description of Entities

### **2.3.2)** Description of relations

### **2.3.3)** Drawing ERD.

An entity-relationship model (ERM) is an abstract and conceptual representation of data. Entity-relationship modeling is a database modeling method, used to produce a type of conceptual schema or semantic data model of a system, often a relational database, and its requirements in a top-down fashion. Diagrams created by this process are called entity-relationship diagrams, ER diagrams, or ERDs.

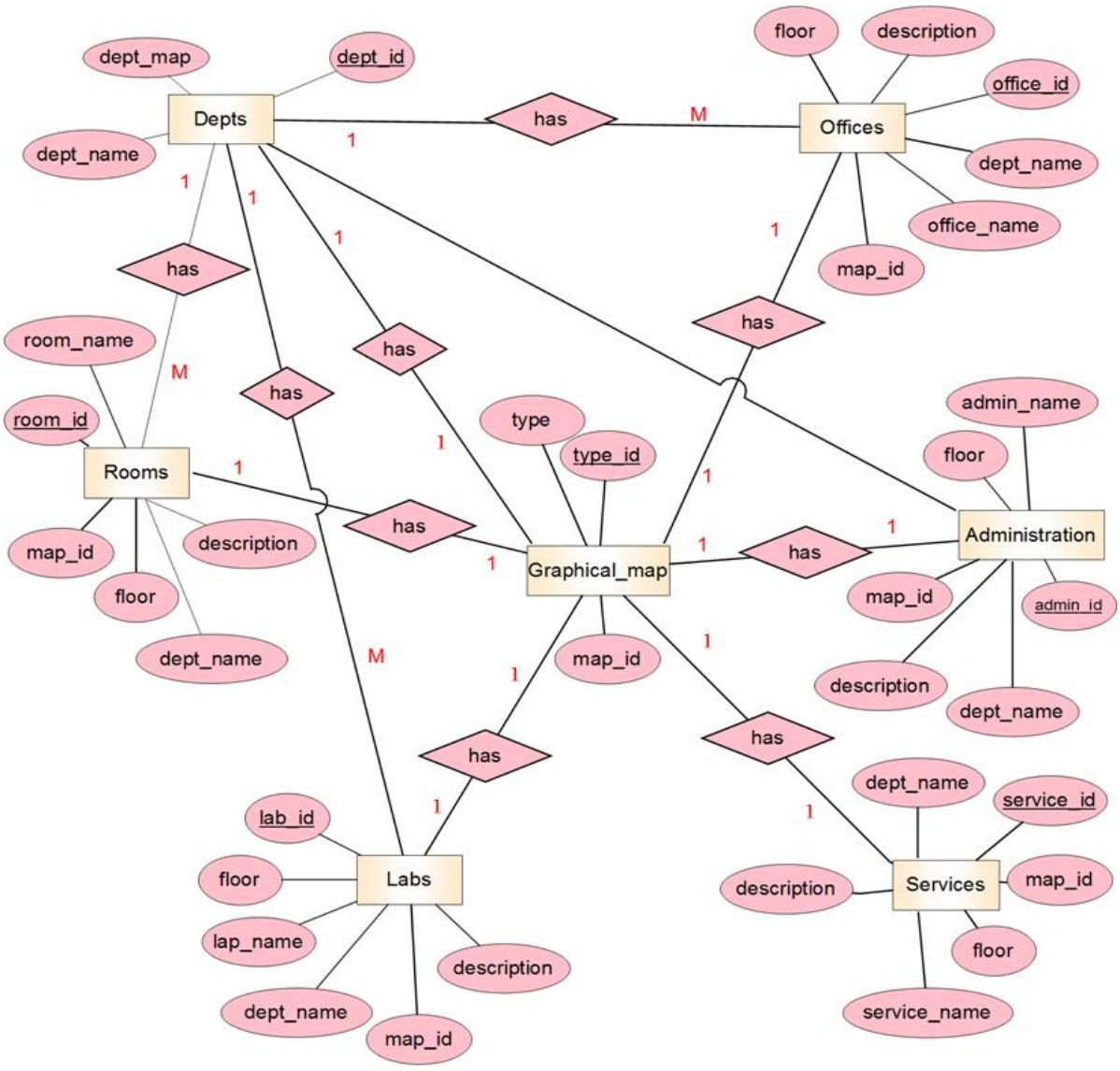



Figure 2. 9 Entity Relationship Diagram

# 3. Chapter 3: System design

## 3.1. Description of procedures and function

This section will introduce the main functions of the proposed system in details. Our system has some functions available for new students and visitors to the faculty.

## 3.2. Relation database schema

In this section we will describe every table and relationship between database tables

### 3.2.1) Tables

**Table 1. Department Table**

| Dep_id | Dep_name | Floor | Dep_map | Description |
|---|---|---|---|---|
| | | | | |

The faculty is consisting of some departments every department has its own rooms, offices and lecture rooms and so on.

In this table Dep_id is the primary key for this table, dep_name is the name of the department and dep_map is the graphical map that enable any visitor to reach this department.

