Smart Travel Planner

Final Year Project – Mid Report

Session 2017-2021

A project submitted in partial fulfilment of the
COMSATS University Degree
of
BS in Computer Science / Software Engineering (CUI)

Department of Computer Science

COMSATS University Islamabad, Lahore Campus

08 January 2021

Evaluation

		Project Group Members			
Sr.#	Reg. #	Student Name	Email	*Signature	Obtained Marks (Total Marks: 10)
(i)	FA17-BCS-087	Abbas Ali	abbasashraff12313@gmail.com		
(ii)	FA17-BCS-127	Muhammad Saeed	saeedartists@gmail.com		
(iii)	FA17-BCS-152	Ibad Ur Rehman	ibadurrehman78@gmail.com		

*The candidates confirm that the work submitted is their own and appropriate credit has been given where reference has been made to work of others.

Supervisor Name: Dr. Hamid Turab Supervisor Signature: ________________

Remarks (if any): ____

____

____

____

Abstract

Travelling is crucial in human race for recreation and exploring cultures, trend of tourism is increasing day by day, and people across the globe start take interest in new places. Its scope is equally prevalent in Pakistan but meet the shortcomings due to lack of knowledge about proper place and inadequacy of proper guideline of travelling areas for visitors. The purpose is to reduce user the time by helping them to finding different tourist route and providing proper information like distance, pictures and reviews about that place and suggest places according to user's interest. Many applications have been developed but none of them fulfilled the needs of travelling. The propose project is supposed to a working application like Roadtrippers, Waze, CultureTrip and Skyscanner. For this purpose, the application (IOS & Android) is to develop which not only suggests routes but also with user's interest and proper reviews. Therefore, we create large dataset by mining data of travel locations from google and different websites. Then digital image processing is applied to the pictures of the dataset to obtain better quality images and computer vision algorithm to assign a particular label to all the images.

1 Introduction 1
1.1 Problem statement 1
1.2 Objective 2
1.3 Assumptions and Constraints 2
1.3.1 Assumptions 2
1.3.2 Constraints 4
1.4 Project Scope 4
1.5 Chapter Summary 4
2 Requirement Analysis 5
2.1 Literature Review 5
2.2 Stakeholders 5
2.3 Requirements Elicitation 5
2.3.1 Functional Requirements 5
2.3.2 Non-Functional Requirements 8
2.4 Use Case Design 9
2.4.1 Use case descriptions 10
2.5 Software Development Lifecycle Model 10
2.5.1 Requirements 10
2.5.2 Plan 10
2.5.3 Design and Develop 11
2.5.4 Release 11
2.5.5 Track & Monitor 11
2.6 Chapter Summary 11
3 System Design 12
3.1 Work Breakdown Structure 12
- Research 12
- Development 12
3.2 Model Architecture 13
3.3 Activity Diagram 14
3.4 Class Diagram 15
3.5 Collaboration Diagram 15
3.6 Network Diagram 16
3.7 Chapter Summary 16
4 Implementation 17
4.1 User Interfaces 17
4.2 Algorithms 20
4.3 Chapter Summary 25
5 System Testing 26
5.1 Test Cases 26
5.2 Unit/ Acceptance Testing 26
5.3 Chapter Summary 26
6 Conclusion 27
6.1 Problems faced and Lessons Learned 27
6.2 Project Summary 27
6.3 Future Work 27

7 References..28

List of Abbreviations

- **WBS** – Work break down structure
- **SDLC –** Software Development Lifecycle
- **SE** – Software Engineering
- **R&D** – Research and Development

List of Figures

Figure 1 – Dataset.. 3
Figure 2 – Use Case Design 9
Figure 3 - Agile Lifecycle 10
Figure 4 – Work Breakdown Structure... 12
Figure 5 - Model Architecture 13
Figure 6 - Activity Diagram 14
Figure 7 - Class Diagram.. 15
Figure 8 - Collaboration Diagram... 15
Figure 9 - Network Diagram 16
Figure 10 – Sign up Screen 17
Figure 11 – Login Screen 17
Figure 12 – Home Screen 18
Figure 13 – Map Screen 19

List of Tables

Table 1 – Functional Requirements 6

Table 2 – Test Cases 26

1 Introduction

Travelling and seeing places is an important part of our lives. One cannot believe a place to be true unless one sees it. Anyone can visit any places around the globe but one cannot know the place he/she is going to visit is of his/her interest. This can cause major issues for the traveler. The traveller can see the photos and reviews of the place he/she going to visit but he/she have to go a tons of different platforms and website and have to ask questions to make sure he/she is going to visit the right place. This is very time consuming and take a lot of effort.

People nowadays prefer to travel only places, which are of their interest, they tend to have some prior knowledge of the place they are going to visit. The knowledge of the places can be get from social media and different travelling websites. However, when a random person travel he do not know the website to look for and social media accounts to search for. So, due to this reason he/she may not be able to explore the way he/she want to. Reviews can help to find out the latest trending places and to find out which place is best for them. Reviews affect many things of a place, it can affect the mind-set of a person visiting it, But one can travel unexpectedly that is having no knowledge of place whatsoever, There should be some mechanism to entertain all sort of travelers with proper knowledge and reviews so they can visit the places of their interests.

The Project will be a cross platform mobile app in which the user mention the route of travelling and app will suggest user the places of their interests in the mentioned route. The Project will work on a real-world example of the dataset. Data will be scraped of reviews, pictures of different places, suggest users only places which are in their routes, and match their interests. The Project will take help from social media accounts and travelling websites to have proper reviews and pictures. The Project will contain an AI recommendation model [1] to suggest places; there will be extraction of user behavior and interests [2] according to their search history using supervised and unsupervised machine learning techniques. This way app can improve the overall experience of all sort of travelers whether he/she is a professional traveler and a random person the complete working of the proposed system is elaborated in Figure 1.

1.1 Problem statement

The trend of travelling is increasing day by day. So does the problem of choosing the right place to go. A Person who travels a lot have some prior knowledge about the place that will help better plan his travel schedule but in case of a person who does not travel a lot, the need of knowing the place, which match their interest, is very important. Otherwise, the time and resources of a person will waste. The reviews about a place can help a user had better understand that the place he is going to visit match his interest. The Person can see the photos of places and can read the reviews about it from different persons. Then the problem arises of finding reviews by going to different platforms that will also cost a lot of time.

Let us consider a user story to understand the problem more in depth. Let us say that a person is travelling from Lahore to Islamabad, He has one complete day to travel and he is travelling all be himself. As, he has enough time to visit places of his interests, but he does not know the places worth seeing in his route. He goes to social media accounts like Facebook, Twitter, Instagram and surfs different websites to know the places in the routes. In this whole process, a lot of time is being wasted. He may have to ask few questions on social media about some place. Sometime also used in this process now, you can see how much valuable time of him is wasted and he has very less time to explore the places.

Therefore, we should have a mechanism to address this problem. This is where our application comes in and tackle the problem. The proposed system will keep track of the user search history, will get user's interest from that, and will suggest him the places in his route that matches his interest. In this way user will more focus on travelling rather to worry about where to go.

1.2 Objective

The project's goal is to reduce the time wasted by user because of the time wasted in searching for the right place for him. For that purpose, the end goal of the project would be to make a software prototype that would manage user's travel using techniques and methods researched by us. The basic flow of the prototype would be something like:

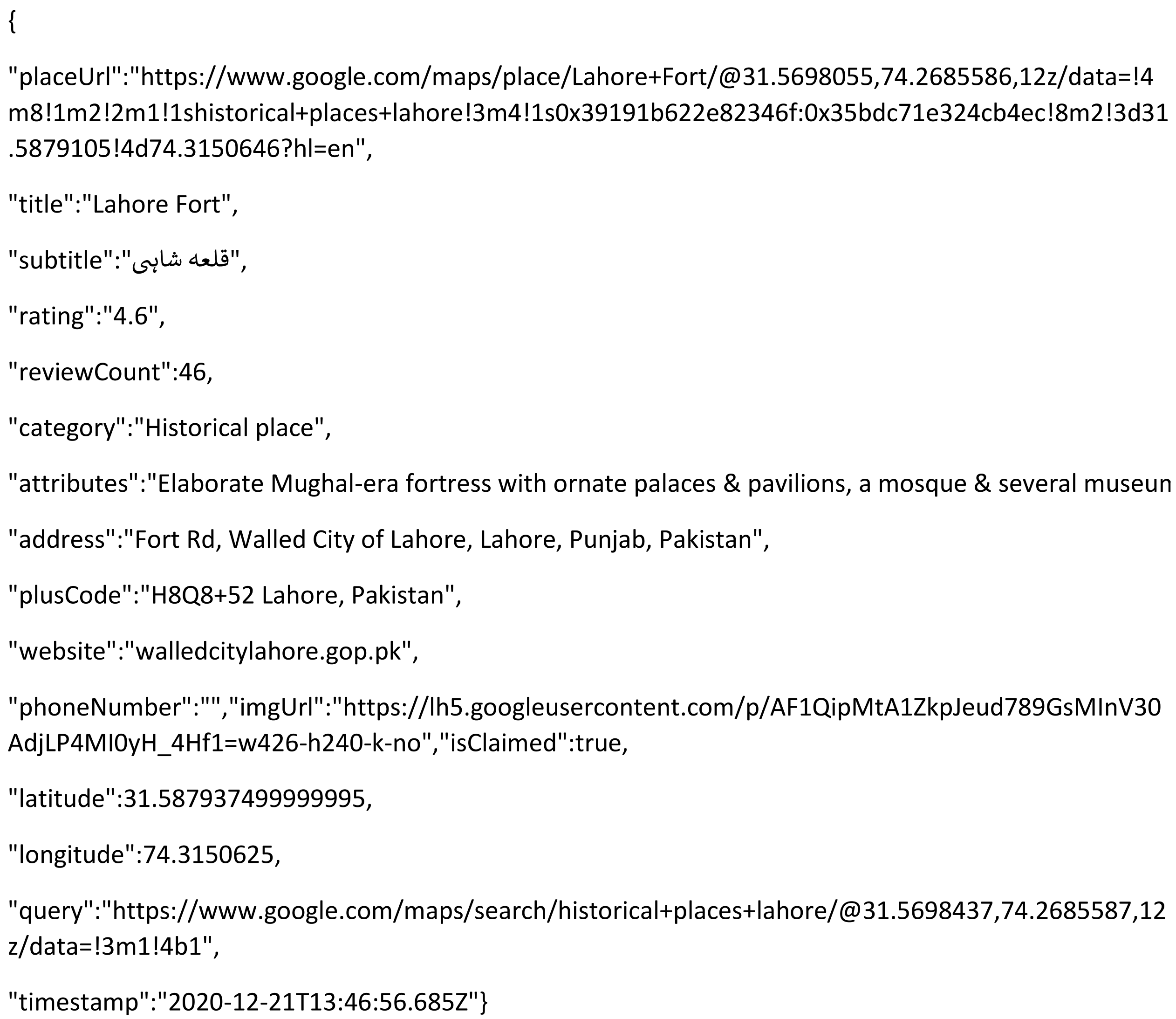
- Data pre-processing: -
  - Crawling dataset
  - Classifying dataset into Reviews and Pictures
  - Converting the Reviews and Pictures into model
- Taking an in-depth dive into already done and on-going research on this problem
- Extracting user behaviour according to its search history
- Designing an accurate AI recommendation models
- Developing our designed models
- Training and testing our dataset on our models
- Developing a mobile app of our project

