



# Visit Bangladesh

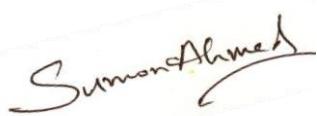
Report on  
**Visit Bangladesh**  
**An Android App**

**Course: SE 505**

Submitted by

**S.M. Shahriyar (BSSE 0902)**

**Sudip Kumar Sah (BSSE 1040)**



Supervised by

**Dr. Sumon Ahmed**

**Associate Professor**



Institute of Information Technology,  
University of Dhaka  
**Date: 26/08/2021**

# Group Member Description

## **Member 1:**

**Name:** Sudip Kumar Sah

**Roll:** BSSE-1040

**Registration Number:** 2016-718-933

**Contact:** 01323279848

**Email:** [bsse0934@iit.du.ac.bd](mailto:bsse0934@iit.du.ac.bd)

## **Member 2:**

**Name:** S M Shahriyar

**Roll:** BSSE-1044

**Registration Number:** 2016-115-031

**Contact:** 01521432437

**Email:** [bsse0902@iit.du.ac.bd](mailto:bsse0902@iit.du.ac.bd)

## Acknowledgement

We are highly indebted for getting such a tremendous opportunity to prepare the report on Tour guide application. We would like to thank whole-heartedly our supervisor Dr. Sumon Ahmed, Associate Professor, Institute of Information Technology, University of Dhaka, for giving us guidelines about how we can prepare this report. We are also thankful of those people who gave to do this job.

## **Abstract**

At present, we are living in a world where technology rules almost every place and every person. Internet offers us to grab anything from any corner of the world now. In almost twinkling of an eye, people enjoy a wide range of facilities given by internet through a small smartphone, tab or laptop. Following the stream, we have developed a Visit Bangladesh application for android smartphones. When using this app, a user will be able to overview and select better for their tours. This will help user to search division wise places of Bangladesh.

# Table of Contents

Acknowledgement .....	4
Abstract .....	5
Table of Figures .....	8
INTRODUCTION .....	12
Background Study .....	13
Methodology .....	14
Software Requirement Specification Document.....	15
3.1 Purpose.....	15
3.1.2 Intended Audience .....	15
3.2 Inception .....	16
3.2.1 Introduction.....	16
3.2.2 Conclusion .....	17
3.3 Elicitation.....	18
3.3.1 Introduction.....	18
3.3.2 Eliciting requirements.....	18
3.3.3 Requirements gathering .....	19
3.3.4 Quality function deployment.....	19
3.3.5 User scenario .....	21
3.3.6 Elicitation work product .....	22
3.4 Scenario Based Model .....	23
3.4.1 Introduction.....	23
3.4.2 Use case diagram .....	23
3.4.3 Use Case Scenario .....	24
3.5 Data Based Model.....	48
3.5.1 Introduction.....	48
3.5.2 Data object .....	48
3.6 Class Based Model.....	56

3.6.1 Grammatical Parsing and Analysis .....	56
3.6.2 Attributes and Methods of Potential Classes.....	61
3.6.3 Class Responsibility Collaborator .....	68
3.7 DATA FLOW MODEL.....	73
7.1.0: level 0.....	73
7.1.1: Level 1.1 .....	74
7.1.2: Level 1.2 .....	74
7.3.1: Level 2.1 .....	75
7.3.2 Level 2.2 .....	75
7.3.3 Level 2.3 .....	76
7.3.4 Level 2.4 .....	76
7.3.5 Level 2.5 .....	77
3.8 BEHAVIORAL MODEL .....	78
3.8.1 State diagram of Account .....	78
3.8.2 State diagram of Admin.....	79
3.8.3 State diagram of Notification .....	79
3.8.4 State diagram of User: .....	80
3.8.5 State diagram of System .....	81
3.8.6 State diagram of Database .....	81
3.8.7 State diagram of Search .....	82
3.8.8 State diagram of Location.....	82
3.9 Sequence diagram .....	83
3.9.1 Sequence diagram of Sign Up .....	83
3.9.1 Sequence diagram of Sign In.....	84
3.9.1 Sequence diagram of Search Location .....	85
3.9.1 Sequence diagram of Edit Location.....	85
3.9.1 Sequence diagram of Delete Location.....	86
8.2.6 Sequence diagram of Browse Location .....	87
3.9.7 Sequence diagram of Add Location .....	88

Designs.....	89
Activities .....	89
<b>USER MANUAL.....</b>	<b>92</b>
4.1 Login	92
4.2 My Account .....	94
4.3 Weather Update .....	95
4.4 Currency Converter.....	96
4.5 Adding new tourist places.....	97
4.5.1 Google Maps.....	98
4.5.2 Upload Images .....	99
4.6 Details Of Places .....	100
4.6.1 Route Details.....	101
4.6.2 Guide Details .....	102
4.6.3 Police Station Details.....	103
4.6.4 Airplane icon .....	104
4.7 Search Places	105
4.8 Emergency Contact.....	106
4.9 show newly added place .....	107
Conclusion, Implementation and Limitation .....	108
5.1 Achievements	108
5.2 Obstacles	108
5.3 Future Plan .....	108
5.4 Limitations .....	109
References .....	109

# Table of Figures

Figure 1:Level 0 Use Case Diagram	25
Figure2:Level 1 Use Case Diagram	26
Figure 3: Level 1.1 Use Case Diagram of Authentication Subsystem	27
Figure 4: Level 1.1.1 Use Case Diagram of Authentication Subsystem	28
Figure 5: Level 1.2 Use Case Diagram of Browse Subsystem	29
Figure 6: Level 1.3 Use Case Diagram of Update Subsystem	30
Figure 7: Level 1.4 Use Case Diagram of Search Subsystem	31
Figure 8: Level 1.5 Use Case Diagram of Setting Subsystem	32
Figure 9: Level 1.5.1 Use Case Diagram of Subsystem's task edit account	33
Figure 10: Level 1.6 Use Case Diagram of Emergency Contact Subsystem	34
Figure 11: Level 4.3.1 Activity Diagram of Sign Up	35
Figure 12: Level 4.3.2 Activity Diagram of Log In	36
Figure 13: Level 4.3.2 Activity Diagram of Guest user	37
Figure 14: Level 4.3.3 Activity Diagram of Browse	38
Figure 15: Level 4.3.4 Activity Diagram of Edit Contact	39
Figure 16: Level 4.4.1 Swim-lane Diagram of Register	41
Figure 17: Level 4.4.2 Swim-lane Diagram of Log In	42
Figure 18: Level 4.4.3 Swim-lane Diagram of Guest user	43
Figure 19: Level 4.4.4 Swim-lane Diagram of Browse	44
Figure 20: Level 4.4.5 Swim-lane Diagram of Update	45
Figure 21: Level 4.4.5 Swim-lane Diagram of Edit contact	46
Figure 22: Level 4.4.6 Swim-lane Diagram of emergency contact	47
Figure 23: Relationship among Data Objects	52
Figure 24: Entity Relationship Diagram	53
Figure 25: level 0 for Visit Bangladesh	73
Figure 26: level 1.1 for Visit Bangladesh	74
Figure 27: level 1.2 for Visit Bangladesh	74
Figure 28: level 2.1. for Visit Bangladesh	75

Figure 29: level 2.2 for Visit Bangladesh	75
Figure 30: level 2.3 for Visit Bangladesh	76
Figure 31: level 2.4 for Visit Bangladesh	76
Figure 32: level 2.5 for Visit Bangladesh	77
Figure 33:State diagram of Account	78
Figure 34:State diagram of Admin:	79
Figure 35: State diagram of notification	79
Figure 36:State diagram of User	80
Figure 37:State diagram of System	81
Figure 38:State diagram of Database	81
Figure 39:State diagram of Search	82
Figure 40:State diagram of Location	82
Figure 41:Sequence diagram of Sign Up	83
Figure 42:Sequence diagram of Sign In	84
Figure 43:Sequence diagram of Search Location	85
Figure 44:Sequence diagram of Edit Location	85
Figure 45:Sequence diagram of Delete Location:	86
Figure 46:Sequence diagram of Browse Location	87
Figure 47:Sequence diagram of Add Location	88
Figure 48:Splash and Sign In	92
Figure 49:Home Page	93
Figure 50:User-Account Info	94
Figure 51: Shows weather	95
Figure 52:Currency Converter Interface	96
Figure 53: Form Adding new places	97
Figure 54:Maps Navigation	98
Figure 55: Uploading Images	99
Figure 56:Details Of Places	100
Figure 57:Three means of Transportation	101

Figure 58:Information of Tour Guide	102
Figure 59:Local Police Station Details	103
Figure 60: Navigate to Selected Tourist place	104
Figure 61: Search by Placename, District and Division	105
Figure 62:Emergency Contact	106

# INTRODUCTION

This project is a tourist guide app named as “VISIT BANGLADESH” app can be used to visit and know about the various tourist of places of Bangladesh.

The project “Visit Bangladesh” is a tour guide mobile application developed to help domestic and foreign tourists in Bangladesh. With the developing world, people want to get everything to their fingertips. Various tours and travel agencies have defined expensive package which most of people can't afford. Sometimes, they might not have valid information and keep tourists in vain. Lack of proper information about certain places may result in problem like lack of transportation facilities, lodging and fooding, lack of proper tour guide. Loss of time and effort for looking to visit places around you.

In order to overcome this heck, we came up with the idea to resolve this. An android application where user go for looking nearby places easily. At same place, user get all required details about place without going anywhere on their fingertips.

In every sector, digitalization is reaching its peak. Our goal is to provide service to the tourist, on the other hand, promotes the tourist sector as well as uplift the economy of local people and country.

With proper user- friendly interface and exciting features like currency, weather report, emergency contacting so on. An app already has some pre updated tourists' site further can be added by user related places in their locality. Navigation with google maps and calculating distances to tourist places and can be updated with various route to reach selected destination by various means of transportation.

## Background Study

We have gone through detailed information of most of the tourists' places in Bangladesh by division, district and Upazila wise to sort the places for easy searching. Another detailed information is gathered about nearby police station of tourist places to help tourist at the time of need as well as route to reach there. Gathered famous name, interesting facts and their images (source: GoogleSearch).

Language involved and needed to learn are:

- Dart/Flutter

Flutter is main component for creating expressive ui in the application. Flutter helps to generate attractive and intuitive ui. Stateful widget and stateless widget is very useful.

- Firebase Integration

Cloud firestore is used to store data, needed to learn about collection, document and about generating restapi in JSON format. SHA-key and webapi-key is used to parse information. Enabled firebase plugin in android studio for authentication and storing data to database.

- Using Google api

Google api was an important for functional requirements. Google cloud platform is used to enable different library to provide api and services for an application.

- Maps integration, route finding

Maps integration was most important task done by api-key generated by Google cloud platform and various feature allowed to calculate distance, searching place in map and route and location.

- Rest api and Widget

Along with google api, we have also used rest-api for getting district, division and upazila using dropdown box. Learnt use of widget for quality UI.

Tools/Software involved are android studio, firebase database etc.

# Methodology

Following methodologies are applied while working on this project.

## Architecture and Strategy

We have identified the app users as well as drawn rough sketch of app. On choosing strategy, mostly we have used agile methodologies and tried to work features wise.

## Analysis and Planning

Defined use cases and capturing detailed functional requirements. Choosing name for app and platform where app runs

## User-Interface Design

Developed interactive, intuitive and user-friendly like splash showing visit Bangladesh. Later, loading with login/signup page and finally, attractive landing page. Keeping in mind, better intuitive design sees better user adoption. Usability of application is as important as functionality of application.

## Development

We have collected Identified technology stack like flutter, kotlin or java. For our project, we used flutter as it provides flexible and expressive UI. We have used rest api for fetching information about division, district etc. which provides communication between app and backend. We have decided to use .net web api but due to need for more functional requirements we have used firebase database for storing and fetching data.

# Software Requirement Specification Document

## 3.1 Purpose

This document is based on the Software Requirement Analysis (SRS) for Visit Bangladesh android application. It includes all necessary requirements to develop this application no matter whether they are functional or non-functional. The information about the requirements here have been organized systematically. so that everyone can easily figure out a summarized concept about Visit Bangladesh. This SRS serves as the official means of communicating user requirements to the developer and provides a common reference point for both the developer and stakeholder community. It will evolve over time as users and developers work together to validate, clarify and expand its contents.

### 3.1.2 Intended Audience

This report is intended for several audiences, including the customer, as well as the project managers, designers, developers, and testers.

- The customer will use this document to ensure that whatever he requires has been fulfilled by the project teams.
- The project managers of the developer team will use this SRS to fix a milestone and time to deliver the software and to ensure that the teams working on this project are on the right path.
- The designers will use this document as a basis for creating the system's design. The designers will continually refer back to this SRS to ensure that the system they are designing will fulfill the customer's needs.
- The developers will use this report as a basis for developing the system's functionality. The developers will link the requirements defined in this SRS to the software they create to ensure that they have created software that will fulfill all of the customer's documented requirements.
- The testers will use this SRS to derive test plans and test cases for each

documented requirement. When portions of the software are complete, the testers will run their tests on that software to ensure that the software fulfills the requirements documented in this SRS. The testers will again run their tests on the entire system when it is complete and ensure that all requirements documented in this SRS have been fulfilled.

## 3.2 Inception

In this chapter, we will discuss about the first step of Software Requirements Specifications Analysis, that is, Inception.

### 3.2.1 Introduction

Requirement Engineering comprises several sequential steps. Inception is the first one among them. Inception creates the entrance to the project for the requirements analysts. It refers them how the project should get started. It also provides a basic idea to the engineers about the problems ahead which are needed to be solved and how critical obstacles may come during the project. The main target of Inception phase is to identify the people related to the project and their needs. In order to complete this phase, we have focused on-

- Identifying Stakeholders
- Recognizing multiple viewpoints
- Working towards collaboration
- Asking the first questions

#### 3.2.1.1 Identifying stakeholders

Stakeholders are entities that have an interest in a given project. These stakeholders may be inside or outside an organization which:

- Sponsor a project, or
- Have an interest or a gain upon a successful completion of a project,
- May have a positive or negative influence in the project completion

There is one stakeholder for this system. That is an android app supplier who requested for the app to develop and design as well as specified the requirements for the app.

#### *3.2.1.2 Requirements:*

The requirements that have to be fulfilled in the application are:

- The app should be able to launch on any version of android smartphone.
- User can activate or deactivate the app from functioning.
- User can perform search and browse operation for various tourist places
- User can create their profile and add comments and reviews.
- The app should maintain a database for faster query.

#### *3.2.1.3 Asking first questions*

We set our first set of context-free questions focuses on the stakeholder, overall project goals and benefits. These questions helped us to identify the measurable benefit of the successful implementation and possible alternatives to custom software development. Next set of question helped us to gain a better understanding of problem and allows the customer to voice his perception about the solution. The final set of question focused on the effectiveness of the communication activity itself.

### *3.2.2 Conclusion*

Inception phase helped us to establish basic understanding about the Visit Bangladesh application, identify the people who will be benefited if the application is implemented, define the purpose of the project and establish a preliminary communication with the stakeholder.

## 3.3 Elicitation

In this chapter, we will briefly discuss about the Elicitation phase of Visit Bangladesh application.

### 3.3.1 Introduction

Requirement's elicitation is recognized as one of the most critical, knowledge-intensive activities of software development; poor execution of elicitation will almost guarantee that the final project is a complete failure. Since project failures are so rampant, it is quite likely that improving how the industry performs elicitation could have a dramatic effect on the success record of the industry. Improving requirements elicitation requires us to first understand it. Although many papers have been written that define elicitation, or prescribe a specific technique to perform during elicitation, nobody has yet defined a unified model of the elicitation process that emphasizes the role of knowledge.

### 3.3.2 Eliciting requirements

Earlier we have seen that the methodology used in Inception phase is Question and Answer approach. But Elicitation is quite different in this point of view. The elicitation phase follows a format of eliciting requirements which combines the other four phases namely problem solving, elaboration, negotiation and specification. In order to elicit requirements, we have followed four steps:

- Collaborative Requirements gathering
- Quality Function Deployment (QFD)
- Usage Scenarios
- Elicitation work product

### 3.3.3 Requirements gathering

Many different approaches to collaborative requirements gathering have been proposed. Each makes use of a slightly different scenario. We completed following steps to do it:

- Meetings have been conducted with the android app supplier and he was questioned about his requirements and expectations from the Visit Bangladesh application.
- He was asked about the existing problems that are being faced by users without the application.
- Based on the meetings and response of him, we finally selected the requirements.

### 3.3.4 Quality function deployment

Quality Function Deployment (QFD) is a technique that translates the needs of the customer into technical requirements for software. It concentrates on maximizing customer satisfaction from the Software engineering process.

#### 3.3.4.1 Normal requirements

Normal requirements consist of objectives and goals that are stated during the meeting with the customers. Normal requirements of our project are:

- Allowing all user to search various tourist places
- Allowing all users to update about their locale tourist places
- connection require (Online)
- Allowing user to add favorite contacts as emergency contact

#### *3.3.4.2 Expected requirements*

These requirements are implicit to the system and may be so fundamental that the customer does not explicitly state them. Their absence will be a cause for dissatisfaction. The expected requirements of our app are:

- Finding route to destined places
- User can create their profile and, add comments and tourist places detailed information like local guide, police station information.
- User can reset their username and password.
- User can upload images of various tourist places

#### *3.3.4.3 Exciting requirements*

These requirements are for features that go beyond the customer's expectations and prove to be very satisfying when present. The exciting requirements for our app are following:

- can add five favorite contacts and use it during journey at the time of emergency.
- Can use our currency converter features at any time
- Can get updated about weather of any tourist places

### **3.3.5 User scenario**

The aim of the app is to guide local and foreign tourist while visiting Bangladesh.

#### **Authentication:**

At first user must provide name, unique id, and password to create account. System will create a unique account for the user with the given information.

To use the app user does not need to enter own id and password to login each time. If any other user wants to login, the logged in user has to log out at first. Guest user do not need to create account and they can directly view places and route.

#### **Browse:**

user can browse place name and interesting facts, can view images and look for route to reach that tourist place.

#### **Update:**

All users can add new places and their route to reach tourist places. User can also add new places available in their locality and assures route with special authentication.

#### **Search:**

When user logged in, they can perform search operation by division/state name or place name.

#### **Edit contact:**

In edit contact option, a user can update their profile picture and can reset their username/password.

#### **Emergency contact:**

A user can add five favorite contacts in their profile at the time of creation of account. Later on, app will help user to send notification or alert to those favorite contacts people at the time of emergency during their journey.

### 3.3.6 Elicitation work product

The output of the elicitation task can vary depending on size of the system or product to be built. Our elicitation work product includes:

- Set of usage scenarios.
- Description of the system's technical environment.
- Make a bounded statement of scope for our system.
- Make a list of user and other stakeholder who participated in requirements.
- Make a statement of the requirements for Visit Bangladesh.

## 3.4 Scenario Based Model

This chapter is about the scenario-based model of Visit Bangladesh.

### 3.4.1 Introduction

Scenario based modeling is an inexpensive rapid prototyping technique. This method is effective when systems are being built with the requirements vaguely known at the outset. Users are involved right from the start, to build prototypes evolving towards the final product. The users are also involved with the testing of the prototypes which is essential for the validation of requirements and help the users to gain an initial experience of the final system during the development itself. This method involves techniques which are applied by one or more professionals working alongside users who are expected to provide and specify their requirements at the beginning as well as evaluate and approve the system upon completion. The user (in a passive capacity) and the designer/builder (an active partner) cooperate to reach a working model where the means of communications are by the examination of preliminary models such as the initial narratives, paper models and graphical representations built to represent the final system functions.

### 3.4.2 Use case diagram

A use case diagram is a graphic depiction of the interactions among the elements of a system.

The purposes of use case diagrams are:

- Gathering requirements of a system.
- Getting an outside view of a system.
- Identifying external and internal factors influencing the system.
- Showing the interaction among actors.

The first step in writing a use case is to define the set of actors that will be involved in the story.

Actors are of two types. They are:

1. **Primary Actors:** Primary actors are the actors using the system to achieve a goal. They both consume data and produce information.
2. **Secondary Actors:** Secondary actors are the actors that the system needs assistance from to achieve the primary actor's goal. They either consume data or produce information.

Once actors have been identified, use cases can be developed.

### 3.4.3 Use Case Scenario

Level 0	Level 1	Level 2	Level 3
Visit Bangladesh	Authentication	Login	Reset Password
		Register	
		Guest User	
	Browse	Images of tourist places	
		Name and facts of places	
		Route	
		Comments and Reviews	
	Update	Add Route	Add places
		Add new place	Add images
	Search	Search by state/division	
		Search by place name	
	Settings	Edit account	Change profile picture
			Reset username/password
	Emergency Contact	Add contacts	
		Send notifications	

### *3.4.3.1 Use case diagrams and description of subsystems*

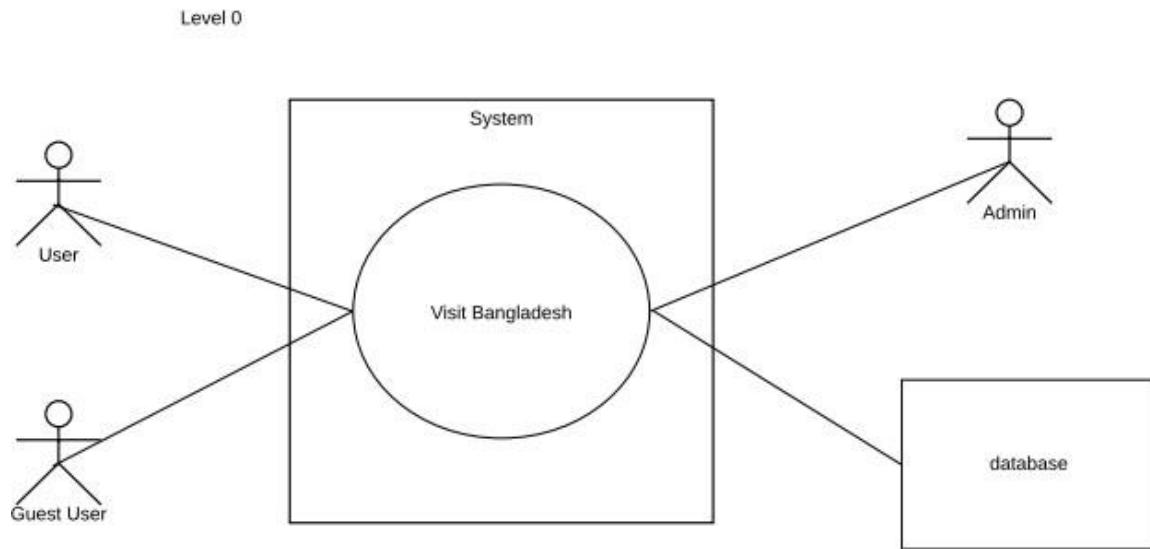
There are two actors in the system- user and system. The use case diagram of each subsystem and their description will be discussed in this section.

#### *Level 0: Visit Bangladesh*

**Primary actor:** User, Admin, Guest user

There are no secondary actors in Visit Bangladesh application.

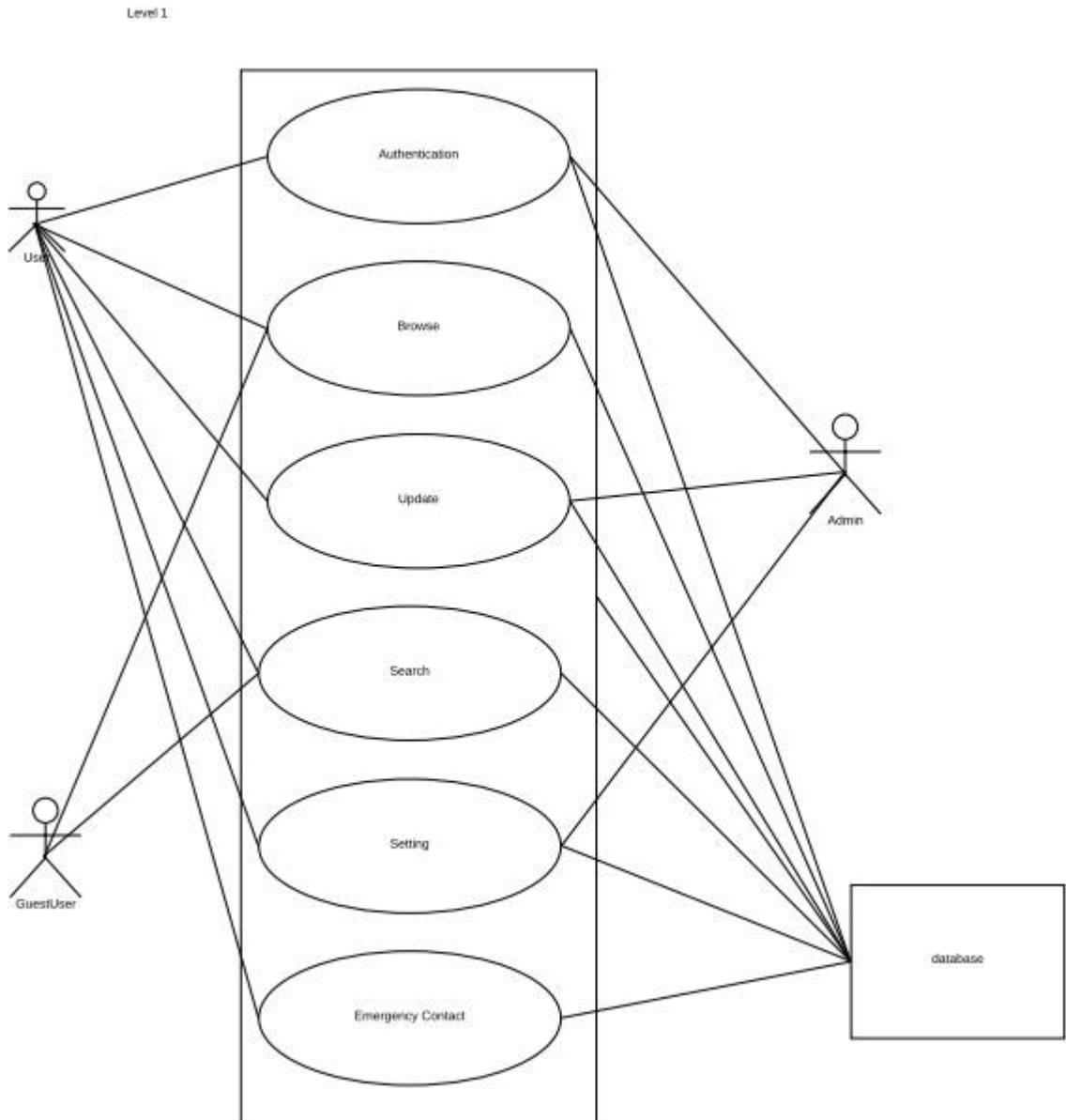
**Goal in context:** The diagram in Figure 1 represents the whole Visit Bangladesh.



*Figure 1:Level 0 Use Case Diagram*

### *Level 1: Visit Bangladesh*

There are six subsystems in Visit Bangladesh application. These are- Authentication, Browse, Update, Search, Setting and Emergency Contact. Figure 2 shows these subsystems.



*Figure2:Level 1 Use Case Diagram*

### 3.4.3.1.1 Authentication:

In this subsystem, the user inputs name, id and password where system verifies this information. Once registration is completed, user needs to login to own account. Here, both user and admin are primary actors. Guest user can view the app with particular restriction. Figure 3 shows Authentication subsystem.