**Table 2. Labs Table**

| lab_id | Dep_name | Map_id | floor | Lab_name | Description |
|---|---|---|---|---|---|
| | | | | | |

There are many laboratory rooms for every department in the faculty and every laboratory has its own graphical map so, there are relations between the lab and the map table and the department table. Lab_id is the primary key and Map_id is the foreign key.

**Table 3. Rooms Table**

| Room_id | Dep_name | Map_id | floor | Room_name | description |
|---|---|---|---|---|---|

Room_id is the primary key and Map_id is the foreign keys.

**Table 4. Offices Table**

| Office_id | Dep_name | Map_id | floor | Office_name | description |
|---|---|---|---|---|---|

Office_id is the primary key and Map_id are the foreign keys.

**Table 5. Administrations Table**

| Administration_id | Dep_name | Map_id | floor | Adminstration_name | description |
|---|---|---|---|---|---|

Adminstration_id is the primary key and Map_id are the foreign keys.

**Table 6. Services Table**

| Service_id | Dep_name | Map_id | floor | Office_name | description |
|---|---|---|---|---|---|

Service_id is the primary key , Map_id is the foreign keys.

**Map Table**

**Table 7. Map Table**

| Map_id | type_name | Type |
|---|---|---|

Map_id is the primary key for this table

Attributes

Every table has its attributes or fields. in this section we will describe some of these attributes. Every place in the faculty has some attributes like its id, name, description about this place, graphical map, Department and floor.

The id is the number of this place, the last id number in the table is the total number of rooms, Name is an attribute for the room name like (Physics lab), description contains some information about the place, floor is the layer in which the room is in like layer 1,2,3 and so on, there are some foreign keys most tables like dep_id, map_id and

floor_id , map_id refers to the graphical map for this place from the map table.

### 3.2.2) Attributes

### 3.2.3) Relations

Form the following figure you can show relations between the tables above.

**Department table**

| Dep_id | Dep_name | floor | Dep_map | description |
|---|---|---|---|---|

**Lab_Room**

| lab_id | Dep_name | Map_id | floor | Lab_name | description |
|---|---|---|---|---|---|

**Rooms Table**

| Room_id | Dep_name | Map_id | floor | Room_name | description |
|---|---|---|---|---|---|

**Offices Table**

| Office_id | Dep_name | Map_id | floor | Office_name | description |
|---|---|---|---|---|---|

**Administrations Table**

| Admin_id | Dep_name | Map_id | floor | Admin_name | description |
|---|---|---|---|---|---|

**Services Table**

| Service_id | Dep_name | Map_id | floor | Office_name | description |
|---|---|---|---|---|---|

**Map Table**

| Map_id | type_name | Type |
|---|---|---|

### 3.3. Hardware and software requirements

In this section we will present the hardware components and software systems we need to use this application. Firstly, the application will be an android application, so it requires a smart phone with android system.

So, the only hardware component is the smart phone and the software is the Android system installed in this mobile.

The software tools that are required in the system Maps of the architectural of Campus :

- PHP language
- My SQL databases
- Android software programs

- HTML Software applications
- CSS
- GPS applications

- Figure 3.2 is a smart phones that uses the android system on it to help us achieve our goal.

- 

- Figure 3. 1 Android Device

-

# Chapter 4: Implementation and Testing

**4.1.** Introduction

**4.2.** Procedures

## Connect to database

```
package com.example.util;

import java.io.Serializable;

public class Constant implements Serializable{

   /**
    *
    */    private static final long serialVersionUID = 1L;

    //server url
    public static final String SERVER_URL="http://192.168.1.3/univGuide/";

    //images url
    public static final String SERVER_IMAGE_UPFOLDER=SERVER_URL+"images/";

   //images gallery url
   public static final String SERVER_IMAGE_GALLERY=SERVER_URL+"images/gallery/";

    //imagesthumb url
    public static final String SERVER_IMAGE_UPFOLDER_THUMB=SERVER_URL+"images/thumb/";

    //category  url
    public static final String CATEGORY_URL =SERVER_URL+"api.php";

    //category list
    public static final String LISTING_URL=SERVER_URL+"api.php?cat_id=";

    //categorylist  details
    public static final String LISTING_DETAILS_URL=SERVER_URL+"api.php?place_id=";

    //about  details
    public static final String RATING_URL=SERVER_URL+"api_rating.php?place_id=";

   //rating  url
   public static final String ABOUT_DETAILS_URL=SERVER_URL+"api.php?settings";

   public static final String FACEBOOK_LOGIN_URL =
SERVER_URL+"user_social_register_api.php?user_type=fb&name=";

   public static final String NORMAL_LOGIN_URL = SERVER_URL+"user_login_api.php?email=";

   public static final String REGISTER_URL =
SERVER_URL+"user_register_api.php?user_type=normal&name=";

   public static final String GMAIL_LOGIN_URL =
SERVER_URL+"user_social_register_api.php?user_type=google&name=";

   public static final String FORGET_PASSWORD_URL = SERVER_URL+"user_forgot_pass_api.php?email=";

   public static final String PROFILE_URL = SERVER_URL+"user_profile_api.php?id=";

   public static final String PROFILE_UPDATE_URL = SERVER_URL+"user_profile_update_api.php?user_id=";

    public static final String CATEGORY_ARRAY_NAME="Place_App";
    public static final String CATEGORY_CID="cid";
    public static final String CATEGORY_NAME="category_name";
    public static final String CATEGORY_IMAGE="category_image";
    public static String CATEGORYID;
    public static String CATEGORYNAME;

    public static final String LISTING_H_ID="p_id";
    public static final String LISTING_H_NAME="place_name";
    public static final String LISTING_H_IMAGE="place_image";
    public static final String LISTING_H_VIDEO="place_video";
    public static final String LISTING_H_DES="place_description";
    public static final String LISTING_H_MAP_LATI="place_map_latitude";
    public static final String LISTING_H_MAP_LONGI="place_map_longitude";
    public static final String LISTING_H_ADDRESS="place_address";
    public static final String LISTING_H_EMAIL="place_email";
    public static final String LISTING_H_PHONE="place_phone";
    public static final String LISTING_H_WEBSITE="place_website";
   public static final String LISTING_H_RATING="place_rate_avg";
    public static String LISTING_H_IDD;

    public static final String ABOUT_NAME="app_name";
    public static final String ABOUT_LOGO="app_logo";
    public static final String ABOUT_EMAIL="app_email";
```

```
    public static final String ABOUT_WEB="app_website";
    public static final String ABOUT_DESC="app_description";

   //rate
   public static final String RATE_MSG="MSG";
   public static String DEVICE_ID;

   public static int GET_SUCCESS_MSG;
   public static final String MSG="msg";
   public static final String SUCCESS="success";
   public static final String USER_NAME="name";
   public static final String USER_ID="user_id";
   public static final String USER_EMAIL="email";
   public static final String IMAGE="image";
 }
```

