A PROJECT REPORT

ON

TRAVEL OFFLINE VIRTUAL SUPPORT THROUGH LOCATIONAL REMINDER AND SUGGESTIONS USING DATA ANALYTICS

Submitted to
UNIVERSITY OF MUMBAI

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN
COMPUTER ENGINEERING
BY

Khan Mahir Ali Hamid Ali Parveen Taj	**15DCO50**
Shaikh Arfat Izharul haq Zahida	**15DCO63**
Shaikh Farhan Firoz Ahmed Fatima	**15DCO65**

UNDER THE GUIDANCE OF
PROF. KALPANA BODKE

DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

2017-2018
AFFILIATED TO
UNIVERSITY OF MUMBAI

A PROJECT II REPORT
ON

TRAVEL OFFLINE VIRTUAL SUPPORT THROUGH LOCATIONAL REMINDER AND SUGGESTIONS USING DATA ANALYTICS

Submitted to
UNIVERSITY OF MUMBAI

In Partial Fulfilment of the Requirement for the Award of

BACHELOR'S DEGREE IN COMPUTER ENGINEERING

BY

Khan Mahir Ali Hamid Ali Parveen Taj	**15DCO50**
Shaikh Arfat Izharul haq Zahida	**15DCO63**
Shaikh Farhan Firoz Ahmed Fatima	**15DCO65**

UNDER THE GUIDANCE OF
PROF. KALPANA BODKE

DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

2017-2018
AFFILIATED TO

UNIVERSITY OF MUMBAI

Anjuman-I-Islam's Kalsekar Technical Campus

Department of Computer Engineering

SCHOOL OF ENGINEERING & TECHNOLOGY

Plot No. 2 3, Sector - 16, Near Thana Naka,

Khandagaon, New Panvel - 410206

CERTIFICATE

This is certify that the project entitled

Travel Offline Virtual Support Through Locational Reminder and Suggestions Using Data Analytics

submitted by

Khan Mahir Ali Hamid Ali Parveen Taj	**15DCO50**
Shaikh Arfat Izharul haq Zahida	**15DCO63**
Shaikh Farhan Firoz Ahmed Fatima	**15DCO65**

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Engineering) at *Anjuman-I-Islam's Kalsekar Technical Campus, Navi Mumbai* under the University of MUMBAI. This work is done during year 2017-2018, under our guidance.

Date: /   /

Prof. KALPANA BODKE   **Prof. KALPANA BODKE**
  Project Supervisor   **Project Coordinator**

Prof. TABREZ KHAN   **DR. ABDUL RAZAK HONNUTAGI**
HOD, Computer Department   **Director**

External Examiner

Acknowledgements

We would like to take the opportunity to express our sincere thanks to our guide **KALPANA BODKE**, Assistant Professor, Department of Computer Engineering, AIKTC, School of Engineering, Panvel for his invaluable support and guidance throughout my project research work. Without his kind guidance & support this was not possible.

We are grateful to him/her for his timely feedback which helped me track and schedule the process effectively. His/her time, ideas and encouragement that he gave is help me to complete my project efficiently.

We would like to express deepest appreciation towards **DR. ABDUL RAZAK HONNUTAGI**, Director, AIKTC, Navi Mumbai, **Prof. TABREZ KHAN**, Head of Department of Computer Engineering and **Prof. KALPANA BODKE**, Project Coordinator whose invaluable guidance supported us in completing this project.

At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped me directly or indirectly during this course of work.

Khan Mahir Ali Hamid Ali Parveen Taj

Shaikh Arfat Izharul haq Zahida

Shaikh Farhan Firoz Ahmed Fatima

Project II Approval for Bachelor of Engineering

This project entitled *Ϊravel Offline Virtual Support Through Locational Reminder and Suggestions Using Data Analytics¨* by **Khan Mahir Ali Hamid Ali Parveen Taj (Roll No: 15DC050), Shaikh Arfat Izharul haq Zahida (Roll No: 15DC063), Shaikh Farhan Firoz Ahmed Fatima (Roll No: 15DC065)** is approved for the degree of **Bachelor of Engineering in Department of Computer Engineering.**
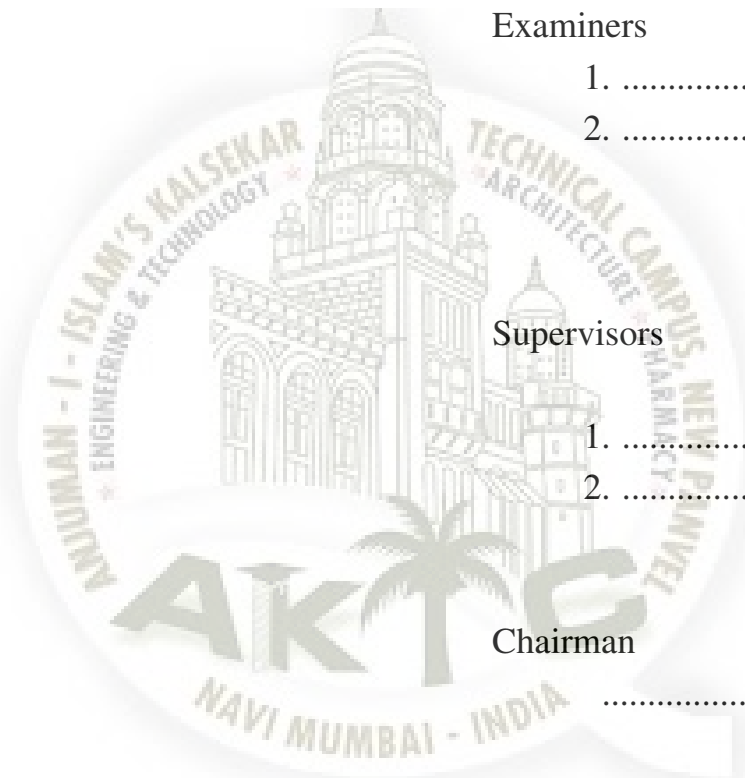
Examiners

1. ....
2. ....

Supervisors

1. ....
2. ....

Chairman

....

Declaration

We declare that this written submission represents our ideas in our own words and where others ideas or words have been included, we have adequately cited and referenced the original sources. we also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.
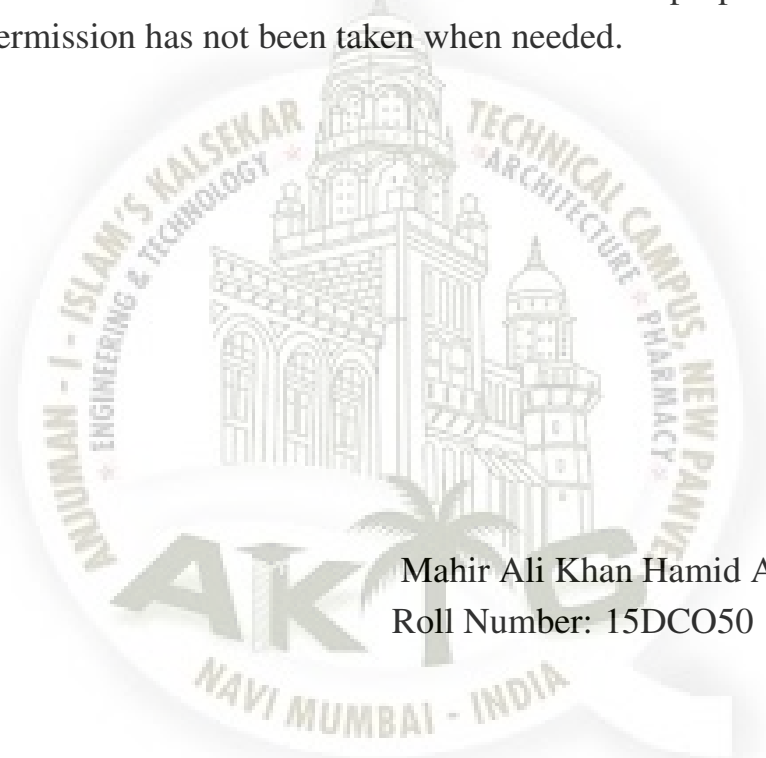
Mahir Ali Khan Hamid Ali Parveen Taj
Roll Number: 15DCO50

Arfat Shaikh Izharul haq Zahida
Roll Number: 15DCO63

Farhan Shaikh Firoz Ahmed Fatima
Roll Number: 15DCO65

ABSTRACT

The combination of smart phone and web service is that the trend of the longer term data development and software applications. As the tourists are provided with printed guide booklet to find out interesting places during the visit. As the tourists are unaware of up to date information and proper navigation, tourists are not able to visit all the destinations. Mobile phones were normally used for communication purpose. Travel offline virtual support through locational reminder and suggestions using data analytic. The web and Android application developed to show how the architecture works and has a lot of important features for tourists such as book- ing, maps and places, events, weather, travel partner and many others discussed in this paper. Nowadays mobile phones are used for various technologies like GPS and browsing over Inter- net. Mobile phones are equipped with various functionality.

We can use this functionalities in our system. The proposed system provides many services to the users like displaying the short- est route between the sources and des- tinations the tourist specify. We are using data and web scrapping to get the data from the websites .we will be using various API available in market for maps and navigation purposes. The application gets the current location of the user through GPS in the form of longitude and latitude and this information is send to the server. Goal of this project is to guide the tourists to travel on their own. The web and An- droid application developed to show how the architecture works
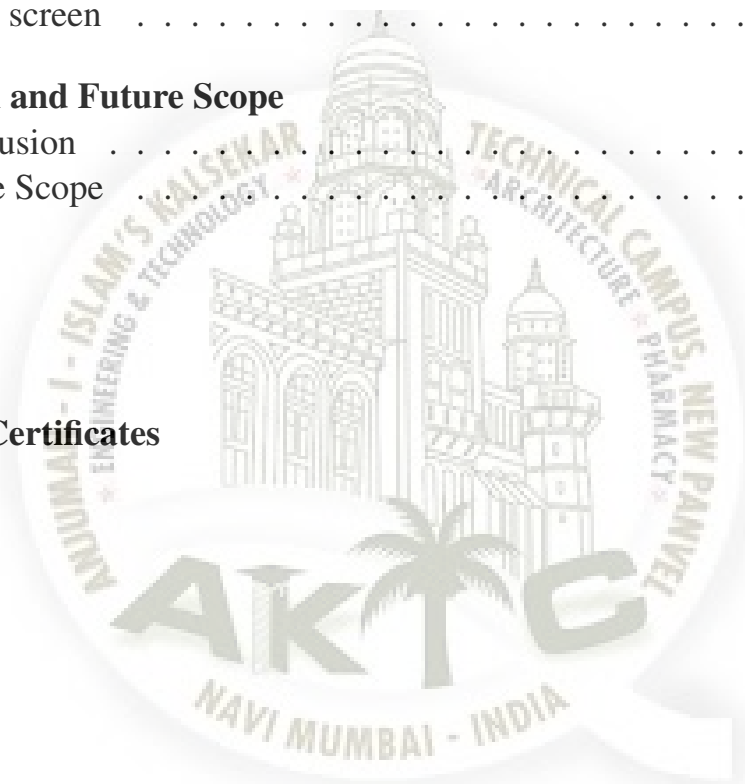
Keywords: Mobile tourist guide, Tourist Information System, Tourist Application, Android, GPS- Global Positioning System.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With the recent advances in Internet and mobile technologies, there are increasing demands for electronic access to tourist information systems for service coordination and process integration. Mobile computing and mobile devices are used to implement various tourist services (e.g. electronic tourist guides, digital interactive maps, and tourist e-commerce transactions). However, due to disparate tourist information and service resources such as airlines, hotels, tour operators; it is still difficult for tourists to use them effectively during their trips or even in the planning stage. Neither can current tourist portals assist tourists proactively. To overcome this problem, we propose the analysis, design, and implementation of the "Mobile tourist guide" system, that access through wireless devices and use Semantic Web technologies for effective organization of information resources and service processes.

1.1 Purpose

Travelling is a part of our daily life.We often travel with our family, friends and many times alone.As in the recent survey conducted by many travel agencies and travel websites the travel rate has increase in very large rate. Mostly friends makes plans to visit new places and explore new places. So the main aim of this project is to make people aware of new places and makes them easy and comfortable to visit and enjoy the trip. The project also focuses on the budget of the trip as students travel so there budget will be mostly low. This system will help to explore new places and with in low and cheap rates.

1.2 Project Scope

Our business idea is to create an mobile application which can do all the tasks for the user that he/she is expecting.our application will store all the information regarding maps, hotel details,near by attractions, dinning options etc Offline so that user can all the information even when internet connection is not available.Before heading towards the trip the users will enter the location he/she wish to go based

on that location all the information will be gathered and stored on user phones.The technical domain about this project is mainly all about data analysis.Data analysis is a process which is required to analyze a given set of data. This process is very much needed today where information is extracted and transformed into a format which is easy to process and interpret. For preparing a prototype of our system, we will first crawl the data.Then, it will match the entered string with the content of the website, display options and then extract data from the links and extract the data from the apis that we would be using throughout our system.

1.3 Project Goals and Objectives

1.3.1 Goals

Now a days people like to travel to a new place they have never been before.The tourists use to carry a bookltes regarding about all the information needed during the journey or note everything down on a piece of paper.Later then technology emerge now people use internet specially google to solve all thier queries regarding trip.But internet will answer only what you ask about.To overcome all this problem our idea comes into picture.

1.3.2 Objectives

Popular perceptions of domestic tourists tend to be quite narrowly defined, persons traveling to leisure resorts or tourist destinations or on religious pilgrimage. However, the internationally accepted definitions are broader, and are the ones adopted the nowadays people now only travel to holy places. The main objective of this project is that it makes people aware of new trekking and hiking places in India. Further the project can be expanded for hotel booking and many more purpose

1.4   Organization of Report

In Chapter 1, we have considered Project overview under which we have explained various important terminologies like introduction of the project, motivation (what exactly motivates us to create travel guide application),goals of this project.

In chapter 2, We have discussed about various papers that we have referred for our project, we have mentioned the description, pros and cons, and how to overcome the problems of each paper. 3 papers have been referred before the development of this project.

In chapter 3, We have done with the project planning in which every members' capabilities and responsibilities have been detailed. Assumptions and constraints have been discussed and project management approach has been given and also the ground rules for the project has been detailed.

In Chapter 4, We have discussed about the requirement analysis, under which we have discussed about platform requirement(supporting OS for the software), Software requirement and hardware requirement along with feasibility study.

In Chapter 5, We can see the system design and its architecture, various diagrams can be seen in this chapter which represent the software, diagrams included are System architecture, class diagram, sequence diagram, component diagram and deployment diagram.

In Chapter 6, We discussed about the implementation details of the system. This part contains details of the implementation of various modules. In short we describe how the system works.

In Chapter 7, We have shown the test cases and results along with analytical discussion. This part contains the results of the output of our project.

In Chapter 8, We have shown various screenshots of the project.

In Chapter 9,We have concluded the whole project and future scope along with the limitations. Followed by references and appendix.