Figure  
3:  
Level  
1.1 Use  
Case

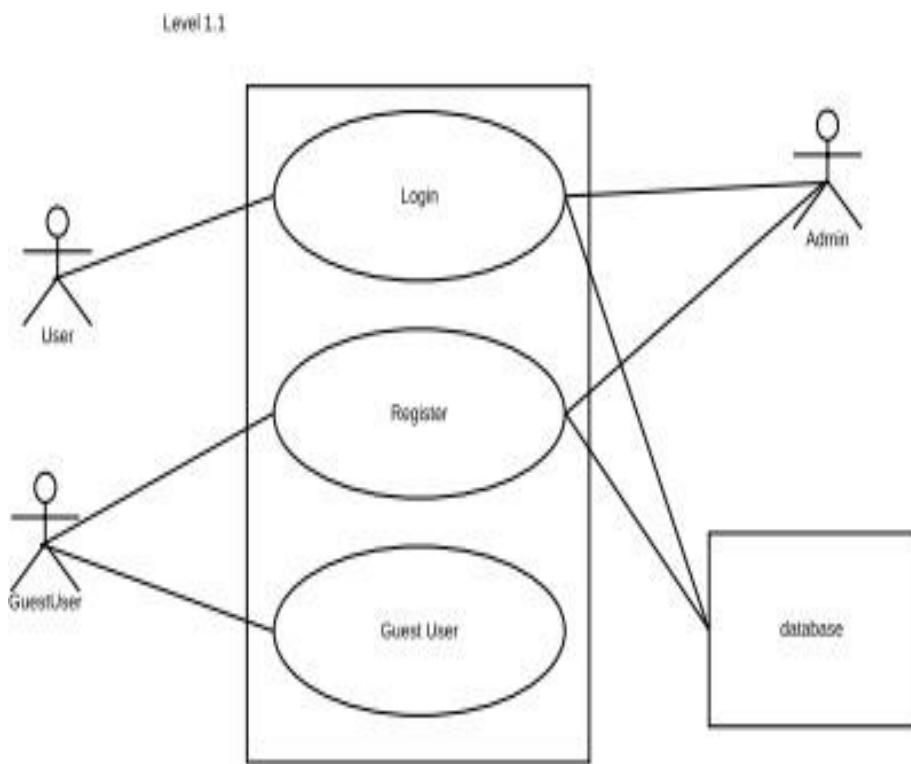
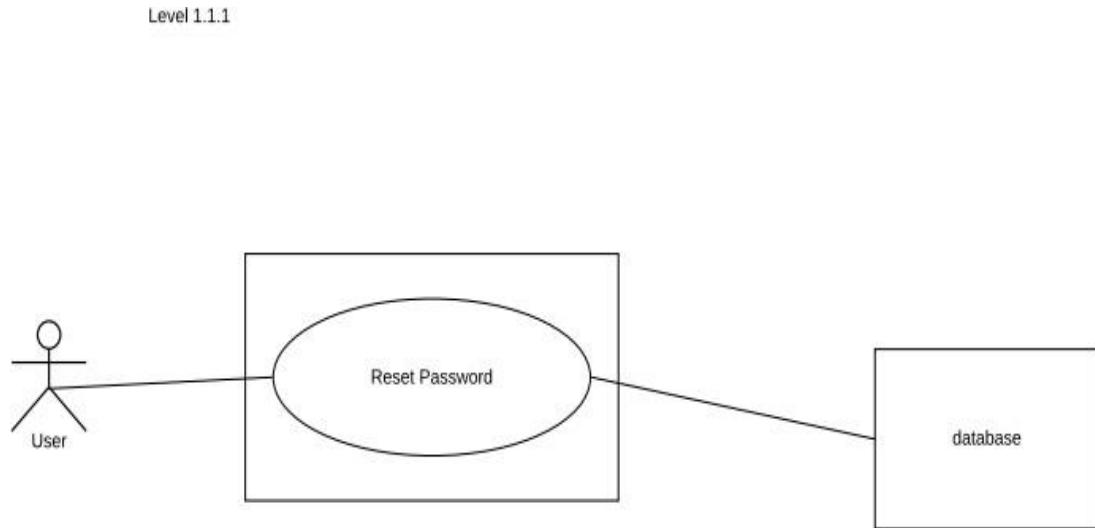


Diagram of Authentication Subsystem

### 3.4.3.1.1 Reset Password

This diagram shows the level 3 reset password in figure 4.



*Figure 4: Level 1.1.1 Use Case Diagram of Authentication Subsystem*

### 3.4.3.1.2 Browse:

In this subsystem, the user can browse images, interesting facts and names of tourist places, route to reach those places. User can add their review and comments. This diagram shows Level 1.2 about Browse subsystems in figure:5.

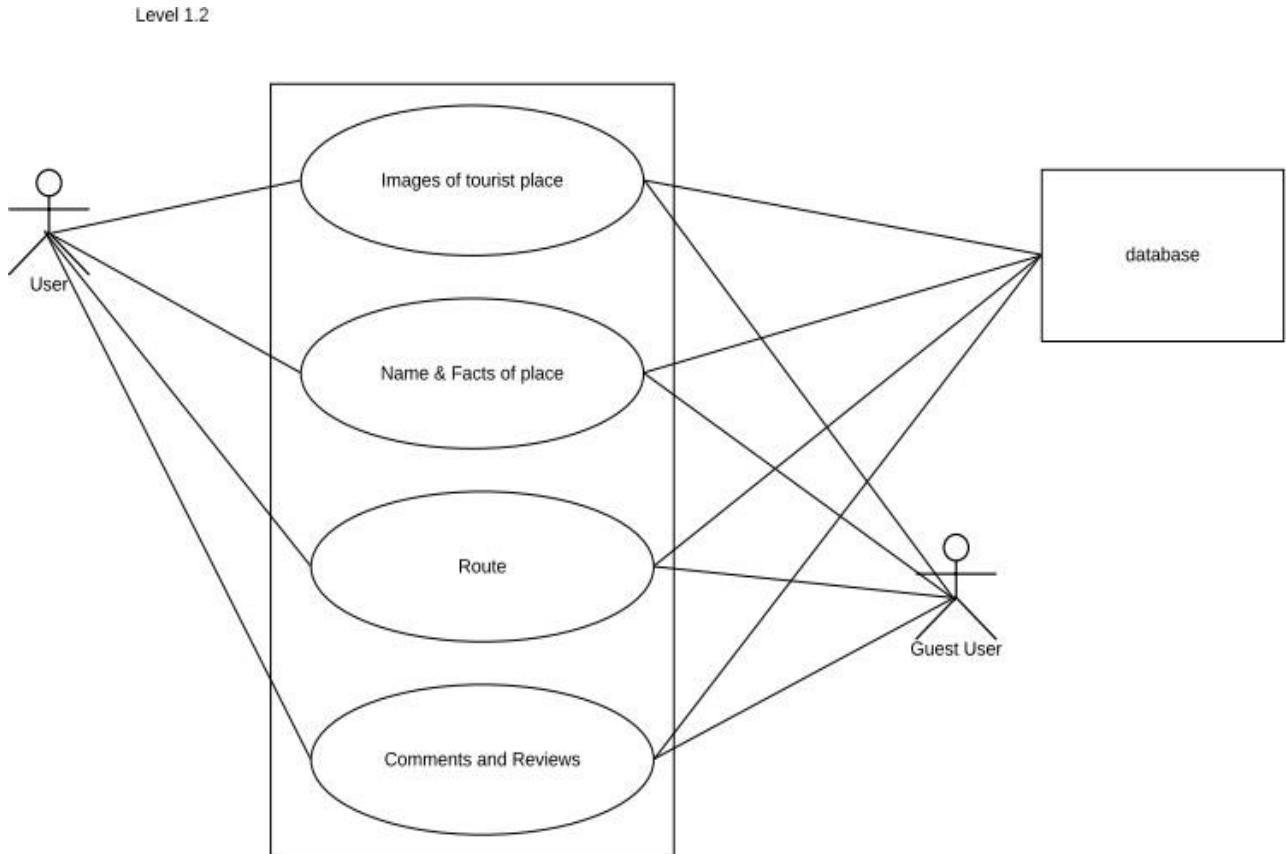
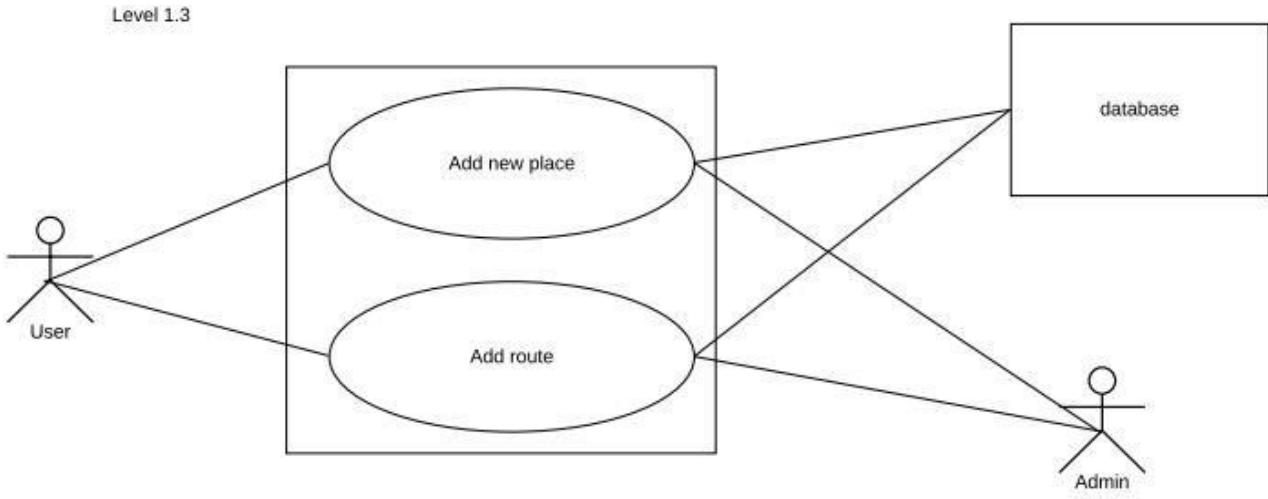


Figure 5: Level 1.2 Use Case Diagram of Browse Subsystem

### 3.4.3.1.3 Update:



*Figure 6: Level 1.3 Use Case Diagram of Update Subsystem*

In this subsystem, admin can add new more tourist places and route for user. Also, user can add new visiting places in their locality. Figure 6 shows the update subsystem

#### 3.4.3.1.4 Search:

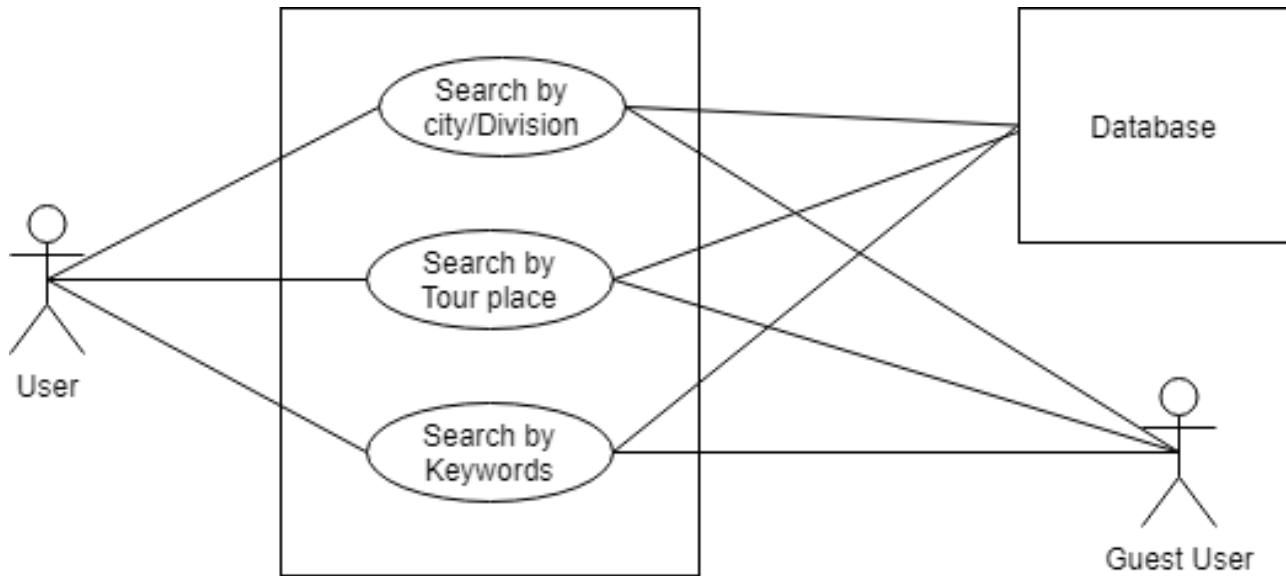
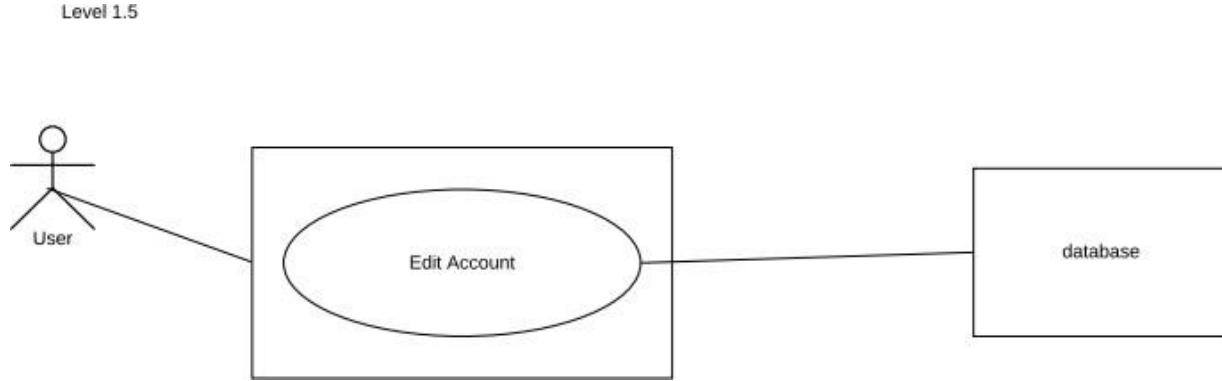


Figure 7: Level 1.4 Use Case Diagram of Search Subsystem

In this subsystem, user can search various tourist places by searching with the division/state name or directly tourist place name. Figure:7 shows Search subsystem.

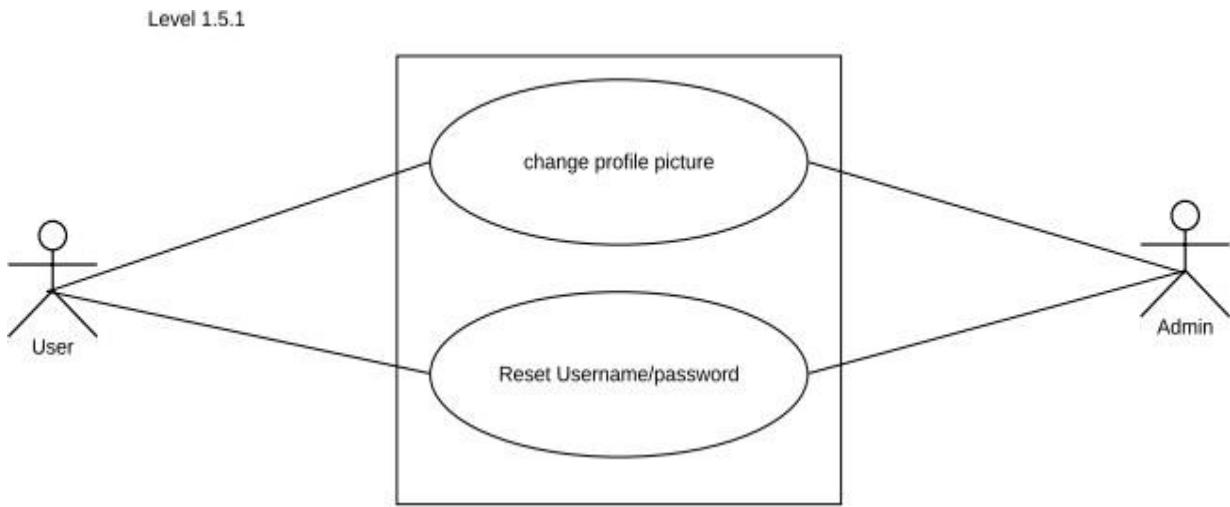
### 3.4.3.1.5 Setting:



*Figure 8: Level 1.5 Use Case Diagram of Setting Subsystem*

In this subsystem, user can edit their account, user can change their profile picture and can reset username and password. Figure 8 shows setting subsystem.

### 3.4.3.1.6 Edit Account:



*Figure 9: Level 1.5.1 Use Case Diagram of Subsystem's task edit account*

In this subsystem's tasks, the user can change their profile picture and can reset their password and username.

Figure 9 shows the subsystem's tasks.

### 3.4.3.1.7 Emergency Contact:

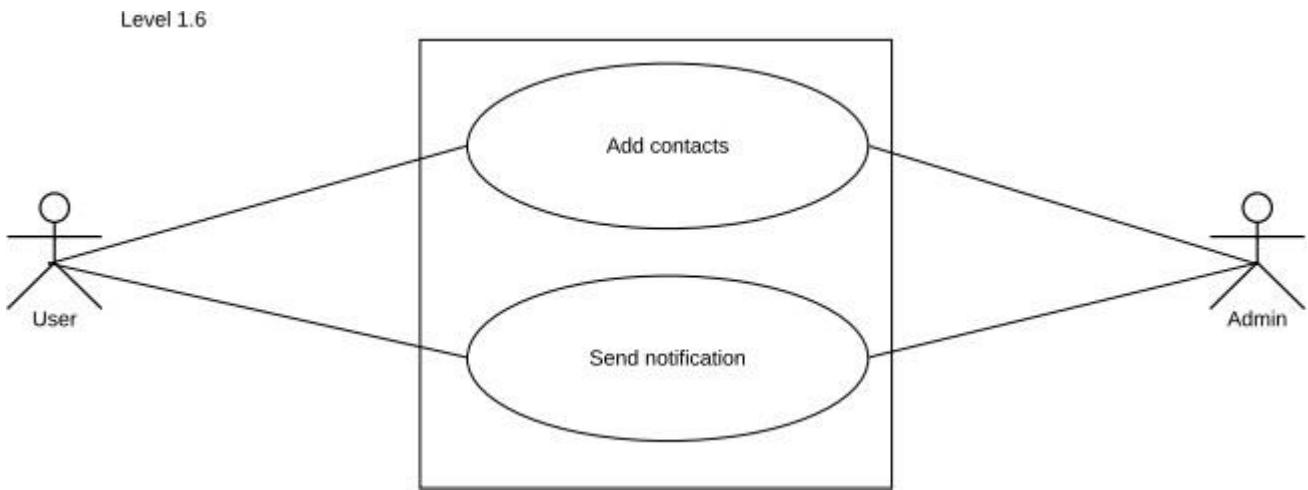


Figure 10: Level 1.6 Use Case Diagram of Emergency Contact Subsystem

In Emergency contact subsystem, the users can add their five favorite contacts and send notifications to all. Following use case diagram in Figure 10 shows emergency contact subsystem works –

### 3.4.3.2 Activity Diagrams of Visit Bangladesh

Activity diagrams are graphical representations of workflows of step-wise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e. workflows).

The activity diagrams of the modules described in the previous chapter is shown in the following figures:

#### 3.4.3.2.1 Activity diagram of Register:

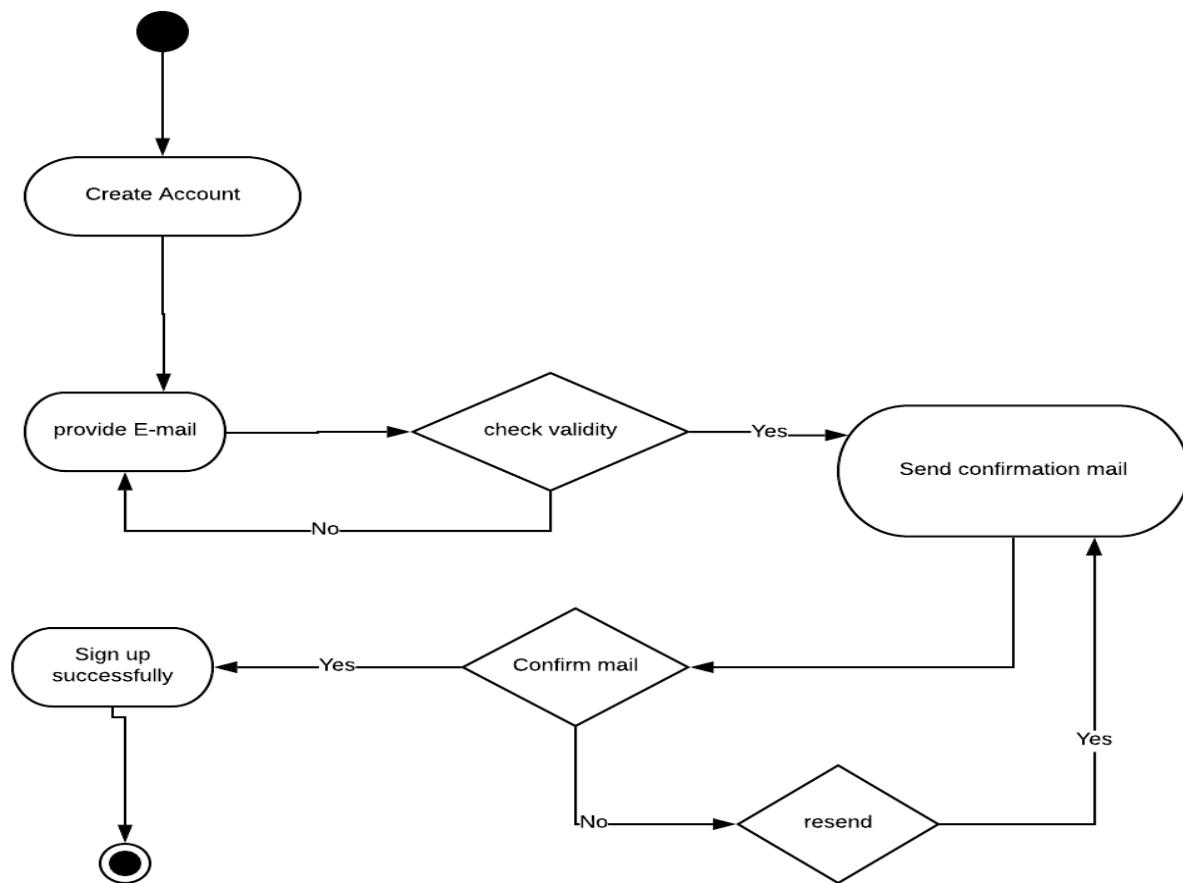
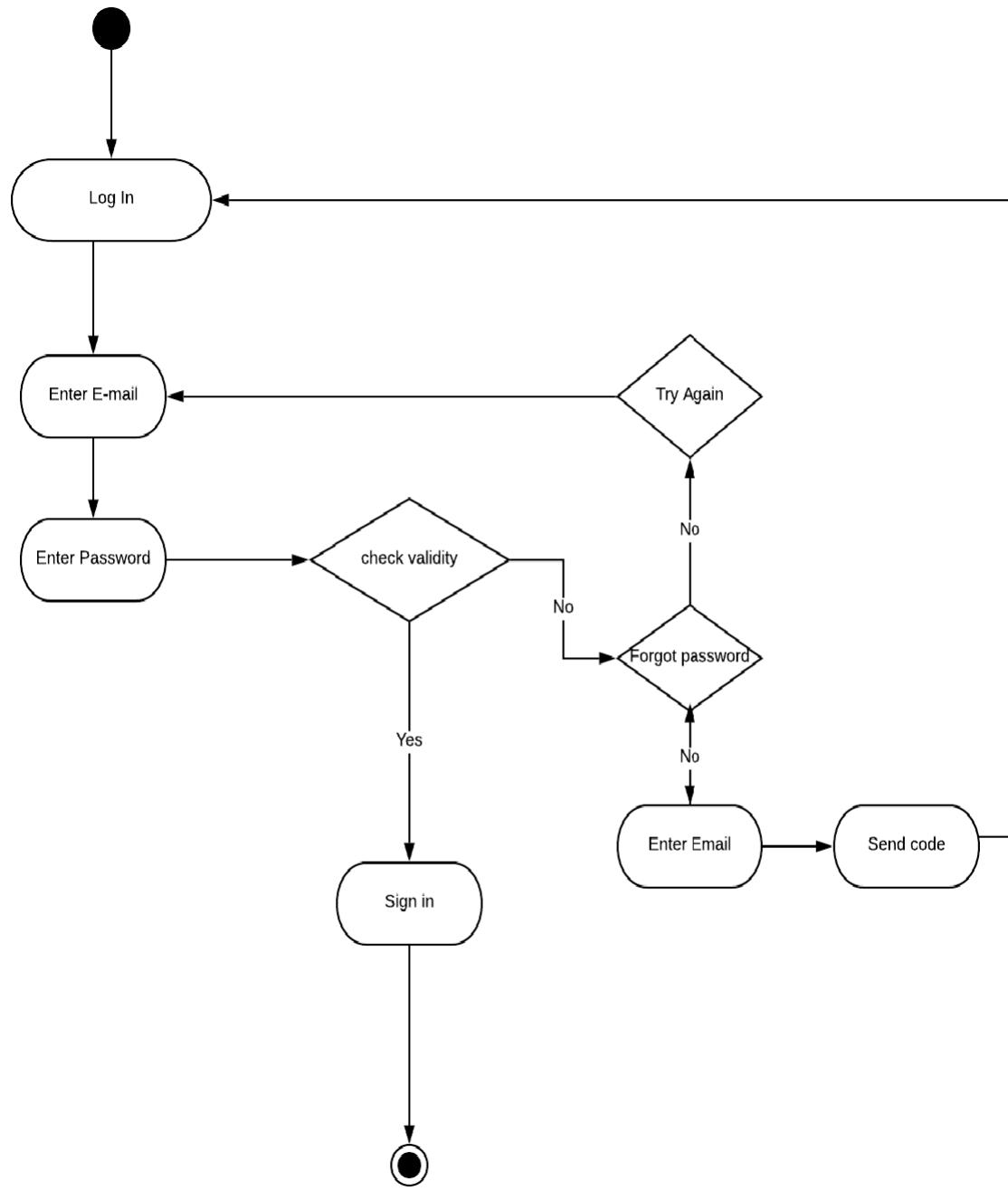


Figure 11: Level 4.3.1 Activity Diagram of Sign Up

### 3.4.3.2.2 Activity diagram of Log In:



*Figure 12: Level 4.3.2 Activity Diagram of Log In*

### 3.4.3.2.3 Activity diagram of Guest user:

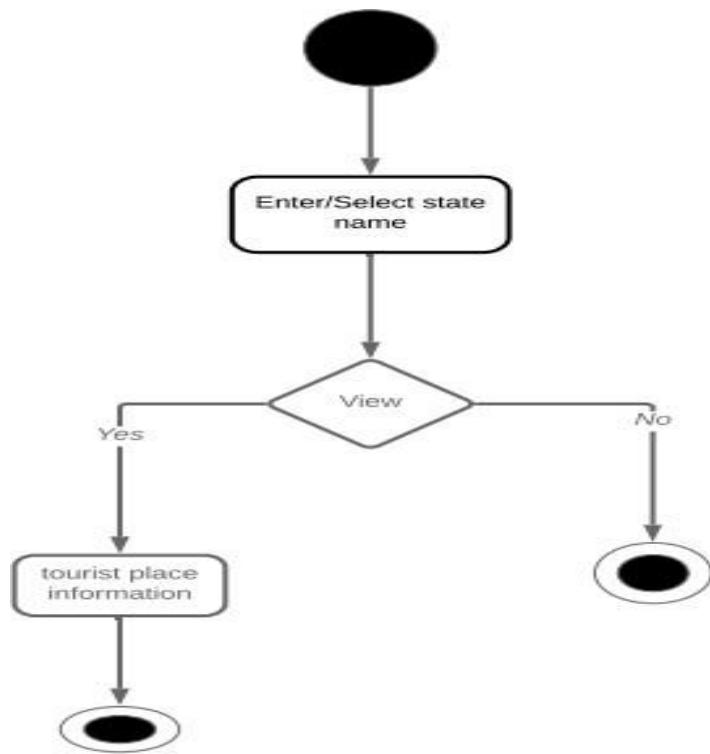
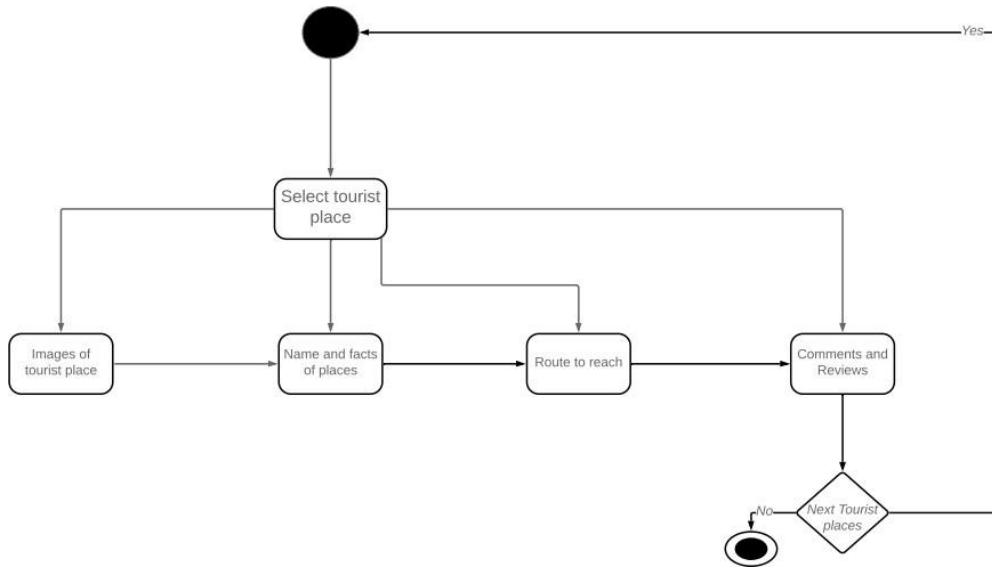