## Map Activity

```
public class MapActivity extends ActionBarActivity implements OnMapReadyCallback {

   private GoogleMap googleMap;
   int position;
   String alllatitude,alllongitude,allplacename;
   Toolbar toolbar;
   LatLng TutorialsPoint;
   @SuppressLint("NewApi") @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.map_activity);
      toolbar = (Toolbar) this.findViewById(R.id.toolbar);
      toolbar.setTitle(getString(R.string.app_name));
      this.setSupportActionBar(toolbar);
      getSupportActionBar().setDisplayHomeAsUpEnabled(true);
      getSupportActionBar().setDisplayShowHomeEnabled(true);
      if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
         getWindow().addFlags(WindowManager.LayoutParams.FLAG_DRAWS_SYSTEM_BAR_BACKGROUNDS);
         getWindow().setStatusBarColor(getResources().getColor(R.color.m_color_primary_dark));
      }

      Intent i=getIntent();
      position=i.getIntExtra("POSITION", 0);

      alllatitude=i.getStringExtra("LATITUDE");
      alllongitude=i.getStringExtra("LONGITUDE");
      allplacename=i.getStringExtra("PLACENAME");
       TutorialsPoint = new LatLng(Double.parseDouble(alllatitude), Double.parseDouble(alllongitude));
      try {
         if (googleMap == null) {
            //googleMap = ((MapFragment)
getFragmentManager().findFragmentById(R.id.map)).getMapAsync();
            MapFragment mapFragment = (MapFragment) getFragmentManager()
                  .findFragmentById(R.id.map);
            mapFragment.getMapAsync(this);
         }


      } catch (Exception e) {
         e.printStackTrace();
      }


   }

   @Override
   public boolean onOptionsItemSelected(MenuItem menuItem)
   {
      switch (menuItem.getItemId())
      {
      case android.R.id.home:
         onBackPressed();
         break;
      default:
         return super.onOptionsItemSelected(menuItem);
      }
      return true;
   }

   @Override
```

```
   public void onMapReady(GoogleMap googleMap) {
      googleMap.setMapType(GoogleMap.MAP_TYPE_NORMAL);
      googleMap.animateCamera(CameraUpdateFactory.newLatLngZoom(TutorialsPoint, 15.0f));
      googleMap.addMarker(new MarkerOptions().position(TutorialsPoint)
            .title(allplacename));
   }
}
```