1.3 Assumptions and Constraints

The made assumptions for the project and the constraints that bounds this project are listed below:

1.3.1 Assumptions

- Following are the assumptions for this project:
- Processing of text data will be alphabetical characters' form.
- The entered search location will be recorded as user's interest.
- The user's interest will then be used to train model more efficiently.
- If the user does not search for any particular location then users current coordinates will be used as a reference point to show interest based locations.
- All data will have the structure in Figure 1.



```
{

"placeUrl":"https://www.google.com/maps/place/Lahore+Fort/@31.5698055,74.2685586,12z/data=!4
m8!1m2!2m1!1shistorical+places+lahore!3m4!1s0x39191b622e82346f:0x35bdc71e324cb4ec!8m2!3d31
.5879105!4d74.3150646?hl=en",

"title":"Lahore Fort",

"subtitle":"قلعه شاہی",

"rating":"4.6",

"reviewCount":46,

"category":"Historical place",

"attributes":"Elaborate Mughal-era fortress with ornate palaces & pavilions, a mosque & several museun

"address":"Fort Rd, Walled City of Lahore, Lahore, Punjab, Pakistan",

"plusCode":"H8Q8+52 Lahore, Pakistan",

"website":"walledcitylahore.gop.pk",

"phoneNumber":"","imgUrl":"https://lh5.googleusercontent.com/p/AF1QipMtA1ZkpJeud789GsMInV30
AdjLP4MI0yH_4Hf1=w426-h240-k-no","isClaimed":true,

"latitude":31.587937499999995,

"longitude":74.3150625,

"query":"https://www.google.com/maps/search/historical+places+lahore/@31.5698437,74.2685587,12
z/data=!3m1!4b1",

"timestamp":"2020-12-21T13:46:56.685Z"}
```

Figure 1 - Dataset

Where

1. **placeUrl** – Url of the place
2. **title** – Title of place
3. **subtitle** – Name of the place in Urdu
4. **rating** – Rating of place
5. **reviewCount** – No. of reviews given
6. **Category** – Category of place
7. **Attributes** – Characteristics of place
8. **Address** – Address of place

9. **plusCode** – Code of place

10. **website** – Website of place

11. **phoneNumber** – Phone number of care taker

12. **imgUrl -** Image of place

13. **latitude –** Latitude of place

14. **longitude –** Longitude of place

15. **query –** Search query

16. **Timestamp –** Time of search

1.3.2 Constraints

The project has following constraints:

1. Duration for the accomplishment of the project is one year.
2. The efficiency and correctness of the prototype can vary.

1.4 Project Scope

The growth of travelling industry is increasing and so that the need to finding a good place to travel. For that purpose, a need of well-mannered environment is necessary where user does not have to worry about the reviews and whether or not the place if of his interests because the app will have the capability to automatically detect user's interests and suggest accordingly.

The project would be creation of cross platform mobile app that will be used in iOS and ANDROID. This is mainly because of the fact that majority of people prefer to use mobile apps instead of website. First we will scrap the data then on the basis of scrap data an AI model will be created that will suggest the user the places that match their interest. We will be using maps [3] and React native [4] platform for app creation. We will be using artificial intelligence [5] and machine learning [6] in the app. We believe people will use our system if they want to plan their route of traveling. It will boost the tourism potential of Pakistan and interest in traveling and tourism.

1.5 Chapter Summary

In this section, the introduction to the proposed system is provided, after that, the problem system is discussed in detail, which completely explain why the proposed system is being developed. Every system has some constraint and assumptions so does the proposed system has which are discussed in depth in that section. After that the scope of the system is discussed that is what is the system have what it will be trying to achieve.

2 Requirement Analysis

2.1 Literature Review

A literary review is being made on the work previously done to determine how much work has been done and what enhancement our work can make. Tourism has become a popular research topic recently. When the research started of how to create a travel app, the first problem that arises is how to get user's current location and then provide a travel guide to it. This problem was solved by GPS. Mobile solved many problems like finding exact place from the desired location, weather forecasting, user's location map. Many software have been created but researchers suggested that the specific needs of travelling are not being fulfilled. Keeping in mind, the problems many authors have proposed different apps to address these issues. One of them is **Roodtrippers [7]**, it can suggest user routes, pinpoint hotels and attractions but the app only allow mapping seven routes and no user's reviews are available about the places. Another app is **TripIt [8]**; it can provide flight confirmation, hotel reservation and tourist places.

Another app is **Skyscanner [9]**, it can provide suggestion according to user budget and travel timeframe but very few filters are available for places search and suggestions. Another app is **Waze [10]**, this app makes it easy for user to avoid congestion, blocked roads, police and accidents but biggest drawback of Waze is the ugly UI and cluttered layout of the maps. Another app is **CultureTrip [11]**, this app provides hyper-local information and recommendations near your location but user can only explore places in one city/area. Most research trends concentrate on travel patterns mining and travel patterns extraction. The project will try to cover the lacks and limitations of this previously done work and try to propose better solutions.

Analysis of Literature Review
As far as location based travel apps are concerned, many choices are available, but currently, there are no applications used in smart phones which uses interest based artificial intelligence techniques.

2.2 Stakeholders

The stakeholders are listed as follows:

- Researchers.
- Developers.
- Travelers.
- Hotel owners.
- Supervisors.

2.3 Requirements Elicitation

The assembling of requirements has the most critical and vital part in any lifecycle. Missing some requirements can have a key outcome on overall model; this is why any ambiguity should be dealt with before hand.
In this part, both functional and non-functional are listed below.

2.3.1 Functional Requirements

Functional requirements will describe main function in our system. Without these requirements, we cannot define our project properly

Priority → 1= High 2=Medium 3=Low

Section/ Requirement ID	Requirement	Description	Priority
T_R_1	Login	It will be first screen where user will enter his credential and login to system. User can also signup by social media (Facebook & Google). In case of wrong credentials, system will give error message.	1
T_R_1.1	Forget Password	If user forget his password this screen will help. User will enter his email here and his password will be sent to his email	1
T_R_01.2	Signup	On signup screen User will make his new account. User can make account by entering his email or he can sign up by social media (Facebook & Google)	2
T_R_2	Home Screen	In home screen user will see these option 1. Search bar on which user can search routes 2. Filter option on which user can filter his choice like (Hotels, places restaurants etc. 3. User will see nearby places according to his recommendation model	2
T_R_2.1	Search Bar	In search bar user will search routes. User will set starting location and destination location.	2

		System will give shortest route according to that location and mark all places in this route	
T_R_2.2	Filter Bar	In filter bar user can filter location according to his interest. After apply filter system suggest location according to filter	3
T_R_2.3	Neary by interested location	User will see all nearby interested location on his home screen. User can click any location to view it more clearly	2
T_R_3	Map screen	In map screen user will see map of all location User will see following option on map screen 1.Search bar 2.mark up places	2
T_R_3.1	Search Bar	In search bar user will search routes. User will set starting location and destination location. System will give shortest route according to that location and mark all places in this route	2
T_R_3.2	Markup places	System will markup all places that come in route	2
T_R_4	Profile screen	In profile screen user will see following option 1.Update profile picture 2.Update Email 3.Update password 4.Update First & last name 5. All visited places	3
T_R_4.1	Update profile picture	User can update his profile picture at any time	3
T_R_4.2	Update Email	User can update Email. A confirmation email sent to new email and then its verified	3
T_R_4.3	Update password	User can update password. To update password user, need to enter password two times	3

T_R_4.4	Update First & last name	User can update his first and last name	3
T_R_4.5	All visited places	In profile screen user will see all his visited places	3
T_R_5	Place Screen	In this screen user will see following information about place 1. Place full name and description 2. Places pictures 3. Reviews about place	2
T_R_5.1	Place full name and description	User will see place full name and its description	2
T_R_5.2	Places pictures	User will see all pictures of specific place	2
T_R_5.3	Reviews about place	User will see reviews about places. User will see all comments about place	2

Table 1 – Functional Requirements

2.3.2 Non-Functional Requirements

Usability:
Intelligent travel app is very user friendly. Any user can use to achieve his goal. User simply need to search route all interest location will be markup according to user interest.
Availability:
We will host our system on amazon ec2 it available all time. Our system will be available for ever user for 24 hours. Whenever user search route it gives result according to user recommended model.
Security & Privacy:
Security and privacy are key value for our system. We secured our system by number of ways. We will apply proxy to our IP address so no one can directly attack on system. In addition, we will apply JWT token in system to protect user data. We are performing most of operation on server side instead of client side for security reason.
Appearance:
Our frontend is attractive and user friendly for all types of users. All UI is self-explanatory. It usually promotes user confidence.
Scalability:
In our system, we consider all types of user. Any type of user can interact with it in user-friendly environment. All features are used in daily life activities.
Performance
Performance is key feature for your system. We are performing most of operation on server side to gives users best performance. Whenever user search, we give suggestion to user according to his trained model also we trained our model whenever user search in app. We have collected most of data of all major routes and trained our model so in this way we can give quick response.

