Visit Bangladesh

An android app

Software Requirement

Specification and Analysis

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

Assistant Professor

Institute of Information Technology

University of Dhaka

Date: 22/10/2020

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 requirements 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

Chapter One **9**

Introduction **9**

1.1 Purpose 9

1.2 Intended Audience 9

Chapter Two **11**

Inception **11**

2.1 Introduction **11**

2.1.1 Identifying stakeholders 11

2.1.2 Requirements: 12

2.1.3 Asking first questions 12

2.2 Conclusion **12**

Chapter Three **13**

Elicitation **13**

3.1 Introduction **13**

3.2 Eliciting requirements **13**

3.3 Requirements gathering **15**

3.4 Quality function deployment **15**

3.4.1 Normal requirements 15

3.4.2 Expected requirements 16

3.4.3 Exciting requirements 16

3.5 User scenario **16**

3.6 Elicitation work product **17**

Chapter Four **19**

Scenario Based Model **19**

Introduction **19**

4.1 Use case diagram **19**

4.2 Use Case Scenario **20**

4.2.1 Use case diagrams and description of subsystems 21

Level 0: Visit Bangladesh 21

Figure 1: Level 0 Use Case Diagram 22

Level 1: Visit Bangladesh 22
Figure 2: Level 1 Use Case Diagram 23
Figure 3: Level 1.1 Use Case Diagram of Authentication Subsystem 24
Figure 4: Level 1.1.1 Use Case Diagram of Authentication Subsystem 25
4.2.1.2 Browse: 26
Figure 5: Level 1.2 Use Case Diagram of Browse Subsystem 26
4.2.1.3 Update: 27
Figure 6: Level 1.3 Use Case Diagram of Update Subsystem 27
4.2.1.4 Search: 28
Figure 7: Level 1.4 Use Case Diagram of Search Subsystem 28
4.2.1.5 Setting: 29
Figure 8: Level 1.5 Use Case Diagram of Setting Subsystem 29
4.2.1.5.1 Edit Account: 30
Figure 9: Level 1.5.1 Use Case Diagram of Subsystem's task edit account 30
4.2.1.6 Emergency Contact: 31
Figure 10: Level 1.6 Use Case Diagram of Emergency Contact Subsystem 31
4.3 Activity Diagrams of Visit Bangladesh 32
4.3.1. Activity diagram of Register: 33
Figure 11: Level 4.3.1 Activity Diagram of Sign Up 33
4.3.2 Activity diagram of Log In: 34
Figure 12: Level 4.3.2 Activity Diagram of Log In 34
4.3.3 Activity diagram of Guest user: 35
Figure 13: Level 4.3.2 Activity Diagram of Guest user 35
4.3.4 Activity diagram of Browse: 36
Figure 14: Level 4.3.3 Activity Diagram of Browse 36
4.3.5 Activity diagram of Update: 37
Figure 13: Level 4.3.4 Activity Diagram of Update 37
4.3.6 Activity diagram of edit contact: 38
Figure 14: Level 4.3.5 Activity Diagram of edit contact 38
4.3.7 Activity diagram of Emergency Contact: 39
Figure 15: Level 4.3.4 Activity Diagram of emergency contact 39
4.4 Swim-lane diagram of Visit Bangladesh 40
4.4.1 Swim-lane diagram of Register 40
Figure 16: Level 4.4.1 Swim-lane Diagram of Register 41
4.4.2 Swim-lane diagram of Log In 42

Figure 17: Level 4.4.2 Swim-lane Diagram of Log In 42
4.4.3 Swim-lane diagram of Guest user 43
Figure 18: Level 4.4.3 Swim-lane Diagram of Guest user 43
4.4.4 Swim-lane diagram of Browse 44
Figure 19: Level 4.4.4 Swim-lane Diagram of Browse 44
4.4.5 Swim-lane diagram of Update 45
Figure 20: Level 4.4.5 Swim-lane Diagram of Update 45
4.4.7 Swim-lane diagram of Edit contact 46
Figure 21: Level 4.4.5 Swim-lane Diagram of Edit contact **46**
4.4.8 Swim-lane diagram of Emergency contact 47
Figure 22: Level 4.4.6 Swim-lane Diagram of emergency contact 47
Chapter Five **48**
Data Model **48**
5.1 Introduction 48
5.2 Data object 48
5.3 ER diagram 53
5.4 Data schema table 54
Chapter Six **57**
Class Based Model Of Visit Bangladesh Android Application **57**
6.1 Grammatical Parsing and Analysis **57**
6.1.1 CLASS IDENTIFICATION WITH GENERAL CLASSIFICATION 57
6.1.2 CLASS IDENTIFIED WITH SELECTION CRITERIA 60
6.2.1 Attributes and Methods of Potential Classes 63
6.2.3 Attribute Selection 64
6.2.3 Method Identification: 67
6.2.4 Finalizing Classes 70
6.3 Class Responsibility Collaborator modeling **70**
6.3.1 Class cards 70
6.3.2 UML Diagram 75
CHAPTER 7 ..... **76**
DATA FLOW MODEL OF 'Visit Bangladesh' A MOBILE APPLICATION **76**
7.1.0 level 0: 76
7.2.1: Level 1.1 77
7.2.2. Level 1.2 77
7.3.1 Level 2.1 78