# SignUpActivity procedure

```
public class SignUpActivity extends ActionBarActivity implements Validator.ValidationListener{

    @Required(order = 1)
    @TextRule(order = 2, minLength = 3, maxLength = 35, trim = true, message = "Enter Valid Full
Name")
    EditText edtFullName;

    @Required(order = 3)
    @Email(order = 4, message = "Please Check and Enter a valid Email Address")
    EditText edtEmail;

    @Required(order = 5)
    @Password(order = 6, message = "Enter a Valid Password")
    @TextRule(order = 7, minLength = 6, message = "Enter a Password Correctly")
    EditText edtPassword;

    @Required(order = 8)
    @NumberRule(order = 9, message = "Enter Phone Number in Numeric", type =
NumberRule.NumberType.LONG)
    @TextRule(order = 10, message = "Enter valid Phone Number", minLength = 10, maxLength = 14)
    EditText edtMobile;


     private Validator validator;

    Button btnsignup;

    String strFullname,strId,strEmail,strPassword,strMessage,strMacAddress,strMobi,strCoun;

    ProgressBar bar;
    ScrollView scrollview;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        // TODO Auto-generated method stub
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_signup);
        Toolbar toolbar = (Toolbar) this.findViewById(R.id.toolbar);
        toolbar.setTitle("Sign Up");
        this.setSupportActionBar(toolbar);
        getSupportActionBar().setDisplayHomeAsUpEnabled(true);
        getSupportActionBar().setDisplayShowHomeEnabled(true);

        edtFullName=(EditText)findViewById(R.id.edt_user);
        edtEmail=(EditText)findViewById(R.id.edt_email);
        edtPassword=(EditText)findViewById(R.id.edt_password);
        edtMobile=(EditText)findViewById(R.id.edt_mobile);


        btnsignup=(Button)findViewById(R.id.button);
        bar=(ProgressBar)findViewById(R.id.progressBar1);
        scrollview=(ScrollView)findViewById(R.id.scrollView);


        btnsignup.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                // TODO Auto-generated method stub
                validator.validateAsync();
            }
        });

        validator = new Validator(this);
        validator.setValidationListener(this);
    }

    @Override
```

```
    public void onValidationSucceeded() {
        // TODO Auto-generated method stub
        strFullname=edtFullName.getText().toString().replace(" ", "%20");
        strEmail=edtEmail.getText().toString();
        strPassword=edtPassword.getText().toString();
        strMobi=edtMobile.getText().toString();


        if (JsonUtils.isNetworkAvailable(SignUpActivity.this)) {
            new
MyTaskRegister().execute(Constant.REGISTER_URL+strFullname+"&email="+strEmail+"&password="+strPassword
+"&phone="+strMobi+"&country="+strCoun);
         } else {
            setSweetDialog(SweetAlertDialog.ERROR_TYPE, getString(R.string.conne_msg1),
getString(R.string.conne_msg2));
        }

    }

    @Override
    public void onValidationFailed(View failedView, Rule<?> failedRule) {
        // TODO Auto-generated method stub
        String message = failedRule.getFailureMessage();
        if (failedView instanceof EditText) {
            failedView.requestFocus();
            ((EditText) failedView).setError(message);
        } else {
            Toast.makeText(this, "Record Not Saved", Toast.LENGTH_SHORT).show();
        }
    }

    private    class MyTaskRegister extends AsyncTask<String, Void, String> {


        @Override
        protected void onPreExecute() {
            super.onPreExecute();
            bar.setVisibility(View.VISIBLE);
            scrollview.setVisibility(View.INVISIBLE);
        }

        @Override
        protected String doInBackground(String... params) {
            return JsonUtils.getJSONString(params[0]);
        }

        @Override
        protected void onPostExecute(String result) {
            super.onPostExecute(result);

            bar.setVisibility(View.GONE);

            if (null == result || result.length() == 0) {
                setSweetDialog(SweetAlertDialog.ERROR_TYPE, getString(R.string.conne_msg1),
getString(R.string.nodata));
                showToast(getString(R.string.nodata));
            } else {

                try {
                    JSONObject mainJson = new JSONObject(result);
                    JSONArray jsonArray = mainJson.getJSONArray(Constant.CATEGORY_ARRAY_NAME);
                    JSONObject objJson = null;
                    for (int i = 0; i < jsonArray.length(); i++) {
                        objJson = jsonArray.getJSONObject(i);
                        strMessage=objJson.getString(Constant.MSG);
                        Constant.GET_SUCCESS_MSG=objJson.getInt(Constant.SUCCESS);
                    }

                } catch (JSONException e) {
                    e.printStackTrace();
                }
                setResult();
            }

        }
    }

    public void setResult() {

        if(Constant.GET_SUCCESS_MSG==0)
        {
```

```
            setSweetDialog(SweetAlertDialog.ERROR_TYPE, "Opps.", strMessage);
            scrollview.setVisibility(View.VISIBLE);
            edtEmail.setText("");
            edtEmail.requestFocus();
        }
        else
        {
            new SweetAlertDialog(SignUpActivity.this, SweetAlertDialog.SUCCESS_TYPE)
                    .setTitleText("Thanks for Register")
                    .setContentText(strMessage)
                    .setConfirmClickListener(new SweetAlertDialog.OnSweetClickListener() {

                        @Override
                        public void onClick(SweetAlertDialog sweetAlertDialog) {
                            // TODO Auto-generated method stub
                            sweetAlertDialog.dismiss();
                            Intent int1=new Intent(getApplicationContext(),SignInActivity.class);
                            int1.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
                            startActivity(int1);
                        }
                    })
                    .show();
        }
    }

    public void showToast(String msg) {
        Toast.makeText(SignUpActivity.this, msg, Toast.LENGTH_LONG).show();
    }

    public void setSweetDialog(int code,String title,String message)
    {
        new SweetAlertDialog(this,code)
                .setTitleText(title)
                .setContentText(message)
                .show();
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem menuItem)
    {
        switch (menuItem.getItemId())
        {
            case android.R.id.home:
                onBackPressed();
                break;
            default:
                return super.onOptionsItemSelected(menuItem);
        }
        return true;
    }
}
```