Portability

Our system design for only smart phone (Android and iPhone).

Integrity

In case of ant failure system notified to admin and error will resolved in 10-15 minutes.

Capacity:

There is no limitation on number of user and not it does not have any limit on any features.

Adaptability

We only show relevant information on screen. Therefore, person belong to any field easily use it.

Usefulness:

We have designed system for any kind if user. Any traveler can use it even he has no technical knowledge. Traveler can see all interested location according to his surrounding and can explore new routes.

2.4 Use Case Design

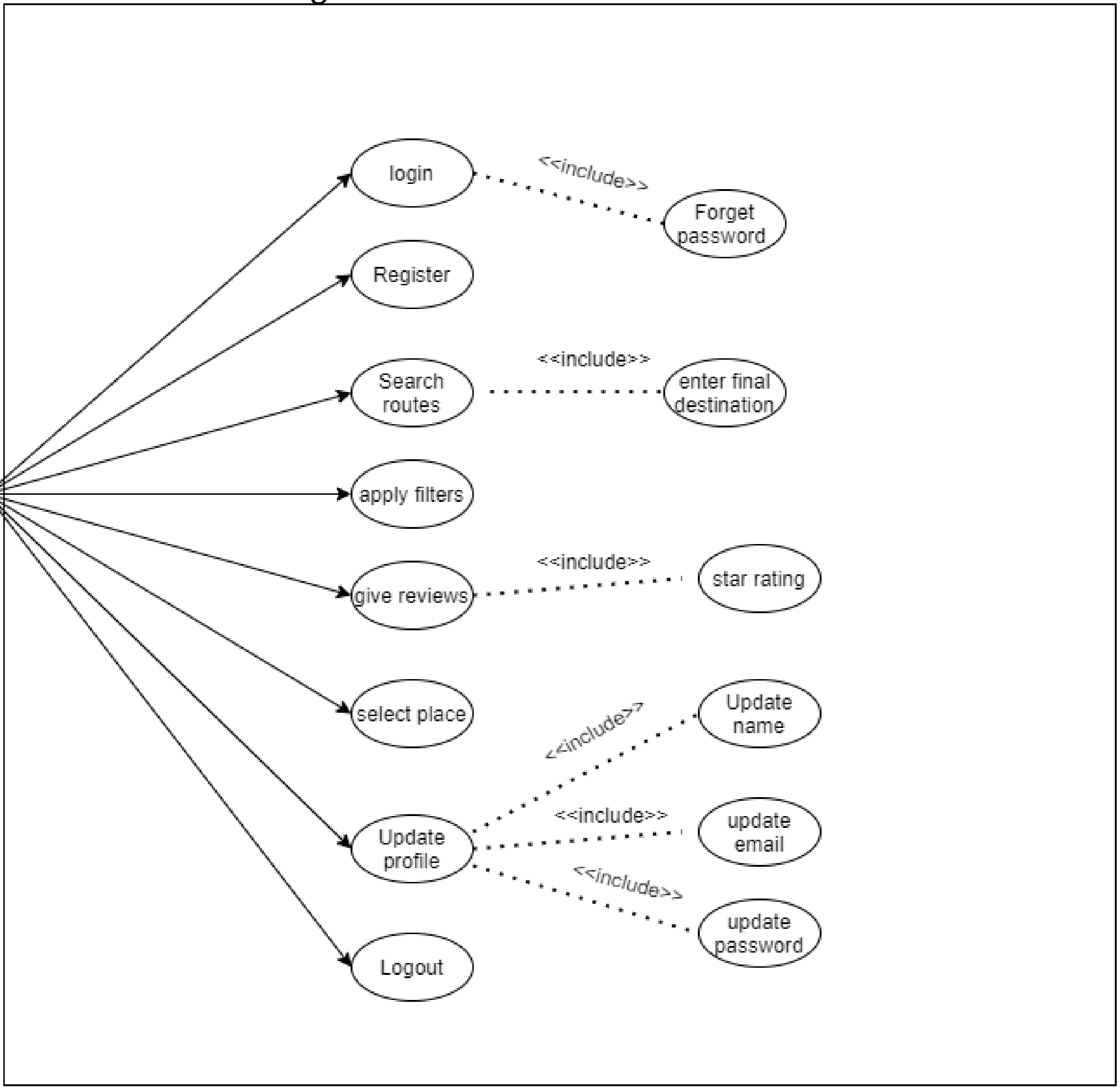


Figure 2 - Use-Case Diagram

2.4.1 Use case descriptions

User (Actor) can login to system by entering his credential. User can also click on forget password if mistakenly he forgets password. User can also search routes and enter his destination. User can also apply multiple filters to filter locations. User can also give reviews about places. In user review section, he can also comment or give star rating reviews. User can also update his profile in which he can update email, password, and his name. User can also logout.

2.5 Software Development Lifecycle Model

Software solution developing companies aspects many problems in selecting the suitable software development life cycle (**SDLC**). [9] Agile development is the utmost effective SDLC not just because of its many benefits over other SE models, but because it will aid us improve the project in future and update it whenever we want. This is because of speed of the change in the field of IT.

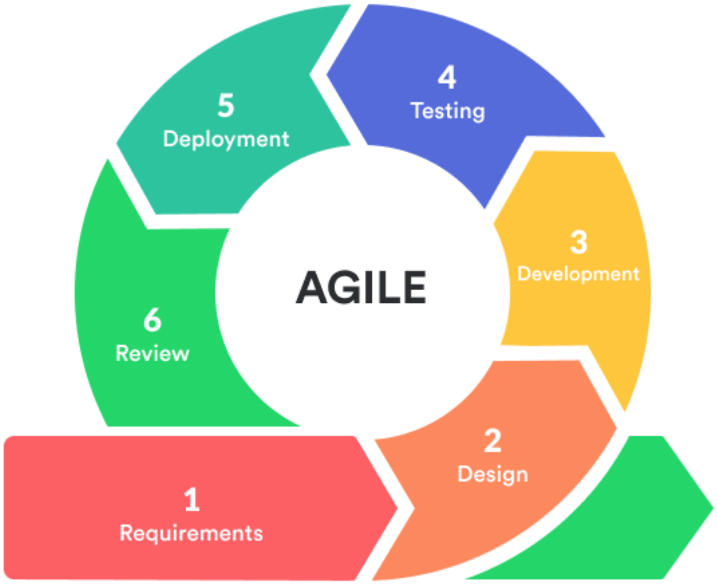


Figure 3 - Agile Lifecycle

2.5.1 Requirements

In this phase, all requirements will be assemble and planned. It is known as the most significant phase in SDLC. In the proposed project, the early requirement was to find large dataset of travelling places, which acquired from Google Maps and social media websites. Some preprocessing will be applied on dataset and a recommendation model to provide us with the desired results.

2.5.2 Plan

In this phase, we debate various methods and carry out a mechanism or plan to get the desired results. The proposed projects will use the dataset to achieve this; a recommendation model will be created to predict the desired location for the user. The study will crosscheck these behavioral features against the groups from the dataset that will provide desired location for the user.

2.5.3 Design and Develop

After the accomplishment of all the preliminary phases, this phase is to implement a prototype of the proposed system and we will be doing that using Python React native and different data sciences techniques.

2.5.4 Release

After completing the prototype, it will be prepare to test and once the testing is done, it will be released in the market.

2.5.5 Track & Monitor

In this phase the tracking and monitoring of the system is completed. The main aim of the system is to acquire with time and produce outputs that are more accurate. So, after the release it will be monitored for any real time errors, misclassifications or any other new modification, so that a new and improved version can be proposed.

2.6 Chapter Summary

In this chapter, the comprehensive discussion about requirement analysis phase is done. The chapter begin after the completion of chapter one at project's scope, In the start of this chapter we discussed around the literature that have been reviewed for this project which contains analyzing various algorithms and research methods. We have done a thorough discussion on dataset and creating a model based on that dataset's features. The list of stakeholders was written who might be fascinated in this research. Elicited requirements were also conversed in terms of functional and non-functional requirements. In the end, SDLC was propose with thorough discussion on its respective step-in relation to this project.

Now, after end of the requirement analysis this project will move to the next chapter i.e. System Design, where the design of our system will be discussed in detail.

3 System Design

After the accomplishment of requirement analysis, the next step is to start designing the system. Diagrams used to explain the system, containing WBS, Activity diagram, System Architecture, Network diagram.

3.1 Work Breakdown Structure

Figure 4 elaborates work breakdown structure for this project, as this is a R&D project so the project will mainly breakdown into two major phases:

- **Research**

The phase, in which study all previous domains new proposed works current or ongoing works all domain specific knowledge conducted on this topic to use all that knowledge in this project.

- **Development**

The phase, which contains the actual implementation and development of the proposed project.