Chapter 2

Literature Survey

2.1   Travel Guide Using GPS

With the recent advances in Internet and mobile technologies, there are increasing demands for electronic access to tourist information systems for service coordination and process integration. Mobile computing and mobile devices are used to implement various tourist services (e.g. electronic tourist guides, digital interactive maps, and tourist e-commerce transactions). However, due to disparate tourist information and service resources such as airlines, hotels, tour operators; it is still difficult for tourists to use them effectively during their trips or even in the planning stage. Neither can current tourist portals assist tourists proactively. To overcome this problem, we propose the analysis, design, and implementation of the "Mobile tourist guide" system, that access through wireless devices and use Semantic Web technologies for effective organization of information resources and service processes.

2.1.1   Advantages of Paper

- Source address need not to be typed manually as it will be taken by gps.

- The application will be made for tablet and mobile phones that will be with the user everytime.

2.1.2   Disadvantages of Paper

- System Does not have Partner Suggesstion by which a user can get to know if there are any people with whom user can travel with.

- If user have a certain amount of budget and user wants to know in that budget which locations he/she can travel.

2.1.3 How to overcome the problems mentioned in Paper

- The system will provide the tourist the type of trip, if the user select the solo trip this feature of our system will as for the user if he/she wants any partner for trekking.

- Our system will also provide emergengy contacts likes women helpline number and hospital contacts based on the nearby locations.

2.2 Mobile Tourist Guide - An Intelligent Wireless System To Improve Tourism ,Using Semantic Web

There are limited technologies used in this sector to address the number of problems faced by the tourists travelling to Fiji. Some of these are bookings for accommodation, retrieving information about different activities in the country, navigation to tourist spots, translation to indigenous Fijian language and access to information on the weather. Tourists also face problems in finding all the tools available in a single application as most of the services are not embedded in a centralized platform. This paper presents a tourism information system that gives a centralized tourist travel guide system for Fiji where a number of services are available. Our proposed system is a centralized system based on web services which provides all necessary information and tools that can be used by tourists to organize their trip. A prototype of the mobile application that demonstrates the proposed system is implemented and tested in order to guide further developments in this area.

2.2.1 Advantages of Paper

- The user can translate the local language into his/her language.

- All the services are centralized in a single system.

2.2.2 Disadvantages of Paper

- Website or Web view of the application is not available.

- Suggested packages are not available.

2.2.3 How to overcome the problems mentioned in Paper

- In our system both the website and application is available which is easy to use.

- In our system we will suggest the packages for the particular locations

2.3 Mobile Travel Guide By Using Android

The combination of the good phone and also the web service is that the trend of the longer term data development and software applications. Mobile phones are the foremost normally used communication tools. Victimization mobile phones to get information isn't solely fast, however conjointly a lot of convenient crosscut to improve people's lives. Within the paper, we tend to propose the code development design supported net services. This framework introduces the three-layer design of net development into mobile code development. Based on the three-layer design, the humanoid primarily based town guide system is developed. The humanoid primarily based town guide system will realize to question data for edifice, scenery, restaurant, traffic and so on. The humanoid primarily based town guide system has a lot of practical significance.

2.3.1 Advantages of Paper

- The system is an application for mobile phones.

- The design of the system is on three layer design such that the services request can be handle much faster than sole system.

2.3.2 Disadvantages of Paper

- The System Does not have a module through which user can select the Type of trip

- There is no emergency contacts in this project which is a very essential feature for a travel guiding application.

2.3.3 How to Overcome

- The system will provide the tourist the type of trip, if the user select the solo trip this feature of our system will as for the user if he/she wants any partner for trekking.

- Our system will also provide emergengy contacts likes women helpline number and hospital contacts based on the nearby locations.

2.4 Technical Review

Our system will be an andriod application.The technologies that we are using in our project are as follows:-1.java, 2.REST API, 3.FirebaseDb

---

2.4.1 Advantages of Technology

- Our system is based on andriod,most of the people uses andriod phones.Thus our system can target large amount of users.

- we will be using firebaseDb,for login purposes and storing the user credentials.The usage of the firebaseDb is very advantageos as the data is stored in cloud instead of users phones.

- Due to its scalability. This protocol stands out due to its scalability. Thanks to the separation between client and server, the product may be scaled by a development team without much difficulty. Due to its flexibility and portability in REST API.

2.4.2 Reasons to use this Technology

- Android creates an App in Efficient way which is easy to use.

- Firebase provides dynamic database for analysis,authentication and storage.

- REST API is used to get the data from different server in simple format.
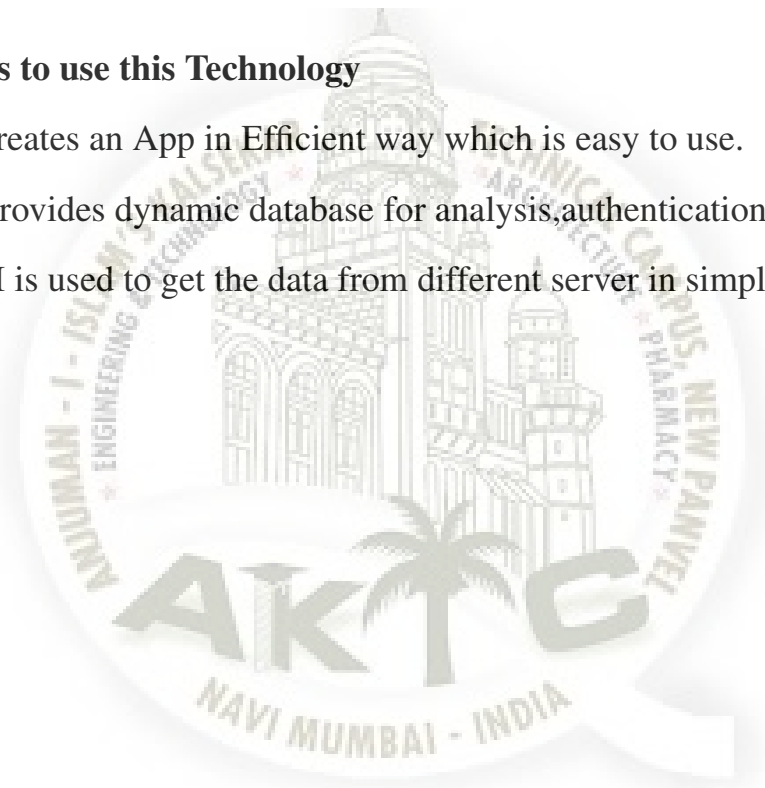
Chapter 3

Project Planning

3.1   Members and Capabilities

Table 3.1: Table of Capabilities

SR. No	Name of Member	Capabilities
1	Shaikh Farhan	Intergrating,Fetching data
2	Shaikh Arfat	UI design
3	Khan Mahirali	Database

Work Breakdown Structure

3.2   Roles and Responsibilities

Table 3.2: Table of Responsibilities

SR. No	Name of Member	Role	Responsibilities
1	Shaikh Farhan	Team Leader	Intergrating,Fetching data
2	Shaikh Arfat	UI deigner	UI design
3	Khan Mahirali	Backend	Database

3.3   Assumptions and Constraints

1. Assumptions
   The team member should know the andriod coding.
   To deveople a system better available in the market.
   No significant changes in technology to change our system.

2. Constraints
   The project should be completed before the deadline.
   The module which is to be added should be known in advance.
   The customer should be able to understand how the system works

3.4   Project Management Approach

We have use Agile methadology for the development of this project.The Agile Project Management Process is a value-centered methods of project management that allows projects to get processed in small phases or cycles. The methodology is one that is extremely flexible and projects that exhibit dynamic traits would benefit from this process as you would find that project managers working in this environment treat milestones the goal being to continuously adapt to abrupt changes from our project guide feedback.

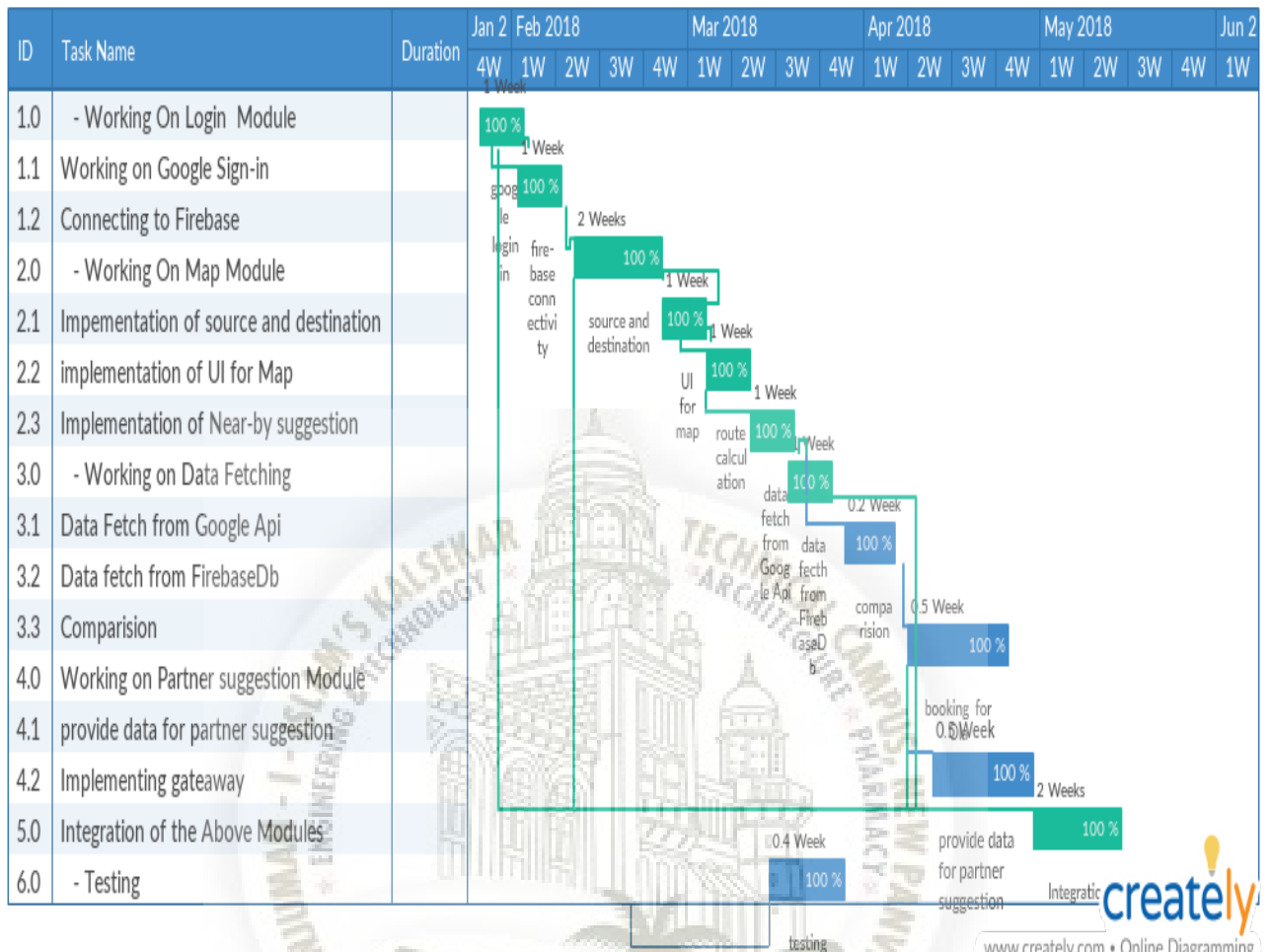## 3.5   Ground Rules for the Project

1. We treat each other with respect.

2. We intend to develop personal relationships to enhance trust and open communication.

3. We value constructive feedback. We will avoid being defensive and give feedback in a constructive manner.

4. As team members, we will pitch in to help where necessary to help solve problems and catch-up on behind schedule work.

5. Additional meetings can be scheduled to discuss critical issues or tabled items upon discussion and agreement with the team leader.

6. One person talks at a time; there are no side discussions

7. When we pose an issue or a problem, we will also try to present a solution.

3.6   Project Budget

The budget for this project is very low as most of the tools we have use are open source.Following are the budget for the project

1. Operating System:linux mint (Open Source).

2. IDE:Andriod Studio (Open Source).

3. API:Google places API,Vision Api,Open Weather Api (Open Source)

3.7   Project Timeline

ID	Task Name	Duration
1.0	- Working On Login Module	
1.1	Working on Google Sign-in	
1.2	Connecting to Firebase	
2.0	- Working On Map Module	
2.1	Impementation of source and destination	
2.2	implementation of UI for Map	
2.3	Implementation of Near-by suggestion	
3.0	- Working on Data Fetching	
3.1	Data Fetch from Google Api	
3.2	Data fetch from FirebaseDb	
3.3	Comparision	
4.0	Working on Partner suggestion Module	
4.1	provide data for partner suggestion	
4.2	Implementing gateaway	
5.0	Integration of the Above Modules	
6.0	- Testing	

Chapter 4

Software Requirements Specification

4.1   Overall Description

4.1.1   Product Perspective

Travel offline virtual support through locational reminder and suggestions using data analytics. . The Android application design to show how the system works and has a lot of functionality for tourists and travellers such as booking, maps , events, weather, travel partner etc. The main attraction of our system would be partner suggestions and trip type.what it really signifies that the person who is travelling would have the option to search for the partner to overcome loneliness throughout the trip.but sometimes its not all about to have a partner just not to feel lonely.Sometimes we would like to explore places like hilly regions which can be turned out to be dangerous while travelling alone .

4.1.2   Product Features

- The location can be taken real time using gps.

- Based on location the restaurants,hospitals etc will shown to the user.

- Social login is incorporated with the app so that user can login through the google account.

- iterations and weekend gateway is added.

- Partner suggestion is a feature through which the user can get a company while travelling to that destination

4.1.3   User Classes and Characteristics

1.Solo Solo will be the kind of user who are travelling alone.
2.couple Couple will be the users who will travel in couple (man and woman).
3.Group This are the types of user who will travel in a group(more than two).

4.1.4   Operating Environment

1.Software
Andriod OS above ice cream sandwich.
2.Hardware
Andriod enabled device with atleast 512mb ram and 200mb of storage.

4.1.5   Design and Implementation Constraints

The major challenge that will hurdle the development of the system is the un-correct gps location,if the location of the user is not fetched correctly than whole system is of no use.Another constraint would be the internet connectivity,if there is no internet connection available than all the services will not be provided to the user.

4.2   System Features

The major features of our system is to provide near by places based on the GPS location,the user can view the hotels,shopping marts,hospitals and ATM around his/her.Another main feature of our project will be the patner suggestion,this feature will allow the user to search for the patner while going on a trip.

4.2.1   System Feature

1. Patner Suggestion.

2. Near-by places.

3. Attractions.

4. Weekends Gateaway.

5. itineraries.