## 4.3. Reports

Tbladmin table

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | id | int(11) | | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | username | varchar(100) | utf8_general_ci | | No | None | | | Change Drop More |
| 3 | password | varchar(100) | utf8_general_ci | | No | None | | | Change Drop More |
| 4 | email | varchar(200) | utf8_general_ci | | No | None | | | Change Drop More |
| 5 | image | varchar(255) | utf8_general_ci | | No | None | | | Change Drop More |

## Tbl_category

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | cid | int(11) | | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | category_name | varchar(255) | utf8_general_ci | | No | None | | | Change Drop More |
| 3 | category_image | varchar(255) | utf8_general_ci | | No | None | | | Change Drop More |
| 4 | path | varchar(255) | utf8_general_ci | | No | None | | | Change Drop More |

## Tbl_places

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | p_id | int(11) | | | No | None | | AUTO_INCREMENT | Change Drop |
| 2 | p_cat_id | int(11) | | | No | None | | | Change Drop |
| 3 | place_name | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 4 | place_image | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 5 | place_video | text | utf8_general_ci | | No | None | | | Change Drop |
| 6 | place_description | text | utf8_general_ci | | No | None | | | Change Drop |
| 7 | place_address | text | utf8_general_ci | | No | None | | | Change Drop |
| 8 | place_email | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 9 | place_phone | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 10 | place_website | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 11 | place_map_latitude | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 12 | place_map_longitude | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 13 | place_status | int(1) | | | No | 1 | | | Change Drop |
| 14 | place_rate_avg | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 15 | place_total_rate | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |

## Tbl_rating

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | id | int(11) | | | No | None | | AUTO_INCREMENT | Change Drop |
| 2 | place_id | int(11) | | | No | None | | | Change Drop |
| 3 | ip | varchar(40) | utf8_general_ci | | No | None | | | Change Drop |
| 4 | rate | int(11) | | | No | None | | | Change Drop |
| 5 | dt_rate | timestamp | | | No | CURRENT_TIMESTAMP | | | Change Drop |

## Tbl_setting

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | id | int(11) | | | No | None | | AUTO_INCREMENT | Change Drop |
| 2 | app_name | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 3 | app_logo | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 4 | app_email | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 5 | app_website | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 6 | app_description | text | utf8_general_ci | | No | None | | | Change Drop |
| 7 | api_cat_order_by | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |
| 8 | api_cat_post_order_by | varchar(255) | utf8_general_ci | | No | None | | | Change Drop |