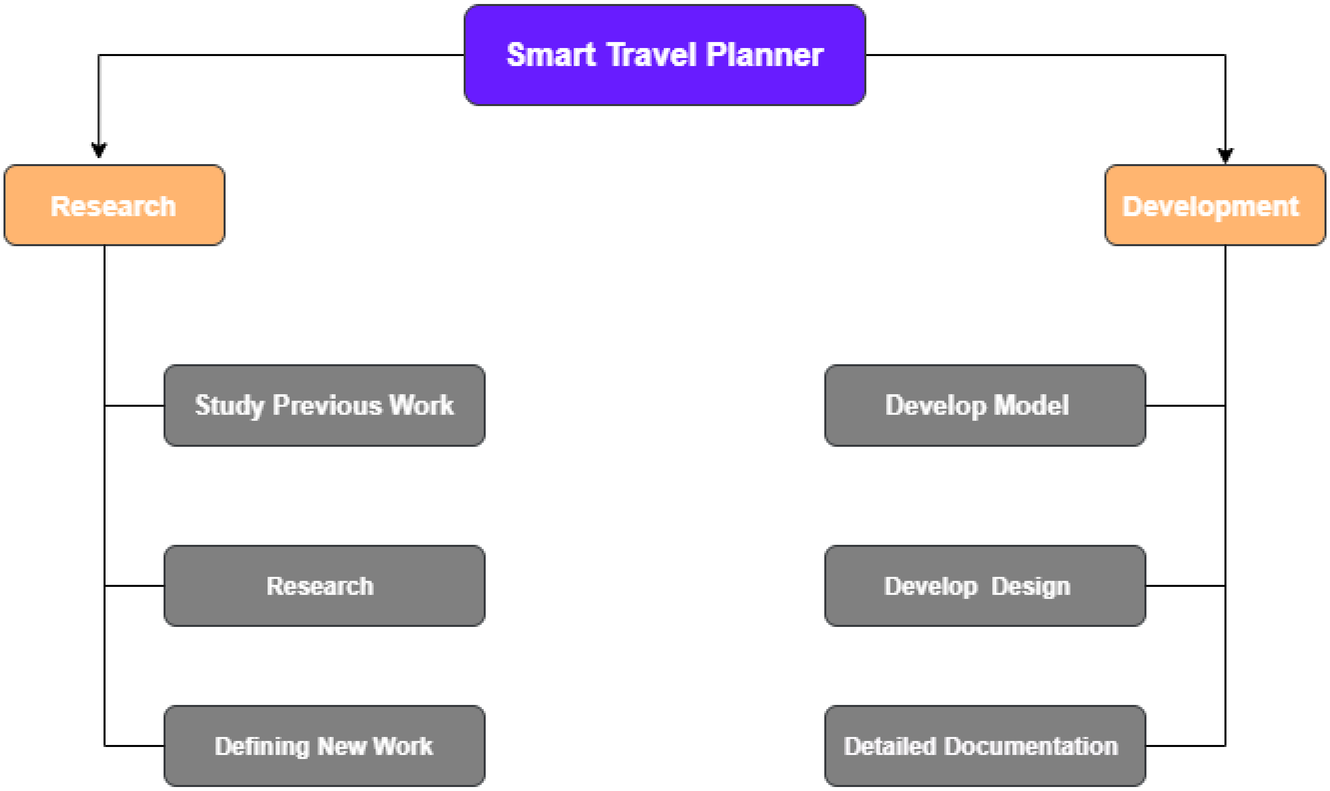


Figure 4 - Work Breakdown Structure

3.2 Model Architecture

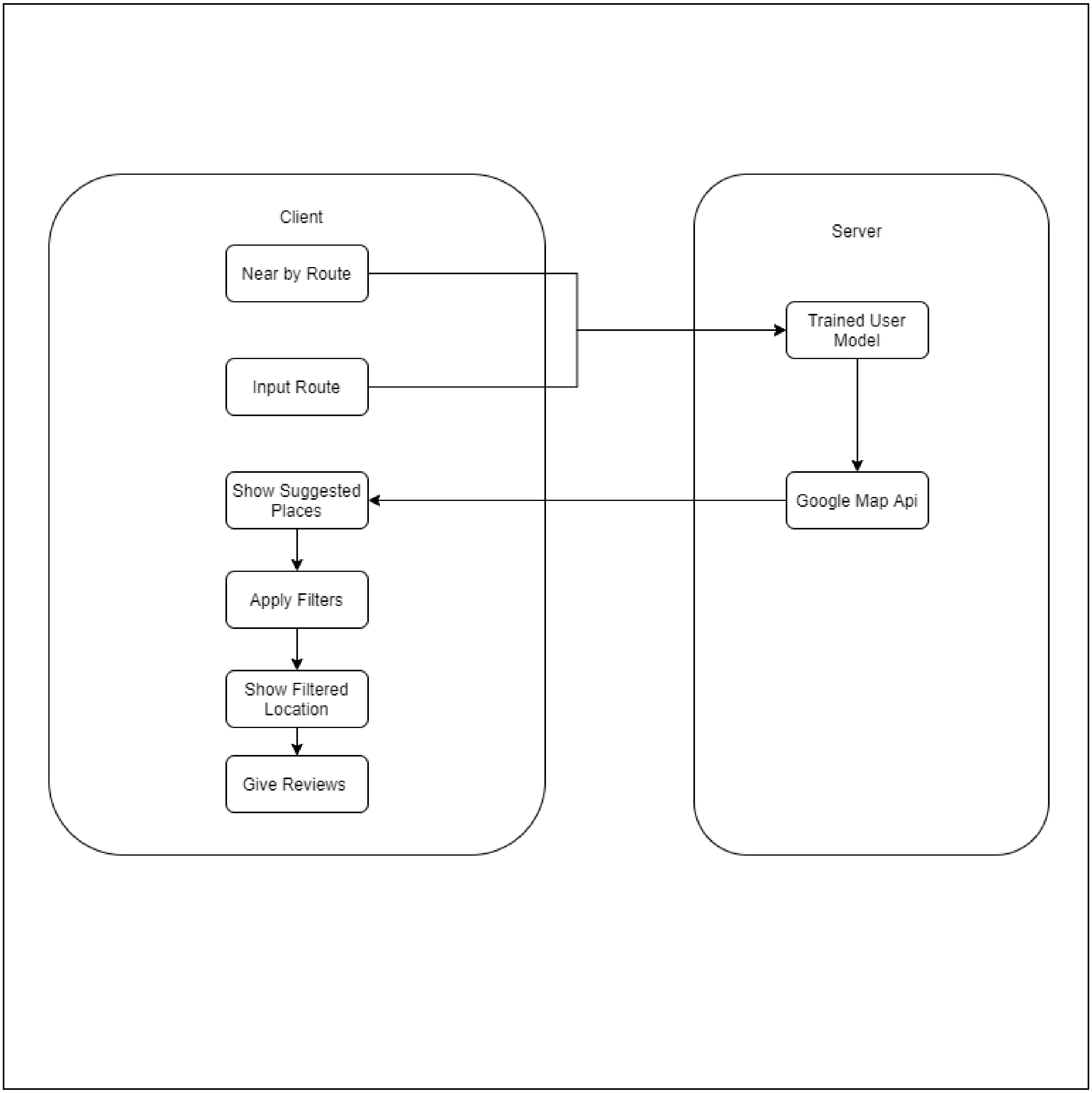


Figure 5 - Model Architecture

3.3 Activity Diagram

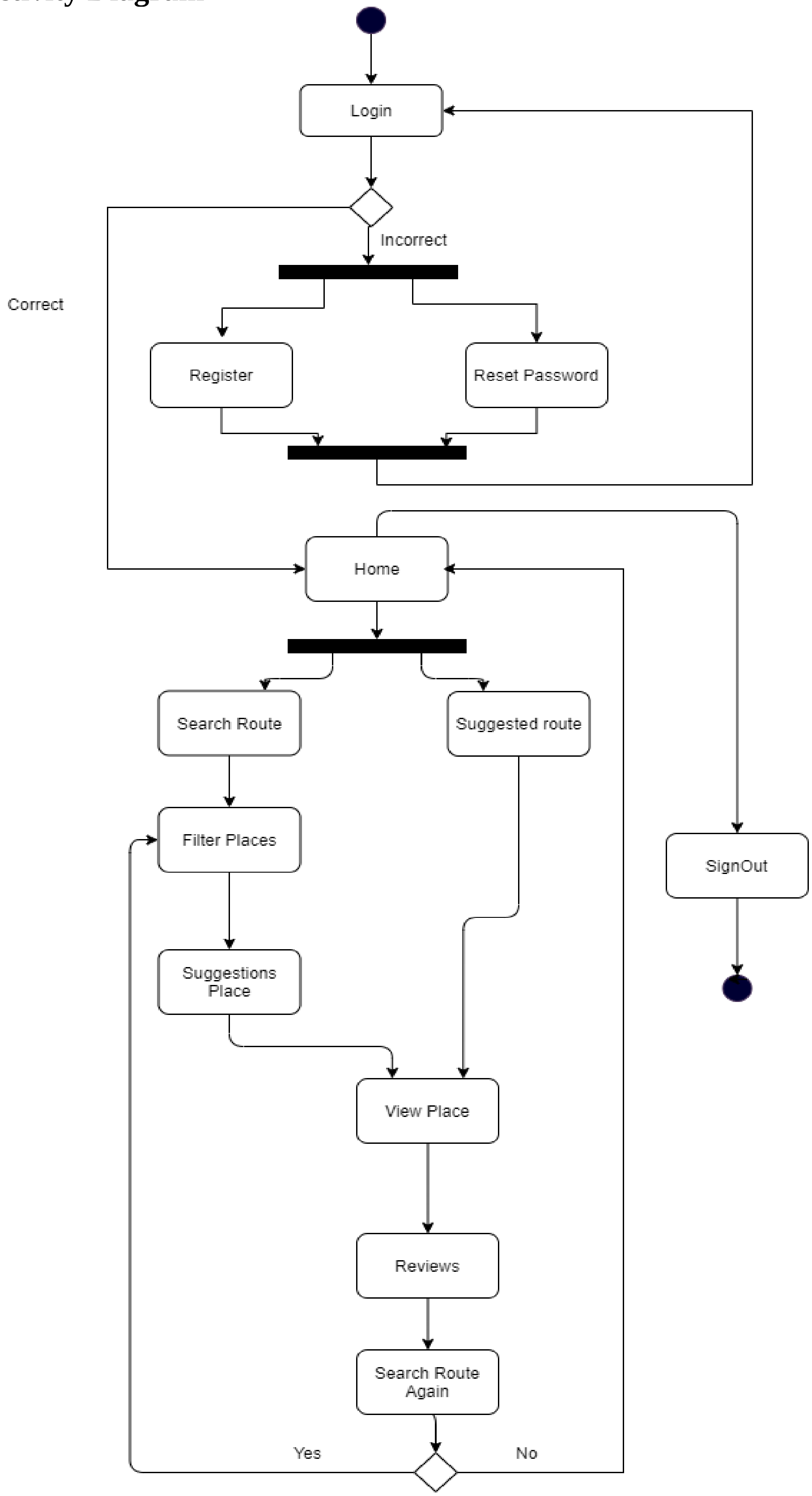


Figure 6 - Activity Diagram

The above diagram represents activities that will be performed to get to the preferred output of proposed system.

3.4 Class Diagram

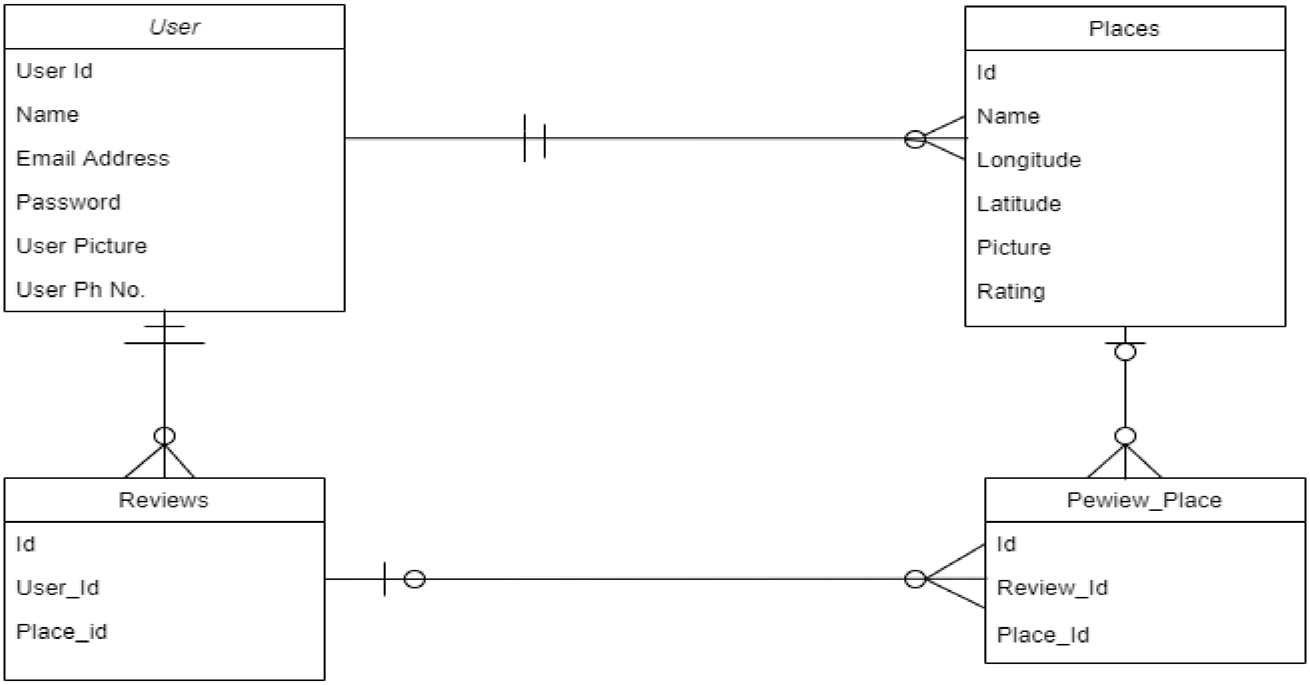


Figure 7 - Class Diagram

3.5 Collaboration Diagram

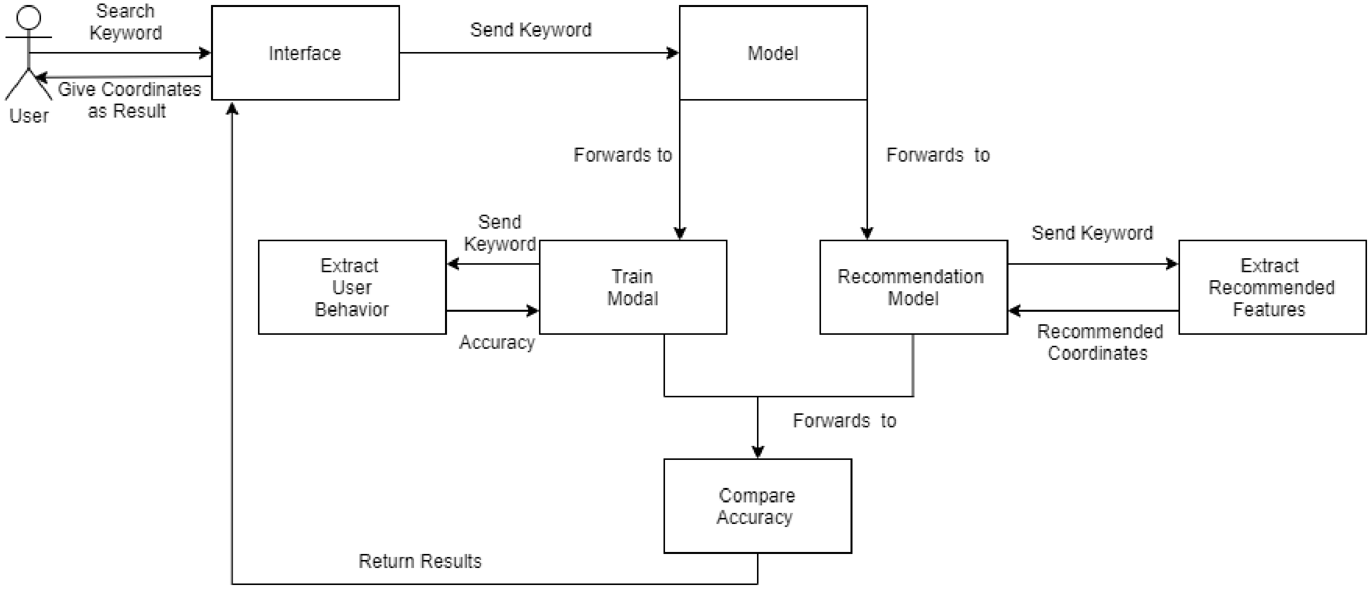


Figure 8 - Collaboration Diagram

3.6 Network Diagram

In network diagram, we have discussed about each phase's duration its start and ending date.

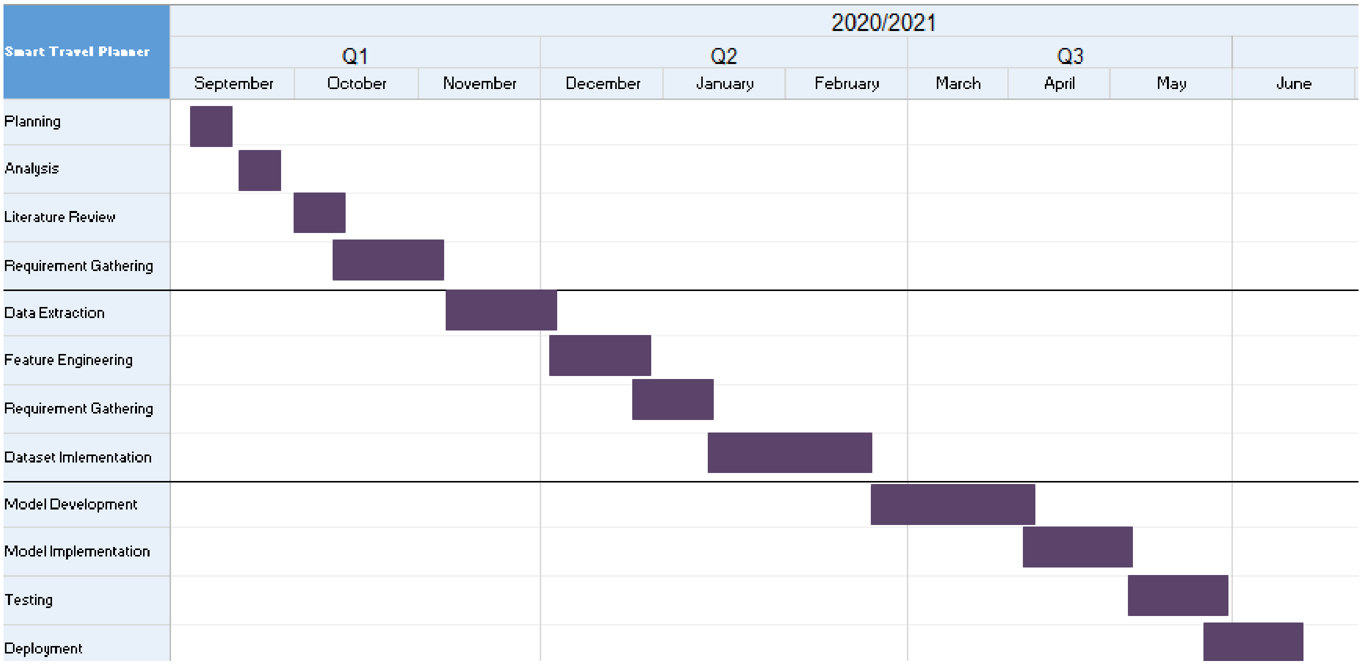


Figure 9 - Network Diagram

3.7 Chapter Summary

The chapter contains discussion on our proposed project's design in terms of diagrams, which will help understand the complex structure of the system and how the total workload is divided for the project. In this system, the proposed system complete infrastructure is also discussed i.e. what will be the work break down structure etc. This section also contain the activity diagram, which explain the overall activity that the system will be going to do. This section also contain the collaboration diagram, which explain how the system will be collaborate with other components. Network diagram is also presented to explain the overall task dividing of the project.