6. Emergency contacts.

Description and Priority

1. Patner suggestion
   This feature of the system will allow the user to search for the patner while going for a trip.
   benefit-8
   risk-8

2. Near-by places
   This feature of the system will show all the near by places around the user.
   benefit-9
   risk-1

Stimulus/Response Sequences

1. The user need to be login in a system.

2. The user need to select the Patner suggestion.

3. The user will enter the location name.

4. The patner will be shown to the user as per the location.

Functional Requirements

1. The user should be able to login in the system.

2. The system should get the correct location of the user.

3. The system will store the correct details of the user.

4. On the basis of the location given by the user,the patner should be suggested.

4.3 External Interface Requirements

4.3.1 User Interfaces

1. User shall be able to login in the system.

2. After the login session shall be maintain.

3. The user can search for destination.

4. The user can view the partner suggested by the system.

5. The user can call to the emergency contacts.

4.3.2 Hardware Interfaces

Andriod enabled device:The andriod enabled device should have andriod version above 5.0

Inorder for the smooth functioning of the application the andriod device must have atleast 512mb of ram and atleast 200mb of free storage on device.

The application can also function on a tablet device.

4.3.3 Software Interfaces

Operating System:Andriod above 5.0.

Databaes:google Firebase.

tools:andriod studio IDE

Apis:google maps,google direction

4.3.4 Communications Interfaces

1.The major communication for location purposes will be done by google api,the data is accessed by the google by using the google apis.

2.The interface between the firebaseDB and the system will be done by using http protocol

4.4 Nonfunctional Requirements

4.4.1 Performance Requirements

Partner suggestion:The performance for this features is mainly depend upon the data in the database,as per the data the suggestion will be given to the user.

Near-by places:The performace of this features is depend on the overall locational access,if the gps is unable to get the correct location than whole system fails

4.4.2 Safety Requirements

If there is any damage to the large amount of the data in the database than the whole system will go down.

The database should be periodically maintained and have to keep upon it.

The data which is updated by the user should be commited in the database.

4.4.3 Security Requirements

The major security requirements thrents for the system will be the safeguarding of the user data from any kind of exploit. Inorder to protect the user data the data is not stored in local databases we will be storing in the cloud for better security.

Chapter 5

System Design

5.1   System Requirements Definition

Our system is an andriod application on a mobile device,the system will function overall on the basis of the location which is captured by the gps location.we have survey various application related to our project.we have decided the system specification for our project.we have studied the end-user requirements and based on that we have decided the functional and non-functional requirement.

5.1.1   Functional requirements

1.The customers must register for create the account and login using username and password to use function in the application.
2.collect data of customers personal information (name, address, tel number, old, behaviour)
3.The customers can view or find the information of hotel, tourist attraction.
4.The customers can search for booking hotel, flight after select date time to go.

Usecase Diagram at its simplest is a representation of a user's interaction with the system that shows the relationship betweeen the user and the different use cases in which the user is involved.
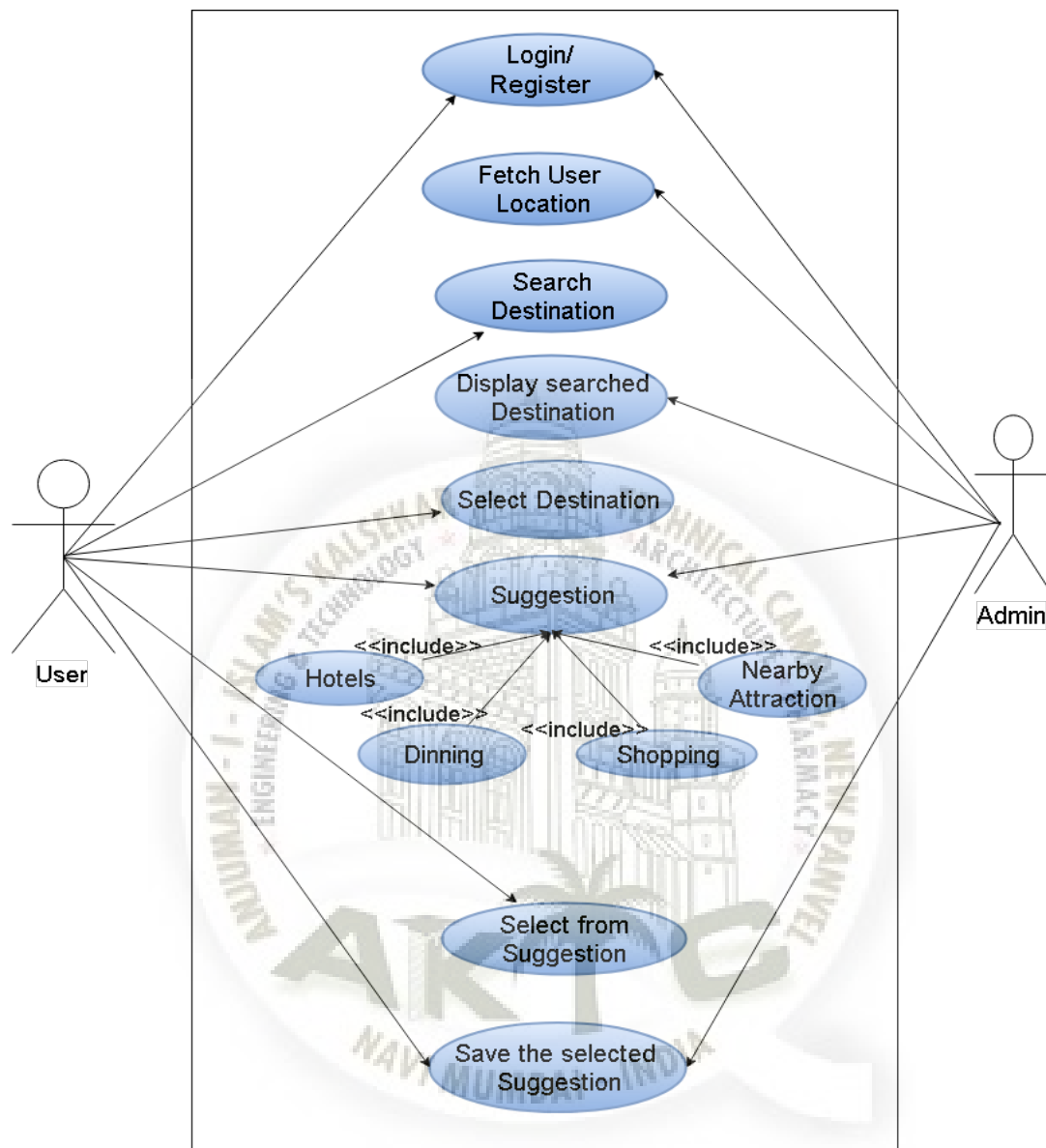
Use Case Diagram

Figure 5.1: Usecase Diagram

Data-flow Diagram

   This is the Data ow diagram level 0 of the system in which the modules which
will be there after the deployment are shown .Server of the rebase database is shown
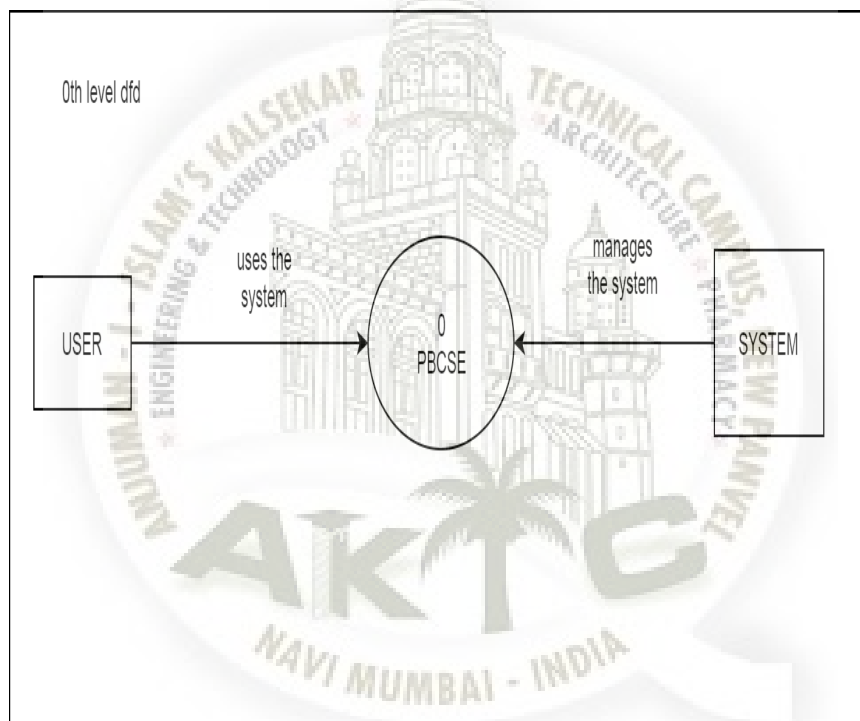for easy understanding of the project.

Figure 5.2: DFD Level 0

This is the Data ow diagram level 1 of the system in which the modules which will be there after the deployment are shown . the server of the respected to the location it will suggest the partner as well as the server of the rebase database is shown for easy understanding of the project.
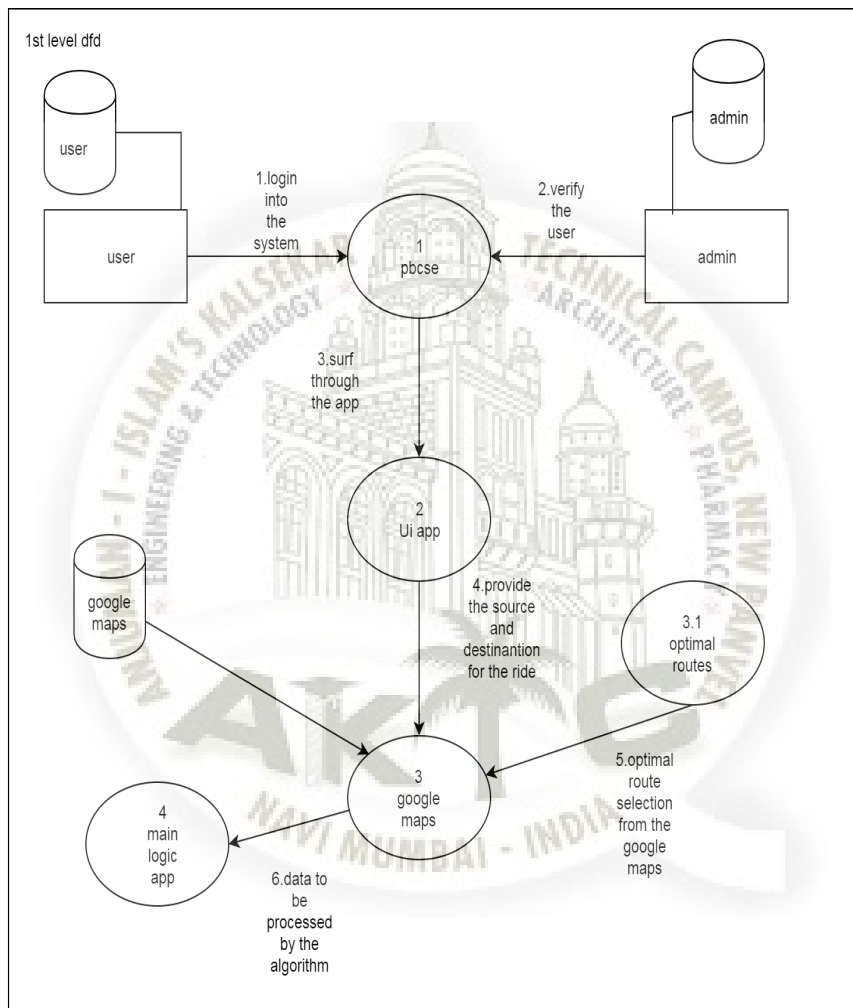

Figure 5.3: DFD Level 1

5.1.2 System requirements (non-functional requirements)

Usability requirement-A user interface for updating the information for travel that would allow the system to better adapt the overall system.

Efficiency requirement-The application should be able to response quickly the user's request.

Performance requirement-The application should be able to response the queries submitted by the user without delay.

Reliabilty requirement-The application should work under all conditions and performed the required functionality.
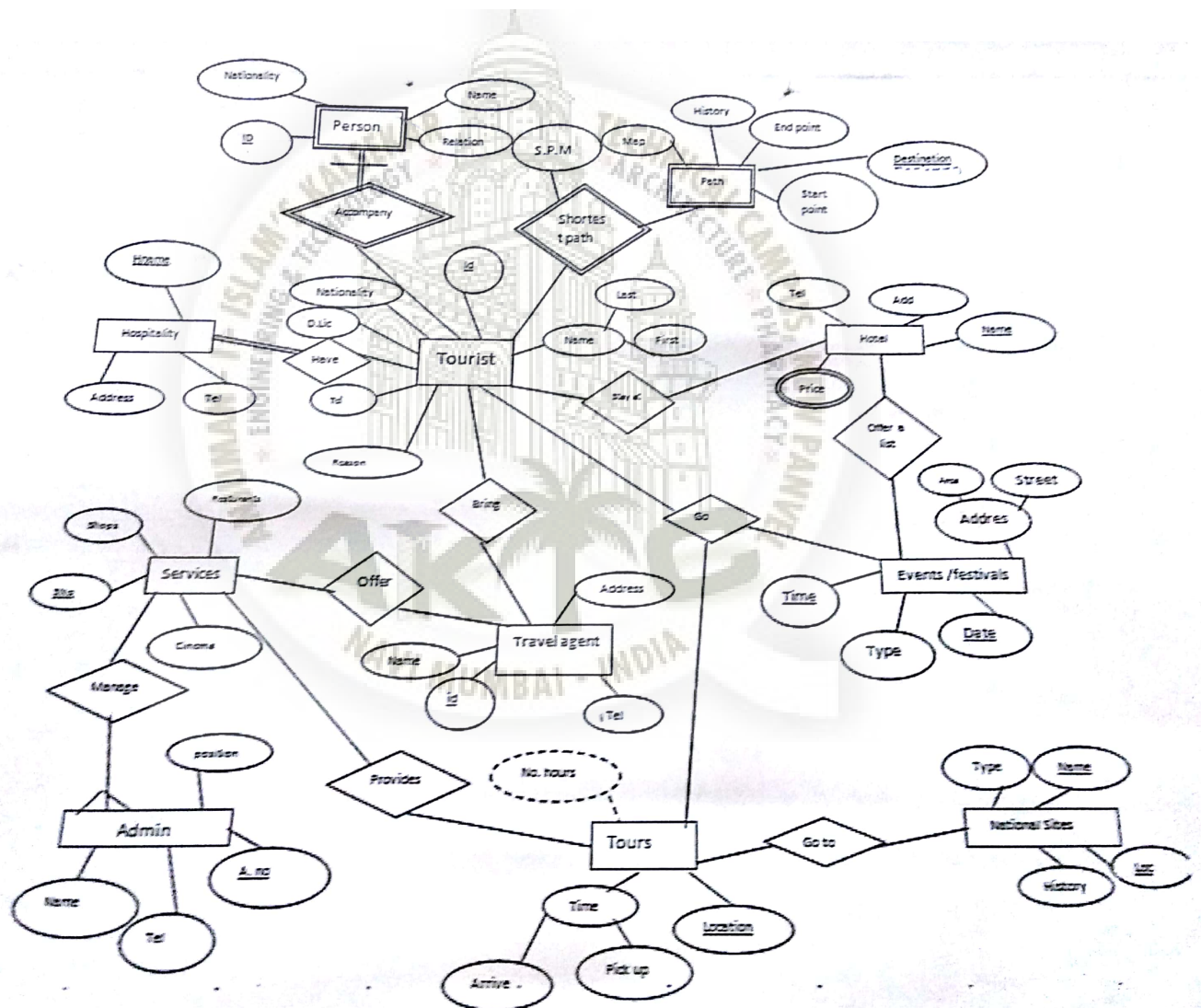
Database Schema/ E-R Diagram