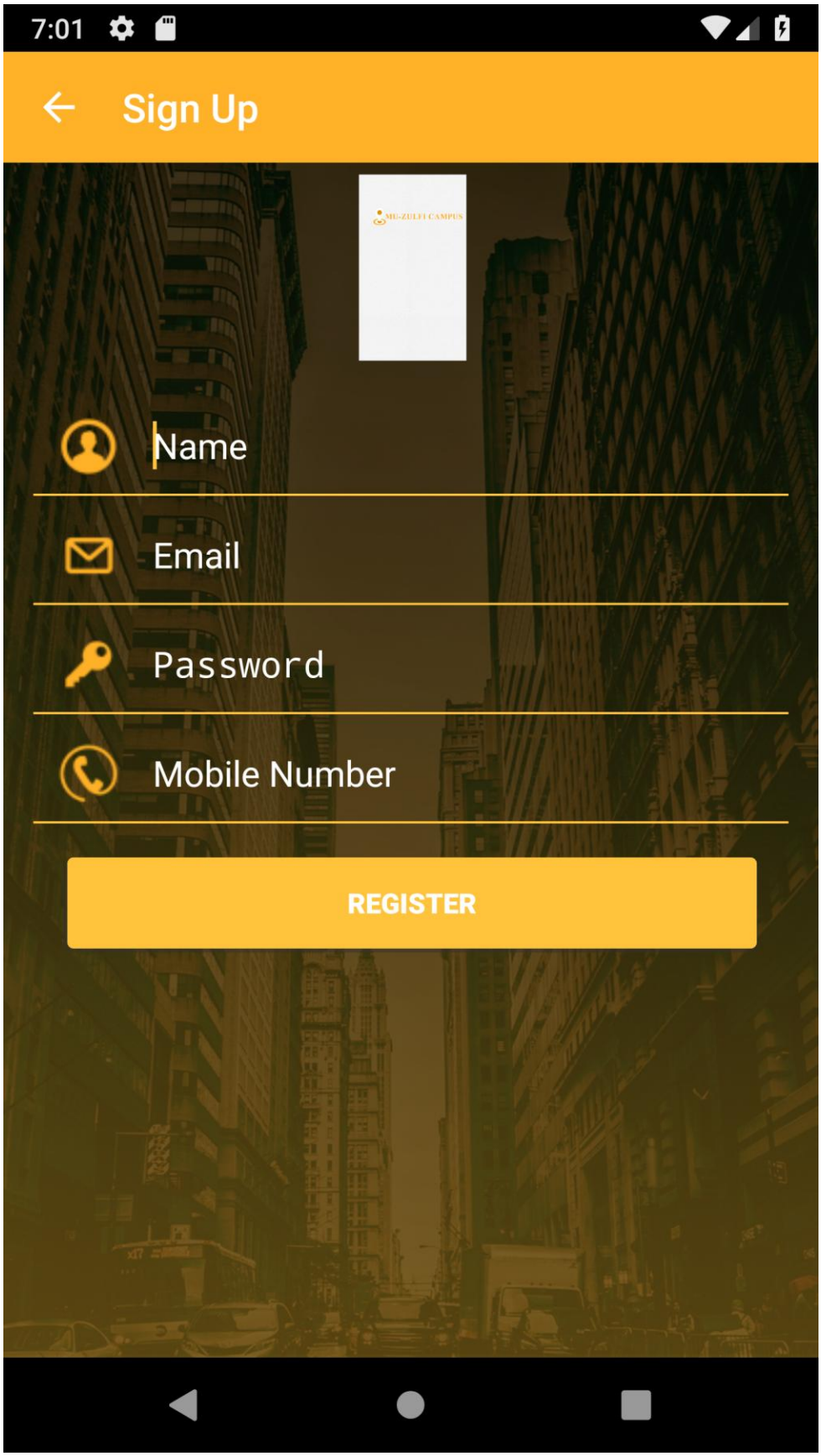
User

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | id | int(11) | | | No | None | | AUTO_INCREMENT | Change Drop More |
| 2 | user_type | varchar(255) | utf8_general_ci | | No | None | | | Change Drop More |
| 3 | name | varchar(255) | utf8_general_ci | | No | None | | | Change Drop More |
| 4 | email | varchar(255) | utf8_general_ci | | No | None | | | Change Drop More |
| 5 | password | varchar(255) | utf8_general_ci | | No | None | | | Change Drop More |
| 6 | phone | varchar(255) | utf8_general_ci | | No | None | | | Change Drop More |
| 7 | confirm_code | varchar(255) | utf8_general_ci | | No | None | | | Change Drop More |
| 8 | status | varchar(255) | utf8_general_ci | | No | 1 | | | Change Drop More |