4 Implementation

4.1 User Interfaces

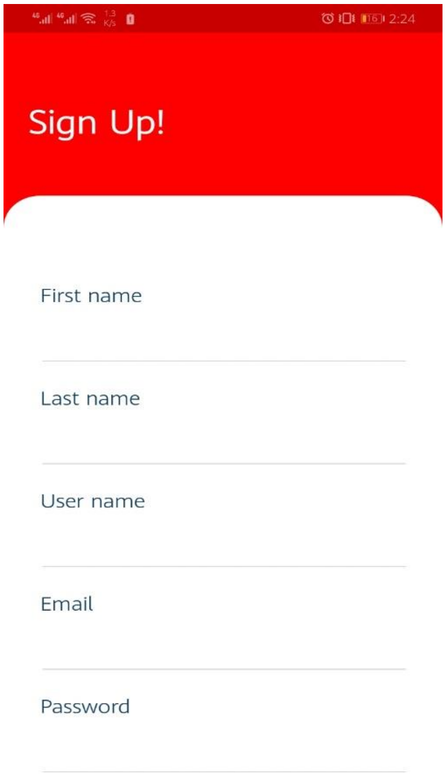


Figure 10 - Sign up Screen

This is the sign up screen of the project asking necessary details from user.

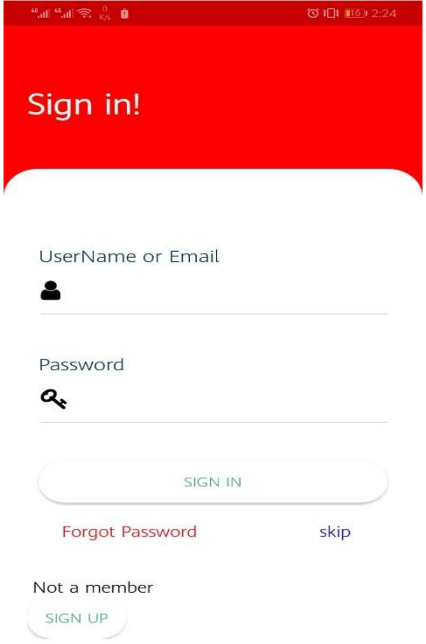


Figure 11 - Login Screen

This is the sign in screen of the project asking user’s username and password with forgot password functionality.

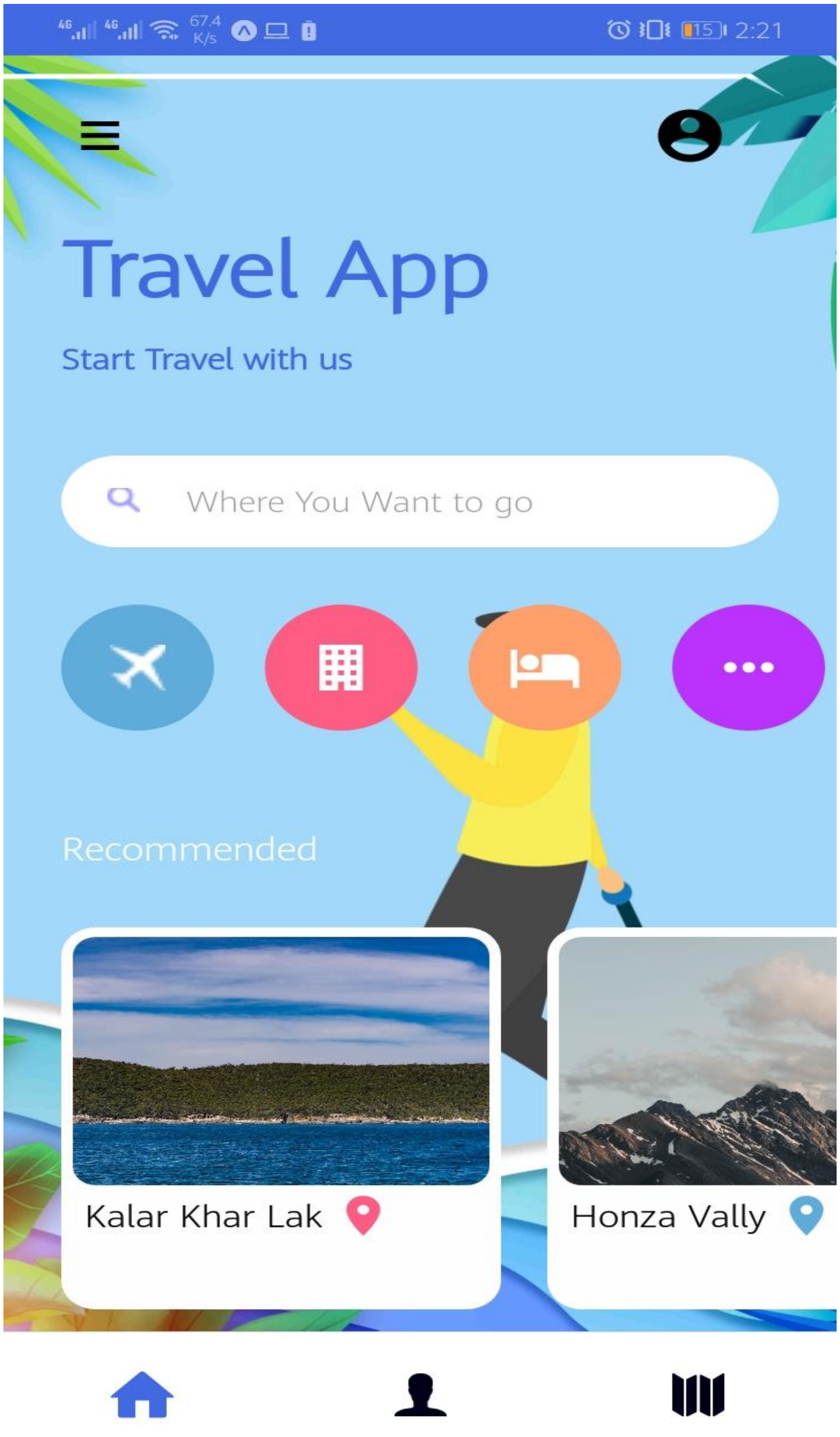


Figure 12 - Home Screen

This is the home screen of the app showing recommended locations. In addition, showing search bar to search for a particular locations.