Figure 5.4: E-R Diagram

This is the ER diagram of the system in which the modules which will be there after the deployment are shown

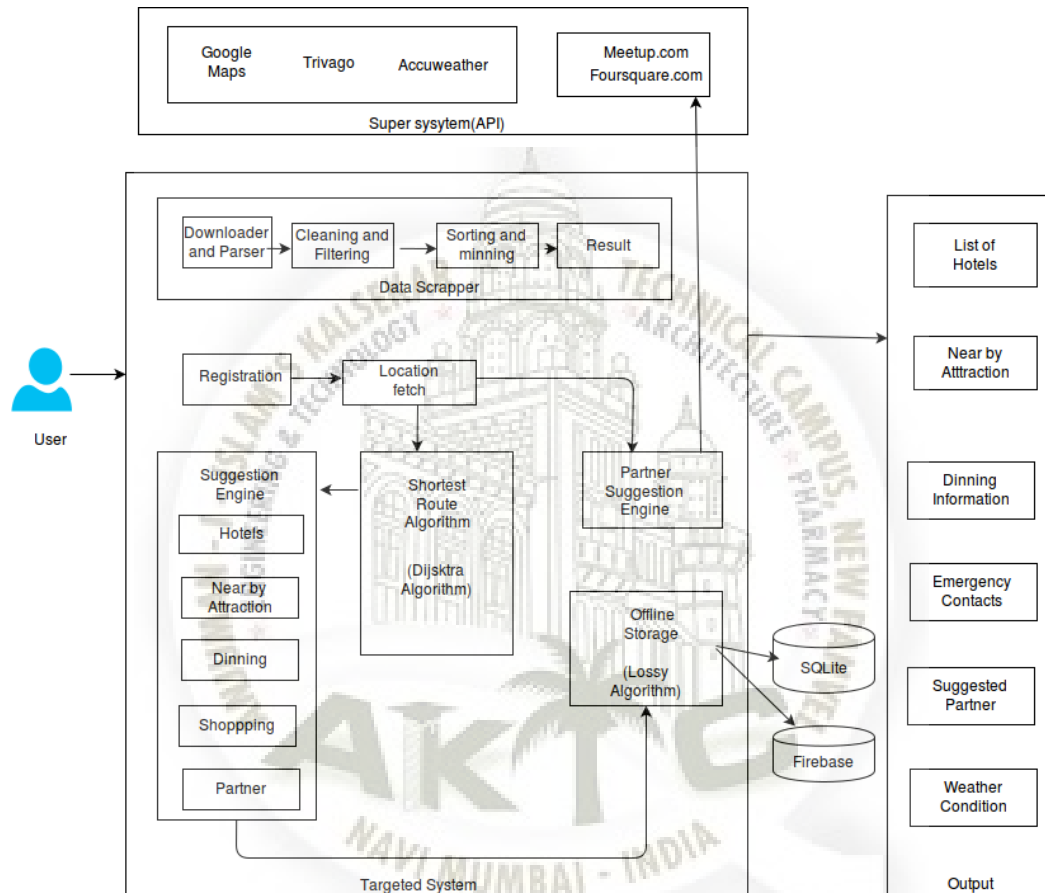## 5.2   System Architecture Design

Figure 5.5: System Architecture

In the below System Architecture,we basically divided the diagram into four parts.First is the super system ,it contain all the apis and websites that we are going to use in our project.Second is the targeted system,it contains all the functions and logics of the system.The web and data scrapping is done here,the data require by the user is extracted from the web here.The flow of process is shown below.As the user register in the system the location of the user is fetched.From that fetched location the shortest route to the destination is calculated.According to the location of the destination specific suggestions given to the user,the user can select any suggestions the data related to the suggestion can be save offline.The data can be save in databases.Later this data can be outputted to the user without the internet connection.

5.3   Sub-system Development

Login:-In login module the authentication part is covered and it provides a safe login into system via google id.

Maps :- In this module the user can get the markets, ATMs, Hospital and Resturants according to the location where the user is current present through the GPS location

Partner: Through this module the user can get the invites or he/she can invite the people who all are going for the same location.This module is mainly design to arrange the trekking partners.

Emergency: Emergency contacts for all kind of emergencies like Police,Ambulance, Fire Brigade,Railway , Women safety, Traffic ,Anti-corruption, Airport Police, Disaster Helpline, Blood bank.

Attractions: This cover up the places you must visit which our system is suggesting which includes Beaches, Historical places,Adventures and parks.

5.3.1   Login

In login module the authentication part is covered and it provides a safe login into system via google id.ocial login is a sign in technology through which the users is authenticated on various applications and sites by connecting through a social site rather than entering again the ID and password on each website. Many people are now using social authenti- cation which are publicly exposed authenticated sites or social media such as Google and Facebook.
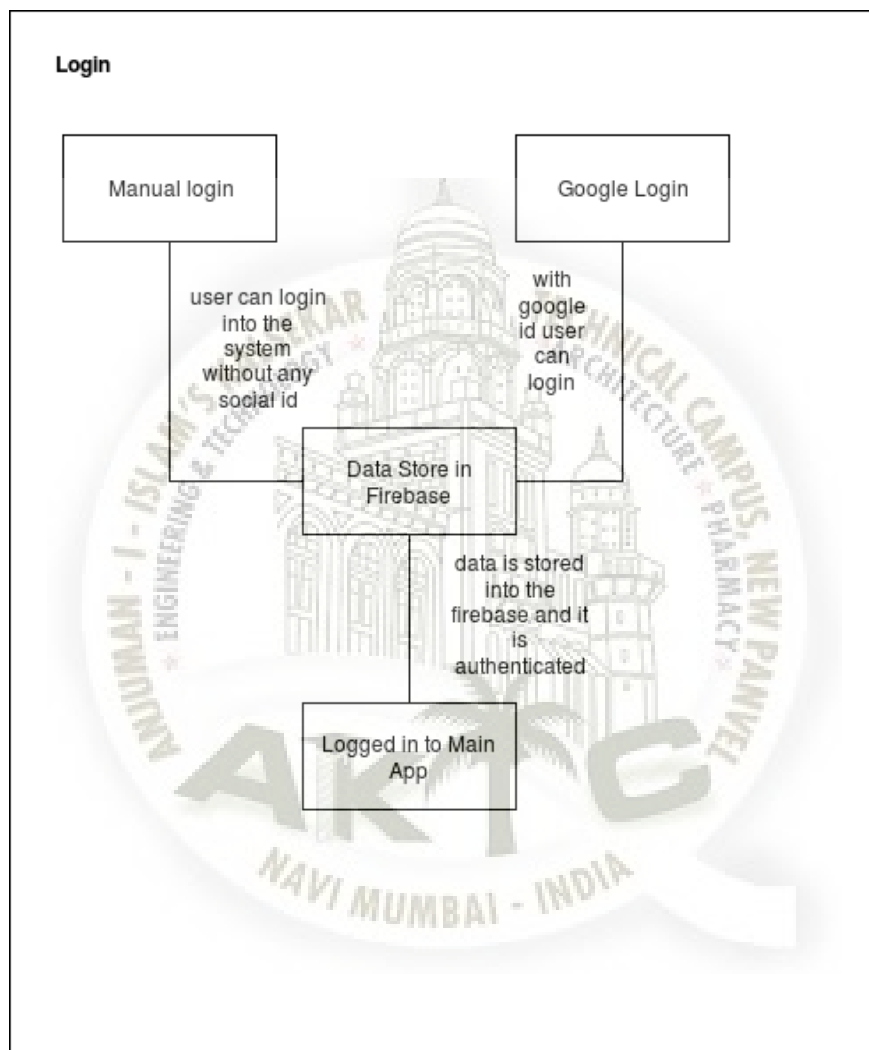

Figure 5.6: Login and Signup

5.3.2 Maps

In this module the user can get the markets, ATMs, Hospital and Resturants according to the location where the user is current present through the GPS location.his feature of the proposed system helps the tourist to find the nearby tourist spots like shopping, adventures activities, hiking location, resorts and amusement park and many more.
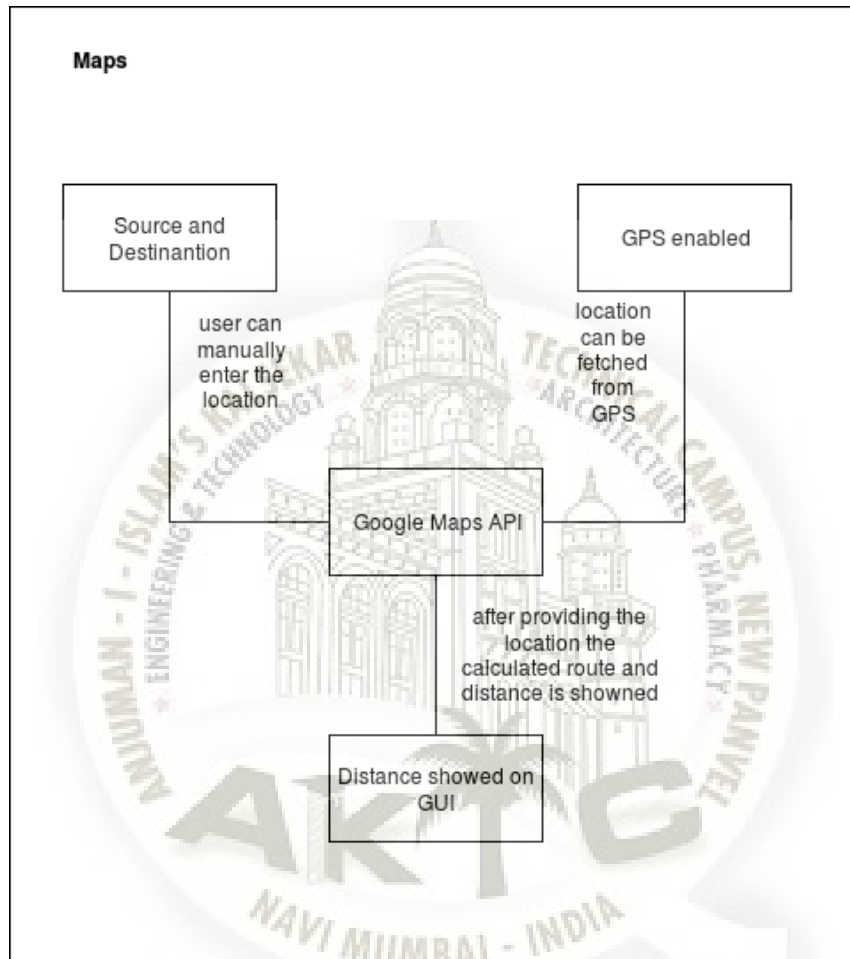

Figure 5.7: Nearby Location

5.3.3   Find Partner

As the tourist selects the type of trip, if the user select the solo trip this feature of our system will as for the user if he/she wants any partner for trekking.
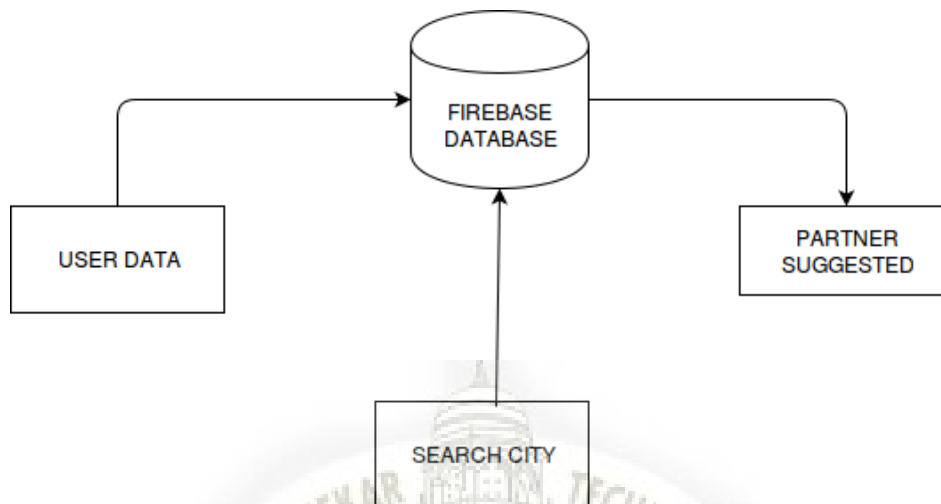

Figure 5.8: Find Partner

5.4   Systems Integration

In the below System Architecture,we basically divided the diagram into four parts.First is the super system ,it contain all the apis and websites that we are going to use in our project.Second is the targeted system,it contains all the functions and logics of the system.The web and data scrapping is done here,the data require by the user is extracted from the web here.The flow of process is shown below.As the user register in the system the location of the user is fetched.From that fetched location the shortest route to the destination is calculated.According to the location of the destination specific suggestions given to the user,the user can select any suggestions the data related to the suggestion can be save offline.The data can be save in databases.Later this data can be displayed to the user without the internet connection