Figure 13: Level 4.3.2 Activity Diagram of Guest user

#### 3.4.3.2.4 Activity diagram of Browse:

Figure 14: Level 4.3.3 Activity Diagram of Browse



### 3.4.3.2.5 Activity diagram of edit contact:

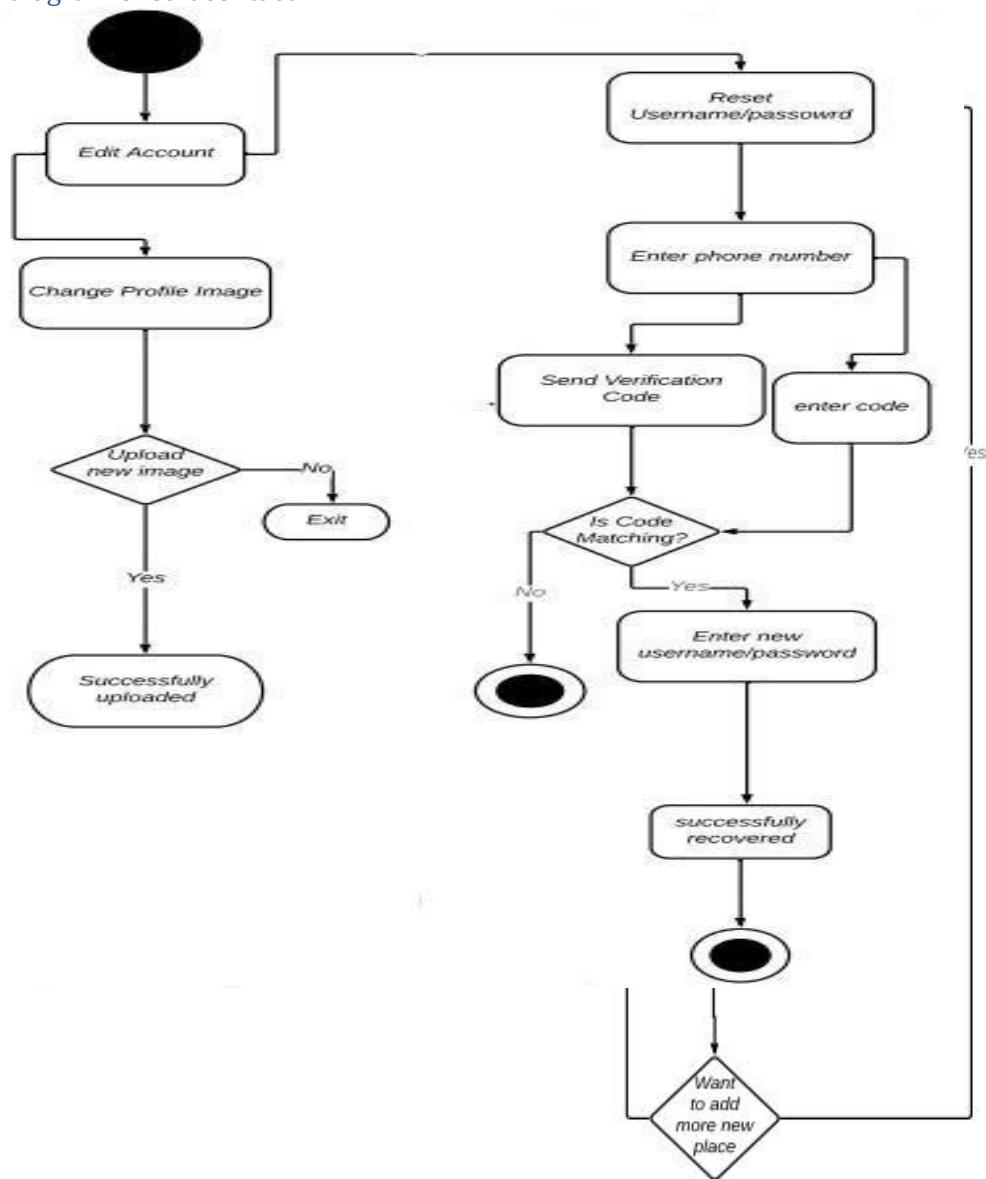
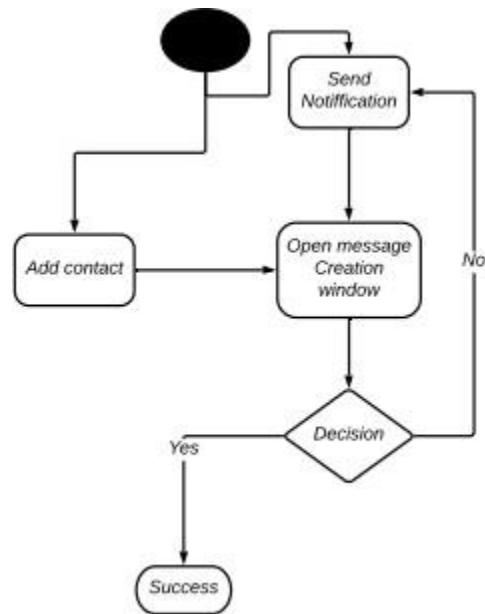


Figure 15: Level 4.3.4 Activity Diagram of Edit Contact

3.4.3.2.6 Activity diagram  
of Emergency Contact:



### 3.4.3.3 Swim-lane diagram of Visit Bangladesh

A swim lane diagram is a visual element used in process flow diagrams, or flowcharts, which visually distinguishes job sharing and responsibilities for sub-processes of a business process. The swim-lane diagrams of the modules described in the previous chapter is shown below:

#### 3.4.3.3.1 Swim-lane diagram of Register

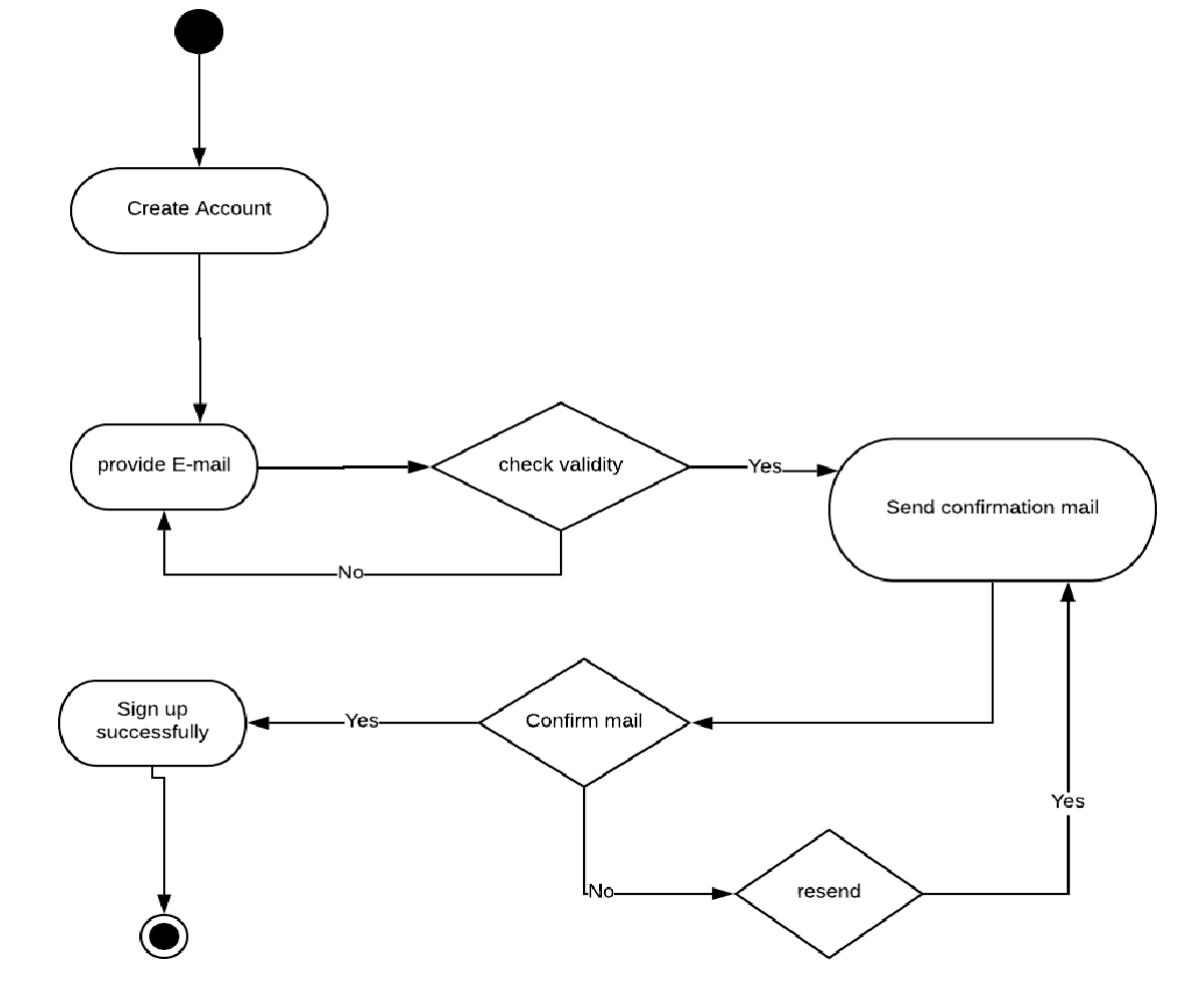


Figure 16: Level 4.4.1 Swim-lane Diagram of Register

### 3.4.3.3.2 Swim-lane diagram of Log In

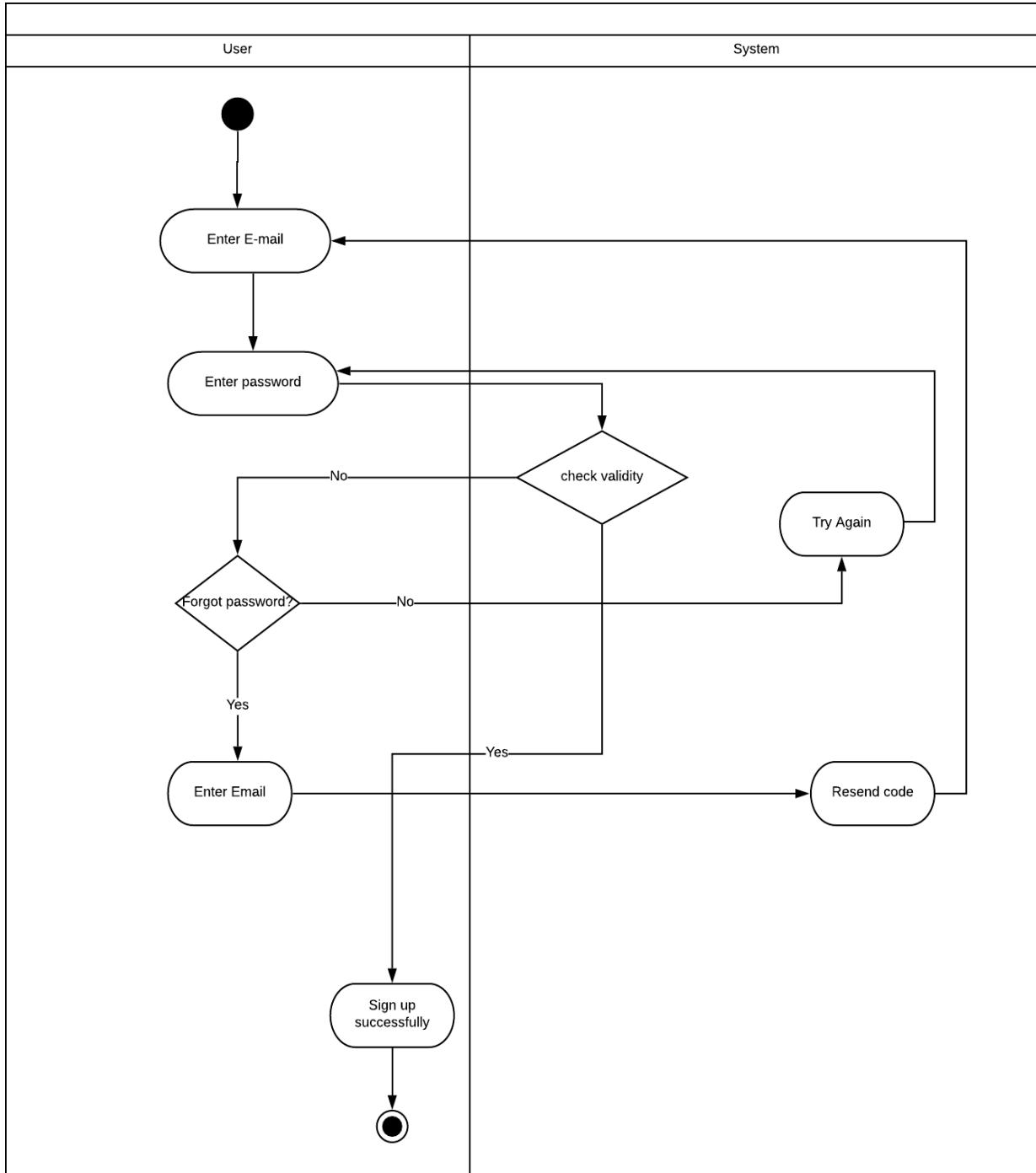
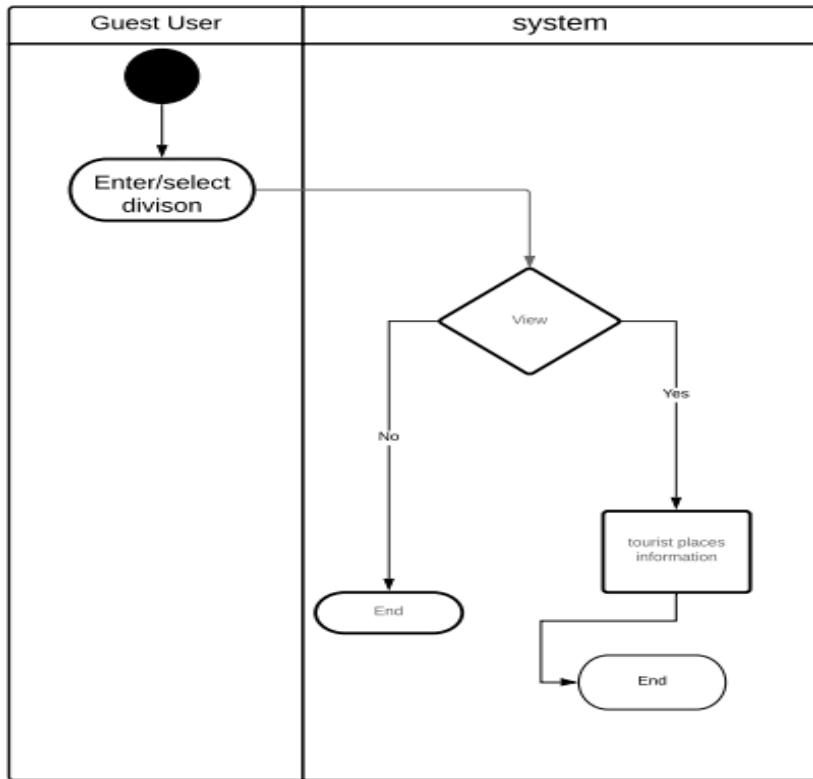


Figure 17: Level 4.4.2 Swim-lane Diagram of Log In

### 3.4.3.3.3 Swim-lane diagram of Guest user



*Figure 18: Level 4.4.3 Swim-lane Diagram of Guest user*

### 3.4.3.3.4 Swim-lane diagram of Browse

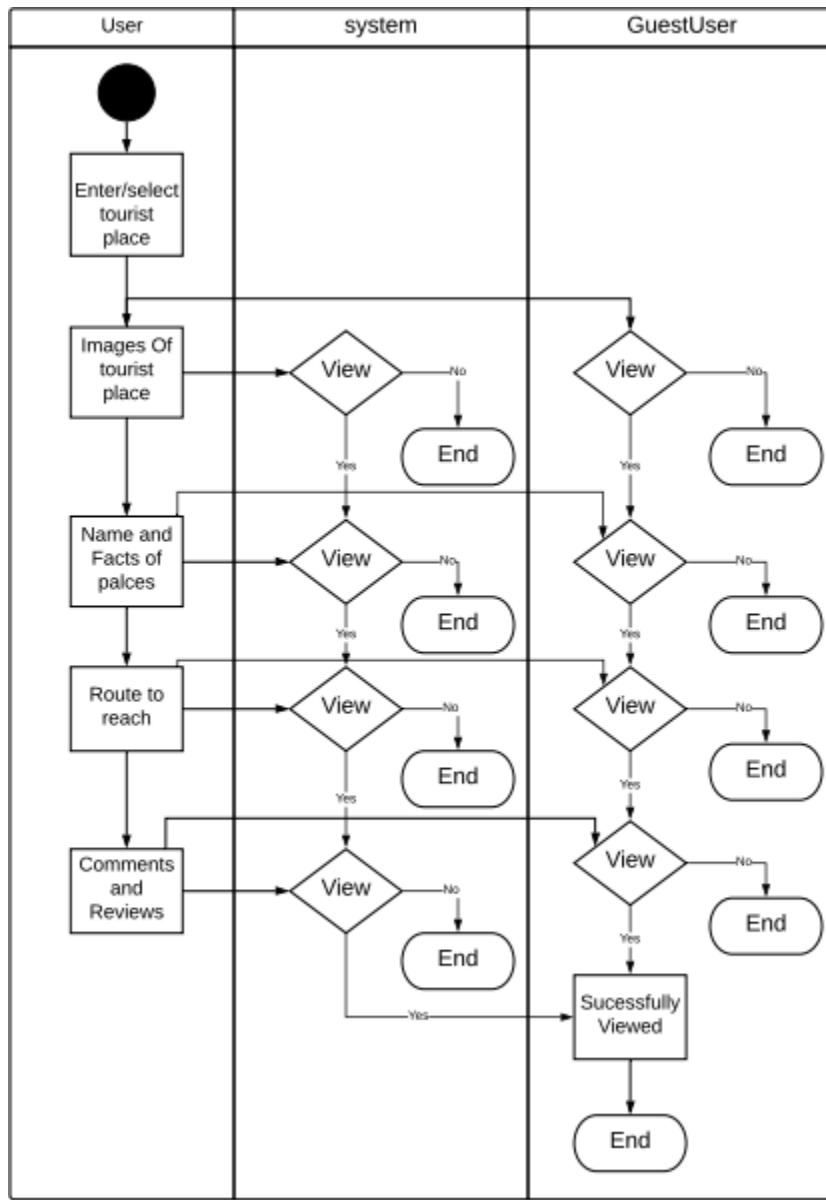
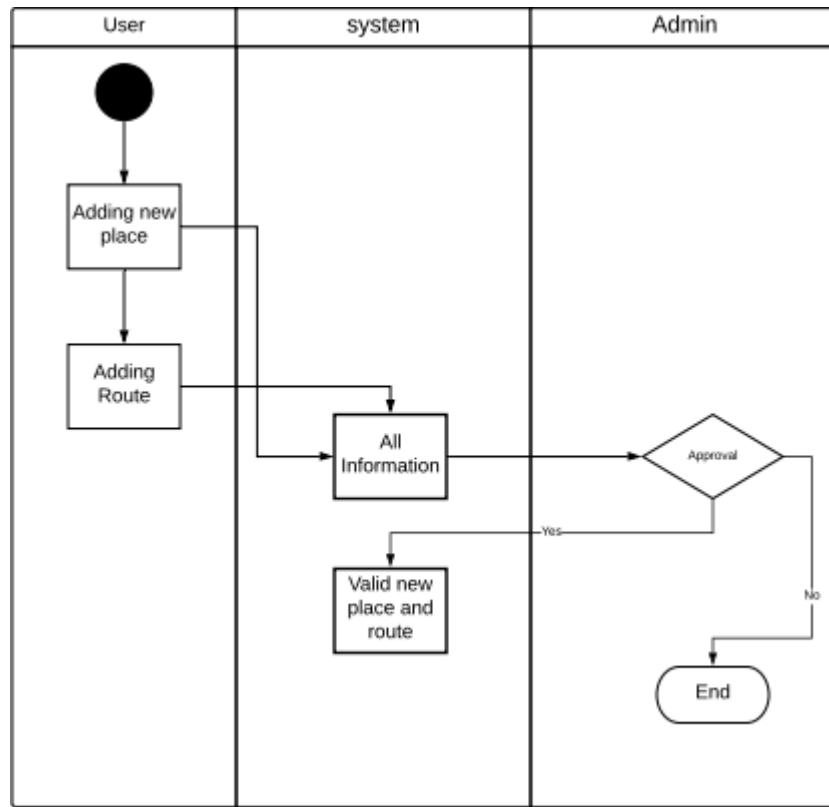


Figure 19: Level 4.4.4  
Swim-lane Diagram of  
Browse

### 3.4.3.3.5 Swim-lane diagram of Update



*Figure 20: Level 4.4.5 Swim-lane Diagram of Update*

### 3.4.3.3.5 Swim-lane diagram of Edit contact

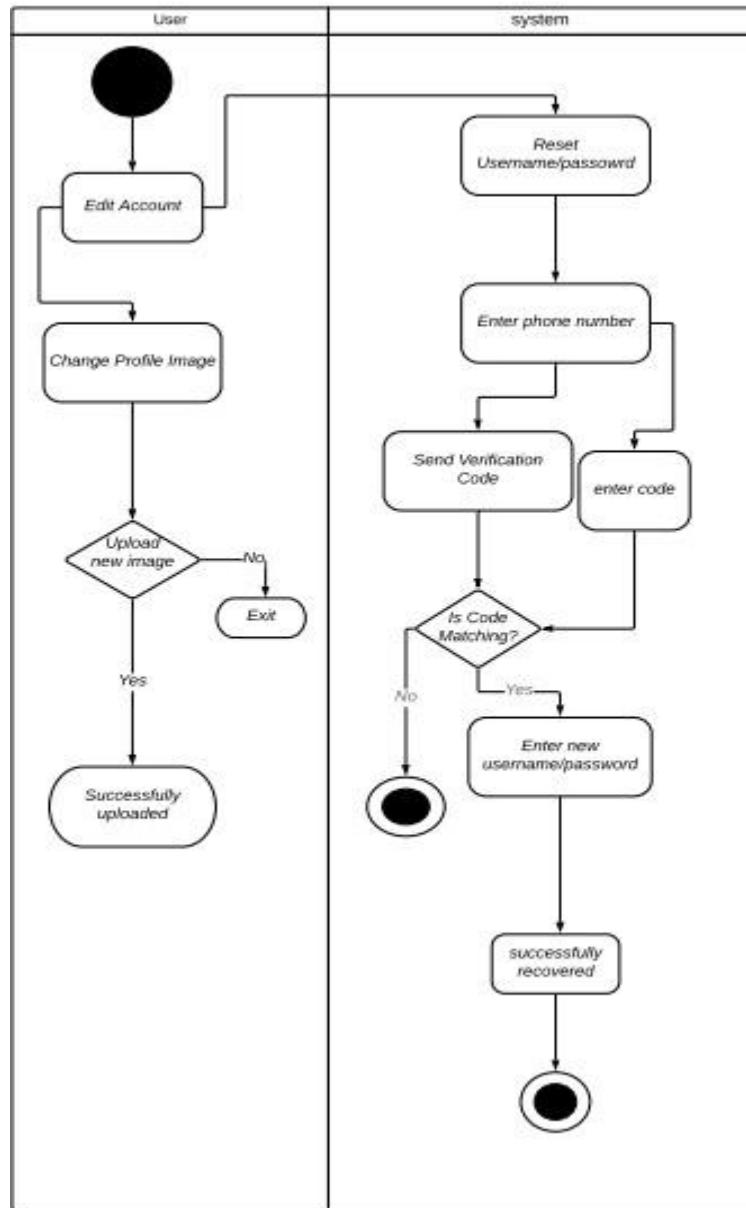
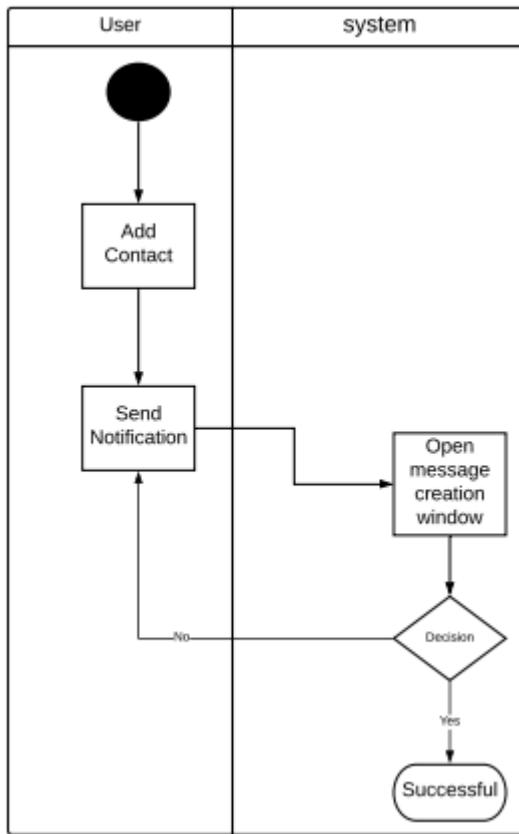


Figure 21: Level 4.4.5 Swim-lane Diagram of Edit contact

### 3.4.3.3.6 Swim-lane diagram of Emergency contact

Figure 22  
Figure 22: Level  
4.4.6 Swim-lane Diagram of  
emergency contact



## 3.5 Data Based Model

### 3.5.1 Introduction

A data model is a conceptual representation of data structures (tables) required for a database and is very powerful in expressing and communicating the business requirements. In this chapter, we will discuss about the data model for Visit Bangladesh.