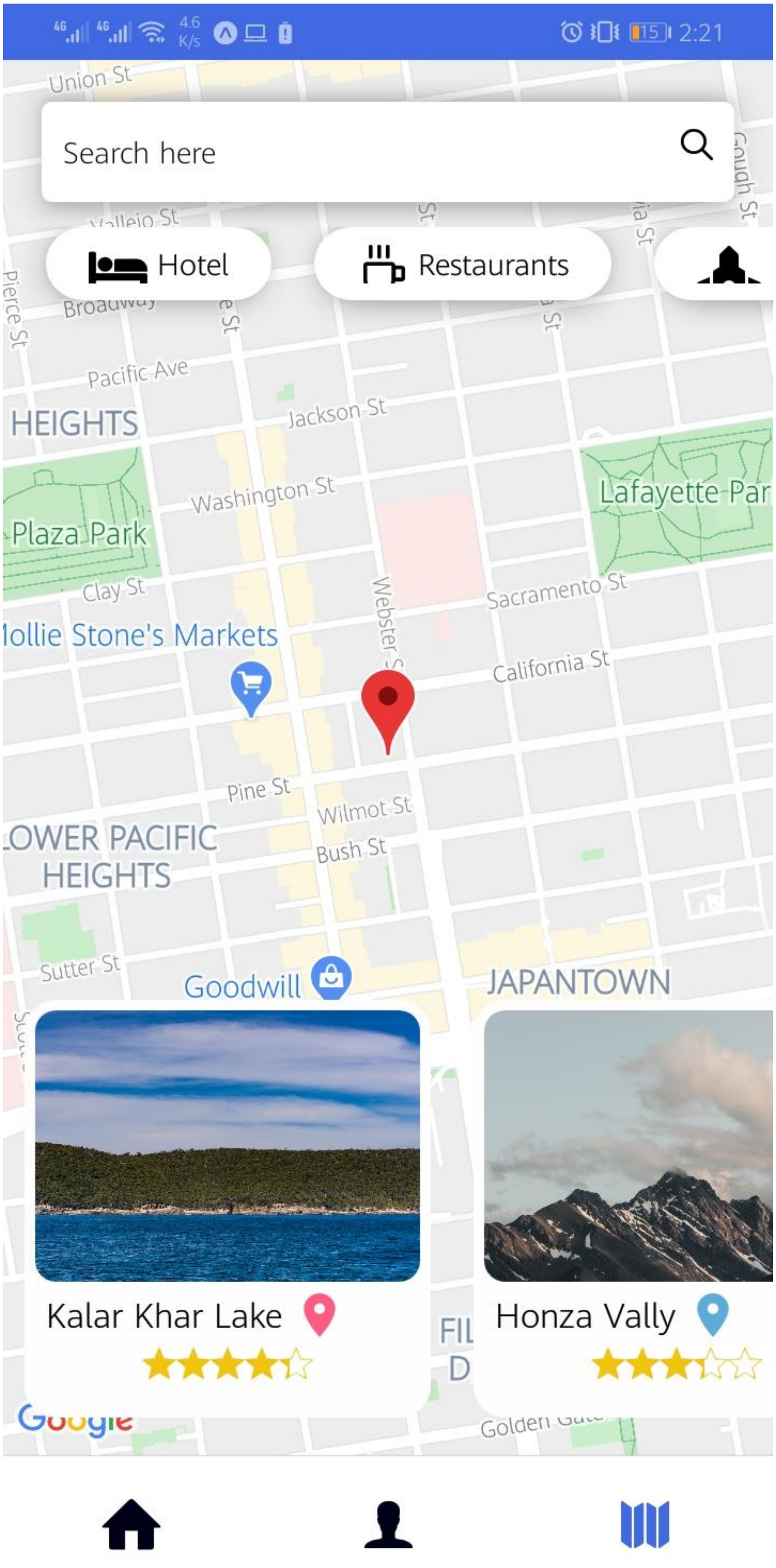


Figure 13 - Map Screen

When user start searching for a location, the location will be showing on the map and location that matches user’s interest.

4.2 Algorithms

In data pre-processing stage, first dataset image is scrapped from google with the help of Image_Scrapper algorithm, which is written in python:

```
import pandas
import time
import numpy as np
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.common.keys import Keys
outputarray1=[]
outputarray2=[]
outputarray3=[]
outputarray4=[]
outputarray5=[]
outputarray6=[]
array=[]
df = pandas.read_csv("Rest Houses.csv")
array=df["title"]
options = webdriver.ChromeOptions()
options.add_argument("--start-maximized")
options.add_experimental_option("excludeSwitches", ["enable-automation"])
options.add_experimental_option('useAutomationExtension', False)
browser = webdriver.Chrome(chrome_options=options, executable_path=r'chromedriver.exe')
browser.get('https://www.google.com/imghp?hl=EN')
for x in array:
    try:
        browser.execute_script("window.open('https://www.google.com/imghp?hl=EN', '_blank')")
        browser.switch_to.window(browser.window_handles[1])
        search = browser.find_element_by_name('q')
        search.send_keys(x)
        search.send_keys(Keys.RETURN) # hit return after you enter search text
        time.sleep(5) # sleep for 5 seconds so you can see the results
        outputarray1.append(browser.execute_script(" return document.getElementsByClassName('rg_i
Q4LuWd')[0].getAttribute('src')"))
       # df["pic1"] = outputarray1
        outputarray2.append(browser.execute_script(" return document.getElementsByClassName('rg_i
Q4LuWd')[1].getAttribute('src')"))
        outputarray3.append(browser.execute_script(" return document.getElementsByClassName('rg_i
Q4LuWd')[2].getAttribute('src')"))
        outputarray4.append(browser.execute_script(" return document.getElementsByClassName('rg_i
Q4LuWd')[3].getAttribute('src')"))
```

```
        outputarray5.append(browser.execute_script(" return document.getElementsByClassName('rg_i
Q4LuWd')[4].getAttribute('src')"))
        outputarray6.append("1")
        raw_data = {'pic1': outputarray1,
        'pic2':outputarray2,'pic3':outputarray3,'pic4':outputarray4,'pic5':outputarray5,'random':outputarray6 }
        df1 = pandas.DataFrame(raw_data, columns = ['pic1','pic2','pic3','pic4','pic5','random'])
        df1.to_csv('saeed1.csv')
        browser.close()
        browser.switch_to.window(browser.window_handles[0])
    except:
        browser.close()
        browser.switch_to.window(browser.window_handles[0])
        outputarray1.append("")
       # df["pic1"] = outputarray1
        outputarray2.append("")
        outputarray3.append("")
        outputarray4.append("")
        outputarray5.append("")
        outputarray6.append("1")
        raw_data = {'pic1': outputarray1,
        'pic2':outputarray2,'pic3':outputarray3,'pic4':outputarray4,'pic5':outputarray5,'random':outputarray6 }
        #df1 = pandas.DataFrame(raw_data, columns = ['pic1','pic2','pic3','pic4','pic5','random'])
        #df1.to_csv('saeed.csv')
        print("smeexcemtoon at")
        print(x)
df["pic1"] = outputarray1
df["pic2"] = outputarray2
df["pic3"] = outputarray3
df["pic4"] = outputarray4
df["pic5"] = outputarray5
df["random"] = outputarray6
df.to_csv('Rest Houses.csv')
##dict = {'pic1': outputarray1, 'pic2': outputarray2, 'pic3': outputarray3,'pic4': outputarray4,'pic5':
outputarray5}
##df = pandas.DataFrame(dict)\
```

After the images obtained, the images seems to be low contrast and some of them was noisy. So, we have use digital image processing techniques first we smooth the images using wiener filter to remove noise in the images. After that the contrast of the image is improved and at the end the image are sharpened. Below is the code of this algorithm named ImageProcessing, which is written in matlab.

```
LinkOfFolder='D:\Image Annotation\fypallrouteswithimage_2\Pictures\';
FolderToSave='D:\Image Annotation\fypallrouteswithimage_2\SaveImages\';
AllImages=dir(LinkOfFolder+"*.jpg");
noOfImages=length(AllImages);
% AllImages.name
for index=1:1:noOfImages
    image=AllImages(index).name;
    imageName=split(image,'.');
    A=imread(strcat( LinkOfFolder,image));
    theVariance = var(double(A(:)));

    [R,C]=size(A);
    A2=A;
    filter=3;

    for i=ceil(filter/2):1:R-floor(filter/2)
        for j=ceil(filter/2):1:C-floor(filter/2)
            temp=A(i-floor(filter/2):i+floor(filter/2),j-
floor(filter/2):j+floor(filter/2));
            localVar=var(double(temp(:)));

            sumValue=sum(temp(:));
            sumValue1=sumValue/(filter^2);

             leftPart=(localVar^2-theVariance^2)/(localVar^2);
            A2(i,j)=A(i,j)-((leftPart)*(A(i,j)-sumValue1));

        end
    end
    if size(A2,3)==3
        LAB = rgb2lab(A2);
        L = LAB(:,:,1)/100;
        L = adapthisteq(L,'NumTiles',[8 8],'ClipLimit',0.005);
        LAB(:,:,1) = L*100;
        A2 = lab2rgb(LAB);

    else
        A2 = adapthisteq(A2,'clipLimit',0.02,'Distribution','rayleigh');
    end

%     A2=adapthisteq(A2);
    LaplacianMask3=[-1 -1 -1;-1 8 -1;-1 -1 -1];
    ApplyingFilter(A2,LaplacianMask3,image,FolderToSave);


end
        function ApplyingFilter(RgbImage,LaplacianMask,ImageName,FolderToSave)
                filterValue=3;
                [R,C]=size(RgbImage);
                A1=RgbImage;

                for i=ceil(filterValue/2):1:R-floor(filterValue / 2)
                            for j=ceil(filterValue/2):1:C-floor(filterValue / 2)
                                temp=RgbImage(i-floor(filterValue /
2):i+floor(filterValue / 2),j-floor(filterValue / 2):j+floor(filterValue / 2));
                                temp=double(temp).*(LaplacianMask);
                                sumValue=sum(sum(temp));
                                sumValue1=sumValue/(filterValue ^ 2);
                                A1(i,j)=sumValue1;
                            end
                end
                ResultantImage=RgbImage+A1;
                fullFileName = fullfile(FolderToSave, ImageName);
                imwrite(ResultantImage, fullFileName);
        end
```

At the end the image are passed to convolutional neural network to assign a particular class to each image. Below is the algorithm named ImageLabelGenerator:

```
import matplotlib.pyplot as plt
import matplotlib.image as mpimg
plt.style.use('dark_background')
from keras.models import Sequential
from keras.preprocessing.image import ImageDataGenerator
from keras.layers import BatchNormalization
from keras.layers import Conv2D, MaxPooling2D, Dense, Flatten,Activation,Dropout
from keras.preprocessing import image
from keras.utils import normalize, to_categorical
from keras import backend as K
import numpy as npimg_width,img_height=150,150
!unrar x "/content/Categoris Images.rar" "/content/drive/path/output_folder/"
!unrar x "/content/Validation Set.rar" "/content/drive/path/output_folder1/"
train_data_dir='/content/drive/path/output_folder/Categoris Images'
validation_data_dir='/content/drive/path/output_folder1/Validation Set'
nb_train_samples=900
nb_validation_samples=90
ephocs=11
batch_size=20
if K.image_data_format()=='channels_first':
  input_shape=(3,img_width,img_height)
else:
  input_shape=(img_width,img_height,3)

train_datagen=ImageDataGenerator(
    rescale=1./255,
    shear_range=0.2,
    zoom_range=0.2,
    horizontal_flip=True
)

test_datagen=ImageDataGenerator(rescale=1./255)

train_generator=train_datagen.flow_from_directory(
    train_data_dir,
    target_size=(img_width,img_height),
    batch_size=batch_size,

)

validation_generator=test_datagen.flow_from_directory(
    validation_data_dir,
    target_size=(img_width,img_height),
    batch_size=batch_size,

)

activation='sigmoid'

model = Sequential()
model.add(Conv2D(32, 3, activation = activation, padding = 'same', input_shape = input_shape))
model.add(BatchNormalization())

model.add(Conv2D(32, 3, activation = activation, padding = 'same', kernel_initializer = 'he_uniform'))
model.add(BatchNormalization())
model.add(MaxPooling2D())

model.add(Conv2D(64, 3, activation = activation, padding = 'same', kernel_initializer = 'he_uniform'))
model.add(BatchNormalization())

model.add(Conv2D(64, 3, activation = activation, padding = 'same', kernel_initializer = 'he_uniform'))
model.add(BatchNormalization())
model.add(MaxPooling2D())

model.add(Flatten())
model.add(Dense(128, activation = activation, kernel_initializer = 'he_uniform'))
model.add(Dense(11, activation = 'softmax'))

model.compile(optimizer = 'rmsprop',loss = 'categorical_crossentropy', metrics = ['accuracy'])
print(model.summary())

FitModel = model.fit_generator(
        train_generator,
        steps_per_epoch = nb_train_samples // batch_size,
        epochs = 30,
        validation_data=validation_generator,
        validation_steps=nb_validation_samples // batch_size
)

img_predict=image.load_img('/content/download (2).jpg',target_size=(150,150))
img_predict=image.img_to_array(img_predict)
img_predict=np.expand_dims(img_predict,axis=0)
print(train_generator.class_indices)
rslt=model.predict_classes(img_predict)
print(rslt)

model.save_weights('/content/second_try.h5')
```