5.4.1 Class Diagram

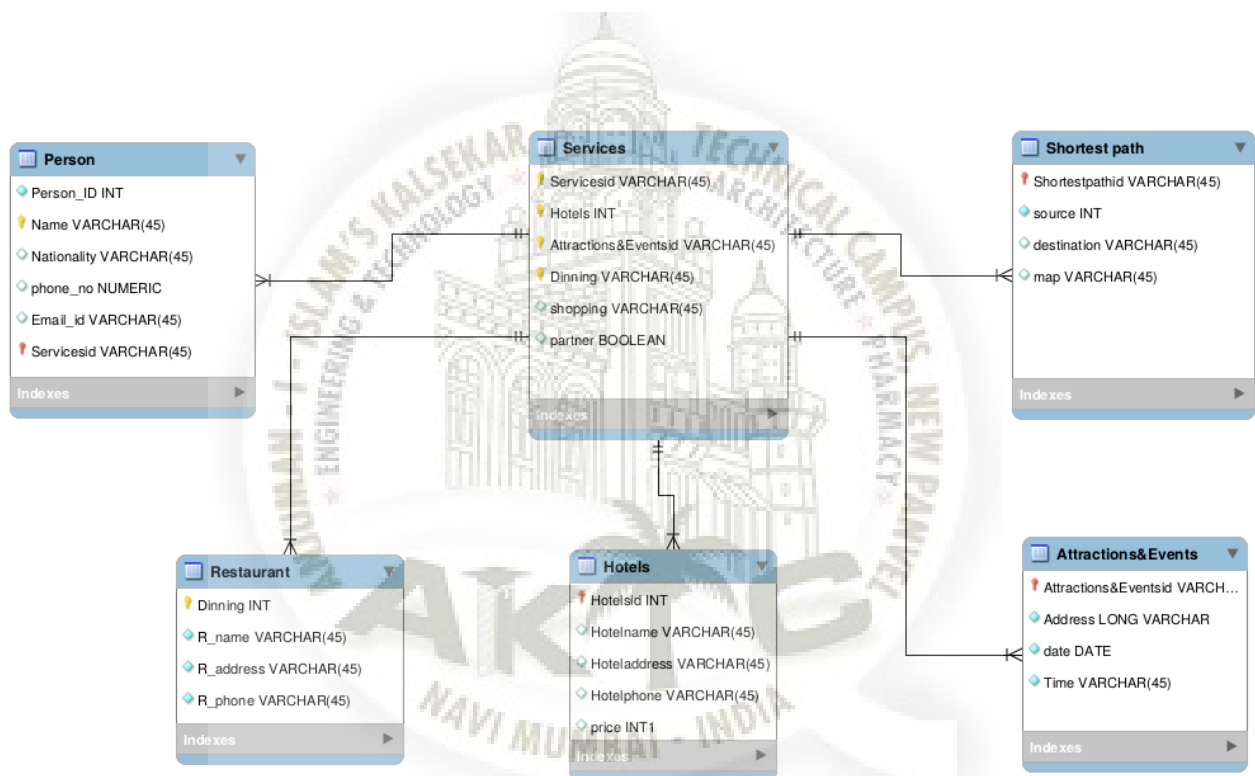As Class is a collection of entity called as objects, so class diagram describes the various modules of our System:

Figure 5.9: Class Diagram

5.4.2 Sequence Diagram

This is the Sequence Diagram for our system which shows the sequential flow of our system when particular user searches or visit a new location this are explained in the below figure:
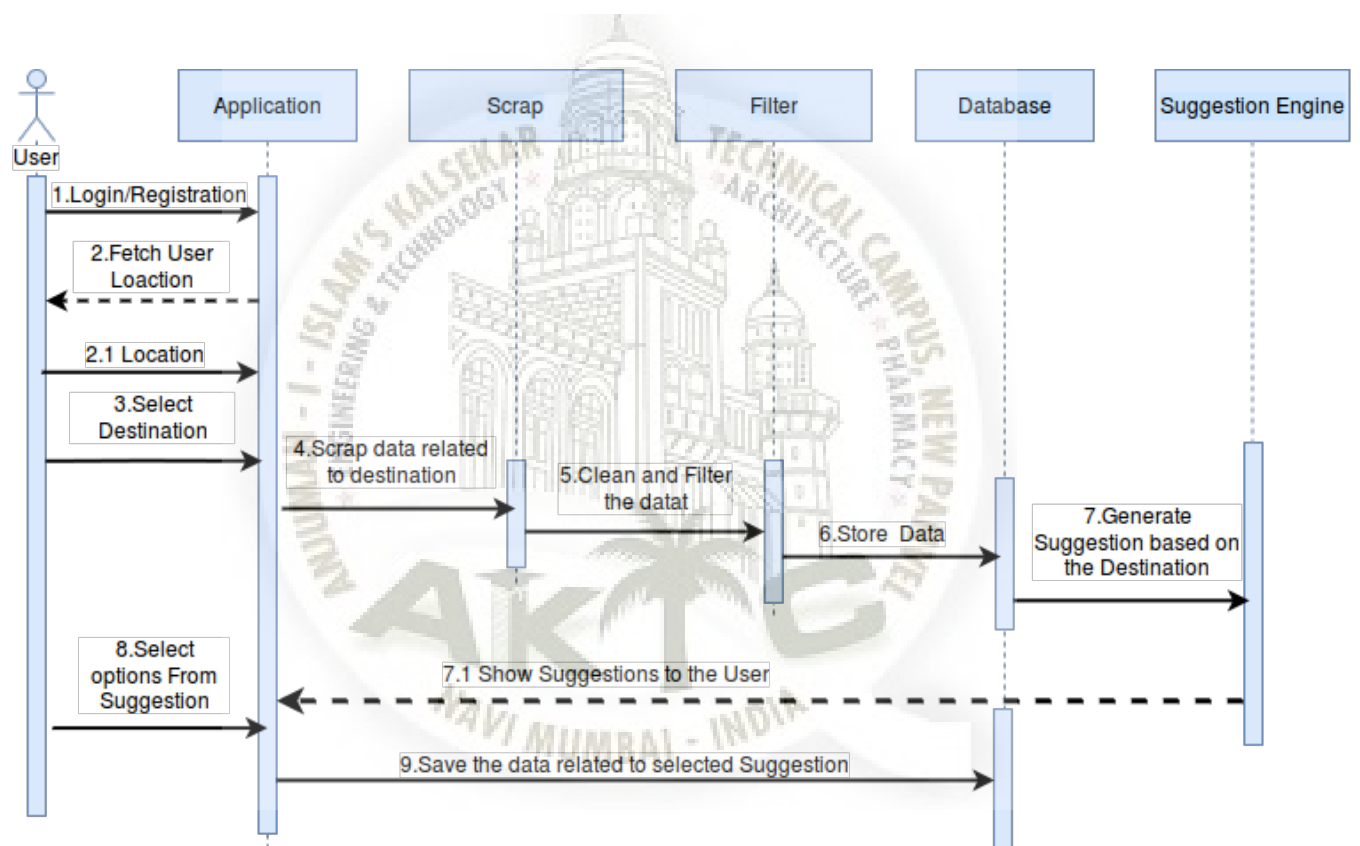

Figure 5.10: Sequence Diagram

5.4.3   Component Diagram

Component diagram is a special kind of diagram in UML. The purpose is also different from all other diagrams discussed so far. It does not describe the functionality of the system but it describes the components used to make those functionalities..This diagram of our system shows the components which are included in our system :
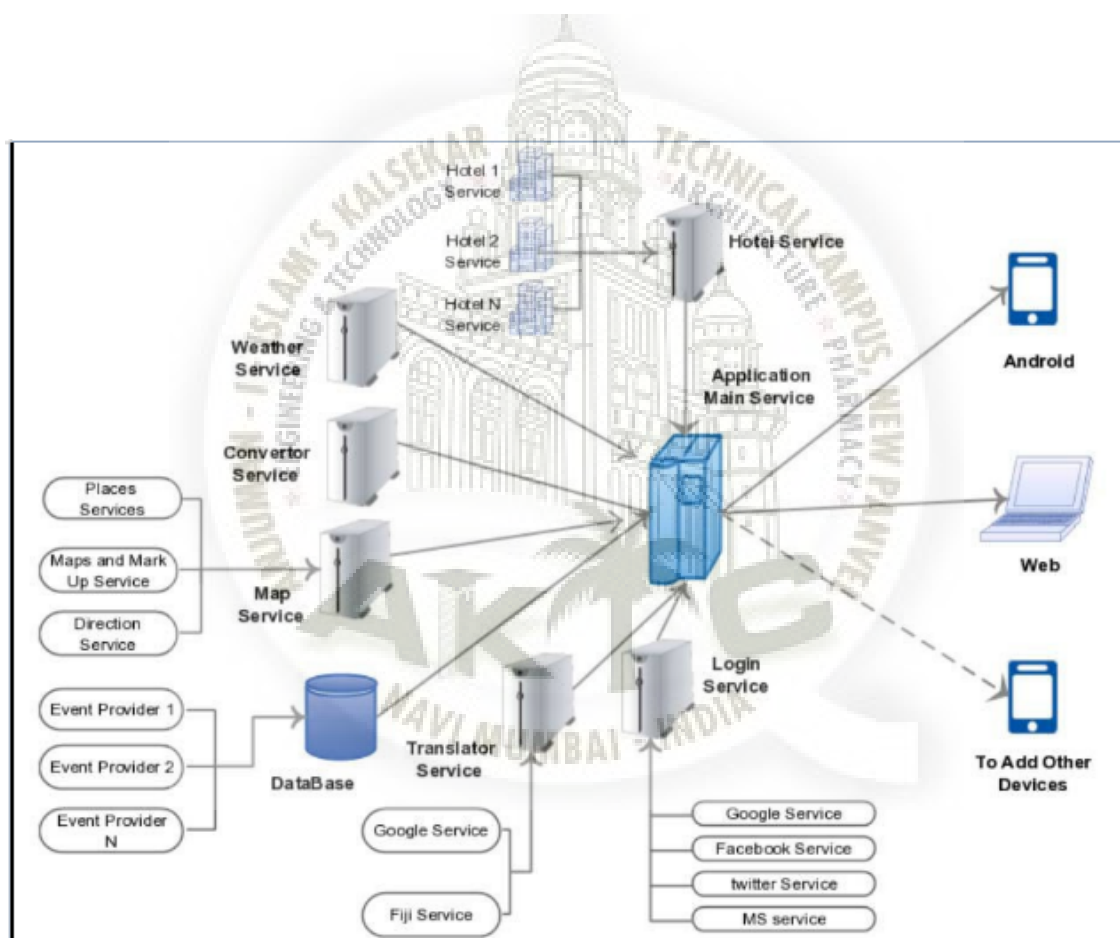

Figure 5.11: Component Diagram

5.4.4 Deployment Diagram

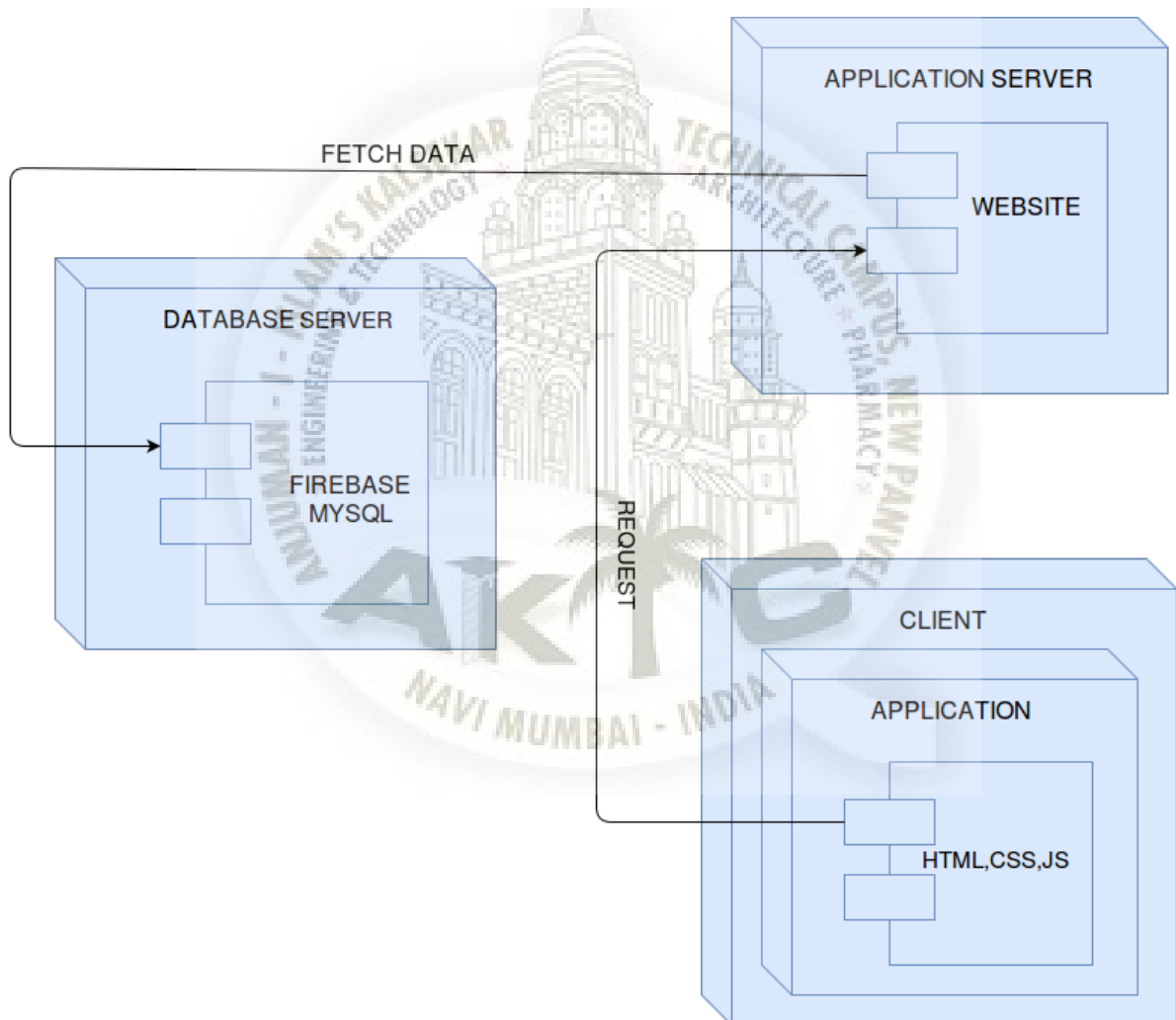This diagram of our system shows the deployment stages of our system which all together makes the whole system:

Figure 5.12: Deployment Diagram

Chapter 6

Implementation

6.1   Login and Signup

In login module the authentication part is covered and it provides a safe login into system via google id.

```java
package com.example.com.mumbai;


public class Activity2 extends AppCompatActivity {
    Button login,signup,google_signin_btn;
    FirebaseAuth mAuth;
    private final static int RC_SIGN_IN=2;
    GoogleSignInClient mGoogleSignInClient;
    FirebaseAuth.AuthStateListener mAuthListener;
    @Override
    protected void onStart() {
        super.onStart();
        mAuth.addAuthStateListener(mAuthListener);
    }

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_2);
        RelativeLayout constraintLayout=(RelativeLayout) findViewById(R.id.act2)
            ;
        AnimationDrawable animationDrawable=(AnimationDrawable) constraintLayout
            .getBackground();
        animationDrawable.setEnterFadeDuration(2000);
        animationDrawable.setExitFadeDuration(2000);
        animationDrawable.start();
        google_signin_btn=(Button) findViewById(R.id.google_btn);
        login=(Button)findViewById(R.id.login_btn);
        signup=(Button)findViewById(R.id.signup_btn);
        mAuth=FirebaseAuth.getInstance();
        google_signin_btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                signIn();
            }
        });
        final String username="",email="";
        mAuthListener=new FirebaseAuth.AuthStateListener() {
            @Override
            public void onAuthStateChanged(@NonNull FirebaseAuth firebaseAuth) {
```

```java
39                      if (firebaseAuth.getCurrentUser() != null){
40                          Intent i=new Intent(Activity2.this, curCity.class);
41                          i.putExtra("name",username);
42                          startActivity(i);
43                      }
44                  }
45              };
46          login.setOnClickListener(new View.OnClickListener() {
47              @Override
48              public void onClick(View view) {
49                  Intent log=new Intent(Activity2.this, login.class);
50                  startActivity(log);
51              }
52          });
53
54          GoogleSignInOptions gso=new GoogleSignInOptions.Builder(
                  GoogleSignInOptions.DEFAULT_SIGN_IN)
55                  .requestIdToken(getString(R.string.default_web_client_id))
56                  .requestEmail()
57                  .build();
58
59          // Build a GoogleSignInClient with the options specified by gso.
60
61          mGoogleSignInClient = GoogleSignIn.getClient(this, gso);
62
63      }
64
65      /*GoogleSignInOptions gso = new GoogleSignInOptions.Builder(
              GoogleSignInOptions.DEFAULT_SIGN_IN)
66                  .requestIdToken(getString(R.string.default_web_client_id))
67                  .requestEmail()
68                  .build();*/
69
70      private void signIn() {
71          Intent signInIntent = mGoogleSignInClient.getSignInIntent();
72          startActivityForResult(signInIntent, RC_SIGN_IN);
73      }
74      @Override
75      public void onActivityResult(int requestCode, int resultCode, Intent data) {
76          super.onActivityResult(requestCode, resultCode, data);
77
78          // Result returned from launching the Intent from GoogleSignInApi.
                  getSignInIntent(...);
79          if (requestCode == RC_SIGN_IN) {
80              Task<GoogleSignInAccount> task = GoogleSignIn.
                      getSignedInAccountFromIntent(data);
81              try {
82                  // Google Sign In was successful, authenticate with Firebase
83                  GoogleSignInAccount account = task.getResult(ApiException.class)
                          ;
84                  firebaseAuthWithGoogle(account);
85              } catch (ApiException e) {
86                  // Google Sign In failed, update UI appropriately
87                  Toast.makeText(Activity2.this,"Google signin failed",Toast.
                          LENGTH_LONG).show();
88                  // ...
89              }
90          }
91      }
```

6.2 Maps Nearby

This module of the proposed system helps the tourist to find the nearby tourist spots like shopping, adventures activities, hiking location, resorts and amusement park and many more.

```java
package com.example.com.mumbai;



public class MapsActivity extends FragmentActivity implements OnMapReadyCallback
    ,
        GoogleApiClient.ConnectionCallbacks ,
        GoogleApiClient.OnConnectionFailedListener ,
        com.google.android.gms.location.LocationListener{

    private static final int MY_PERMISSION_CODE = 1000;
    private GoogleMap mMap;
    private GoogleApiClient mGoogleApiClient;
    private double latitude,longitude;
    private android.location.Location mLastLocation;
    private Marker mMarker;
    private LocationRequest mLocationRequest;

    IGoogleAPIService mService;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_maps);
        // Obtain the SupportMapFragment and get notified when the map is ready
            to be used.
        SupportMapFragment mapFragment = (SupportMapFragment)
            getSupportFragmentManager()
                .findFragmentById(R.id.map);
        mapFragment.getMapAsync(this);

        //Init Service
        mService=Common.getGoogleAPIService();
        //Request Runtime
        if (Build.VERSION.SDK_INT>=Build.VERSION_CODES.M){
            checkLocationPermission();
        }
        BottomNavigationView bottomNavigationView=(BottomNavigationView)
            findViewById(R.id.bottom_navigation);
        bottomNavigationView.setOnNavigationItemSelectedListener(new
            BottomNavigationView.OnNavigationItemSelectedListener() {
            @Override
            public boolean onNavigationItemSelected(@NonNull MenuItem item) {
                //code later
                switch(item.getItemId()){
                    case R.id.hospital:
                        nearByPlace("hospital");
                        Log.d("Navigation","Hospital clicked");
                        break;
                    case R.id.market:
                        nearByPlace("supermarket");
                        break;
                    case R.id.school:
                        nearByPlace("atm");
```

```
50                          break;
51                  case R.id.restaurant:
52                      nearByPlace("restaurant");
53                          break;
54                  default:
55                          break;
56              }
57              return true;
58          }
59      });


62  }

64  public void nearByPlace(final String placeType) {
65      mMap.clear();

67      String url=getUrl(latitude,longitude,placeType);

69      mService.getNearByPlaces(url)
70              .enqueue(new Callback<MyPlaces>() {
71                  @Override
72                  public void onResponse(Call<MyPlaces> call, Response<
                        MyPlaces> response) {
73                      if (response.isSuccessful()){
74                          for (int i=0;i<response.body().getResults().length;i
                            ++){
75                          MarkerOptions markerOptions=new MarkerOptions();
76                          Results googleplace=response.body().getResults()
                            [i];
77                          double lat=Double.parseDouble(googleplace.
                            getGeometry().getLocation().getLat());
78                          double lng=Double.parseDouble(googleplace.
                            getGeometry().getLocation().getLng());
79                          String vicinity=googleplace.getVicinity();
80                          String placeName=googleplace.getName();
81                          //Toast.makeText(getApplicationContext(),
                            placeName,Toast.LENGTH_SHORT).show();

83                          LatLng latLng=new LatLng(lat,lng);
84                          markerOptions.position(latLng);
85                          markerOptions.title(placeName);
86                          if (placeType.equals("restaurant")){
87                              markerOptions.icon(BitmapDescriptorFactory.
                                fromResource(R.drawable.
                                restaurant_marker));
88                          mMarker=mMap.addMarker(markerOptions);
89                          }else if (placeType.equals("supermarket")){
90                              markerOptions.icon(BitmapDescriptorFactory.
                                fromResource(R.drawable.market_marker));
91                          mMarker=mMap.addMarker(markerOptions);
92                          }else if (placeType.equals("atm")){
93                              markerOptions.icon(BitmapDescriptorFactory.
                                fromResource(R.drawable.atm_marker));
94                          mMarker=mMap.addMarker(markerOptions);
95                          } else if (placeType.equals("hospital")){
96                              markerOptions.icon(BitmapDescriptorFactory.
                                fromResource(R.drawable.hospital_market)
                                );
97                          mMarker=mMap.addMarker(markerOptions);
98                          }else{
```

```
99                                            markerOptions.icon(BitmapDescriptorFactory.
                                                   defaultMarker(BitmapDescriptorFactory.
                                                   HUE_RED));
100                                      }

102                                  }
103                          }else{
104                              Toast.makeText(getApplicationContext(), "Error",
                                     Toast.LENGTH_SHORT).show();
105                          }
106                      }

108                      @Override
109                      public void onFailure(Call<MyPlaces> call, Throwable t) {
110                          Log.d("onFailure","MyPlaces Failed");

112                      }
113                  });
114      }

116      private String getUrl(double latitude, double longitude, String placeType) {
117          StringBuilder googlePlacesUrl=new StringBuilder("https://maps.googleapis
                 .com/maps/api/place/nearbysearch/json?");
118          googlePlacesUrl.append("location="+latitude+","+longitude);
119          googlePlacesUrl.append("&radius="+1000);
120          googlePlacesUrl.append("&type="+placeType);
121          googlePlacesUrl.append("&sensor=true");
122          googlePlacesUrl.append("&key="+getResources().getString(R.string.
                 browser_key));
123          Log.d("getUrl",googlePlacesUrl.toString());
124          return googlePlacesUrl.toString();
125      }

127      private boolean checkLocationPermission() {
128          if (ContextCompat.checkSelfPermission(this, Manifest.permission.
                 ACCESS_FINE_LOCATION)!=PackageManager.PERMISSION_GRANTED){
129              if (ActivityCompat.shouldShowRequestPermissionRationale(this,
                     Manifest.permission.ACCESS_FINE_LOCATION))
130                  ActivityCompat.requestPermissions(this,new String[]{
131                          Manifest.permission.ACCESS_FINE_LOCATION
132                  },MY_PERMISSION_CODE);
133              else
134                  ActivityCompat.requestPermissions(this,new String[]{
135                          Manifest.permission.ACCESS_FINE_LOCATION
136                  },MY_PERMISSION_CODE);
137              return false;

139          }
140          else
141              return true;
142      }

144      @Override
145      public void onRequestPermissionsResult(int requestCode, @NonNull String[]
             permissions, @NonNull int[] grantResults) {
146          switch(requestCode){
147              case MY_PERMISSION_CODE:{
148                  if (grantResults.length>0 && grantResults[0]==PackageManager.
                         PERMISSION_GRANTED){
149                      if (ContextCompat.checkSelfPermission(this, Manifest.
                             permission.ACCESS_FINE_LOCATION)==PackageManager.
```

```
                                    PERMISSION_GRANTED){
150                         if (mGoogleApiClient==null)
151                             buildGoogleApiClient();
152                     mMap.setMyLocationEnabled(true);

153
154                     }
155                 }else{
156                     Toast.makeText(this,"Permission Refused",Toast.LENGTH_LONG).
                        show();
157                 }
158                 break;
159             }
160         }
161     }

162
163     @Override
164     public void onMapReady(GoogleMap googleMap) {
165         mMap = googleMap;
166         if (Build.VERSION.SDK_INT>=Build.VERSION_CODES.M){
167             if (ContextCompat.checkSelfPermission(this,android.Manifest.
                permission.ACCESS_FINE_LOCATION)== PackageManager.
                PERMISSION_GRANTED){
168                 buildGoogleApiClient();
169                 mMap.setMyLocationEnabled(true);
170             }
171         }
172         else{
173             buildGoogleApiClient();
174             mMap.setMyLocationEnabled(true);
175         }

176
177
178     }

179
180     private synchronized void buildGoogleApiClient() {
181         mGoogleApiClient =new GoogleApiClient.Builder(this)
182                 .addConnectionCallbacks(this)
183                 .addOnConnectionFailedListener(this)
184                 .addApi(LocationServices.API)
185                 .build();
186         mGoogleApiClient.connect();
187     }
```

6.3   Weather

Time to time updates regarding the climate will notified to the tourists. For regular updates of weather the tourist should have connected to internet.

```
1   package com.example.com.mumbai;
2
3   public class weather extends AppCompatActivity {
4   TextView cityField, detailsField, currentTemperatureField, humidity_field,
5   pressure_field, weatherIcon, updatedField;
6   Typeface weatherFont;
7   @Override
8   protected void onCreate(Bundle savedInstanceState) {
9   super.onCreate(savedInstanceState);
10  setContentView(R.layout.activity_weather);
11  RelativeLayout constraintLayout=(RelativeLayout)
12  findViewById(R.id.weather);
13  AnimationDrawable animationDrawable=(AnimationDrawable)
14  constraintLayout.getBackground();
15  animationDrawable.setEnterFadeDuration(2000);
16  animationDrawable.setExitFadeDuration(2000);
17  animationDrawable.start();
18  weatherFont = Typeface.createFromAsset(getApplicationContext().getAssets(),
19  &quot;fonts/weathericons-regular-webfont.ttf&quot;);
20  cityField=(TextView) findViewById(R.id.city_field);
21  updatedField = (TextView)findViewById(R.id.updated_field);
22  detailsField = (TextView)findViewById(R.id.details_field);
23  currentTemperatureField =
24  (TextView)findViewById(R.id.current_temperature_field);
25  humidity_field = (TextView)findViewById(R.id.humidity_field);
26  pressure_field = (TextView)findViewById(R.id.pressure_field);
27  weatherIcon = (TextView)findViewById(R.id.weather_icon);
28  weatherIcon.setTypeface(weatherFont);
29  Function.placeIdTask asyncTask =new Function.placeIdTask(new
30  Function.AsyncResponse() {
31  public void processFinish(String weather_city, String
32  weather_description, String weather_temperature, String weather_humidity, String
33  weather_pressure, String weather_updatedOn, String weather_iconText, String
34  sun_rise) {
35  updatedField.setText(weather_updatedOn);
36  detailsField.setText(weather_description);
37  currentTemperatureField.setText(weather_temperature);
38  humidity_field.setText(&quot;Humidity: &quot;+weather_humidity);
39  pressure_field.setText(&quot;Pressure: &quot;+weather_pressure);
40  weatherIcon.setText(Html.fromHtml(weather_iconText));
41  }
42  });
43  asyncTask.execute(&quot;19.107956&quot;,&quot;72.848385&quot;); //
44  asyncTask.execute(&quot;Latitude&quot;, &quot;Longitude&quot;)
45
46  }
47  }package com.example.com.mumbai;
48  /**
49  * Created by Mahir on 15/3/2018.
50  */
51
52  public class Function {
53  private static final String OPEN_WEATHER_MAP_URL =
54  &quot;http://api.openweathermap.org/data/2.5/weather?lat=%s&amp;lon=%s&amp;units
       =metric&quot;;;
55  private static final String OPEN_WEATHER_MAP_API =
```

```
56 &quot; ba7e9b6d3d4af71ea42f28f153766bf2&quot ; ;
57 public static String setWeatherIcon ( int actualId , long sunrise , long sunset ) {
58 int id = actualId / 100;
59 String icon = &quot;&quot ; ;
60 if ( actualId == 800) {
61 long currentTime = new Date ( ) . getTime ( ) ;
62 if ( currentTime&gt ;= sunrise &amp;&amp; currentTime&lt ; sunset ) {
63 icon = &quot;&amp;# xf00d;&quot ; ;
64 } else {
65 icon = &quot;&amp;# xf02e;&quot ; ;
66 }
67 } else {
68 switch ( id ) {
69 case 2 : icon = &quot;&amp;# xf01e;&quot ; ;
70 break ;
71 case 3 : icon = &quot;&amp;# xf01c;&quot ; ;
72 break ;
73 case 7 : icon = &quot;&amp;# xf014;&quot ; ;
74 break ;
75 case 8 : icon = &quot;&amp;# xf013;&quot ; ;
76 break ;
77 case 6 : icon = &quot;&amp;# xf01b;&quot ; ;
78 break ;
79 case 5 : icon = &quot;&amp;# xf019;&quot ; ;
80 break ;
81 }
82 }
83 return icon ;
84 }
85
86 public interface AsyncResponse {
87 void processFinish ( String output1 , String output2 , String output3 , String
88 output4 , String output5 , String output6 , String output7 , String output8 ) ;
89 }
90
91 public static class placeIdTask extends AsyncTask&lt ; String , Void , JSONObject&gt
       ; {
92 public AsyncResponse delegate = null ; // Call back interface
93 public placeIdTask ( AsyncResponse asyncResponse ) {
94 delegate = asyncResponse ; // Assigning call back interfacethrough
95 constructor
96 }
97 @Override
98 protected JSONObject doInBackground ( String ... params ) {
99 JSONObject jsonWeather = null ;
100 try {
101 jsonWeather = getWeatherJSON ( params [ 0 ] , params [ 1 ] ) ;
102 } catch ( Exception e ) {
103 Log . d(&quot; Error&quot ; , &quot; Cannot process JSON results&quot ; , e ) ;
104 }
105 return jsonWeather ;
106
107 }
108 @Override
109 protected void onPostExecute ( JSONObject json ) {
110 try {
111 if ( json != null ) {
112 JSONObject details =
113 json . getJSONArray(&quot; weather&quot ; ) . getJSONObject ( 0 ) ;
114 JSONObject main = json . getJSONObject(&quot; main&quot ; ) ;
115 DateFormat df = DateFormat . getDateTimeInstance ( ) ;
```

```
116  String city = json.getString(&quot;name&quot;).toUpperCase(Locale.US) +
117  &quot;, &quot; + json.getJSONObject(&quot;sys&quot;).getString(&quot;country&
     quot;);
118  String description =
119  details.getString(&quot;description&quot;).toUpperCase(Locale.US);
120  String temperature = String.format(&quot;%.2f&quot;,
121  main.getDouble(&quot;temp&quot;))+ &quot;  &quot;;
122  String humidity = main.getString(&quot;humidity&quot;) + &quot;%&quot;;
123  String pressure = main.getString(&quot;pressure&quot;) + &quot; hPa&quot;;
124  String updatedOn = df.format(new
125  Date(json.getLong(&quot;dt&quot;)*1000));
126  String iconText = setWeatherIcon(details.getInt(&quot;id&quot;),
127  json.getJSONObject(&quot;sys&quot;).getLong(&quot;sunrise&quot;) * 1000,
128  json.getJSONObject(&quot;sys&quot;).getLong(&quot;sunset&quot;) * 1000);
129  delegate.processFinish(city, description, temperature,
130  humidity, pressure, updatedOn, iconText, &quot;&quot;+
131  (json.getJSONObject(&quot;sys&quot;).getLong(&quot;sunrise&quot;) * 1000));
132  }
133  } catch (JSONException e) {
134  //Log.e(LOG_TAG, &quot;Cannot process JSON results&quot;, e);
135  }
136
137  }
138  }
139
140  public static JSONObject getWeatherJSON(String lat, String lon){
141  try {
142  URL url = new URL(String.format(OPEN_WEATHER_MAP_URL, lat, lon));
143  HttpURLConnection connection =
144  (HttpURLConnection)url.openConnection();
145  connection.addRequestProperty(&quot;x-api- key&quot;, OPEN_WEATHER_MAP_API);
146  BufferedReader reader = new BufferedReader(
147  new InputStreamReader(connection.getInputStream()));
148  StringBuffer json = new StringBuffer(1024);
149  String tmp=&quot;&quot;;
150  while ((tmp=reader.readLine())!=null)
151  json.append(tmp).append(&quot;\n&quot;);
152  reader.close();
153  JSONObject data = new JSONObject(json.toString());
154  // This value will be 404 if the request was not
155  // successful
156  if(data.getInt(&quot;cod&quot;) != 200){
157  return null;
158  }
159  return data;
160  }catch(Exception e){
161  return null;
```