### 3.5.2 Data object

A data object is a representation of information which has different properties or attributes that must be understood by software. We have found the following data objects in Visit Bangladesh:

#### 3.5.2.1 Noun Identify

Nouns are identified from the usage scenario and shown in the table below:

**Table 1: Identifying Potential Data Objects**

No.	Noun/Noun Phrases	Problem Domain/Solution Domain	Attributes
1.	Authentication	S	
2.	System	P	11,12,25,34
3.	Guest User	S	
4.	User	P	
5.	Username	S	4
6.	Id	S	4,31,35,36,37,20,38,
7.	Password	S	4
8.	Account	P	2
9.	Nearest Place	S	
10.	Information	S	39,37,36,
11.	Sign in	S	

<b>12.</b>	Sign out	S	
<b>13.</b>	Place Image	P	
<b>14.</b>	Place Name	P	
<b>15.</b>	Travel Agency	S	36
<b>16.</b>	Route	P	
<b>17.</b>	Profile	P	4
<b>18.</b>	Search	P	
<b>19.</b>	Location	S	36,38
<b>20.</b>	Division	P	
<b>21.</b>	Address	S	36,38,37
<b>22.</b>	Comment	S	
<b>23.</b>	Emergency Contract	S	
<b>24.</b>	Edit Account	P	
<b>25.</b>	Send Notification	S	
<b>26.</b>	Profile photo	S	4
<b>27.</b>	Settings	S	
<b>28.</b>	Database	P	
<b>29.</b>	Edit Profile	P	
<b>30.</b>	Remove	S	
<b>31.</b>	Email	S	
<b>32.</b>	Phone	S	4,35,38

33.	Date	P	
34.	Update Location	P	
35.	Guide	S	
36.	Tourist place	S	
37.	City	S	
38.	Police Station	S	
39.	Add	P	

### *3.5.2.2 Prospective Data Objects:*

**User:** Id, Username, Password, Email, Phone,

**Location:** Division name, Description, Photo

**City:** name, Description

**Tourist Place:** Place Photos, Name, Description, Comment

**Add Tourist Place:** Place Photos, Name, Description

**Location type:** Mountain, sea area, Fount, Forestland

**Bus route:** route details

**Air route:** route details

**Ship route:** route details

### *3.5.2.3 Analysis on Prospective Data Objects:*

Bus route, Air and Ship have same attributes. So we can merge them into Route. Add Tourist Place and Tourist place have same attributes. So we can merge them into Tourist Place.

#### 3.5.2.4 Actual Data Objects

**User:** Id, Username, Password, Email, Phone, Emergency Contract

**Location:** Division name, Description, Photo

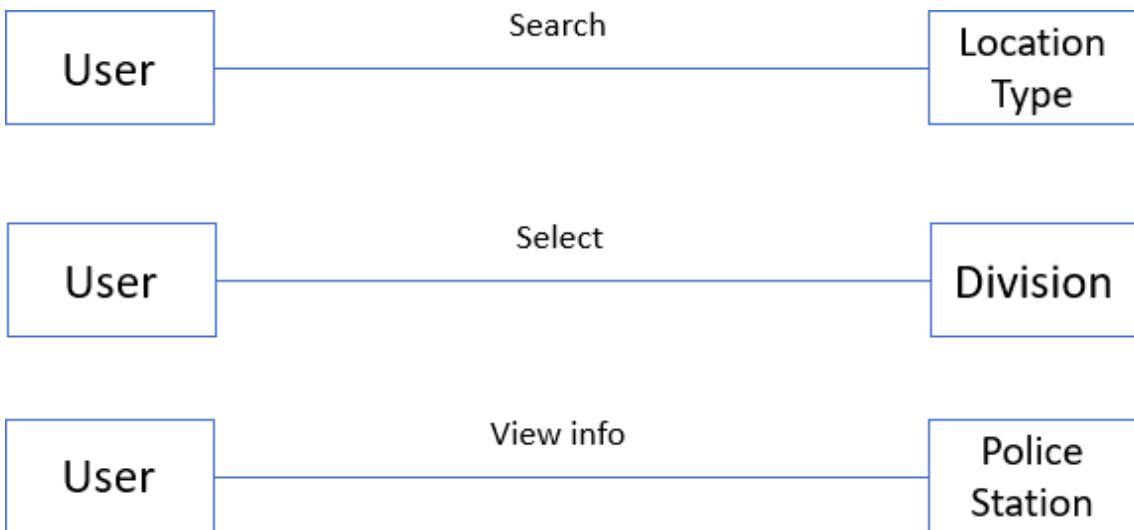
**City:** name, Description

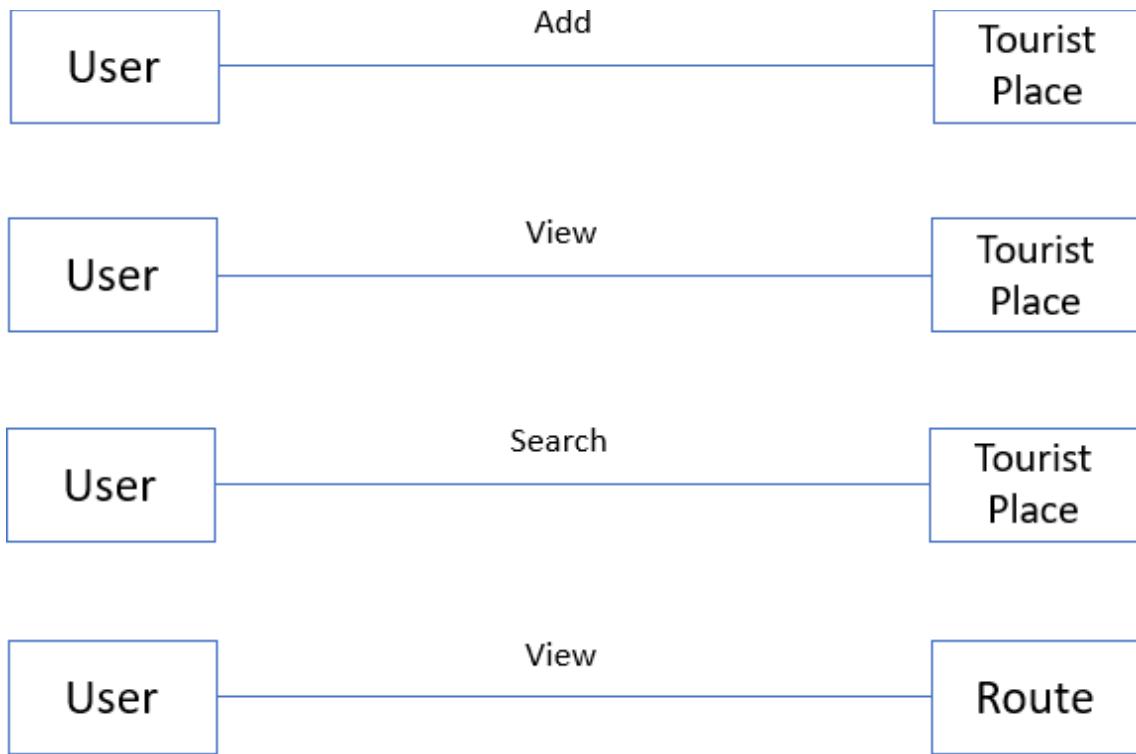
**Tourist Place:** Place Photos, Name, Description, Comment

**Location type:** Mountain, sea area, Fount, Forestland

**route:** route details, Way

##### 3.5.2.4.1 Relationship among Data Objects





*Figure 23: Relationship among Data Objects*

### 3.5.2.5 ER diagram

Based on the data objects found, in order to show the relationship among the data objects, ER diagram that is Entity-Relationship Diagram is used widely. Here, the ER Diagram based on the data objects of Visit Bangladesh is shown below.

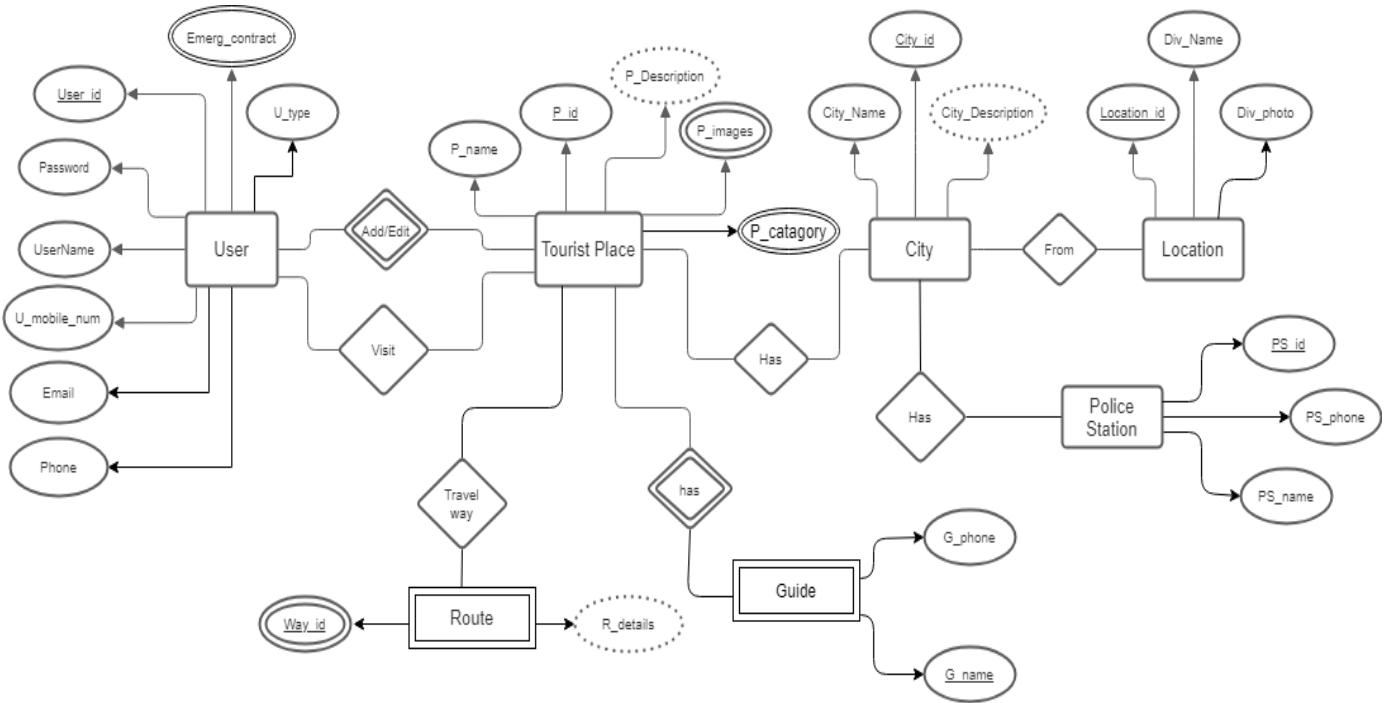


Figure 24: Entity Relationship Diagram

### 3.5.2.6 Data schema table

Based on the data objects, the following data schema tables can be created:

**Table : User**

Attribute	Type	Size
User Id	Varchar	20
User Name	Varchar	20
Emergency Contract	Number	20
Password	Varchar	20
Email	Varchar	10
Phone	Number	10

**Table : Tourist Place**

Attribute	Type	Size
Place Name	Varchar	20
Description	Varchar	300
Place Images	LongBlob	5
City Id	Number	20

**Table : City**

Attribute	Type	Size
City Id	Varchar	10
City Name	Varchar	10
City Description	Varchar	50

**Table : Location**

Attribute	Type	Size
Division Name	Varchar	20
Division Photo	LongBlob	10
Location Id	Varchar	10

**Table : Guide**

Attribute	Type	Size
Guide Id	Varchar	20
Guide Name	Varchar	10

Guide Phone	Number	11
-------------	--------	----

**Table : Police Station**

Attribute	Type	Size
PS Id	Varchar	20
PS Name	Varchar	10
PS Phone	Number	11

**Table : Tourist Location Type**

Attribute	Type	Size
Type Id	Varchar	20
Type Name	Varchar	10

**Table : Route**

Attribute	Type	Size
Way Id	Varchar	20
Route Details	Varchar	300

## 3.6 Class Based Model

Class-based modeling represents the objects that the system will manipulate, the operations that will apply to the objects, relationships between the objects and the collaborations that occur between the classes that are defined.

### 3.6.1 Grammatical Parsing and Analysis

To identify our analysis class, firstly, we grammatically parsed all the nouns and then categorized them according to general classification and selection criteria. We identified potential class by identifying the nouns from the scenery. Then we compared those with the following criteria whether they matched or not. We noted down the number of the fulfilled criteria at the right column.

#### 3.6.1.1 CLASS IDENTIFICATION WITH GENERAL CLASSIFICATION

In table 7, the nouns from the usage scenario are classified by general classification. Also, here, by “P” we meant a noun is in problem domain and by “S” we meant solution space.

##### General Classification:

1. External entities
2. Things
3. Occurrence or events
4. Roles
5. Organizational unit
6. Places
7. Structure

These criteria are used to find Potential Classes in following table-

**Table : Class Identification with General Classification**

No.	Noun or Noun Phrases	P/S	General Classification (GC)
1.	Authentication	S	3,5
2.	System	P	4,5
3.	Guest User	S	4
4.	User	P	4,5,7

<b>5.</b>	Username	S	2
<b>6.</b>	Id	S	2
<b>7.</b>	Password	S	2
<b>8.</b>	Account	P	4,5
<b>9.</b>	Nearest Place	S	4
<b>10.</b>	Information	S	2
<b>11.</b>	Sign in	S	3
<b>12.</b>	Sign out	S	3
<b>13.</b>	Place Image	P	
<b>14.</b>	Place Name	P	
<b>15.</b>	Travel Agency	S	5
<b>16.</b>	Route	P	
<b>17.</b>	Profile	P	2
<b>18.</b>	Search	P	
<b>19.</b>	Location	S	2
<b>20.</b>	Division	P	6
<b>21.</b>	Address	S	6
<b>22.</b>	Comment	S	2
<b>23.</b>	Emergency Contract	S	2
<b>24.</b>	Edit Account	P	
<b>25.</b>	Send Notification	S	2,3

26.	Profile photo	S	2
27.	Settings	S	
28.	Database	P	2
29.	Edit Profile	P	2
30.	Remove	S	
31.	Email	S	
32.	Phone	S	
33.	Date	P	2
34.	Update Location	P	3
35.	Guide	S	4,5
36.	Tourist place	S	6
37.	City	S	6
38.	Police Station	S	5,6
39.	Add	P	3

### 3.6.1.2 CLASS IDENTIFIED WITH SELECTION CRITERIA

The nouns having two or more than two were selected from the general classification list. After that step, we compared them with the following criteria list. Those are

1. Retained information
2. Needed services
3. Multiple attributes
4. Common attributes
5. Common operations
6. Essential requirements

In table, with the help selection criteria we identified whether the noun is accepted for preliminary class or not.

**Table : Class Identification with Selection Criteria**

No.	Noun	Special Classification (SC) Accepted
1.	Authentication	
2.	System	1,2,3
3.	Guest User	
4.	User	1,2,3,4,5
5.	Username	
6.	Id	
7.	Password	
8.	Account	1,2,3
9.	Nearest Place	1,2,6
10.	Information	1
11.	Sign in	
12.	Sign out	
13.	Place Image	2,1,3
14.	Place Name	1
15.	Travel Agency	
16.	Route	
17.	Profile	4,3

<b>18.</b>	Search	1,2,3
<b>19.</b>	Location	2,3,4,5
<b>20.</b>	Division	3,4,2,5
<b>21.</b>	Address	
<b>22.</b>	Comment	
<b>23.</b>	Emergency Contract	
<b>24.</b>	Edit Account	
<b>25.</b>	Notification	2,3,4,6
<b>26.</b>	Profile photo	
<b>27.</b>	Settings	2
<b>28.</b>	Database	1,2,6
<b>29.</b>	Edit Profile	
<b>30.</b>	Remove	
<b>31.</b>	Email	
<b>32.</b>	Phone	
<b>33.</b>	Date	
<b>34.</b>	Update Location	2,
<b>35.</b>	Guide	
<b>36.</b>	Tourist place	
<b>37.</b>	City	
<b>38.</b>	Police Station	

<b>39.</b>	Admin	1,2,3,4,5
<b>40.</b>	Add	

### 3.6.2 Attributes and Methods of Potential Classes

Analyzing the above table, we have categorized the verbs and convert them into method names. We put them to their respective classes and showed them in the table :

**Table :Potential Classes after General and Selection Criteria**

Potential Class	Nouns	Verb
Account	Verification code, User	Log in, Sign up, log out, send Verification code
User	User Id, Name, Email Profile photo, Phone number	Requires an account, can log in to the system, receives message, search, update user information, recover his/her password, add place information, add comment
Admin	Place, Member	Can edit places, can add places, can remove member, can remove comments
System	User, Places	Show popular Place(s), show viewing options, show searched items, can generate SMS and send SMS, sent emergency notification.
Notification	User Id, receiver, type	Will be generated and sent
Search	User, City, Place, Keyword,	Search by keyword, search by popular, search location type wise, search recommended, search by Police Station, search by place and City

Activity	Location, Admin, user	Store authentication information, new and update tour place information
Database	User, product	Update, retrieve, stored, removed
Location	Id, Division Name, Division photo	Division Name, Division photo are visiting by users.
City	Id, city name, City Description, Police Station Name, Police Station details, Police station Name	city name, City Description, Police Station Name, Police Station details, Police station Name are Visiting by Users
Tour Place	Id, Place Name, Type, Photo, Place Details, Guide name, Guide number, Route Details, Route type	Place Name, Type, Photo, Place Details, Guide name, Guide number, Route Details, Route type are update by users and admin

### 3.6.2.1 Attribute Selection

**Table : Attribute Selection of Classes**

Potential Class	Noun
Account	Verification code User
User	User Id Name Email Profile photo Phone number Emergency Contracts

Admin	Place Member
Search	User City Place Keyword
System	User Places
Notification	User Id receiver type
Activity	Location Admin User
Location	Id Division Name Division Photo

City	Id City Name City Description Police station Id Police station name Police station number
Database	User Place
Tour Place	Id Place Name Place Type Place Photos Place Description Place route route Details Guide name Guide Phone

*3.6.2.2 Method Identification:*

**Table : Methods of Class**

No.	Class	Methods
1	Account	<ul style="list-style-type: none"> <li>• signUp()</li> <li>• login()</li> <li>• signOut()</li> <li>• lockAccount()</li> <li>• sendVerificationCode()</li> <li>• recoverPassword()</li> <li>• verifyEmail()</li> <li>• verifyUser()</li> <li>• manageForgotPassword()</li> </ul>
2	User	<ul style="list-style-type: none"> <li>• setUserID()</li> <li>• getUserID ()</li> <li>• setName()</li> <li>• getName()</li> <li>• setEmail()</li> <li>• getEmail()</li> <li>• setPhoneNo()</li> <li>• getPhoneNo()</li> <li>• setEmergencyContractNo()</li> <li>• getEmergencyContractNo ()</li> <li>• addPlace()</li> <li>• Emergency()</li> </ul>
3	Admin	<ul style="list-style-type: none"> <li>• addPlace()</li> <li>• removePlace()</li> <li>• editPlace()</li> <li>• editProfile()</li> </ul>
4	System	<ul style="list-style-type: none"> <li>• viewOptions()</li> <li>• showSearchedItem()</li> <li>• generateSMS()</li> <li>• sendSMS()</li> <li>• setConnection()</li> </ul>

5	Notification	<ul style="list-style-type: none"> <li>• setNotificationID()</li> <li>• getNotificationID()</li> <li>• getSender()</li> <li>• setSender()</li> <li>• setReceiver()</li> <li>• getReceiver()</li> <li>• getDescription()</li> <li>• setDescription()</li> <li>• sendNotification()</li> <li>• generateNotification()</li> </ul>
6	Search	<ul style="list-style-type: none"> <li>• searchbyCity()</li> <li>• searchbyPlace()</li> <li>• searchbykeywords()</li> <li>• searchPopular()</li> <li>• searchTypewise()</li> <li>• searchRecommended()</li> </ul>
7	Activity	<ul style="list-style-type: none"> <li>• storeAuthenticationInfo()</li> <li>• storeLocationModificationInfo()</li> <li>• storeNewLocationInfo()</li> </ul>
8	Database	<ul style="list-style-type: none"> <li>• insertInfo()</li> <li>• updateInfo()</li> <li>• deleteInfo()</li> <li>• retrieveInfo()</li> </ul>
9	Location	<ul style="list-style-type: none"> <li>• getLocationID()</li> <li>• setLocationID()</li> <li>• getDivisionName()</li> <li>• setDivisionName ()</li> <li>• getDivisionPhoto()</li> <li>• setDivisionPhoto ()</li> </ul>

10	City	<ul style="list-style-type: none"> <li>• getCityID()</li> <li>• setCityID()</li> <li>• getCityName()</li> <li>• setCityName()</li> <li>• getCityDetails()</li> <li>• setCityDetails()</li> <li>• getPoliceStationID()</li> <li>• setPoliceStationID()</li> <li>• getPoliceStationName()</li> <li>• setPoliceStationName()</li> <li>• getPoliceStationDetails()</li> <li>• setPoliceStationDetails()</li> </ul>
11	Tour Place	<ul style="list-style-type: none"> <li>• getTourPlaceID()</li> <li>• setTourPlaceID()</li> <li>• getTourPlaceName()</li> <li>• setTourPlaceName()</li> <li>• getTourPlaceImages()</li> <li>• setTourPlaceImages()</li> <li>• getTourPlaceDescription()</li> <li>• setTourPlaceDescription()</li> <li>• getRoute()</li> <li>• setRoute()</li> <li>• getRouteDetails()</li> <li>• setRouteDetails ()</li> <li>• getGuideName()</li> <li>• setGuideName()</li> <li>• getGuideNumber()</li> <li>• setGuideNumber()</li> </ul>

### *3.6.2.3 Finalizing Classes*

To identify the final classes, it was required to check if there can be any hierarchies, merges, additional attributes, methods or classes. These identifications are given below:

1. There are two types of users in the system. So, the all-user class could be the parent class of the admin class and user class. But as the user class and the admin class has different attributes and methods there is no need of all user class.

System and User classes will have an additional method (`viewOptions()`) so that it can be used to get to other methods

### **3.6.3 Class Responsibility Collaborator**

CRC modeling stands for Class Responsibility Collaboration modeling. CRC modeling includes class cards and CRC diagram.

#### *3.6.3.1 Class cards*

Class cards show attributes, methods and collaborating class names along with their responsibility. A responsibility comprises one or more methods together. The potential classes for the system are- Account, User, Admin, System, Notification, Search, Database, Location, City, Tour Place. The proposed class cards of these classes are shown below.

**Table : Account Class Card**

<b>Responsibility</b>	<b>Collaborators</b>
Receiving information for registration	System
Checking availability of user and Admin Id	System
Receiving id and password	Admin, User
Creating Account	Database
Manage activity	Activity
Verifying login	System

**Table : User Class Card**

<b>Responsibility</b>	<b>Collaborators</b>
Authentication	Account
Add Place Info	Database
Request Emergency	System

**Table 22: Admin**

<b>Responsibility</b>	<b>Collaborators</b>
Authentication	Account
Add Place Info	Location
Edit Place Info	Location

**Table 23: System**

<b>Responsibility</b>	<b>Collaborators</b>
Get searched Items	Search
Generate Notifications	Notification
Generate Emergency Message	Notification

**Table : Notification**

<b>Responsibility</b>	<b>Collaborators</b>
Gets Notification	User
Generate Notifications	System

**Table : Search Card**

<b>Responsibility</b>	<b>Collaborators</b>
Generating Search	System
Search for Tour Place	Location
Search for City	Location

**Table : Database Card**

<b>Responsibility</b>	<b>Collaborators</b>
Store Admin Info	Admin
Store User Info	User
Store/Update Tour Place City	Location

**Table : Location Card**

<b>Responsibility</b>	<b>Collaborators</b>
Store/Update Tour Place City	Database
Search for Tour Place	Search
View Tour Place	Admin/User

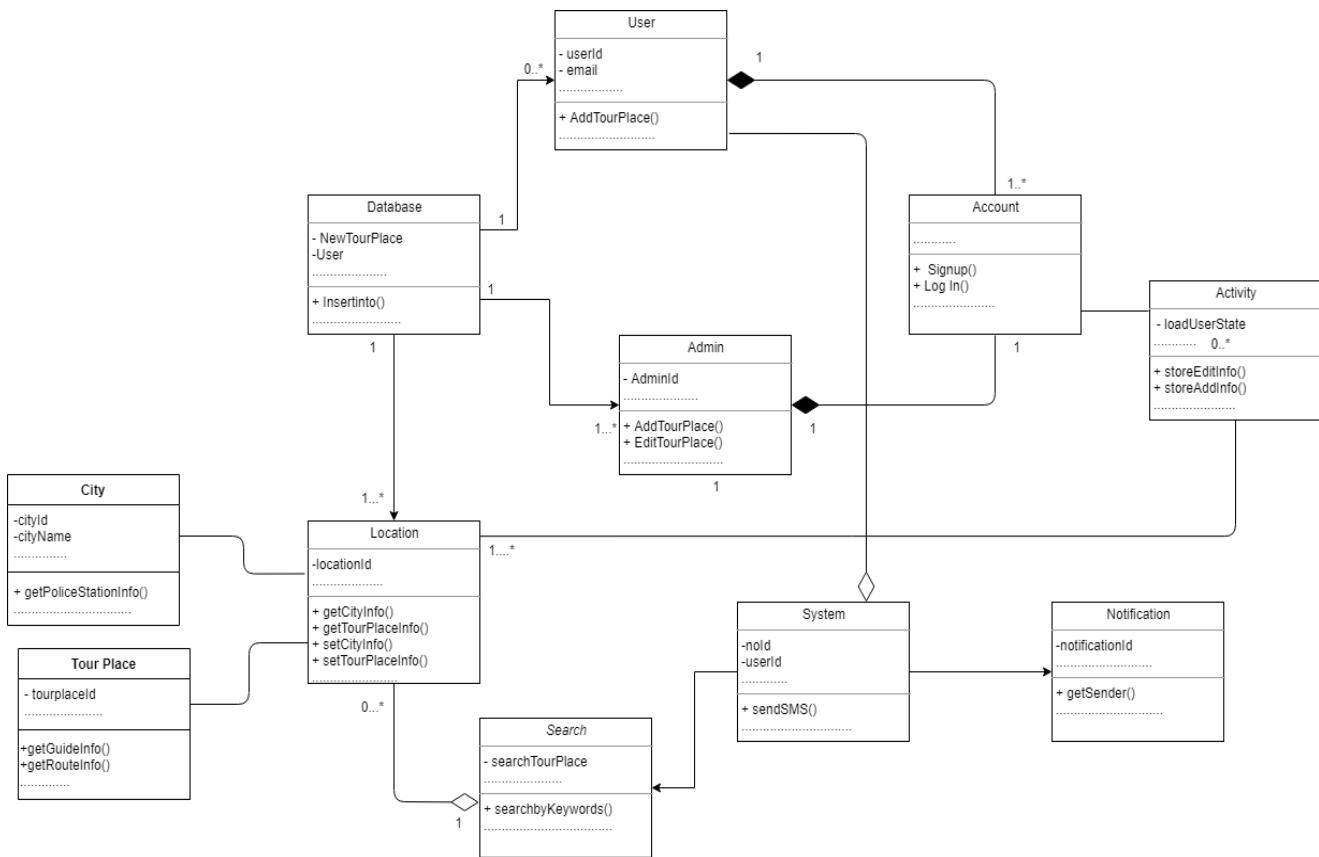
**Table : City Card**

<b>Responsibility</b>	<b>Collaborators</b>
Update/View city Information	Location

**Table : Tour Place Card**

<b>Responsibility</b>	<b>Collaborators</b>
Update/View Tour Place Information	Location

### 3.6.3.2 UML Diagram



## 3.7 DATA FLOW MODEL

A data flow model is a representation of the flow and exchange of information within a system. Data flow models are used to graphically represent the flow of data in an information system by describing the processes involved in transferring data from input to file storage and reports generation. A data flow model may also be known as a data flow diagram (DFD). Data flow modeling can be used to identify a variety of different things, such as:

- Information that is received from or sent to other individuals, organizations, or other computer systems.
- Areas within a system where information is stored and the flows of information within the system are being modeled.
- The processes of a system that act upon information received and produce the resulting outputs.