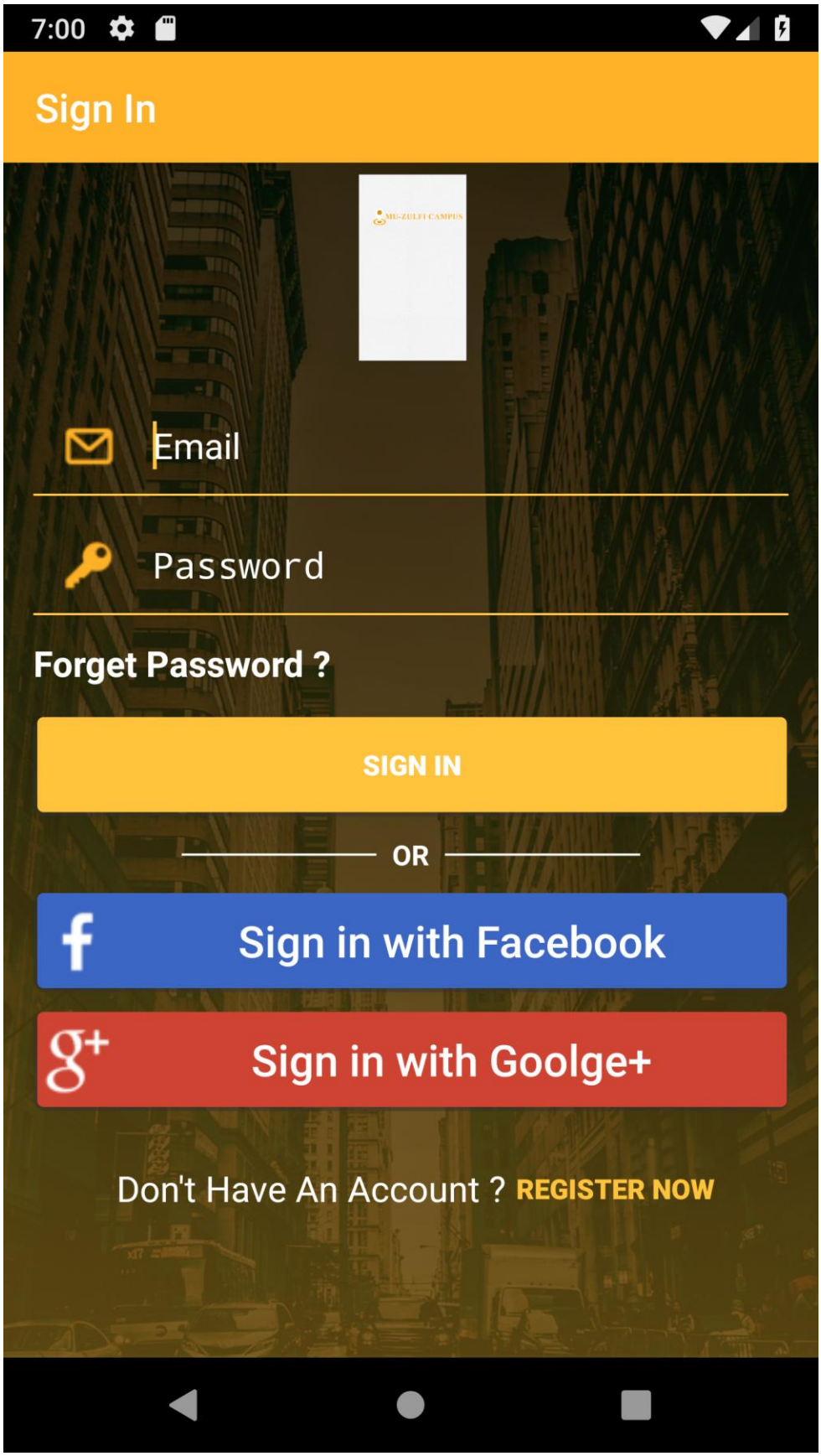
## 4.4. Layouts

Home Interface :


2-Sign Up Interface


7:01
Sign Up
MU-ZULFI CAMPUS
Name
Email
Password
Mobile Number
REGISTER

3- Login Interface


7:00
Sign In
Email
Password
Forget Password ?
SIGN IN
OR
Sign in with Facebook
Sign in with Goolge+
Don't Have An Account ? REGISTER NOW

**4.5.** Reports layouts

# Chapter 6: Conclusion and Future Work

**Conclusion**

In the end of this project, we implemented an android application that can help strangers, visitors and students of the faculty to know their destination like rooms, offices and lecture rooms in the whole faculty. When the person open the application, the app will ask the person for his source place, he should provide the app with the closest room to his place like room name, office number and so on then, the app will ask the visitor for the destination he need to reach. After then, he should click on the find button. After clicking on the find button, the application will show instructions that the visitor should follow to reach his destination. As well as in the same page he can click on the map button to enable the application to provide a graphical map with paths on it to help the visitor to reach his destination.

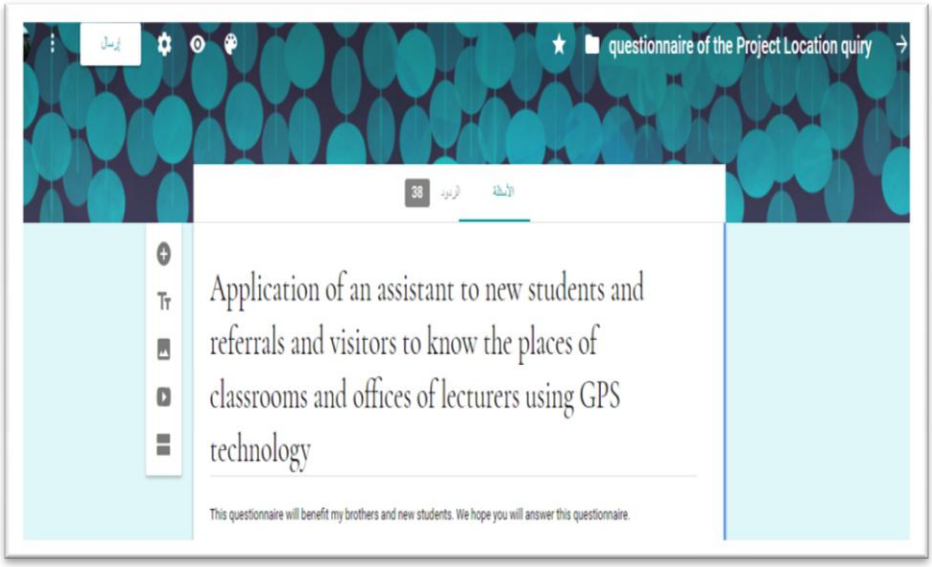
**Future Work**

- We may Apply indoor navigation with Beacons or other facilities may increase interactivity with the application.
- We can use AR or VR map to increase entertainment and interactivity and make the application esy to use.

Also, it is interesting to develop the application using other programming languages to be run on different operating systems such as iOS, Windows phone, ……etc.