Chapter 7

System Testing

The System is tested in every environment,we have done unit testing and make them intergrated together for intergration testing.We have done Beta testing by the person apart from our group.

7.1 Test Cases and Test Results

Test ID	Test Case Title	Test Condition	System Behavior	Expected Result
T01	Login	Should be register user	Will connect to firebase	Should login into system
T02	Maps	should accept location	Will connect to Google API	Calculate the route
T03	Data Fetching	Correct data	REST API call	Display of compare data

7.2 Sample of a Test Case

Title: Login Page – Authenticate Successfully on google
Description: A registered user should be able to successfully login at google

Precondition: The user must already be registered with an email address and password of google acccount.

Assumption: our system must be installed in the user mobile phone .

Test Steps:

1. Start the application

2. Click on login button.

3. Enter the google emailid and password.

4. Click 'Sign In'

Expected Result: The user must be login in the system using google account

Actual Result:

The user was logged into the system,and the session was maintained successfully.

7.2.1 Software Quality Attributes

1. AVAILABILITY: The system should not be down,whenever the user use the system the specific data should be available to the user.

2. CORRECTNESS: As per the user search the correct data should be shown to the user like at time for searching the near by place the system should show only the places around the user.

3. MAINTAINABILITY: The administrators of the system will maintain the system with effective updates though on air update if needed.

4. EXTENSIBILITY: The system is capable to be modified by changing some modules or by adding some features to the existing system

Chapter 8

Screenshots of Project

8.1 Login and Signup
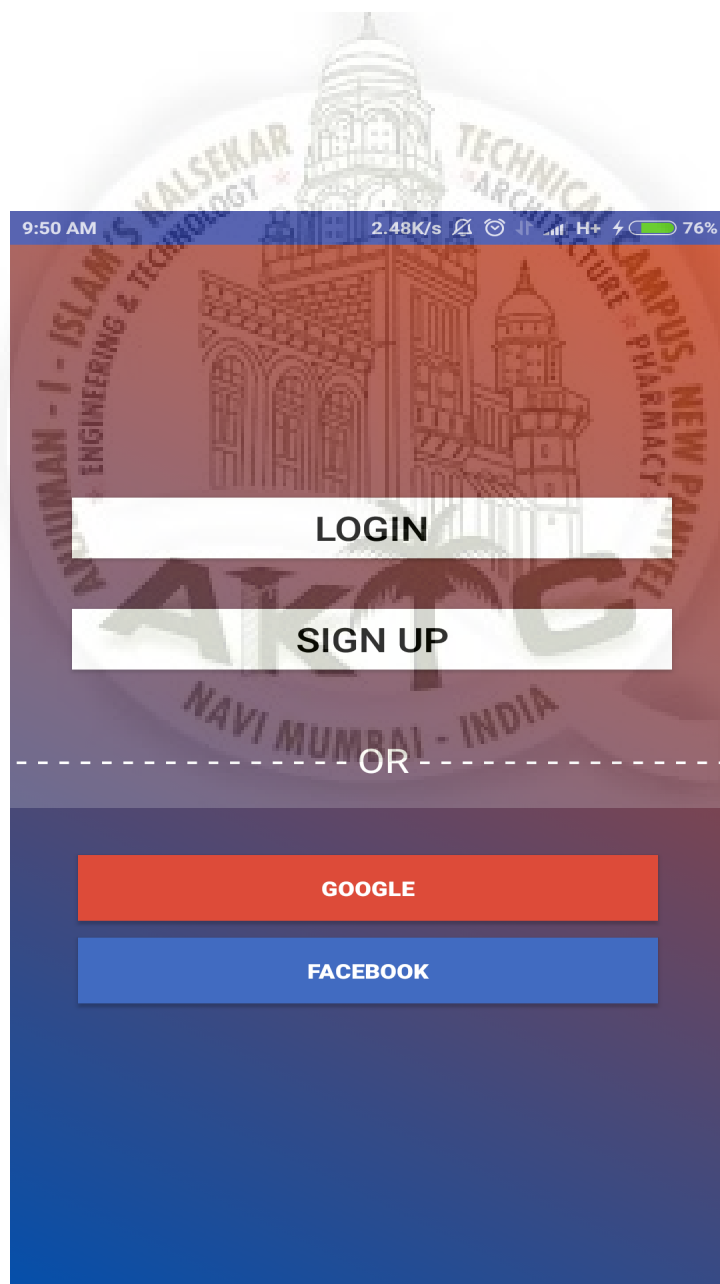
Figure 8.1: Login and Signup

8.2   Home screen

Figure 8.2: Home Screen

Figure 8.3: Travelling details for Partner
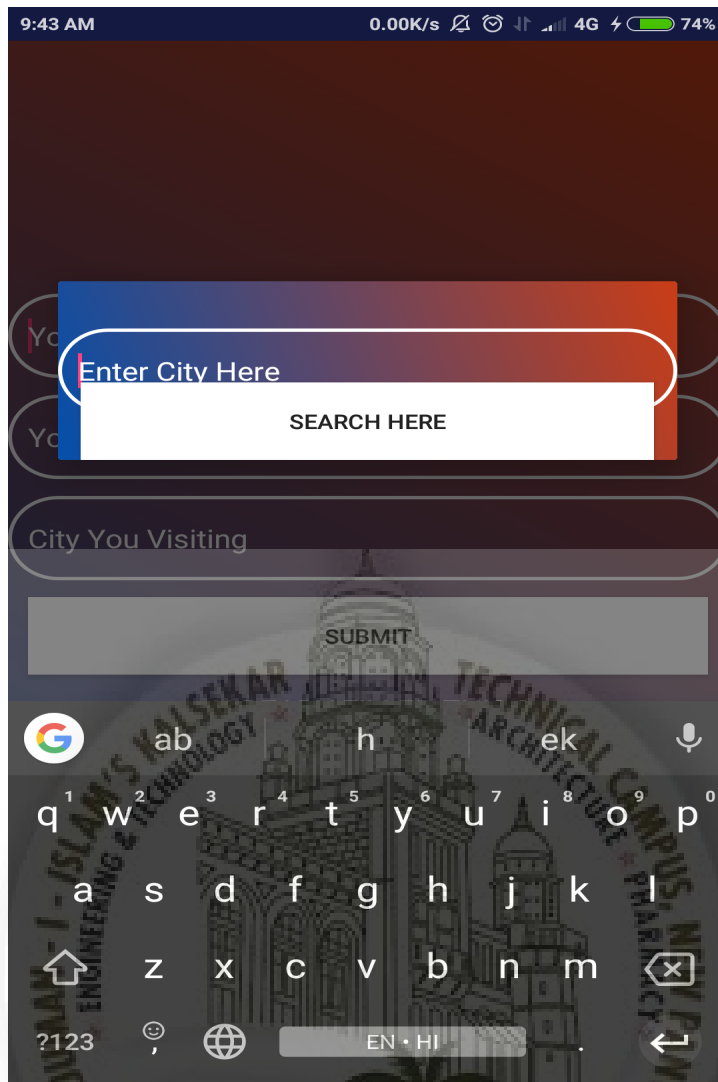
Figure 8.4: Search Partner for specific Destination
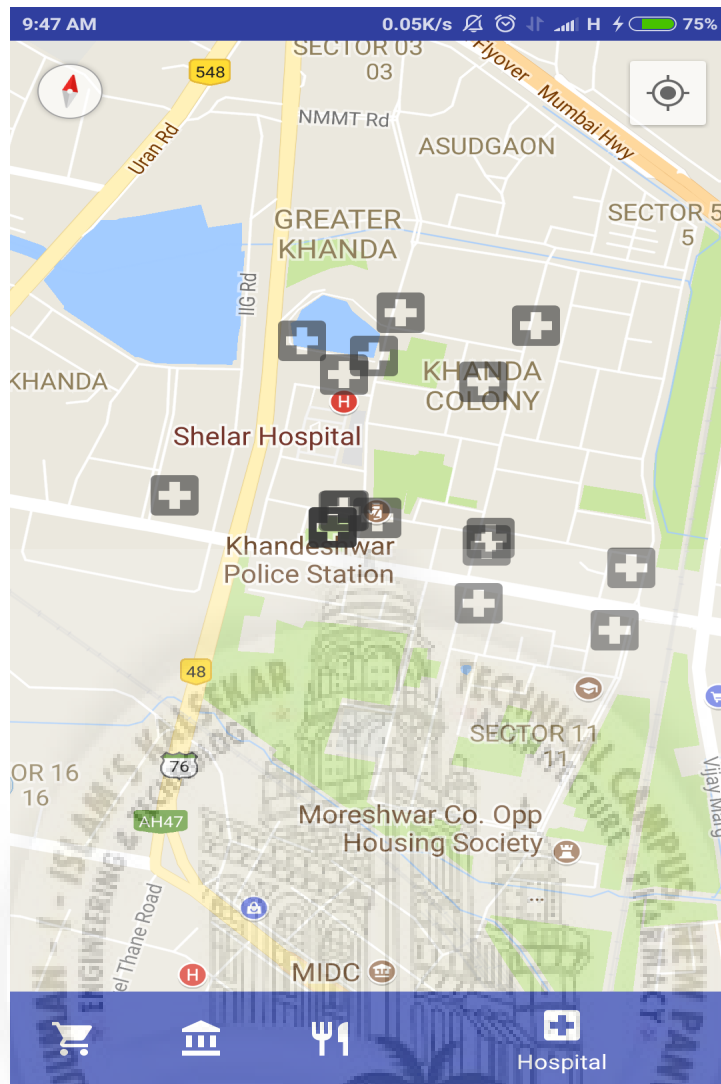
Figure 8.5: Nearby Hospitals