Data Flow Diagrams of “Visit Bangladesh” is given below:

7.1.0 : level 0

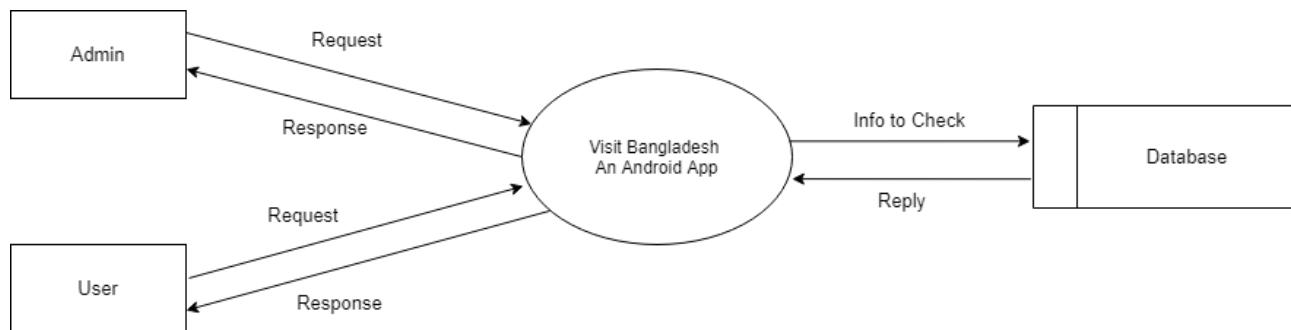
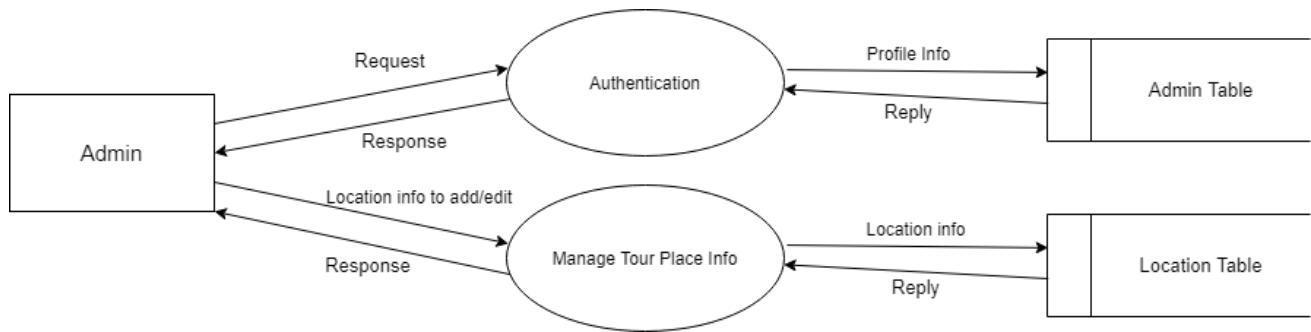


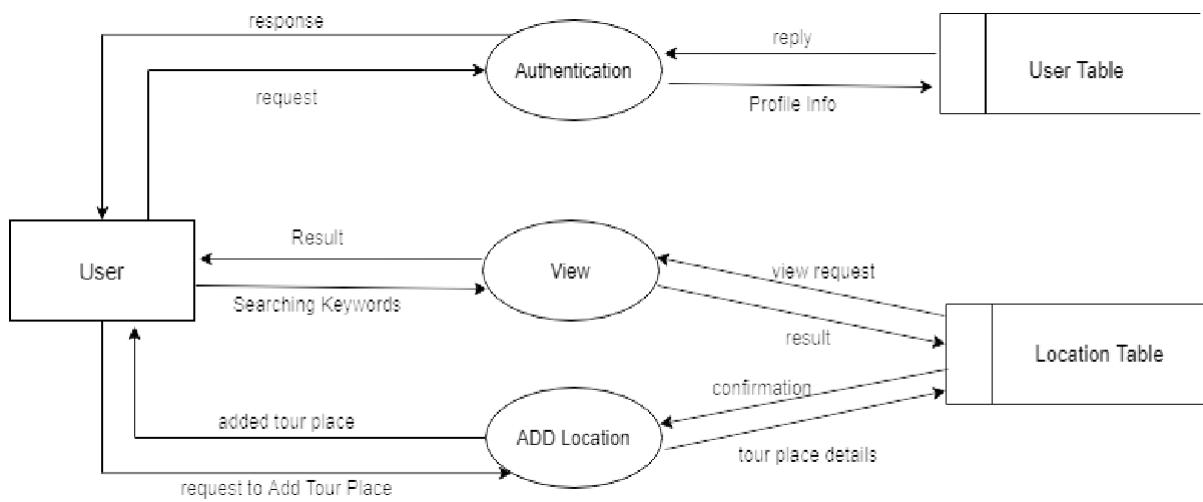
Figure 25: level 0 for Visit Bangladesh

### 7.1.1 : Level 1.1



*Figure 26: level 1.1 for Visit Bangladesh*

### 7.1.2 : Level 1.2



*Figure 27: level 1.2 for Visit Bangladesh*

### 7.3.1 : Level 2.1

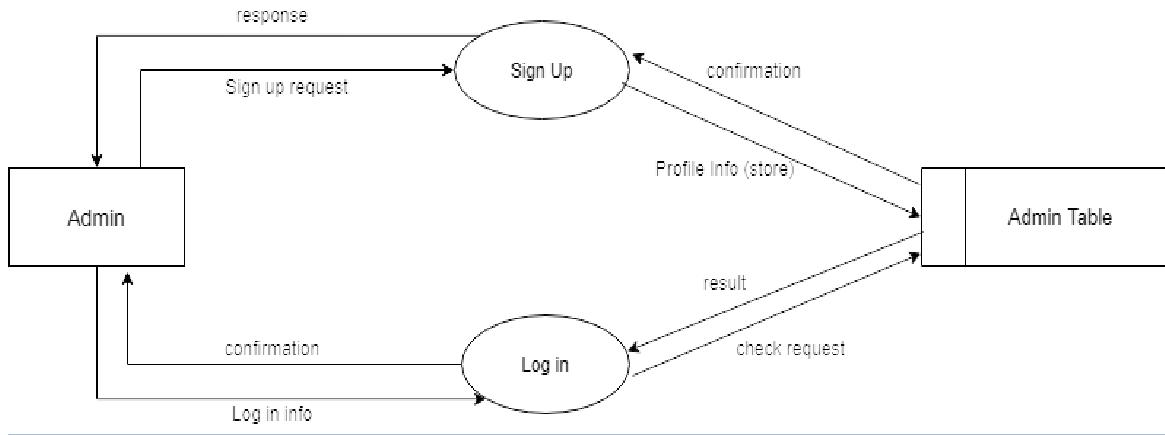


Figure 28: level 2.1. for Visit Bangladesh

### 7.3.2 Level 2.2

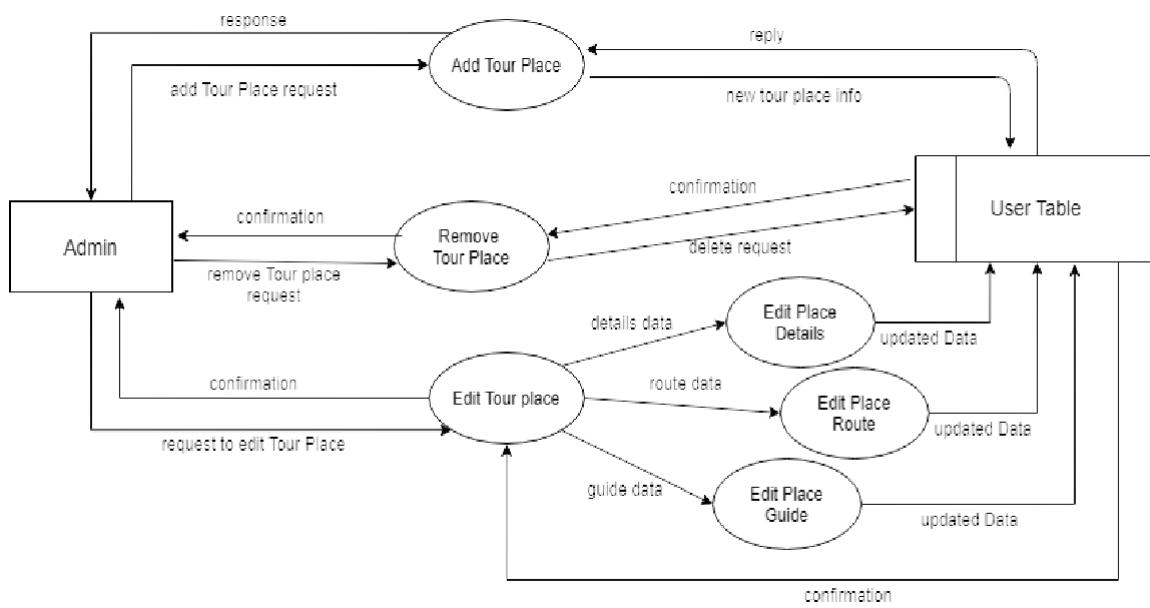


Figure 29: level 2.2 for Visit Bangladesh

### 7.3.3 Level 2.3

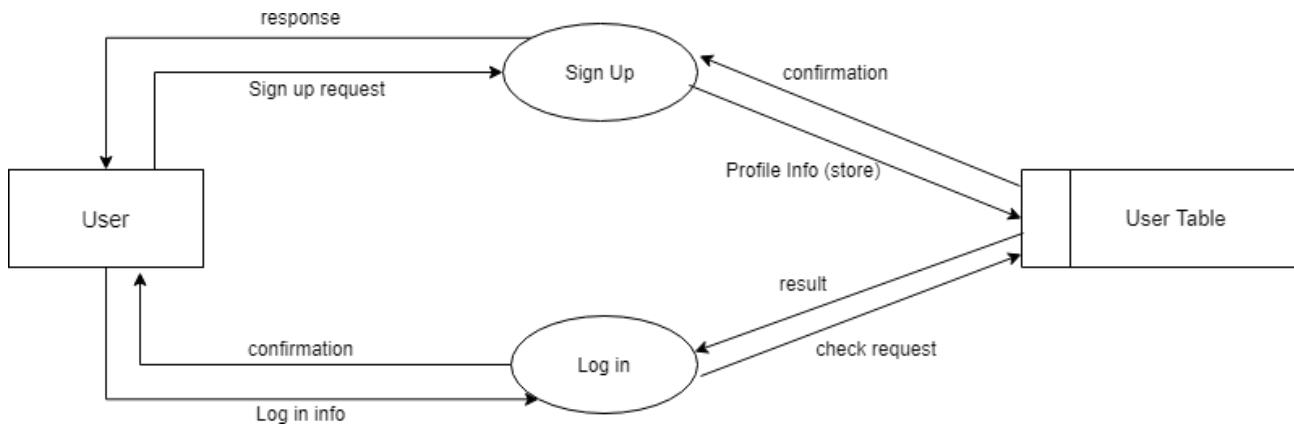


Figure 30: level 2.3 for Visit Bangladesh

### 7.3.4 Level 2.4

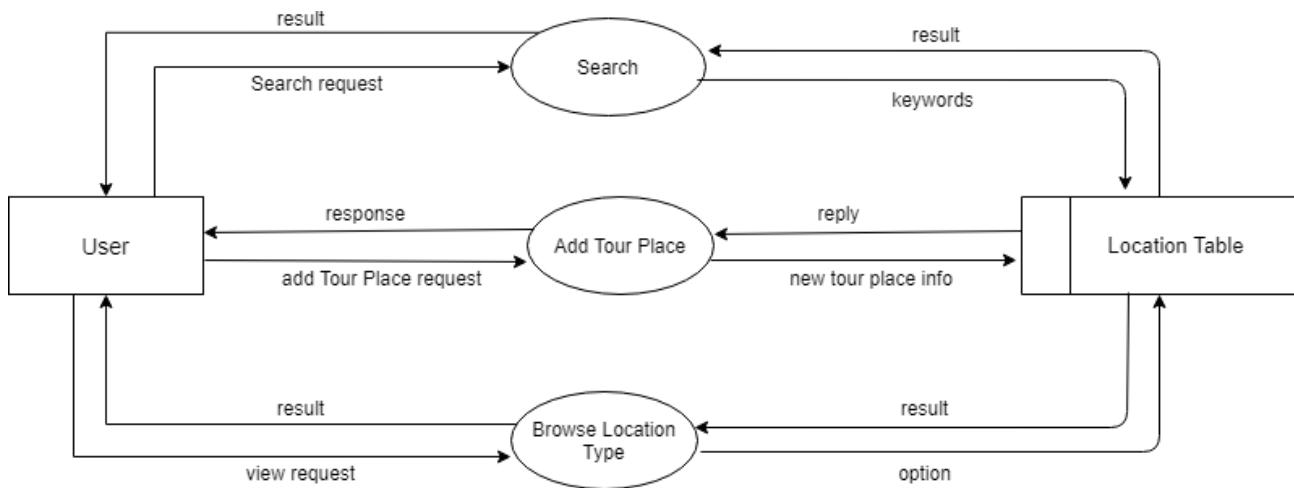
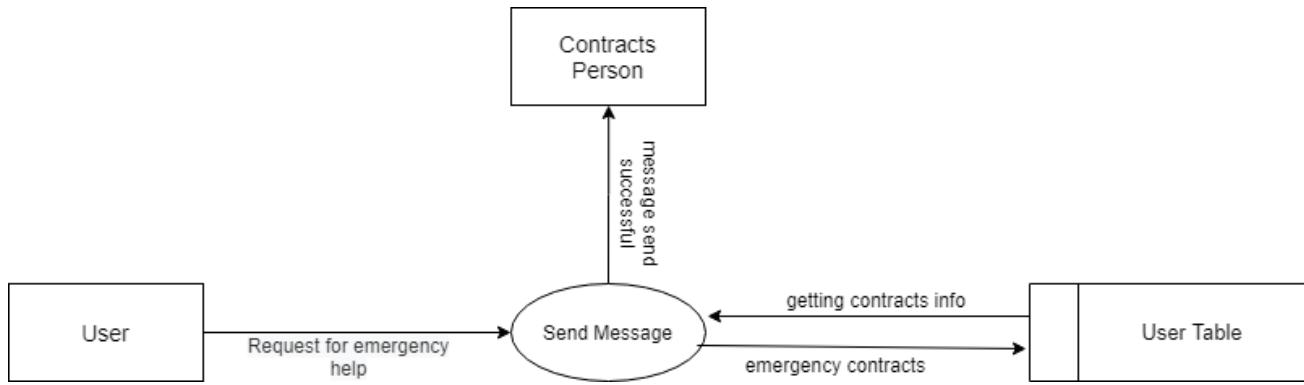


Figure 31: level 2.4 for Visit Bangladesh

### 7.3.5 Level 2.5

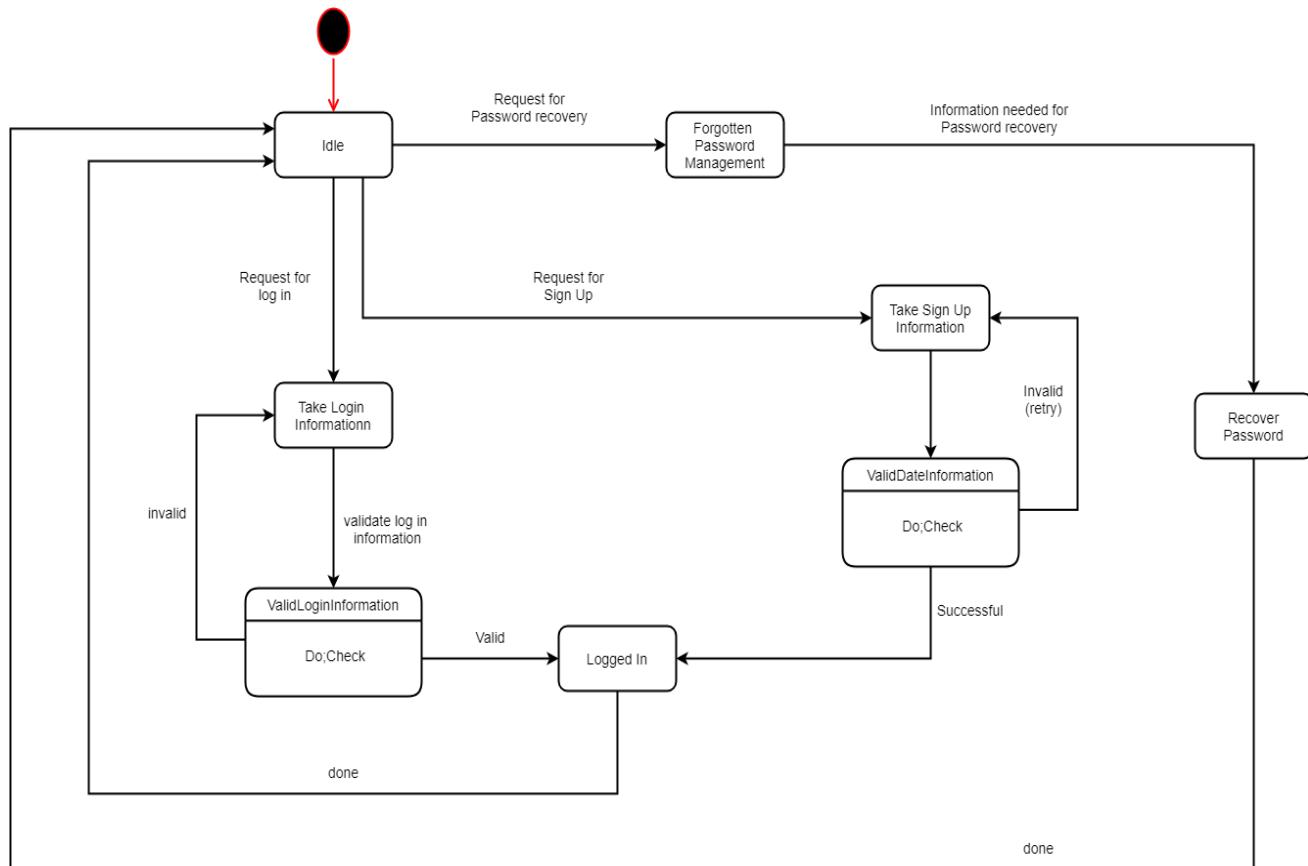


*Figure 32: level 2.5 for Visit Bangladesh*

## 3.8 BEHAVIORAL MODEL

The behavioral model indicates how software will respond to external events. There are two different behavioral representations. The first indicates how individual class changes state based on external events and the second shows the behavior of the software as a function of time. State diagram shows the state in a module of a user. State diagram of this project are following:

### 3.8.1 State diagram of Account:



*Figure 33:State diagram of Account*

### 3.8.2 State diagram of Admin:

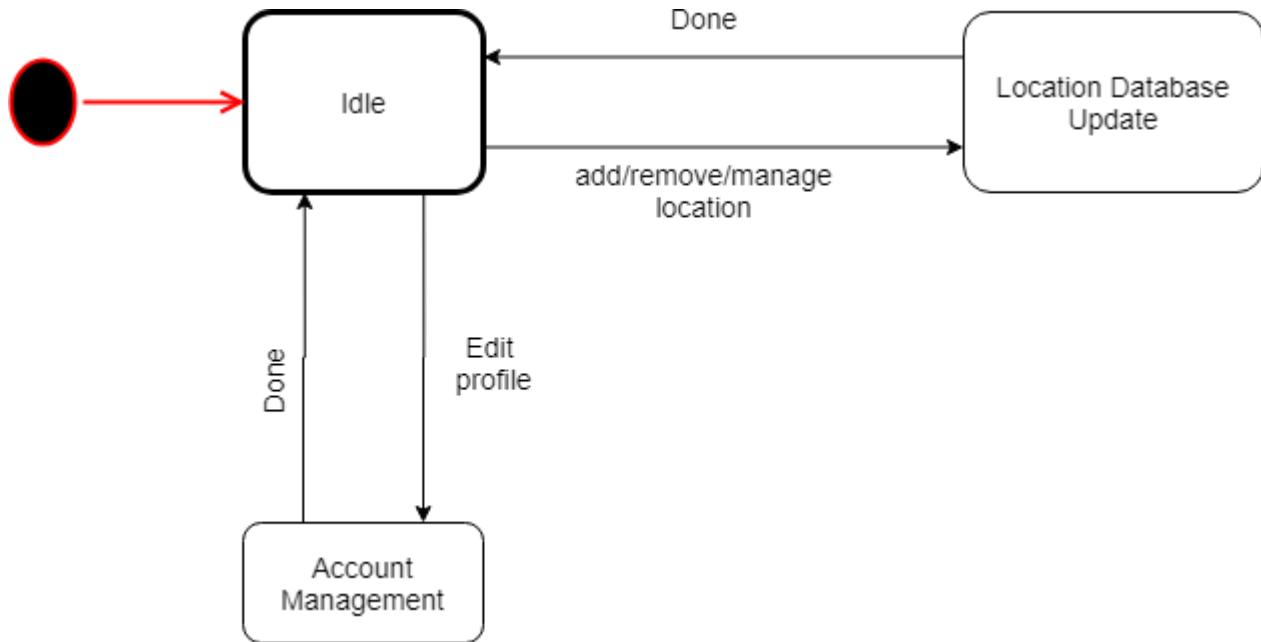


Figure 34: State diagram of Admin:

### 3.8.3 State diagram of Notification:

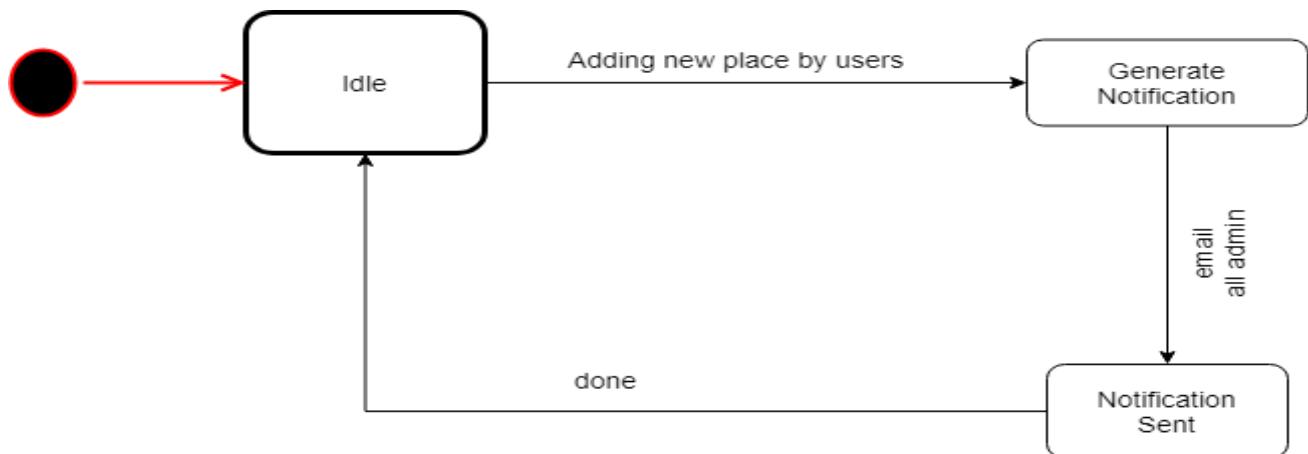


Figure 35: State diagram of notification

### 3.8.4 State diagram of User:

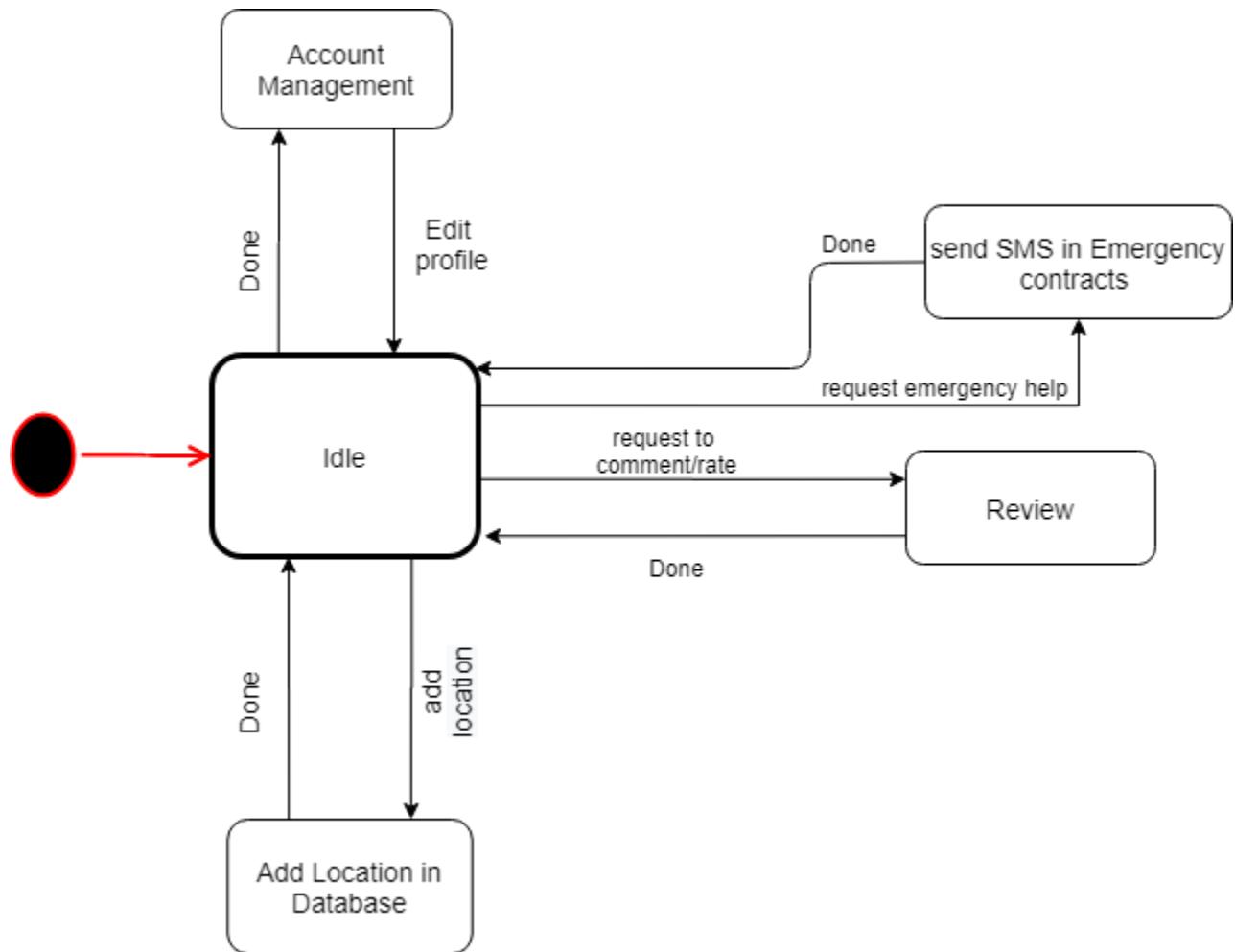


Figure 36:State diagram of User

### 3.8.5 State diagram of System:

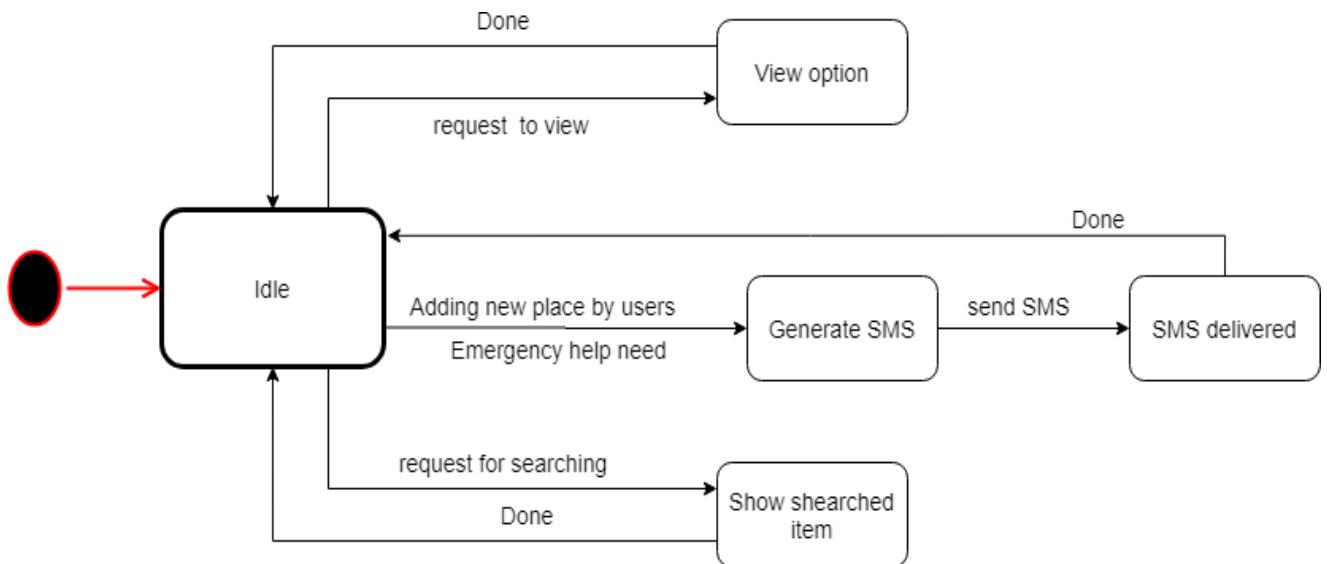


Figure 37:State diagram of System

### 3.8.6 State diagram of Database:

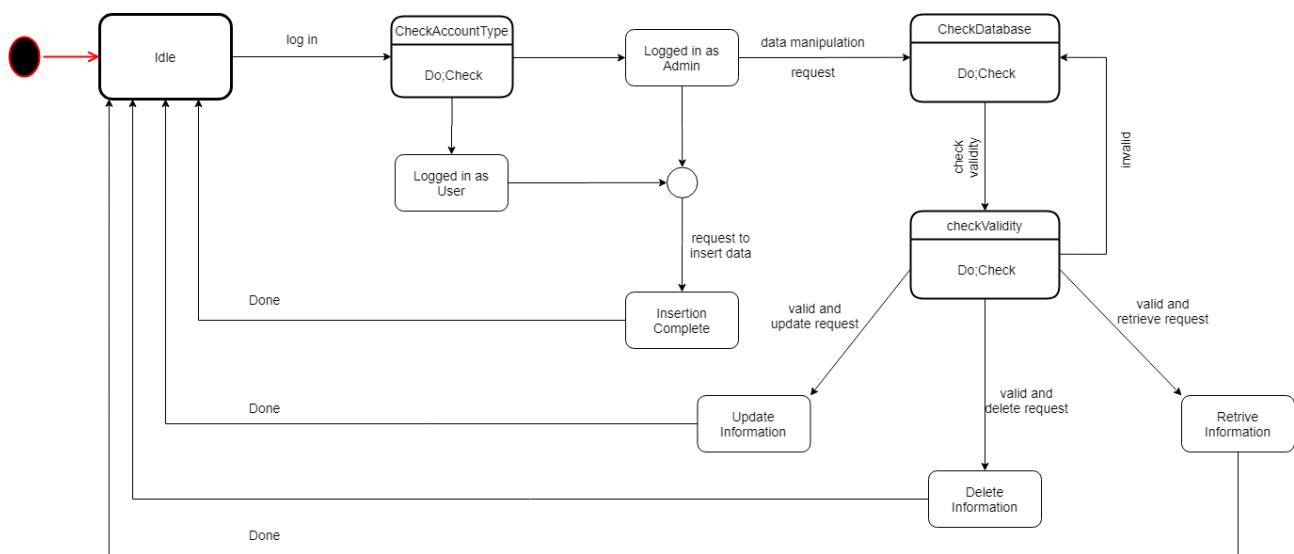


Figure 38:State diagram of Database

### 3.8.7 State diagram of Search:

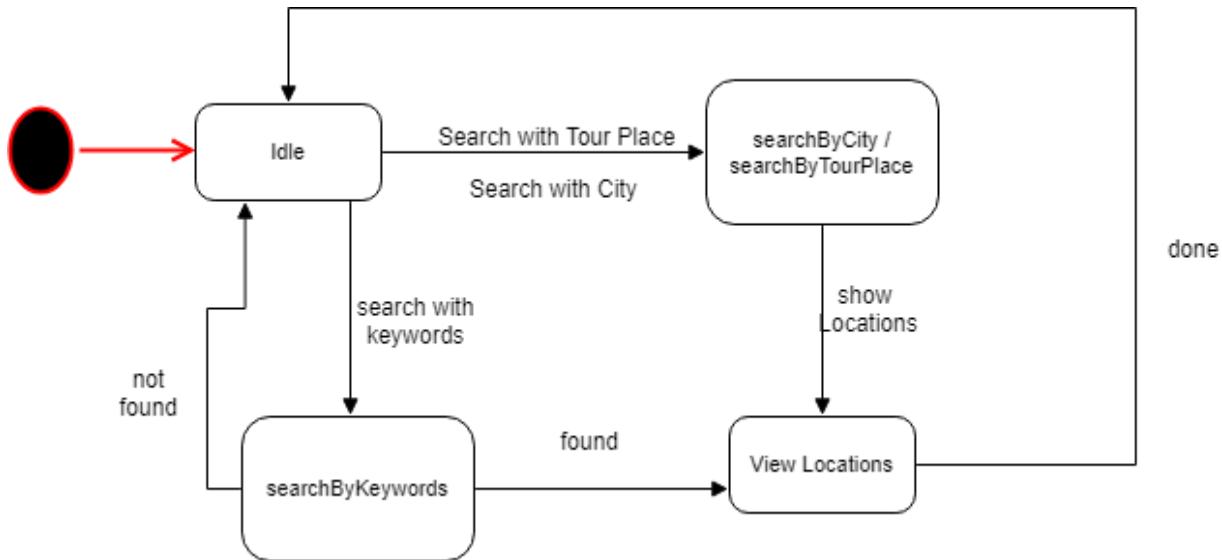


Figure 39:State diagram of Search

### 3.8.8 State diagram of Location:

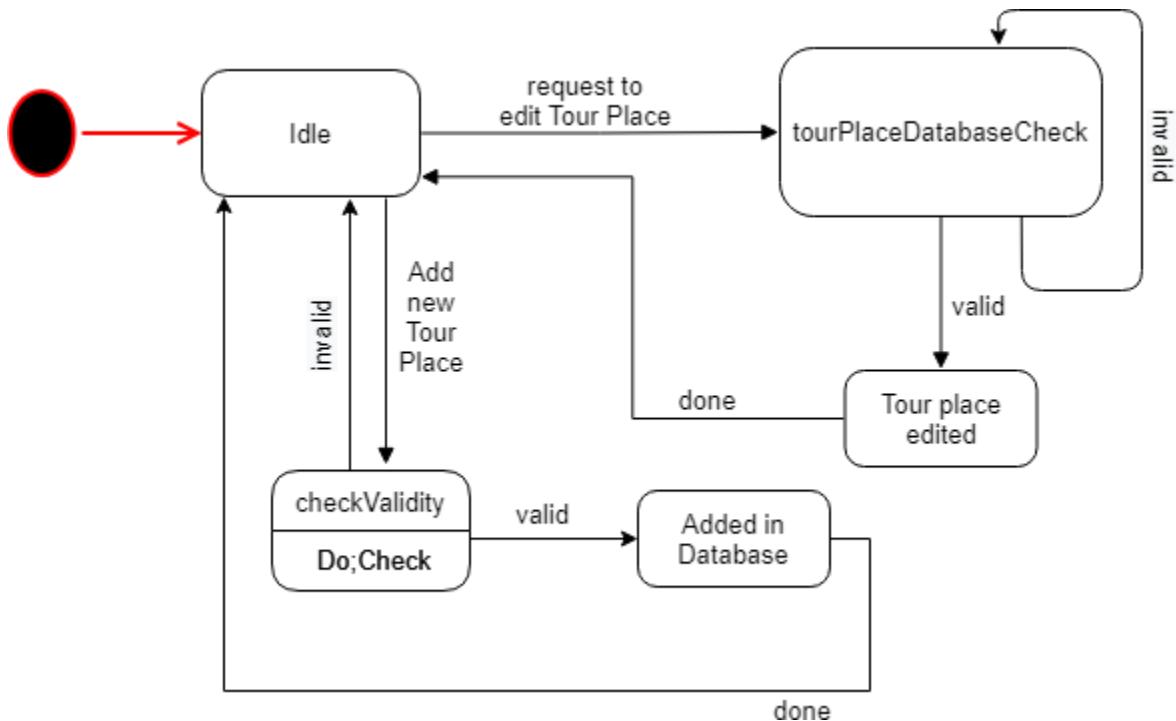
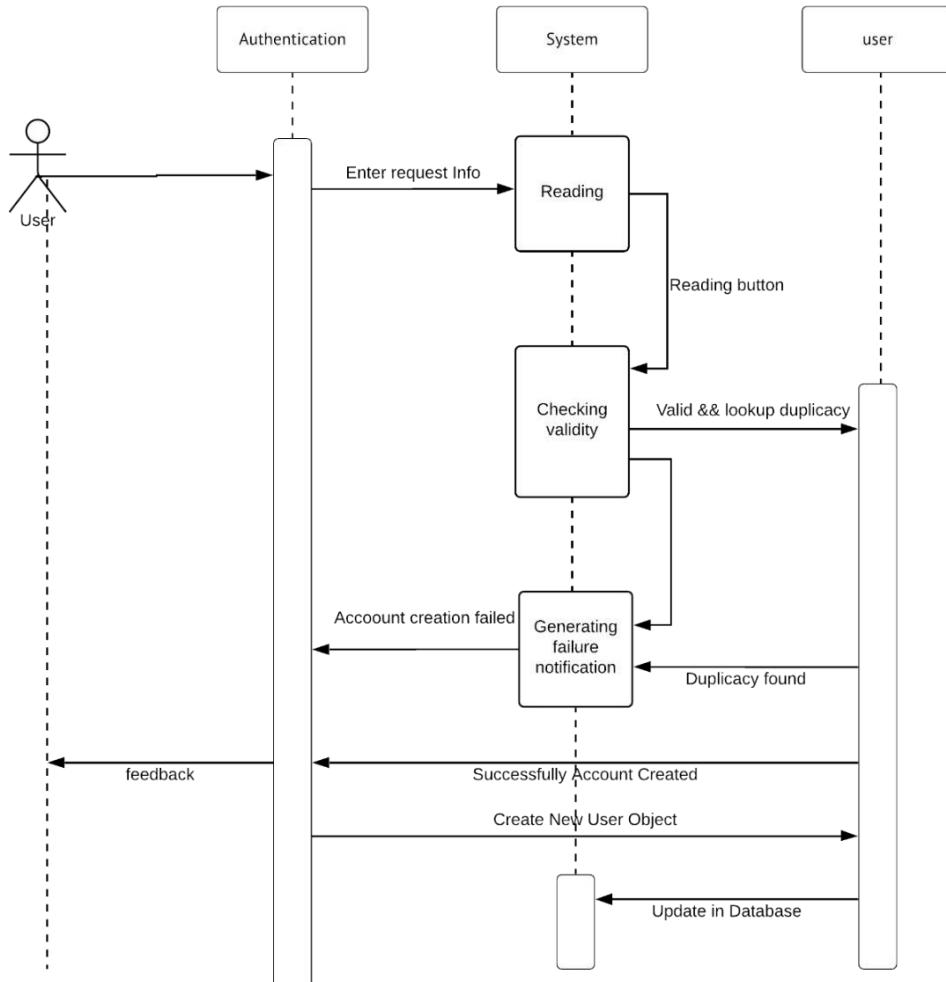


Figure 40:State diagram of Location

## 3.9 Sequence diagram

A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart. A sequence diagram shows object interactions arranged in time sequence. We have shown the sequence diagram of three modules- Homepage, personal account and administration.

### 3.9.1 Sequence diagram of Sign Up



*Figure 41:Sequence diagram of Sign Up*

### 3.9.1 Sequence diagram of Sign In

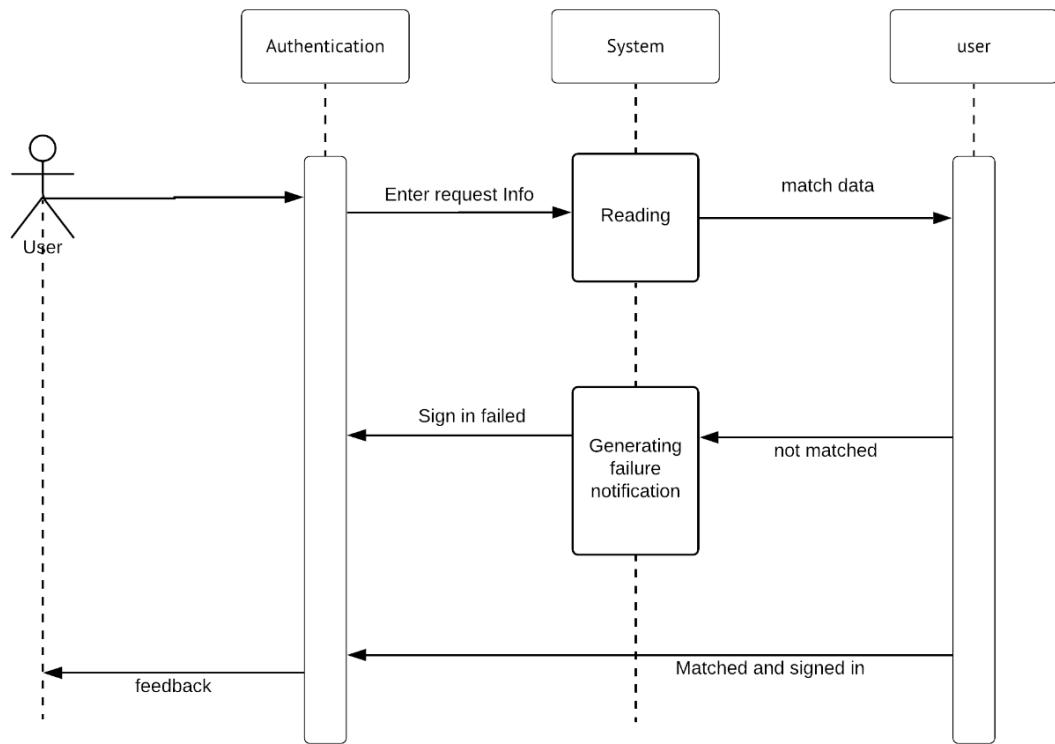


Figure 42: Sequence diagram of Sign In

### 3.9.1 Sequence diagram of Search Location:

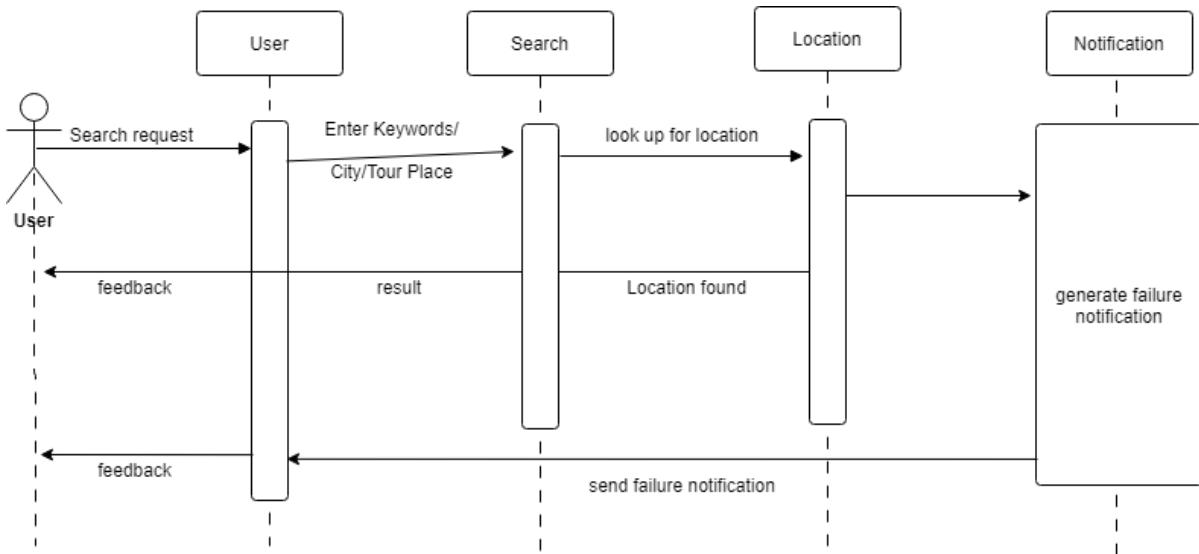


Figure 43:Sequence diagram of Search Location

### 3.9.1 Sequence diagram of Edit Location:

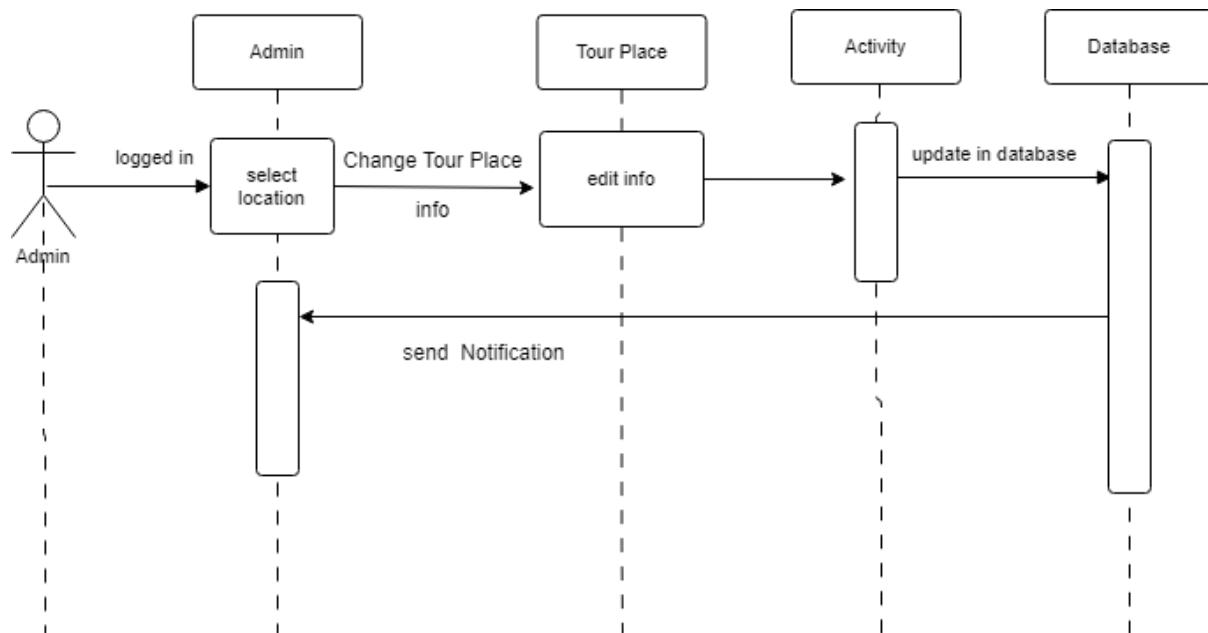


Figure 44:Sequence diagram of Edit Location

### 3.9.1 Sequence diagram of Delete Location:

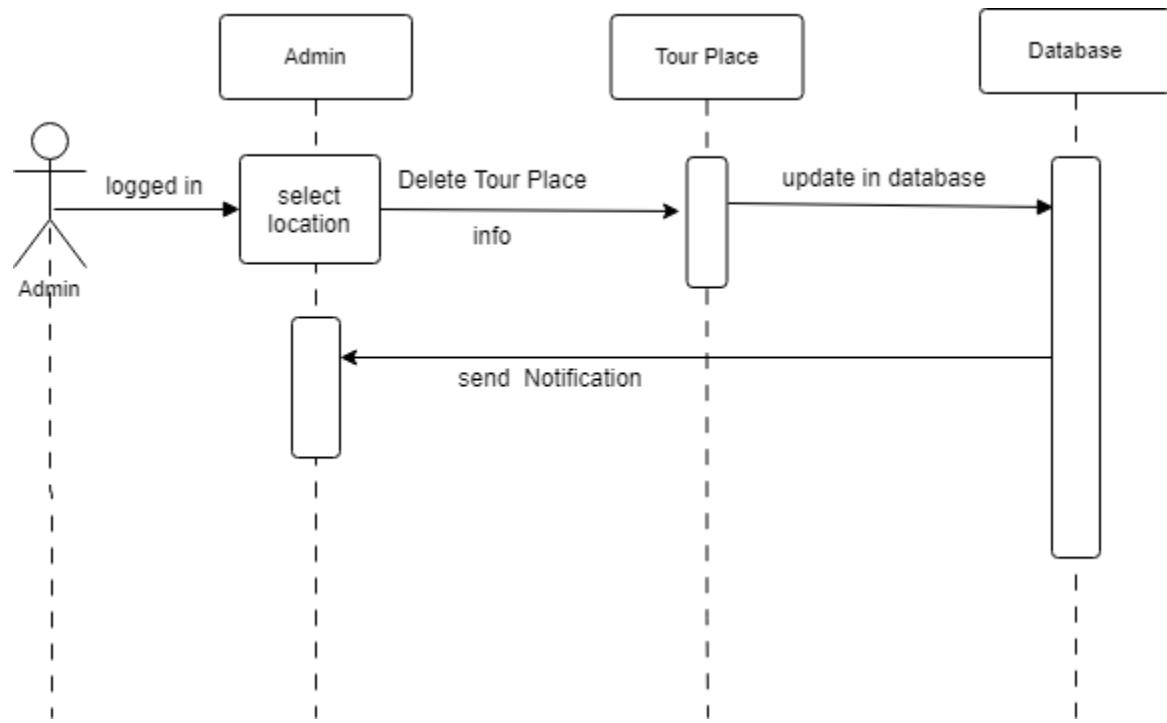


Figure 45: Sequence diagram of Delete Location:

### 8.2.6 Sequence diagram of Browse Location:

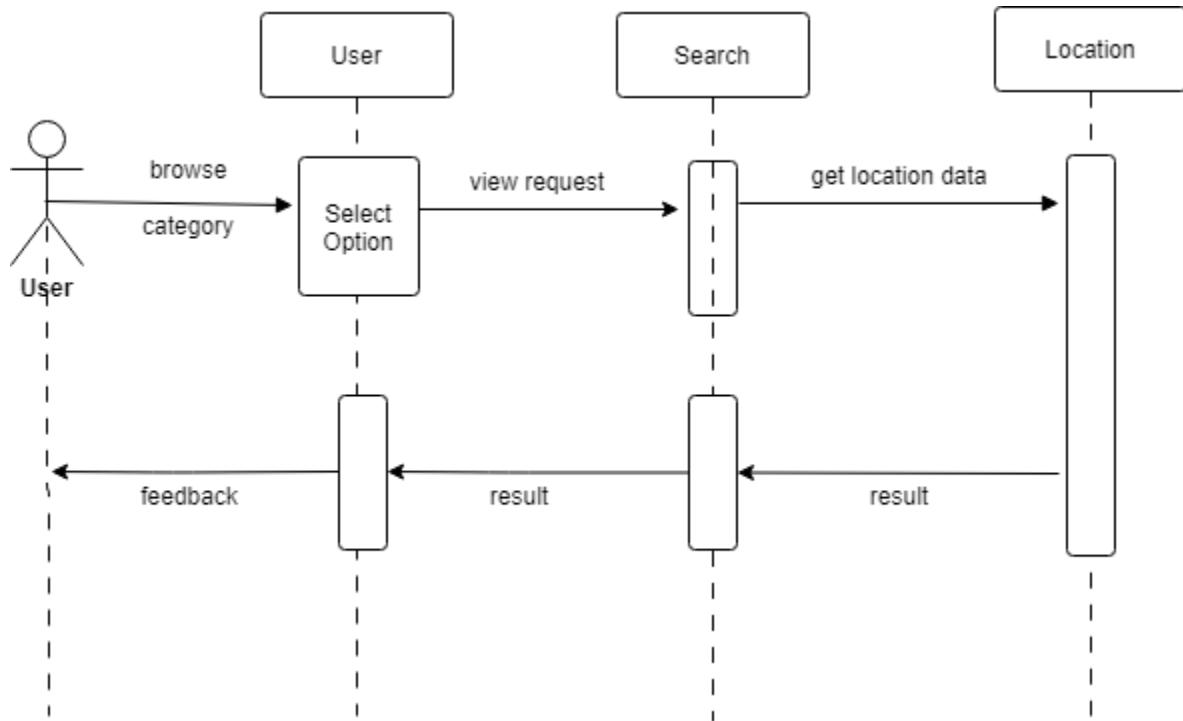
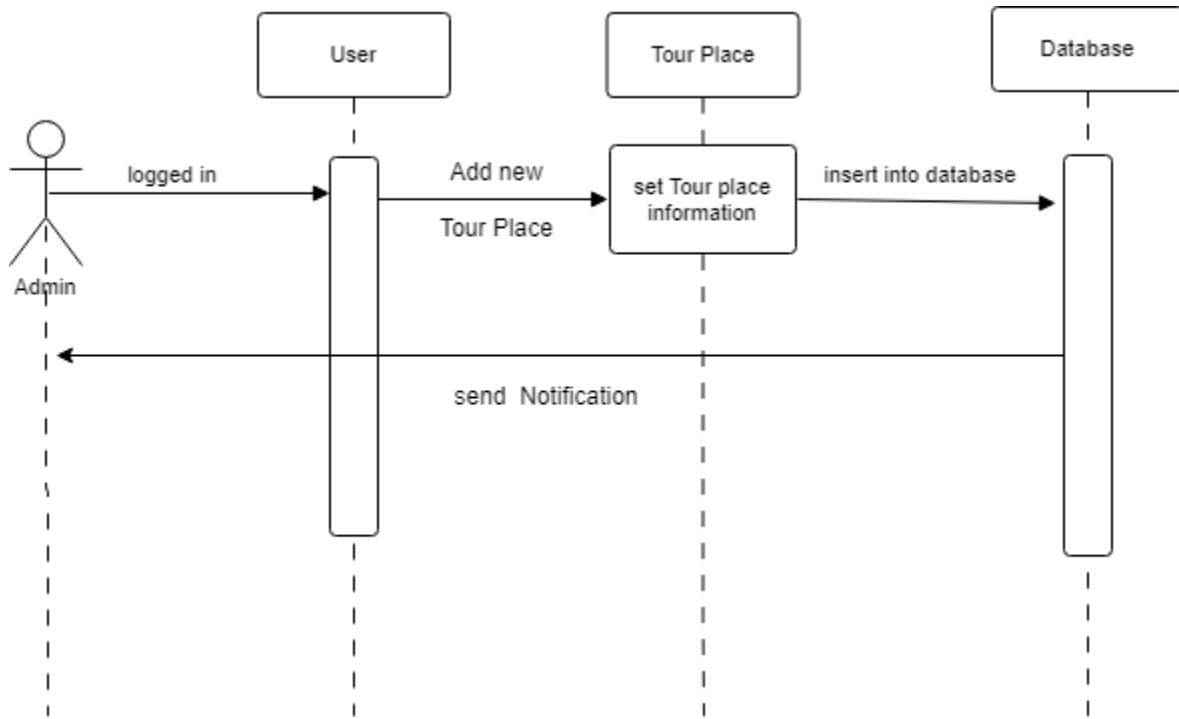


Figure 46:Sequence diagram of Browse Location

### 3.9.7 Sequence diagram of Add Location:



*Figure 47: Sequence diagram of Add Location*

We had developed our android app (“Visit Bangladesh”) through android studio. To set up, this app in any android app. We have Apk file and can be sent and installed. Through apk, app can install.

## Designs

### [Authentication widget](#)

This widget provides interface login/signup and creating user account.

### [Details of Places widget](#)

This widget will fetch all information of specific place and show into the app.

### [Google Maps widget](#)

This will enables google maps and retrieve co-ordinates.

### [Adding new places widget](#)

This widget provides UI for adding new places to firebase cloud firestore.

### [Weather update and Currency Converter](#)

Both widgets run through navigation sidebar and open individually UI for both features.

### [Searching Place](#)

Algorithm and Pages are designed to perform search by famous place name, division/district wise

## Activities

### [Authentication](#)

#### `email_register_page.dart`

This will register new user email and provide username.

#### `email_login_page.dart`

This will help user to login after authentication

#### `MyUserAuth.dart`

This will authenticate user whether previously account is created with same email or not.

### [Adding new places widget](#)

#### `add_tourist_place.dart`

here, users are asked to input various information of tourist place like famous spot

name, description, locally available police station and guide details etc.

#### add\_image.dart

Adding details of places, same time related image needs to be updated from gallery or google drive

#### my\_map.dart

Button click will open google maps and placed marker at map drag latitude and longitude to firebase database.