7.3.2 Level 2.2 78
7.3.3 Level 2.3 79
7.3.4 Level 2.4 79
7.3.5 Level 2.5 80
CHAPTER 8 81
BEHAVIORAL MODEL OF 'Visit Bangladesh' A MOBILE APPLICATION 81
8.1.1 State diagram of Account : 81
8.1.2 State diagram of Admin: 82
8.1.5 State diagram of Notification: 82
8.1.3 State diagram of User: 83
8.1.4 State diagram of System: 84
8.1.6 State diagram of Database: 84
8.1.7 State diagram of Search: 85
8.1.8 State diagram of Location: 85
8.2 Sequence diagram 86
8.2.1 Sequence diagram of Sign Up 86
8.2.2 Sequence diagram of Sign In 87
8.2.3 Sequence diagram of Search Location: 88
8.2.4 Sequence diagram of Edit Location: 88
8.2.5 Sequence diagram of Delete Location: 89
8.2.6 Sequence diagram of Browse Location: 89
8.2.1 Sequence diagram of Add Location: 90
Chapter 9: Conclusion 91

Chapter One

Introduction

This chapter describes the objectives of this report as well as the audiences who should have to go through this report for individual purposes.

1.1 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.

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.

Chapter Two

Inception

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

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

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.

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.

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.

2.2 Conclusion

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

Chapter Three

Elicitation

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

3.1 Introduction

Requirements 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.

3.2 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 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.

3.4 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.

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

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
- Guest user can only view the various tourist places, but cannot add comments and reviews.
- User can create their profile and, add comments and reviews about tourist places.
- User can reset their username and password.

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 the time of emergency.

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:

Admin 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 journey.

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.

Chapter Four

Scenario Based Model

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

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.

4.1 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.

4.2 Use Case Scenario

<table>
<tr><th>Level 0</th><th>Level 1</th><th>Level 2</th><th>Level 3</th></tr>
<tr><td rowspan="8">Visit Bangladesh</td><td rowspan="3">Authentication</td><td>Login</td><td rowspan="3">Reset Password</td></tr>
<tr><td>Register</td></tr>
<tr><td>Guest User</td></tr>
<tr><td rowspan="4">Browse</td><td>Images of tourist places</td><td rowspan="4"></td></tr>
<tr><td>Name and facts of places</td></tr>
<tr><td>Route</td></tr>
<tr><td>Comments and Reviews</td></tr>
<tr><td rowspan="2">Update</td><td rowspan="2">Add Route</td><td>Add places</td></tr>
<tr><td>Add images</td></tr>
</table>

		Add new place	Add comments
	Search	Search by state/division	
		Search by place name	
	Settings	Edit account	Change profile picture
			Reset username/password
	Emergency Contact	Add contacts	
		Send notifications	

4.2.1 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.