# Questionnaire of project

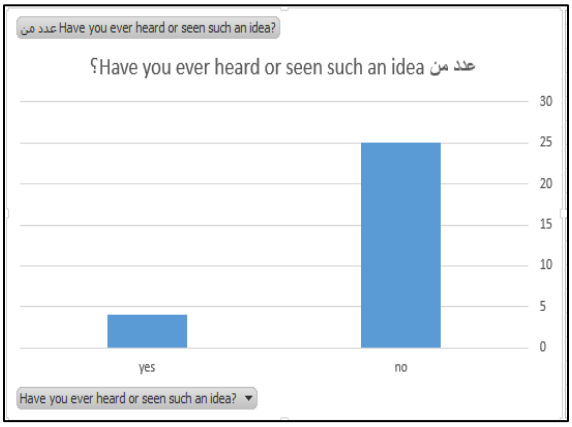
- Idea of project

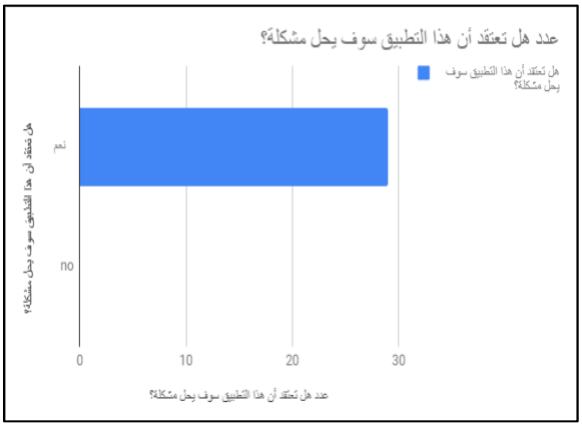



- Five questions:
  1- Have you ever heard or seen such an idea?
  2- Do you think this application will solve a problem?
  3- Do you think the idea is scalable and will be level on the kingdom?
  4- Have you encountered such a problem or anyone you know?
  5- Do you have suggestions to add to the project? mention it.

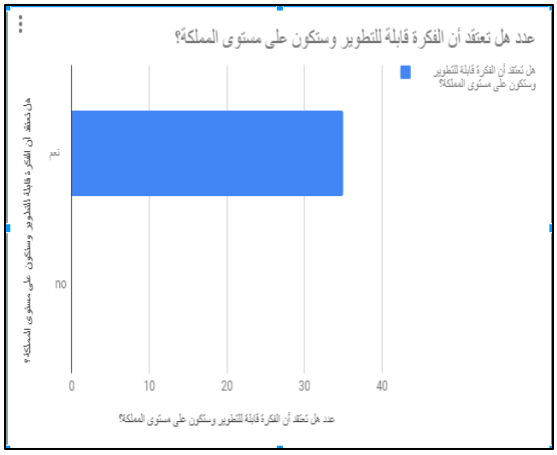
- Analysis

| Have you ever heard or seen such an idea? | Do you think this application will solve a problem? |
|---|---|

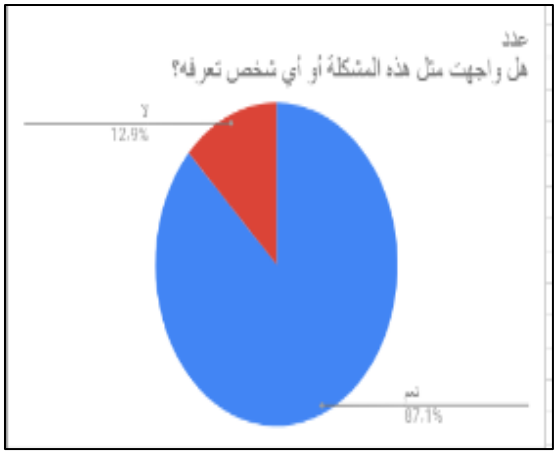

عدد من Have you ever heard or seen such an idea?
عدد من Have you ever heard or seen such an idea؟
30
25
20
15
10
5
0
yes
no
Have you ever heard or seen such an idea?


عدد هل تعتقد أن هذا التطبيق سوف يحل مشكلة؟
هل تعتقد أن هذا التطبيق سوف يحل مشكلة؟
نعم
no
0
10
20
30
عدد هل تعتقد أن هذا التطبيق سوف يحل مشكلة؟

Do you think the idea is scalable and will be level on the kingdom?




Have you encountered such a problem or anyone you know?




Do you have suggestions to add to the project? mention it

Do you have suggestions to add to the project? mention it

12 ردًا

| |
|---|
| فكره جديده وفقهم الله بتوفيقك |
| المشروع داجح جدا وفكره رائعه |
| شكراً الفكره مره مفيده استمر |
| ان يكون متعدد اللغات |
| بالتوفيق |
| اقترح ان تكون الخريطة كالواقع الافتراضي مثل ( google earth) |
| اتمنا منكم الدعم لأخ حمد الدعمي |
| ان يكون هناك برنامج موحد على جميع الجامعات في المملكة . |
| نعم اتمنا اعتماد البرنامج |
| اسال الله لك التوفيق |
| ادخال رقم الشعبة كمفتاح للوصول للموقع |
| ع |

# References


[1] http://m.mu.edu.sa.

[2] http://m.mu.edu.sa/ar/الكليات/بالزلفي-العلوم-كلية.

[3] http://edugate.mu.edu.sa/mu/init.

[4] https://ar.wikipedia.org/wiki/جامعة_المجمعة.

[5] https://www.infsoft.com/solutions/indoor-navigation.

[7] https://www.sqa.org.uk/e-learning/SDM02CD/page_11.htm.

[8] http://ecomputernotes.com/software-engineering/feasibilitystudy.

[9] https://en.wikipedia.org/wiki/Google_Maps.

[10] https://www.lucidchart.com/pages/uml-diagrams.

[11] https://developers.google.com/maps/documentation/android-sdk.

[12] https://www.visual-paradigm.com/guide/uml-unified-modeling/ languag / what-is-uml/

[13] https://www.tutorialspoint.com/android/android_google_maps.htm.


# Appendixes