4.3 Chapter Summary

This chapter overall demonstrate the implementation of proposed project. Several screens of the projects are presented. Along with some algorithms that are used to perform some vital tasks in the project up until now. Based on these algorithms the logic building of the project is done. These algorithms will also be helpful in future activities.

5 System Testing

This chapter contains the test cases for our system along with the unit and acceptance testing.

5.1 Test Cases

Rule #	Test Cases	Steps	Data	Expected Result
1	Input Routes	Go to Model	Processing	Submitted
2	Output		Result	Coordinates of locations

Table 2 – Test Cases

5.2 Unit/ Acceptance Testing

The proposed project will contain a single module that accepts a input routes and that will routes will go to the model and then model will do processing on it provide an output result after performing all analysis on that input. The provide output will be suggested routes and locations to the user.

5.3 Chapter Summary

Testing is an important part of every system. The testing will evaluate the project that how capable it is to solve extensive use cases. In this chapter, unit and system testing will be performed on the proposed project to check its accuracy and compute results to improve logic and performance of the system.

6 Conclusion

6.1 Problems faced and Lessons Learned

Group projects works just well for R&D projects and produces effective outcomes but it is not easy to chain all the ideas in a single project. Outlining the tasks of the group members and synchronizing with the supervisors on the given time is very vital and one thing that is learned here is to work good and more in fewer time.

6.2 Project Summary

Travelling industry is progressing rapidly in this modern era. However, the knowledge to travel where and enjoy their valuable time is very less still. This research aids eliminate this issue. This project spots user's interests using user's search history with high accuracy and then suggesting places and travel routes to user. It does this process using artificial intelligence, computer vision and data mining techniques.

Firstly, a large dataset is created by mining data of travel locations from google and different websites. Then digital image processing is applied to the pictures of the dataset to obtain better quality images and computer vision algorithm to assign a particular label to all the images. That completes the data preprocessing stage of the project. Then that dataset is used to generate an artificial intelligence recommendation model using collaborative and content-based filtering. After that the frontend and backend of the project is created in react native and Node.js to produce a cross platform mobile app to provide an interface to the user.

6.3 Future Work

Currently, the proposed project is restricted by its scope. However, our study can be supportive to researchers for more work and it can be used to shape a well-organized prototype. On the basis of dataset a model will be created that will predict real time locations on map to the user. User will be able to give reviews and according to that its search interest will be updated.

7 References

[1] Roger Chua. (2019, June). A simple way to explain the Recommendation Engine in AI (https://medium.com/voice-tech-podcast/a-simple-way-to-explain-the-recommendation-engine-in-ai-d1a609f59d97)

[2] Zhabiz Gharibshah(2020, Jan). Deep Learning for User Interest and Response Prediction Online Display Advertising (https://link.springer.com/article/10.1007/s41019-019-00115-y)

[3] Shamil. (2019, April). Integrating Google Maps into React Native App on Android (https://medium.com/@samil.mehdiyev/integrating-google-maps-into-react-native-app-on-android-40c984c0e4f2)

[4] Zhabiz Gharibshah (2020, June). React Native at Airbnb: The Technology (https://medium.com/airbnb-engineering/react-native-at-airbnb-the-technology-dafd0b43838)

[5] Povel Kordík. (2018, June). Machine Learning for Recommender systems — Part 1 (algorithms, evaluation and cold start) (https://medium.com/recombee-blog/machine-learning-for-recommender-systems-part-1-algorithms-evaluation-and-cold-start-6f696683d0ed)
(Last Visited 17/09/2020)

[6] Rico Meinl. (2011, May). Recommender Systems: The Most Valuable Application of Machine Learning (Part 1)(https://towardsdatascience.com/recommender-systems-the-most-valuable-application-of-machine-learning-part-1-f96ecbc4b7f5)

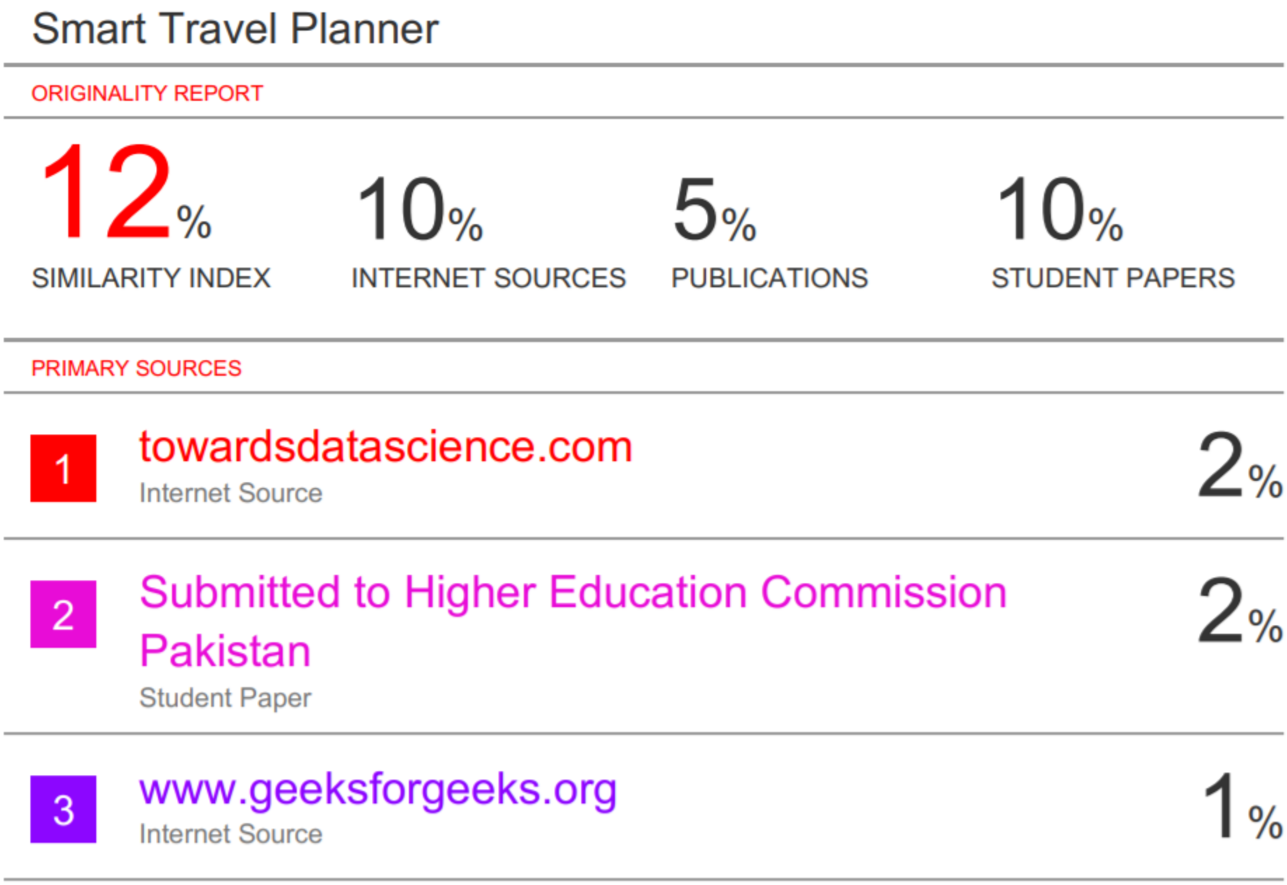
[7] Roadtrippers (2011, August) Roadtrippers is a web based software application, mobile app, and content provider that helps travelers plan road trips (https://roadtrippers.com/)

[8] Tripit (2006, October), checking one place for all your travel details and getting a heads up as things happen (https://www.tripit.com/web)

[9] Skyscanner (2001, March), Find cheap flights, hotels & car hire from over 1200 travel providers. Plan with peace of mind – book flexible flight tickets & hotels with free cancellation. (https://www.skyscanner.net/)

[10] Waze (2016, June), It provides turn-by-turn navigation information and user-submitted travel times and route details, while downloading location-dependent information over a mobile telephone network (https://www.waze.com/)

[11] Culture Trip app (2011, May), makes travel planning easy and fun with award-winning travel stories, insider city guides, and handpicked hotels in the coolest neighbourhoods. (https://theculturetrip.com/)



Smart Travel Planner

ORIGINALITY REPORT

12%	10%	5%	10%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	towardsdatascience.com Internet Source	2%
2	Submitted to Higher Education Commission Pakistan Student Paper	2%
3	www.geeksforgeeks.org Internet Source	1%