#### [Google Maps widget](#)

#### show\_route\_map.dart

When selecting specific tourist to view details, there in bottom right corner, button will navigate to location in map where that place is located. Also calculate distance.

#### Places details widget

#### [Route Details, Police Station Details and Guide Details](#)

#### details.dart

Created button and fetched from database using place model

#### place\_model.dart

Interface for conversion of data into JSON and fetching information from JSON

#### [Currency Converter and Emergency Contact](#)

#### emergency\_contract.dart

In this, there are some predefined help desk contacts and can make call directly from app.

#### currency\_convert.dart

In this, user can convert almost more than 25 countries currencies from their native currencies

#### [Weather Update](#)

#### weatherPage.dart

Provide UI for weather page

#### detailPage.dart

Can change place and get updated weather of tourist place

## Searching Place

grid\_search.dart & search\_widget.dart

Search performed by placename, or by their division/ district

# USER MANUAL

## 4.1 Login

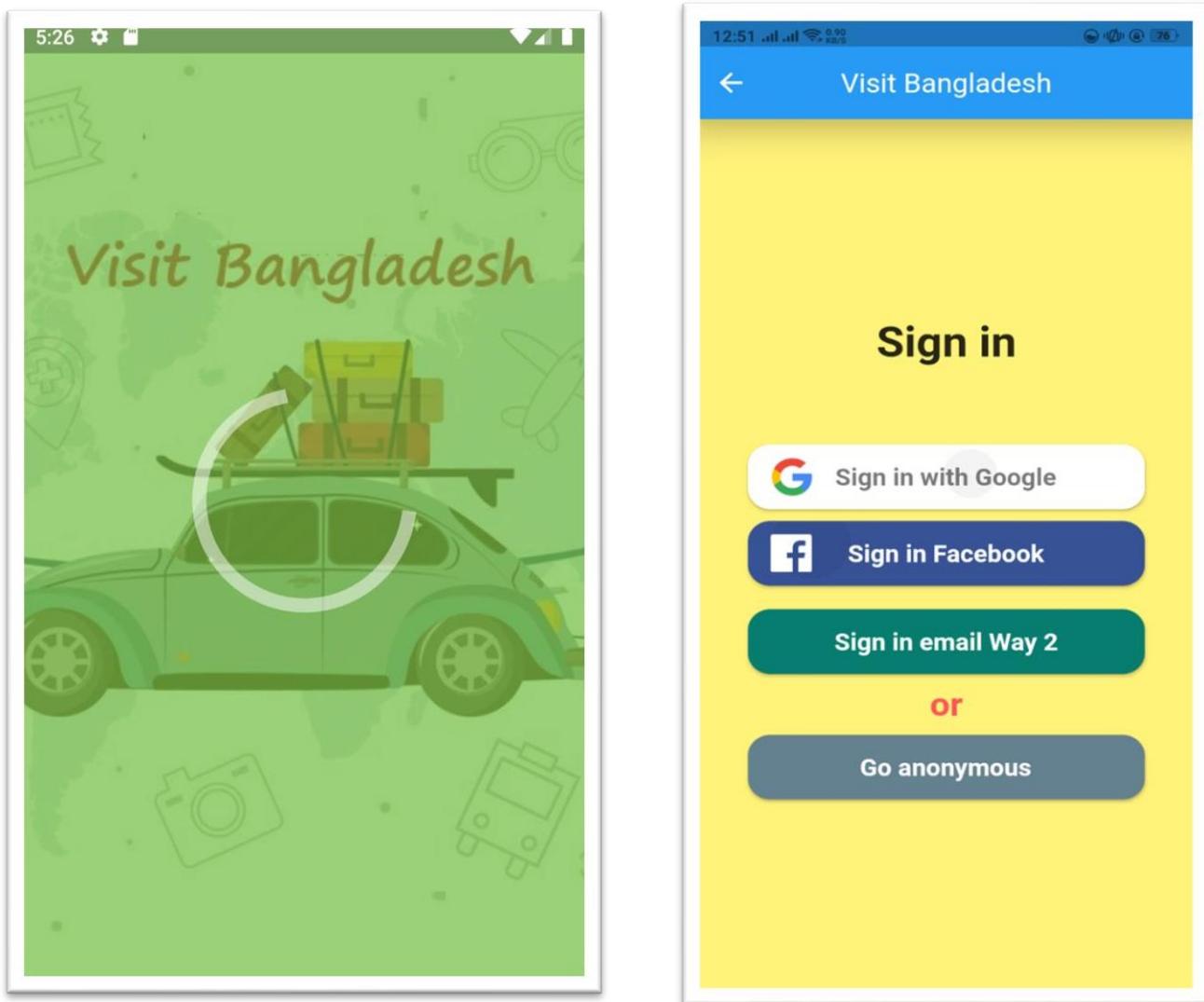


Figure 48:Splash and Sign In

By clicking app icon, we will be opened to login/signup page with splash page. You can enter anonymously or login with Gmail, further you will be opened to home page where there is navigation sidebar and welcome page.

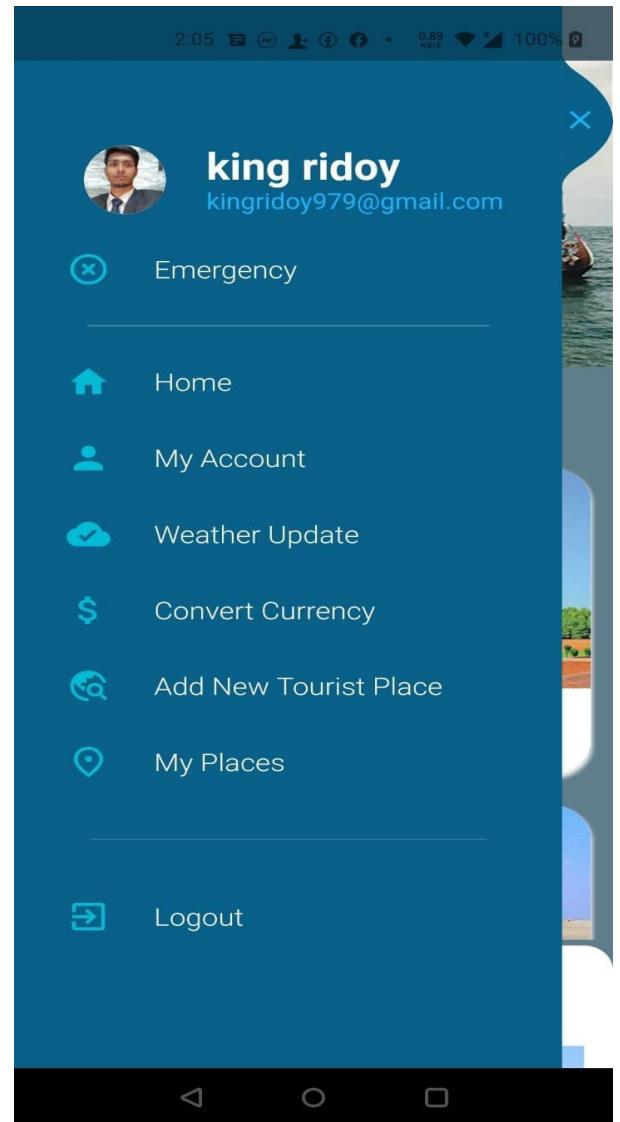


Figure 49: Home Page & sidebar menu

After log in we get the 3rd image, where there is landing page for app where all 8 divisions wise tourist places are available and recommended places with details. Here, navigation side bar has home page, my account details, Weather update, convert currency and add new tourist places pages.

## 4.2 My Account

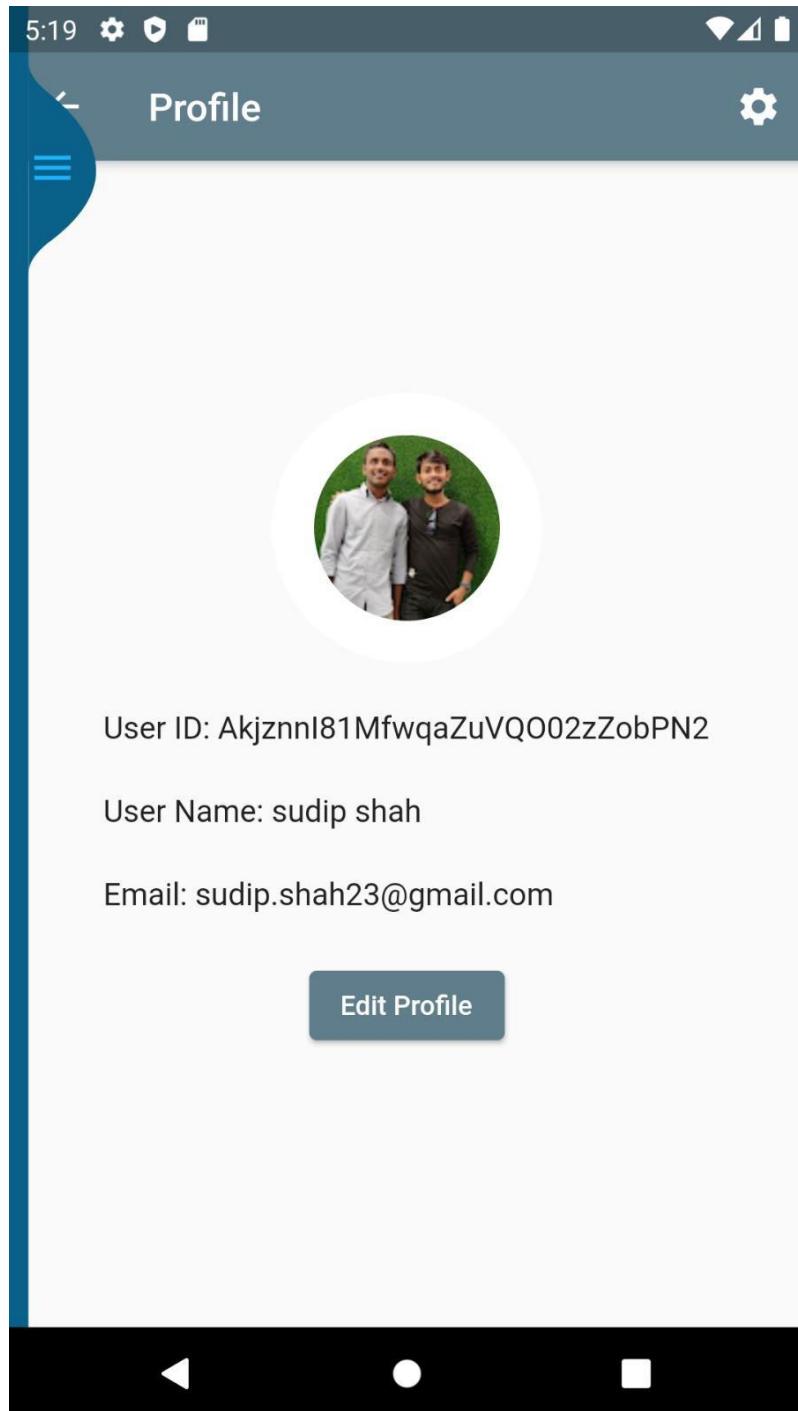


Figure 50:User-Account Info

By clicking, my account in navigation sidebars, we will enter into new page where user details with their image, userid, username and email as well as options to edit profile are available. User must login with their Gmail otherwise, details are unavailable.

## 4.3 Weather Update

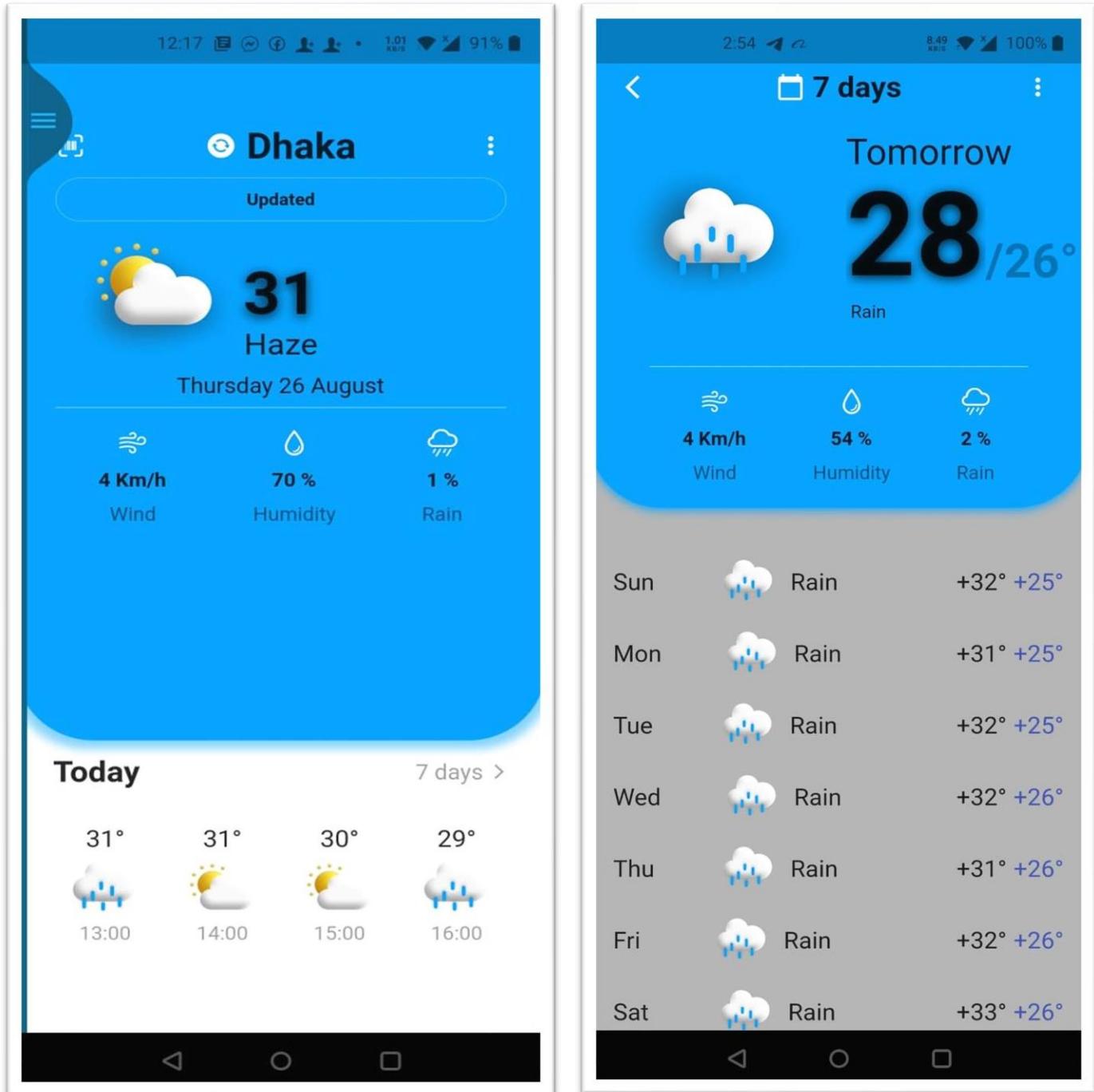
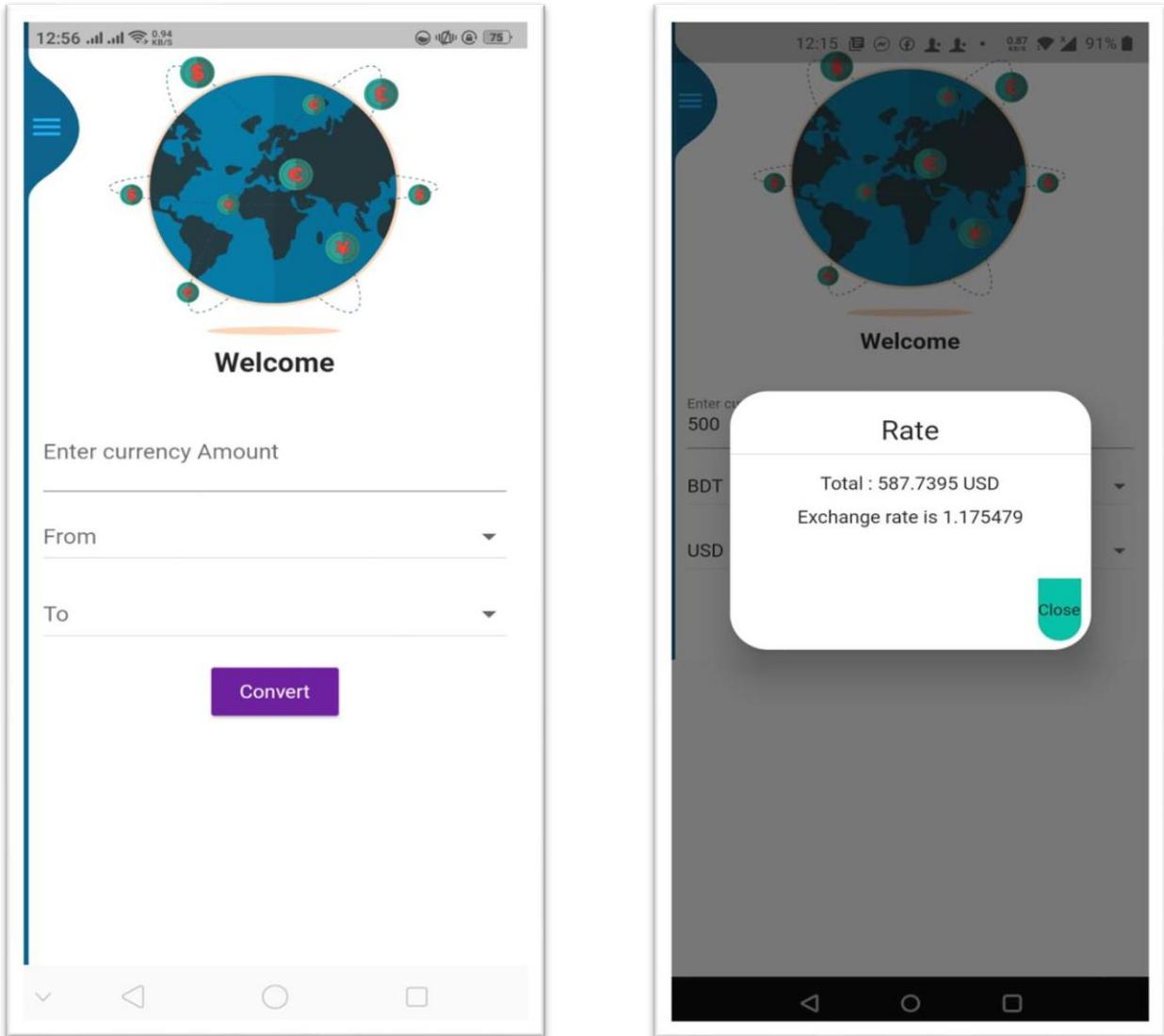


Figure 51: Shows weather

Here, user can see the update about weather of destined places for next days.

## 4.4 Currency Converter



*Figure 52:Currency Converter Interface*

This feature is very helpful for the foreigners. You can easily navigate through sidebar. They have different currencies for which they easily convert to any currencies here entering amount and converting currencies.

## 4.5 Adding new tourist places

The figure consists of two side-by-side screenshots of a mobile application interface for adding tourist places.

**Left Screenshot:**

- Header: "Add Tourist Places".
- Text: "Fill All The Information".
- Form fields:
  - "Select Division"
  - "Select District"
  - "Select Upazilla"
  - "OR"
  - "Enter CityCorporation Ward Name"
  - "Place Name"
  - "Place Details"
  - "Police Station Details":
    - "Police Station Name"

**Right Screenshot:**

- Header: "Add Tourist Places".
- Form sections:
  - "Phone Number"
  - "Incharge Name"
  - "Route":
    - "Bus Details"
    - "Train Details"
    - "Lounge Details"
  - "Tour Guide Information":
    - "Guide Name"
    - "Phone Number"
    - "Guide Address"
  - "Upload" (button)
  - "Google Maps" (button)
- Bottom button: "SAVE INFO".

Figure 53: Form Adding new places

Navigating to this page, you will have a long form which will ask you to input valid details about various valid tourist places in your locality like division, district, route to reach there, famous place name and description about places, available police station details and tour guide information as well as attracting place's images and location from google maps.

## 4.5.1 Google Maps

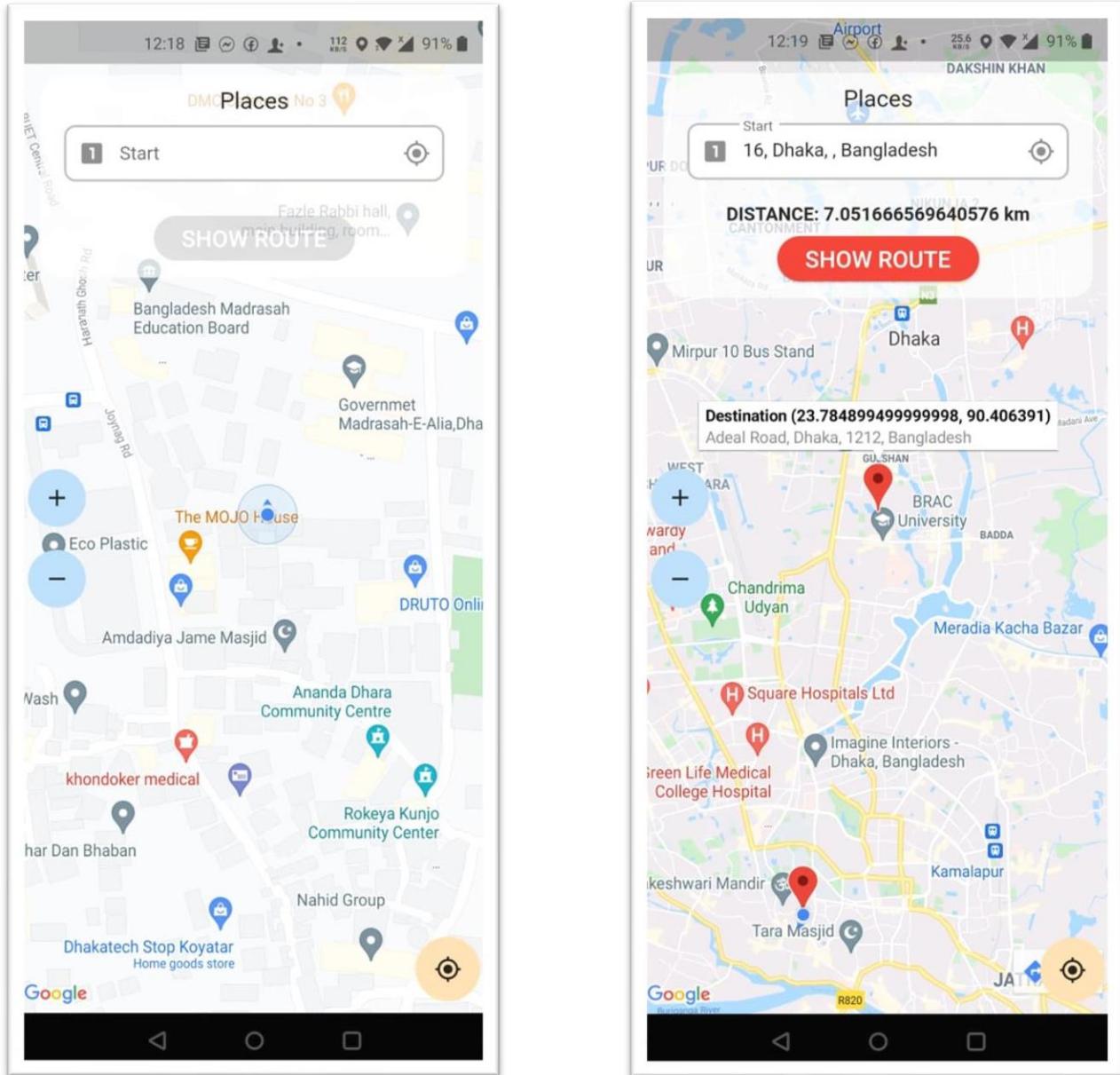
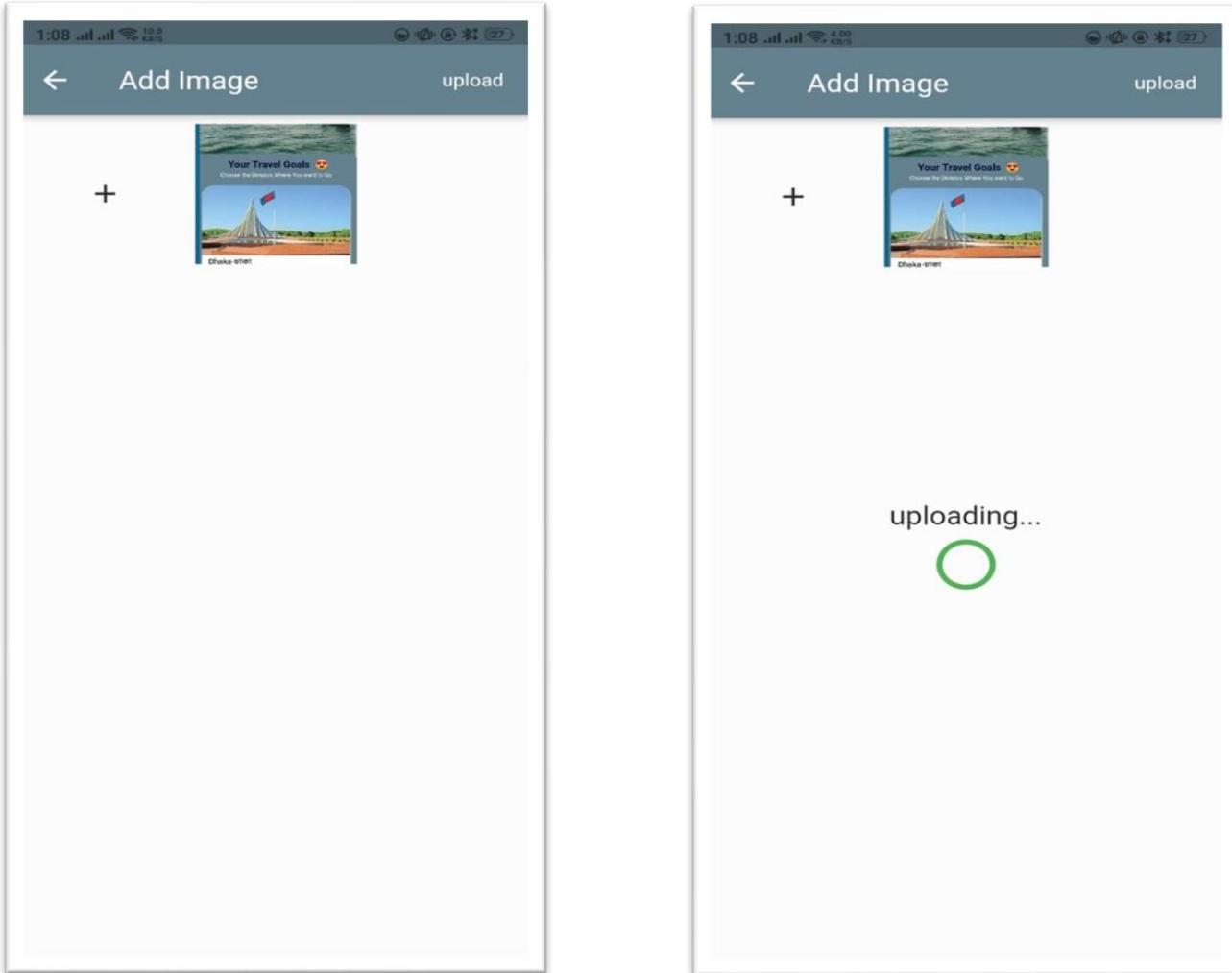


Figure 54:Maps Navigation

Opening Google's maps will help to select places by dragging around tourist places and same coordinates will be updated to database of that tourist places. You can calculate distance.