13 Apr 2018 9:30:00 a.m.

HAZE
Humidity: 58%
Pressure: 1010 hPa

30.45°

Figure 8.6: Weather

Figure 8.7: Nearby Markets
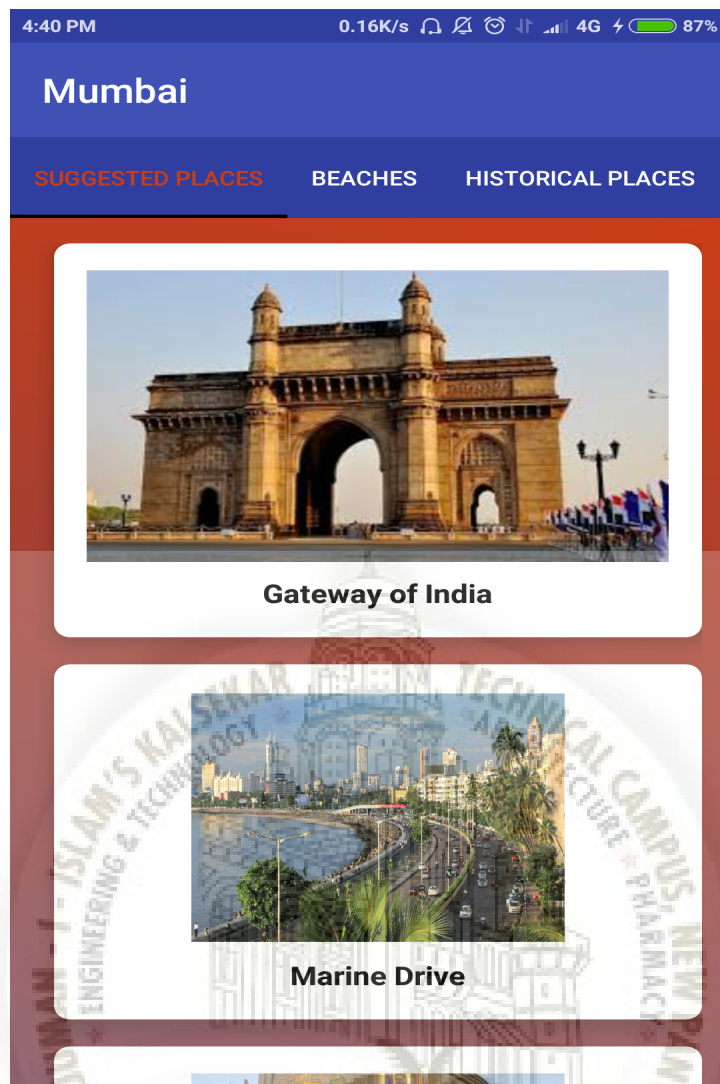
Figure 8.8: Tourist Attraction and Suggested Places
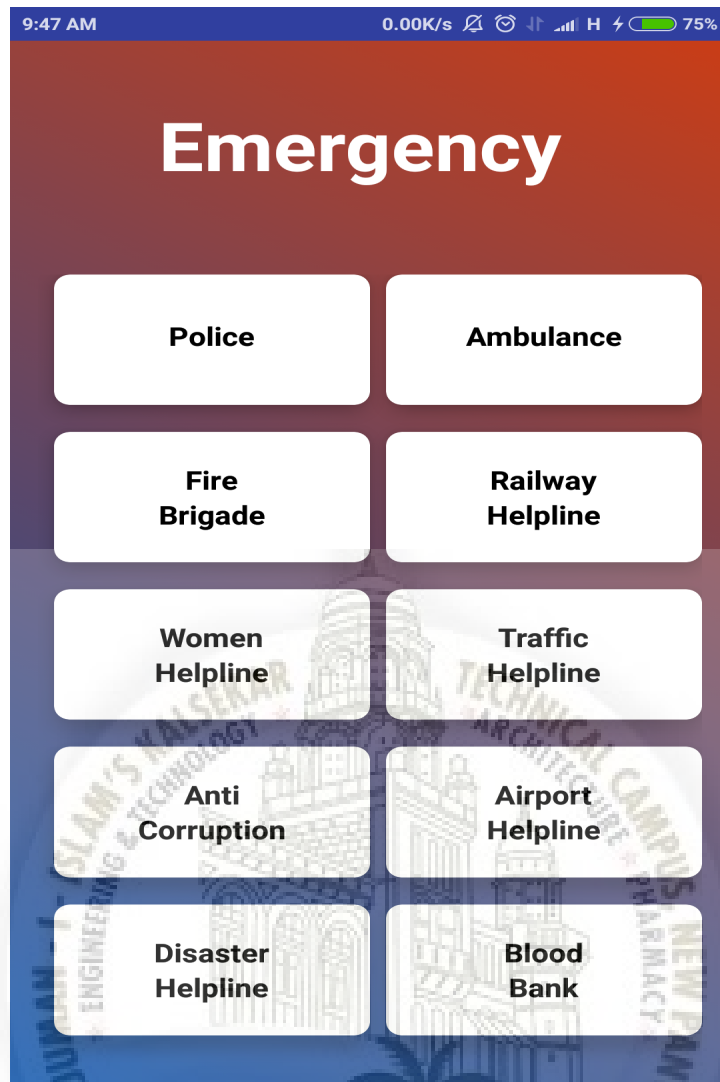
Figure 8.9: Emergency Contacts

Chapter 9

Conclusion and Future Scope

9.1   Conclusion

The main aim of the research project was to develop a tourist guide system for tourist travelling. An important assumption made while developing the system was that the users have the basic idea about using an android mobile device and they are familiar with the English language. In order to use the location based services user needs to be in a place where the mobile device receives GPS data accurately. The system needs a better network connection for communication between the mobile device and the server. Otherwise it takes a long time to receive the data from database. The web and Android application developed to show how the architecture works and has a lot of important features for tourists such as booking, maps and places, events, weather, travel partner and many others discussed in this paper.

An offline version of android application for the system which won't need full time Internet connection get services and as soon as the application gets connected to Internet then the application should be automatically get synchronized with the real time service data. The system provide information query of the hotel, scenery, restaurant, traffic and so on. The system is a combination of smart phone and Internet services and will facilitate tour and life for user. In this paper we have proposed a system or mobile application which will track the current position of the user and send it to the server which will send the requested information for the nearby attractions.

9.2 Future Scope

- The system will work offline.

- The system can distinguish the user on its types whether he/she is solo traveller,family trip or a business trip.

References

[1] Travel Guide Using GPS *Web and Mobile Based Tourist Travel Guide System For Fiji Tourism Industry*; Vineet Singh,Akeshanil Bali,Abhinesh Adhikthikar,Rohitash Chandra,, May 2015

[2] *https://ieeexplore.ieee.org*;Akeshanil Bali,Abhinesh Adhikthikar,Rohitash Chandra,Vineet Singh january 2017

[3] M. Kenteris D. Gavalas D. Economou "Evaluation of Mobile Tourist Guides".

[4] Emilio Ferrara, Pasquale De Meo, Giacomo Fiumara, Robert Baumgartner, "Web Data Extraction, Applications and Techniques: A Survey", June 5, 2014.

[5] Google Places API https://developers.google.com/

Achievements

1. Publications

   (a) *Travel Offline Virtual Support Through Locational Reminder and Suggestions Using Data Analytics*; Khan Mahir Ali , Shaikh Arfat Izharulhaq, Shaikh Farhan Firoz Ahmed ,IJISRT, 1st January 2018 of published(http://ijisrt.com/travel-offline- virtual-support-through- locational-reminder-and-suggestions-using-data-analytics)

2. Conferences

   (a) *Travel offline virtual support through locational reminder using data analytics*; Khan Mahirali,Shaikh Arfat,Shaikh Farhan, Multicon-w 2018 , February 2018(Venue :Thakur College of Engineering)

3. Project Competitions

   (a) *Travel Guide*;Khan MahirAli,Shaikh Arfat,Shaikh Farhan, 4th national level project exhibition cum poster presentation , march 2018(Venue :Universal college of engineering)

INTERNATIONAL JOURNAL OF INNOVATIVE SCIENCE AND RESEARCH TECHNOLOGY

IJISRT   A DIGITAL LIBRARY

ISSN NO :- 2456-2165

AUTHOR CERTIFICATE

THIS IS TO CERTIFY THAT THE MANUSCRIPT, ENTITLED
Travel Offline Virtual Support Through Locational
Reminder and Suggestions using Data Analytics

AUTHORED BY
Khan Mahir Ali

HAS BEEN PUBLISHED IN
Volume 3 | Issue 1 | January - 2018

ARTICLE DIGITAL NO.
IJISRT18JA23

EDITOR IN CHIEF IJISRT

WWW.IJISRT.COM

A DIGITAL LIBRARY

INTERNATIONAL JOURNAL OF INNOVATIVE SCIENCE AND RESEARCH TECHNOLOGY

IJISRT   A DIGITAL LIBRARY

ISSN NO :- 2456-2165

AUTHOR CERTIFICATE

A DIGITAL LIBRARY

THIS IS TO CERTIFY THAT THE MANUSCRIPT, ENTITLED

Travel Offline Virtual Support Through Locational
Reminder and Suggestions using Data Analytics

AUTHORED BY
Shaikh Arfat Izharul haq

HAS BEEN PUBLISHED IN
Volume 3 | Issue 1 | January - 2018

ARTICLE DIGITAL NO.
IJISRT18JA23

EDITOR IN CHIEF IJISRT

WWW.IJISRT.COM

INTERNATIONAL JOURNAL OF INNOVATIVE SCIENCE AND RESEARCH TECHNOLOGY

IJISRT   A DIGITAL LIBRARY

ISSN NO :- 2456-2165

AUTHOR CERTIFICATE

THIS IS TO CERTIFY THAT THE MANUSCRIPT, ENTITLED
Travel Offline Virtual Support Through Locational
Reminder and Suggestions using Data Analytics

AUTHORED BY
Shaikh Farhan Firoz ahmed

HAS BEEN PUBLISHED IN
Volume 3 | Issue 1 | January - 2018

ARTICLE DIGITAL NO.
IJISRT18JA23
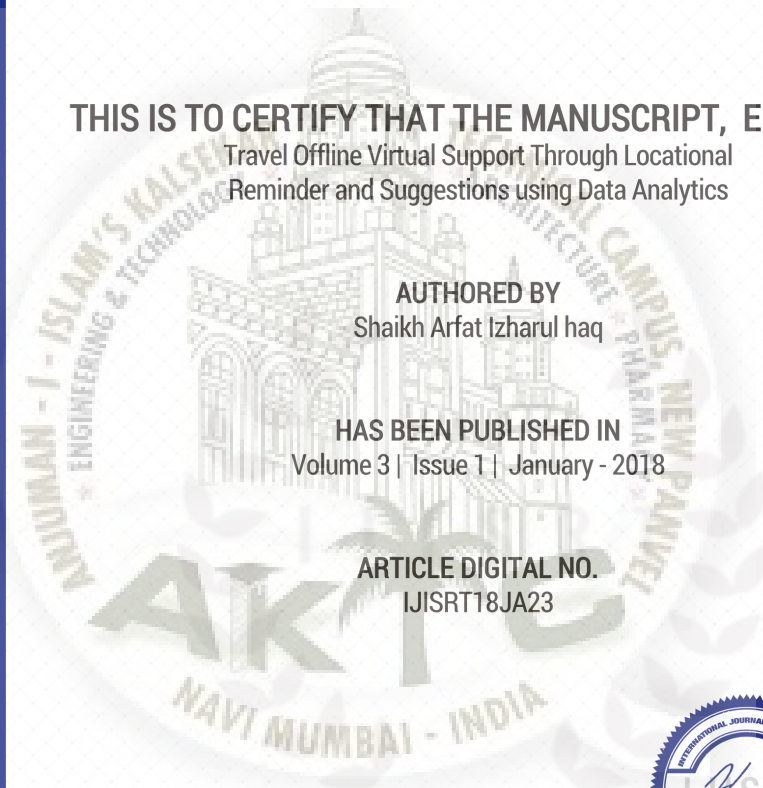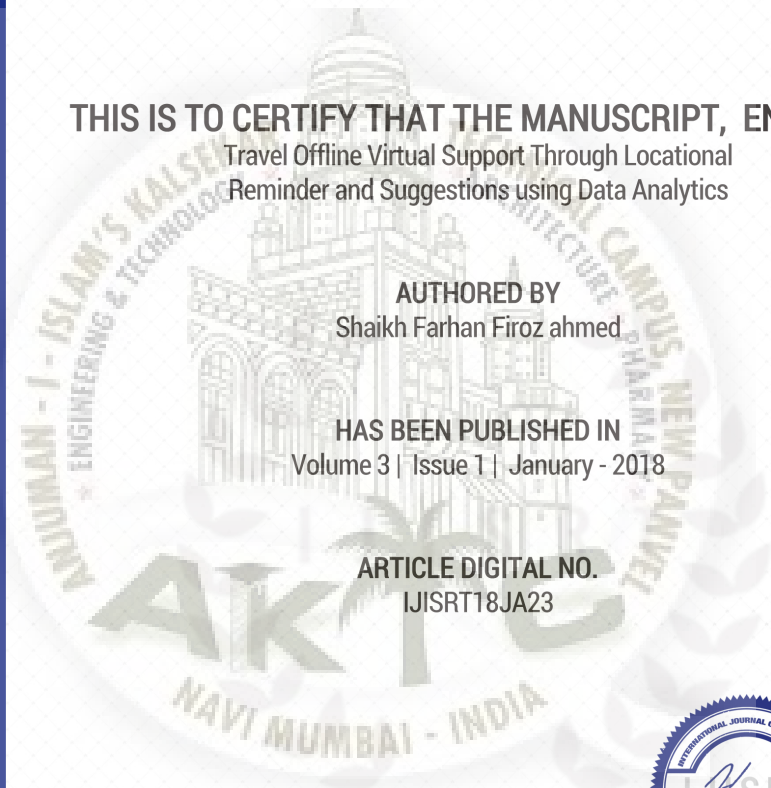
EDITOR IN CHIEF IJISRT

This document certifies that the manuscript listed above was submitted by above said respected author
To verify the submitted manuscript please visit our official website: www.ijisrt.com
Or Email us: editor@ijisrt.com

W W W . I J I S R T . C O M

Universal College of Engineering

DTE Code: 3460

(Permanently Unaided | Approved by AICTE, DTE & Affiliated to University of Mumbai)

Near Bhajansons and Pinyadham, Kaman Bhiwandi Road, Vasai
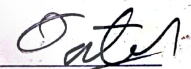
in association with **I.E.T.E. - I.S.F., C.S.I. & I.S.A.**

ISA

4th National Level Project Exhibition Cum Poster Presentation

Certificate of Participation

Awarded to Ms./Mr. _Khan Mahir Ali_

of _A. I. Kalsekar Technical Campus_

College for participating in "4th National Level Project Exhibition Cum Poster Presentation" 2018.

Date: 9th March 2018

Dr. Ajoy Kumar
(Principal)

Dr. J. B. Patil
(Campus Director)

Universal College of Engineering

DTE Code: 3460

(Permanently Unaided | Approved by AICTE, DTE & Affiliated to University of Mumbai)
Near Bhajansons and Pinyadham, Kaman Bhiwandi Road, Vasai
in association with **I.E.T.E. - I.S.F., C.S.I. & I.S.A.**

ISA

4th National Level Project Exhibition Cum Poster Presentation

Certificate of Participation

Awarded to Ms./Mr. _Shaikh   Arfat_

of _A·I   Kalsekar   technical   Campus_

College for participating in "**4th National Level Project Exhibition Cum Poster Presentation**" *2018*.

Date: *9th March 2018*

Dr. Ajoy Kumar
(Principal)

Dr. J. B. Patil
(Campus Director)

Universal College of Engineering

DTE Code: 3460

(Permanently Unaided | Approved by AICTE, DTE & Affiliated to University of Mumbai)
Near Bhajansons and Pinyadham, Kaman Bhiwandi Road, Vasai
in association with **I.E.T.E. - I.S.F., C.S.I. & I.S.A.**

4ᵗʰ National Level Project Exhibition Cum Poster Presentation

Certificate of Participation

Awarded to Ms./Mr. _Shaikh Farhan_

A. I. Kalsekar Technical Campus

...ge for participating in "**4ᵗʰ National Level Project Exhibition Cum Poster Presentation**" 2018.

Date: 9ᵗʰ March 2018

Dr. Ajoy Kumar
(Principal)

(Campus ....)

THAKUR TRUSTS

Zagdu Singh Charitable Trust's (Regd.)
THAKUR COLLEGE OF ENGINEERING & TECHNOLOGY
(Approved by AICTE, Govt. of Maharashtra & Affiliated to University of Mumbai)
Thakur Village, Kandivali (E), Mumbai - 400101.

tcet
Estd. 2001
ISO 9001 : 2015 Certified
NBA and NAAC Accredited

Certificate of Appreciation

This is to certify that Dr./Mr./Ms. __Mahir Khan__ has presented/published

a __Full__ length paper with the title __Travel offline virtual support through locational reminder and suggestions using data analytics__

in the **Conference on Recent Trends in Computer Engineering (CRTCE 2018)** organized during February, 23rd & 24th,

2018 at Thakur College of Engineering and Technology.

(Dr. Sheetal Rathi)
Convenor

(Dr. R. R. Sedamkar)
Technical Chair

(Dr. B. K. Mishra)
Principal & Programme Chair

MULTICON-W 2018
9th National & International Conferences & Workshops

Philanthropic Partners
Platinum — Diamond

Technical Partners

Other Partners
KILLER — Style
FREE PRESS — Print Media
CARNIVAL — Cinema

Collaborators