## 4.5.2 Upload Images

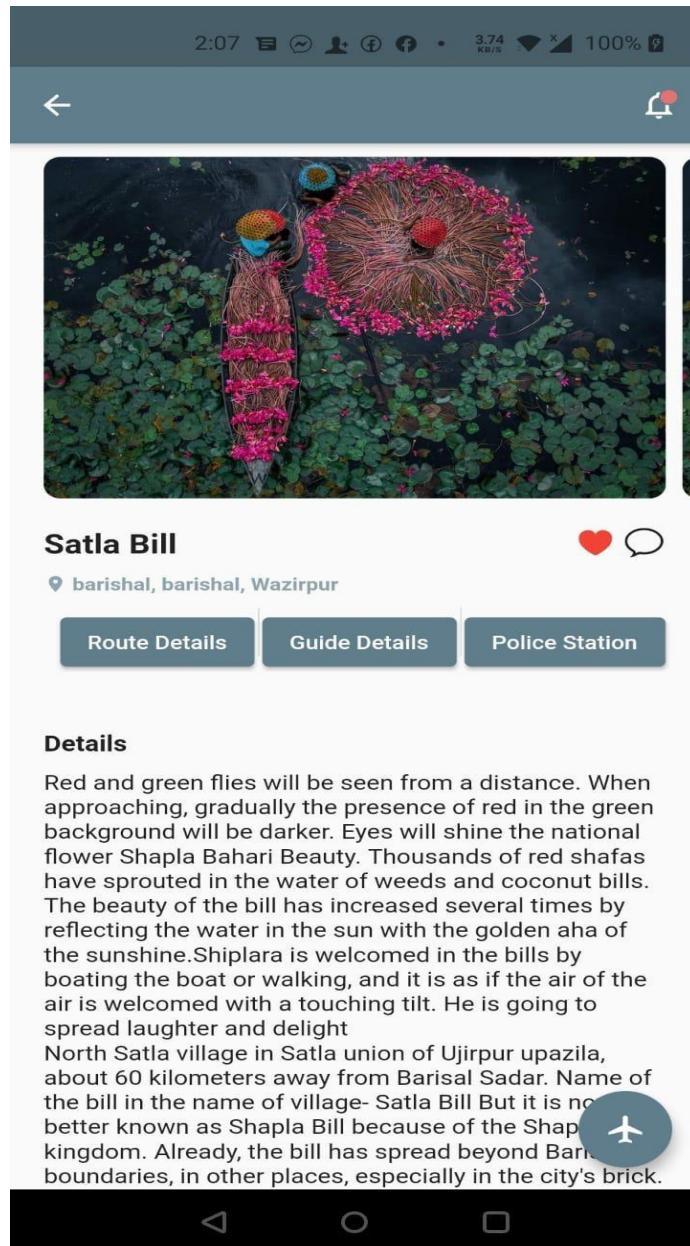


*Figure 55: Uploading Images*

After tapping button, app will open the new page where you need to select images from gallery then, click upload button.

Finally, clicking submit button will update the database with new tourist place details. Navigate to new page.

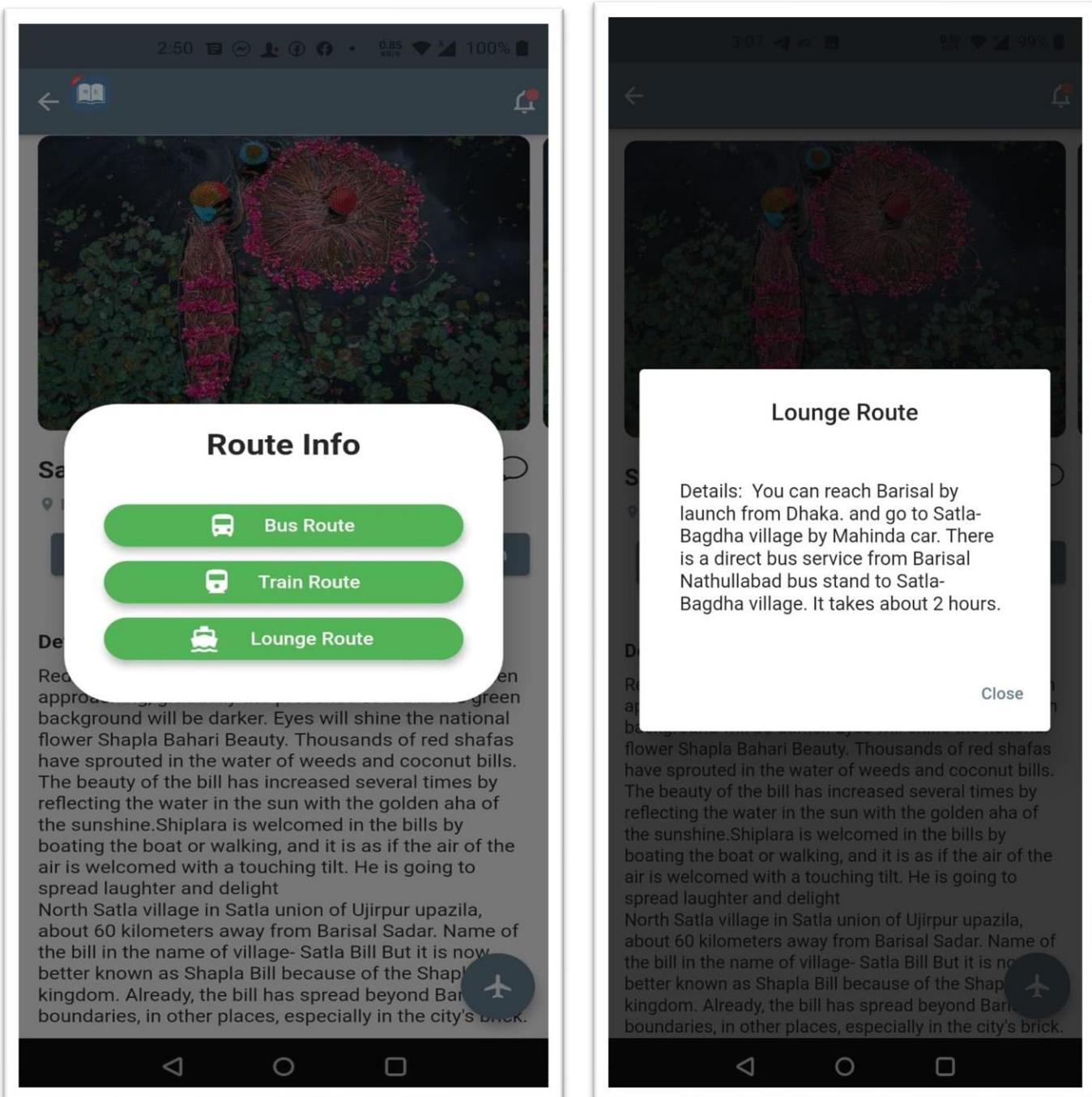
## 4.6 Details Of Places



*Figure 56:Details Of Places*

When user want to know about any tourist places and after clicking there, they will navigate to details of that place. Below images, there are three buttons where user can get Route Details, Guide Details and Police Station Details.

#### 4.6.1 Route Details



*Figure 57:Three means of Transportation*

Here, we will get all kind of route details like Bus Route, Train Route and Lounge Route by clicking on these buttons.

## 4.6.2 Guide Details

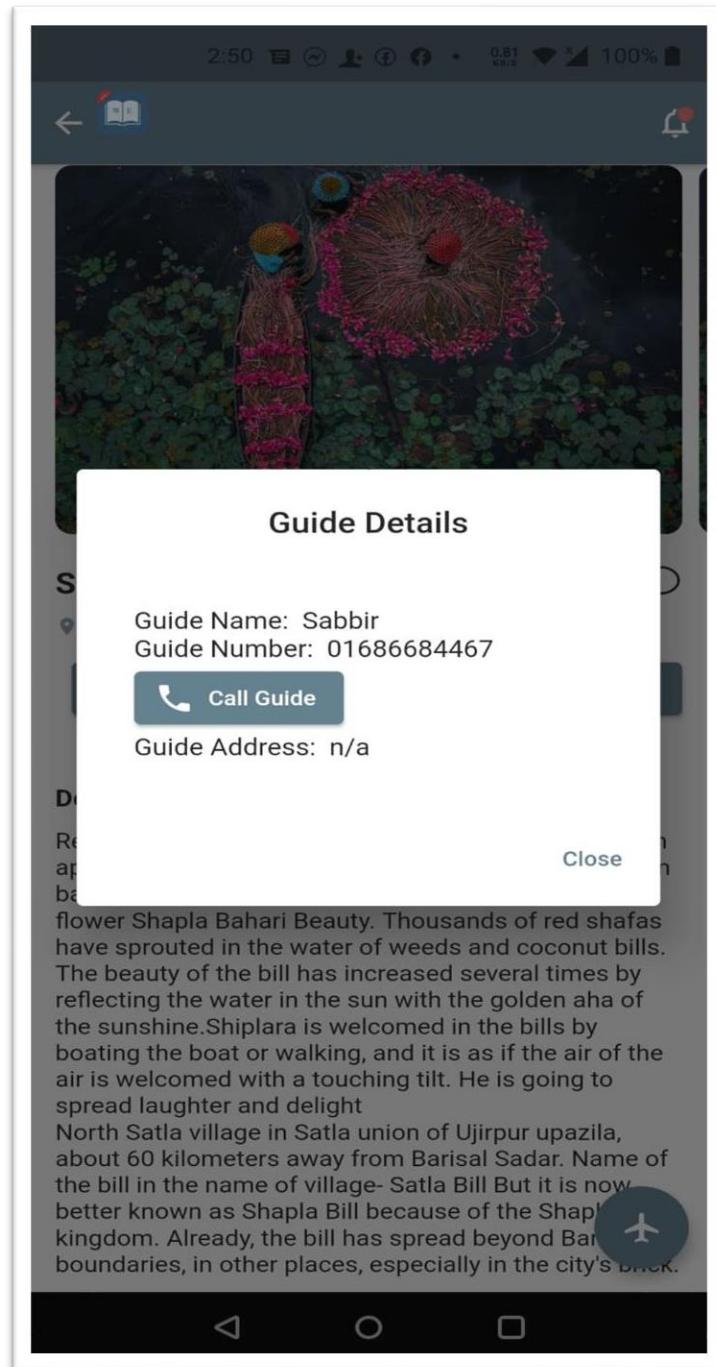


Figure 58:Information of Tour Guide

Next to Button Route details is Guide Details, after clicking that button, user can get guide information about selected tourist place. And when user press call button, he/she will be redirected to inbuilt phone calling interface.

#### 4.6.3 Police Station Details

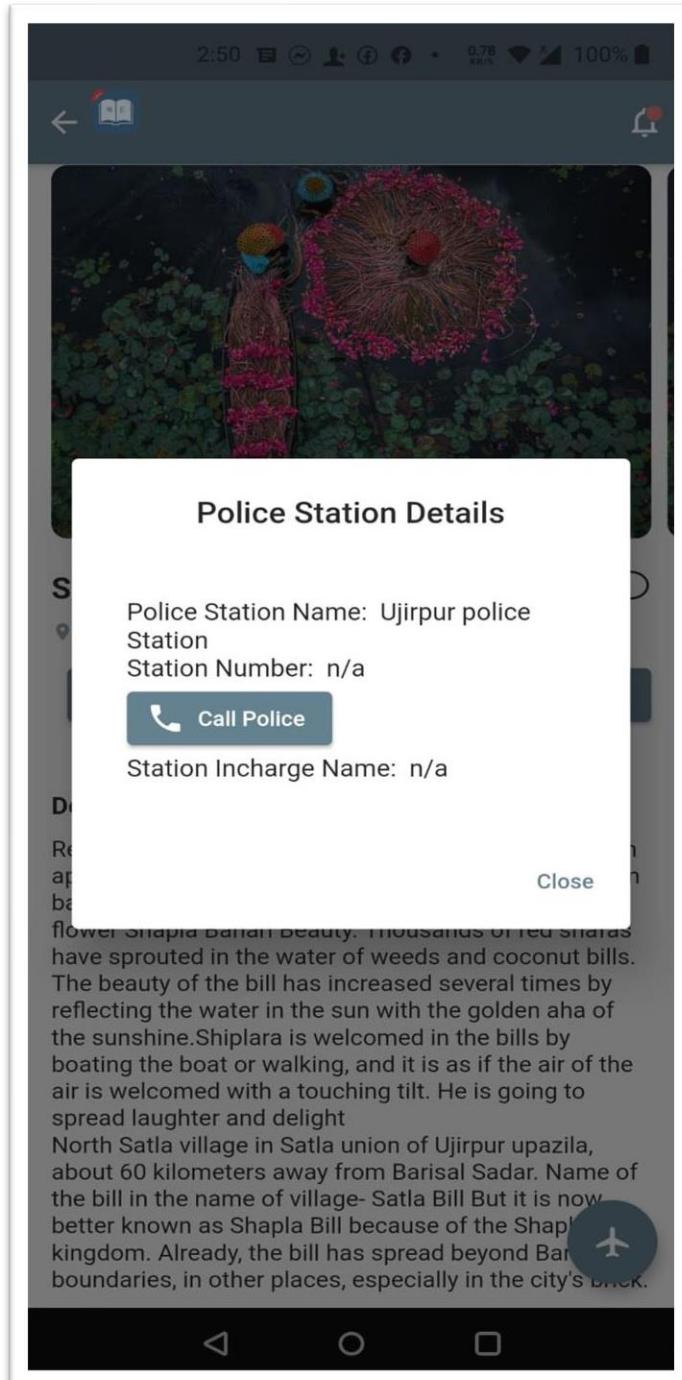
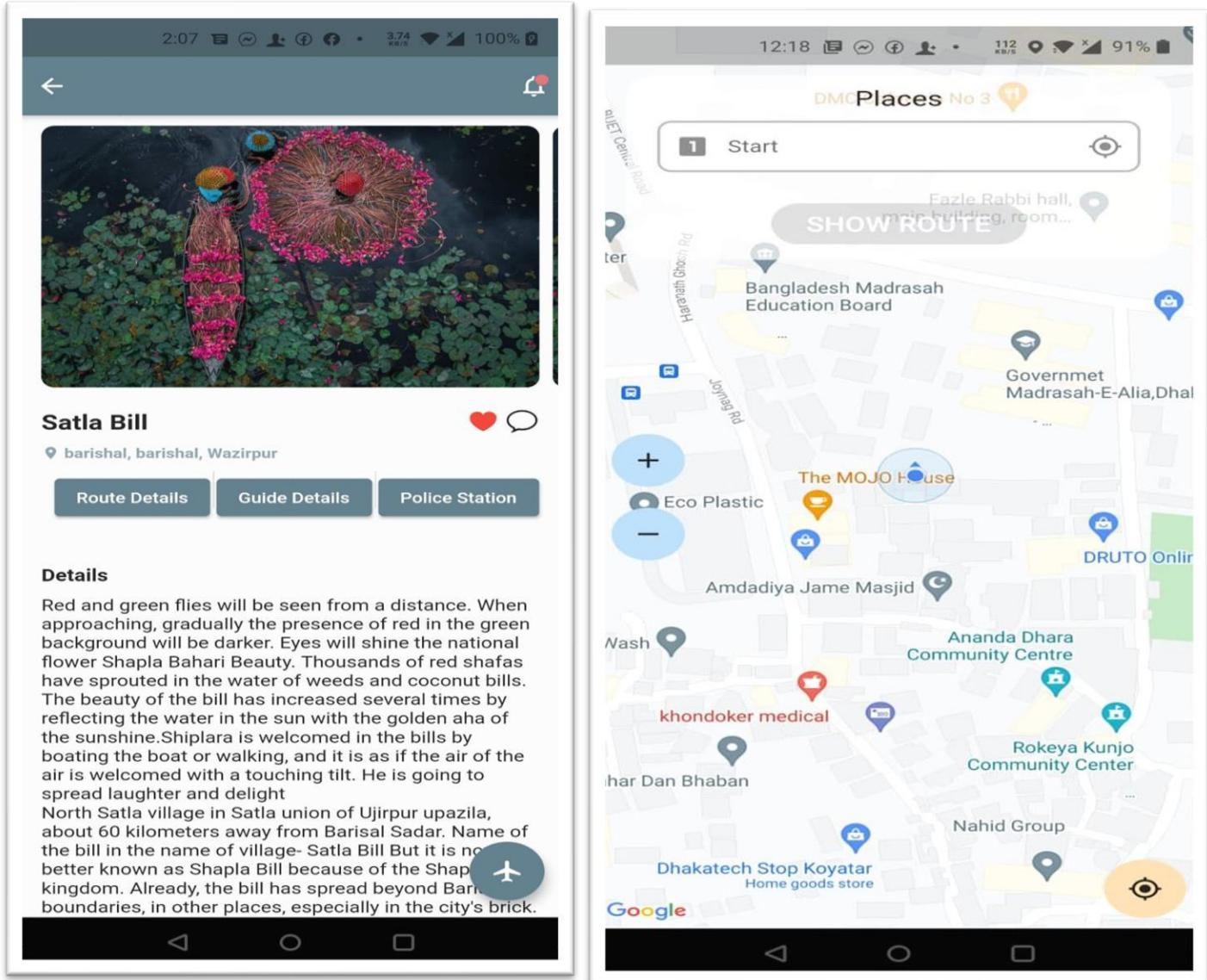


Figure 59:Local Police Station Details

Next to Button Guide details is Police Station Details button, after clicking that button, user can get local police station information about selected tourist place. And when user press call button, he/she will be redirected to inbuilt phone calling interface.

#### 4.6.4 Airplane icon



*Figure 60: Navigate to Selected Tourist place*

When we search or went manually to see more details of specific tourist place then below-right corner has button with Airplane icon after clicking that use will navigated to map where that tourist place is located.

## 4.7 Search Places

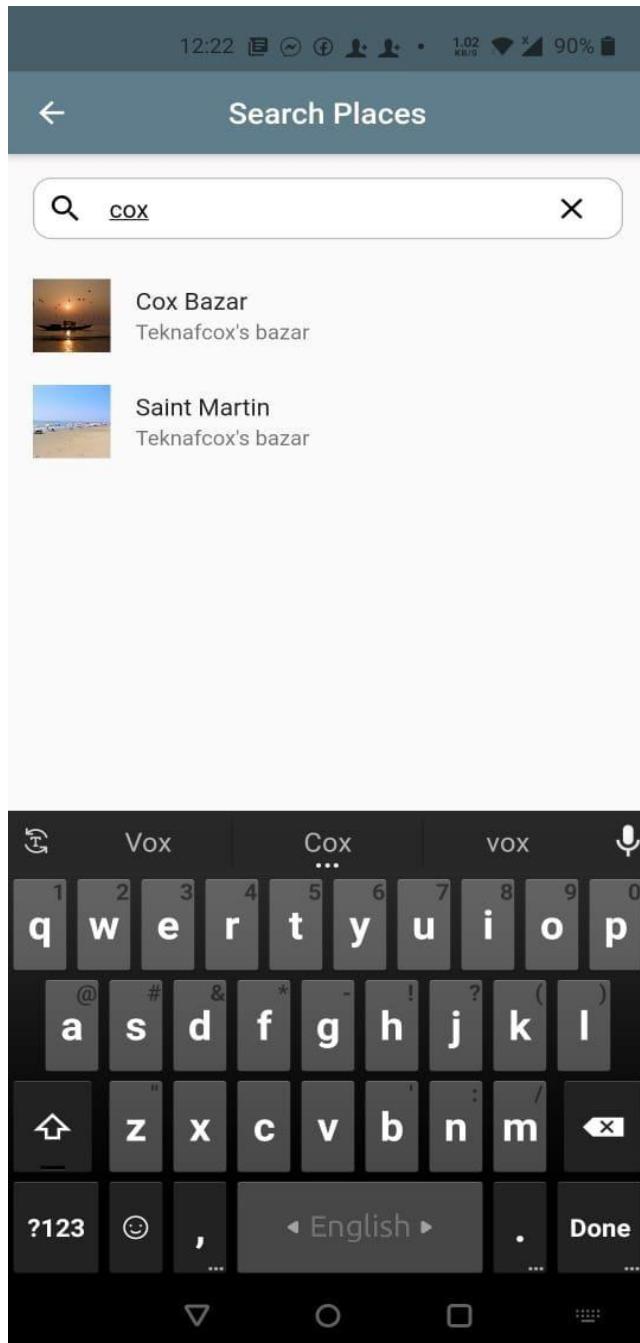


Figure 61: Search by Placename, District and Division

User can search by placename, division and district and view details easily and smoothly by clicking search pane.

## 4.8 Emergency Contact

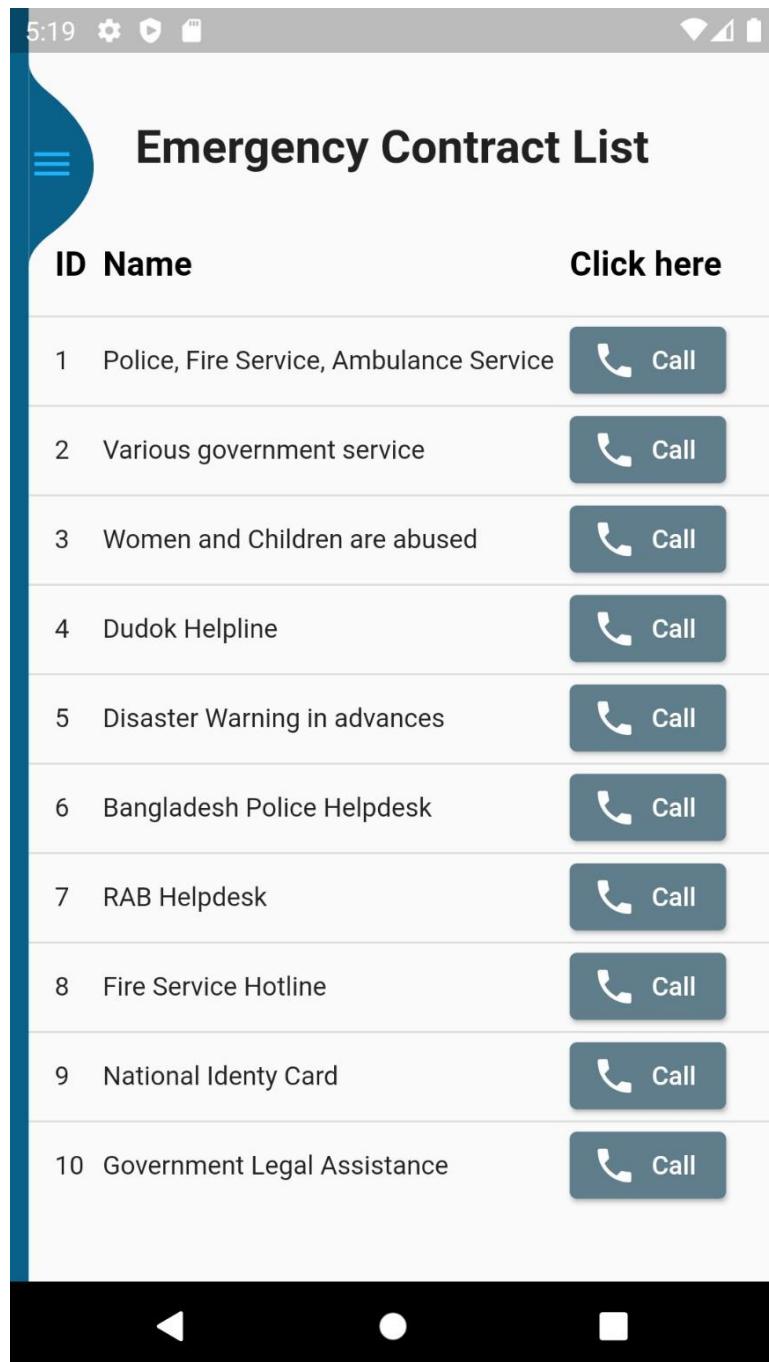


Figure 62:Emergency Contact

User can contact to the above-mentioned helpline numbers at tourist places.

## 4.9 show newly added place

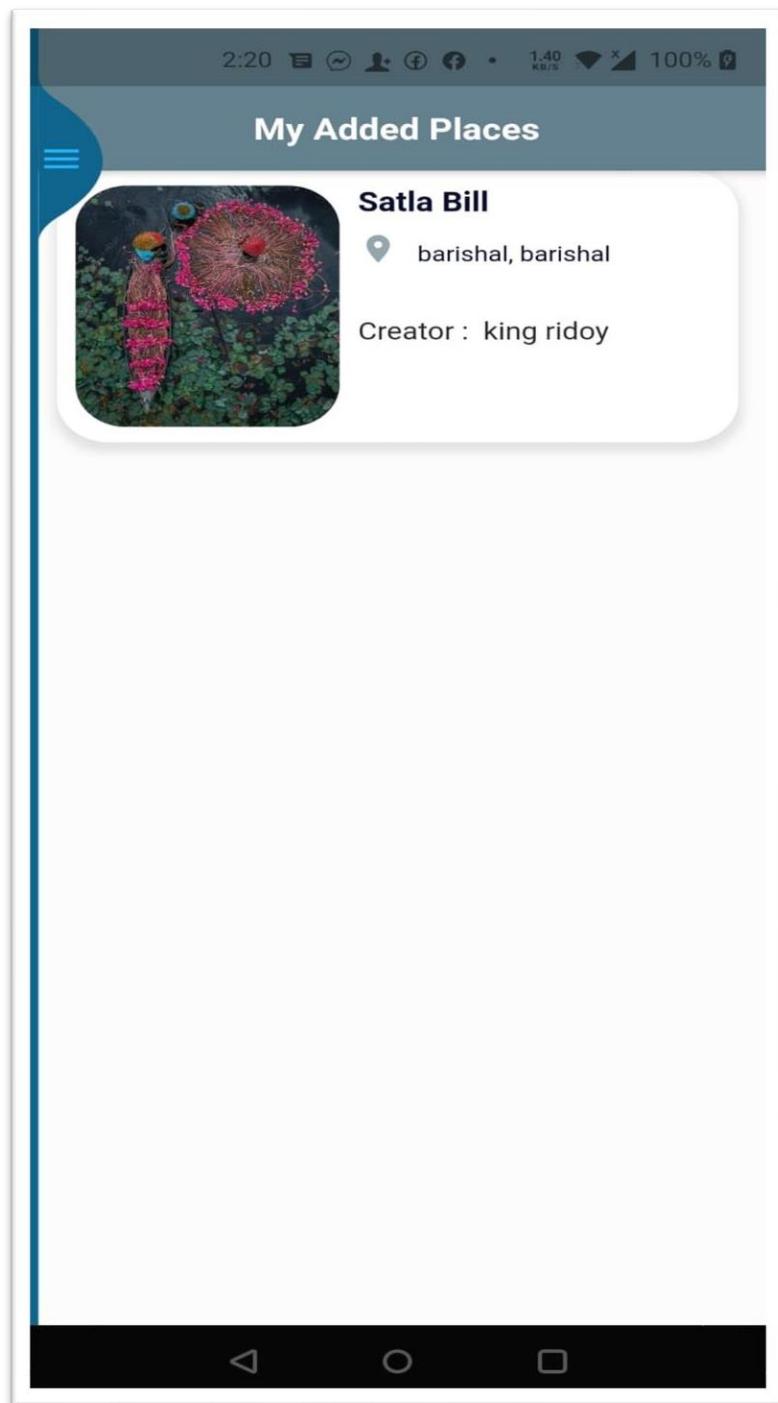


Figure63: show newly added place

This page will help user to review what they uploaded.

# **Conclusion, Implementation and Limitation**

## **5.1 Achievements**

This project has helped us to achieve a lot of experiences are enlisted: -

- Gathering and analyzing the functional requirements
- Working as a team
- Implementation of firebase and Google map with android application
- An app based on SRS-report
- Learnt flexible and expressive UI using flutter

## **5.2 Obstacles**

- Challenges to learn and implement android studio
- Billing for enabled Google API
- Working for backend with .net and firebase
- Render Flex overflow
- Custom route in Google Maps

## **5.3 Future Plan**

- Proper Google Maps implementation
- Uploading to the Play store
- Admin approval system

## **5.4 Limitations**

- User Interface could be improved
- User couldn't search specific places

## **References:**

- 1) <https://firebase.google.com/docs/firestore?authuser=1> (last access 22/08/2021)
- 2) <https://console.firebaseio.google.com/u/1/project/visit-bangladesh-1ecbb/overview>(last access 24/08/2021)
- 3) <https://flutter.dev/> (last access 22/08/2021)
- 4) <https://bdapis.herokuapp.com/> (last access 25/08/2021)