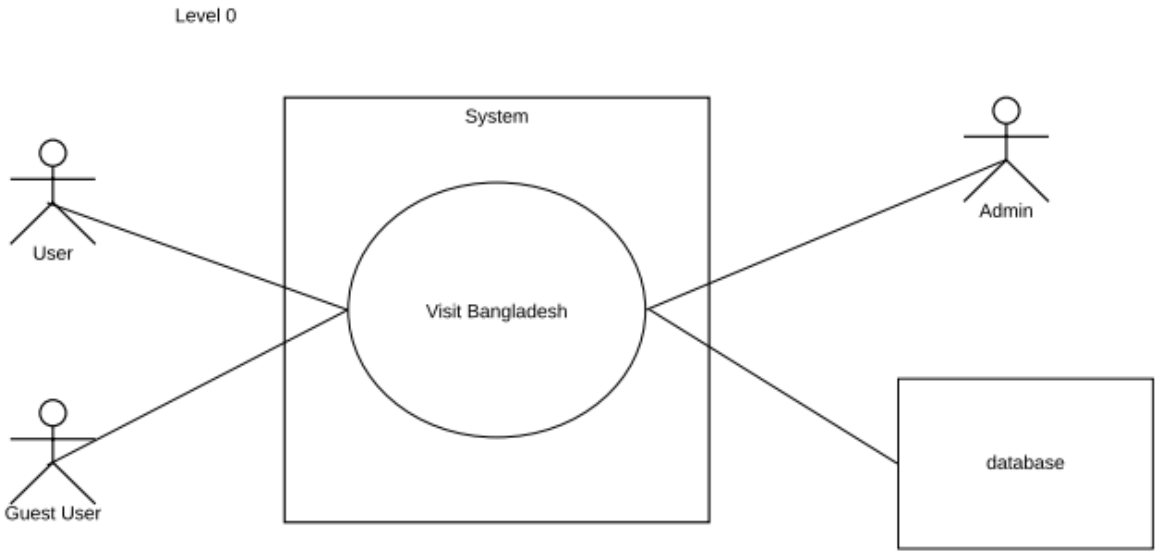


Figure 1: Level 0 Use Case Diagram

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.

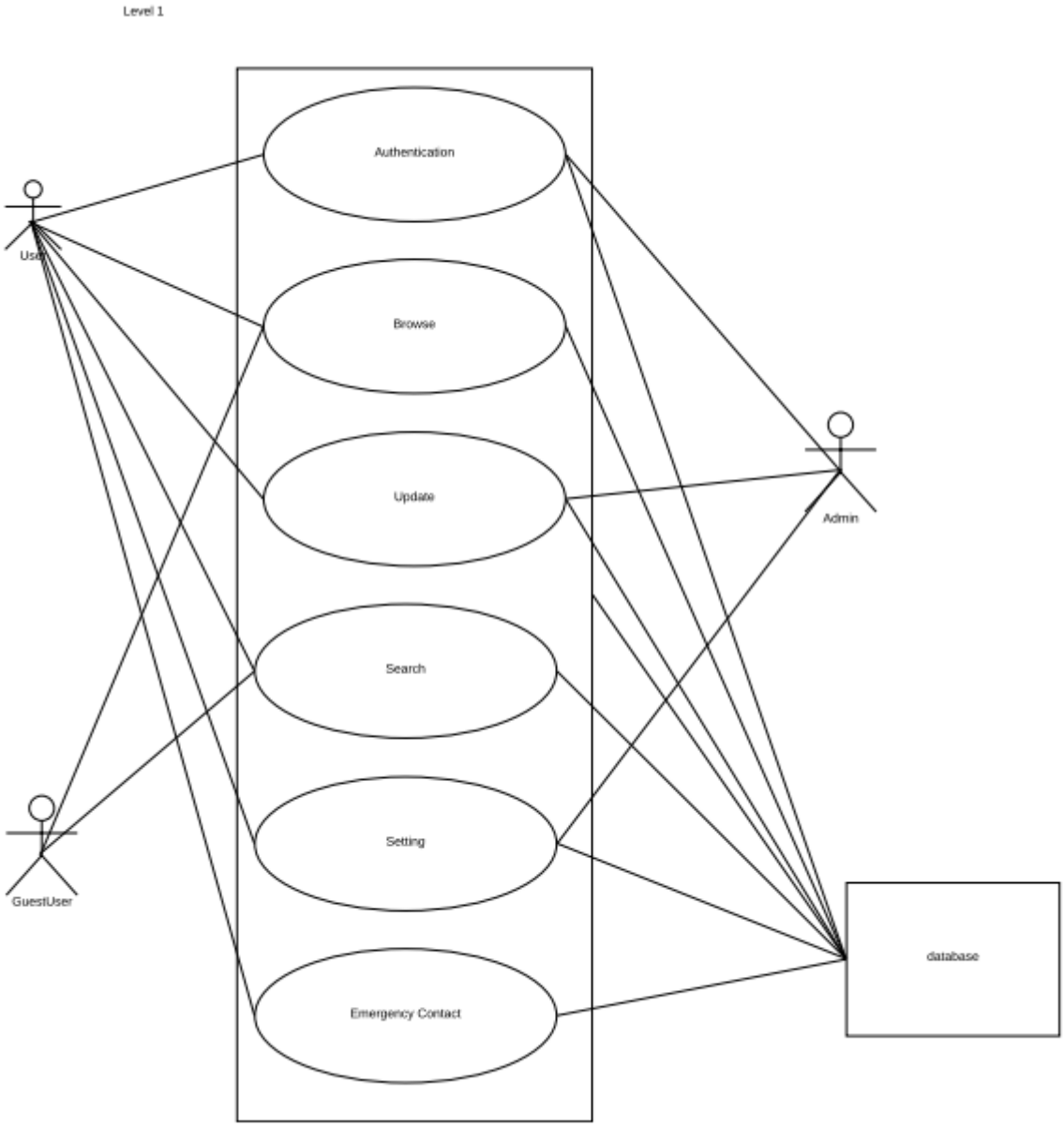


Figure 2: Level 1 Use Case Diagram

4.2.1.1 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.

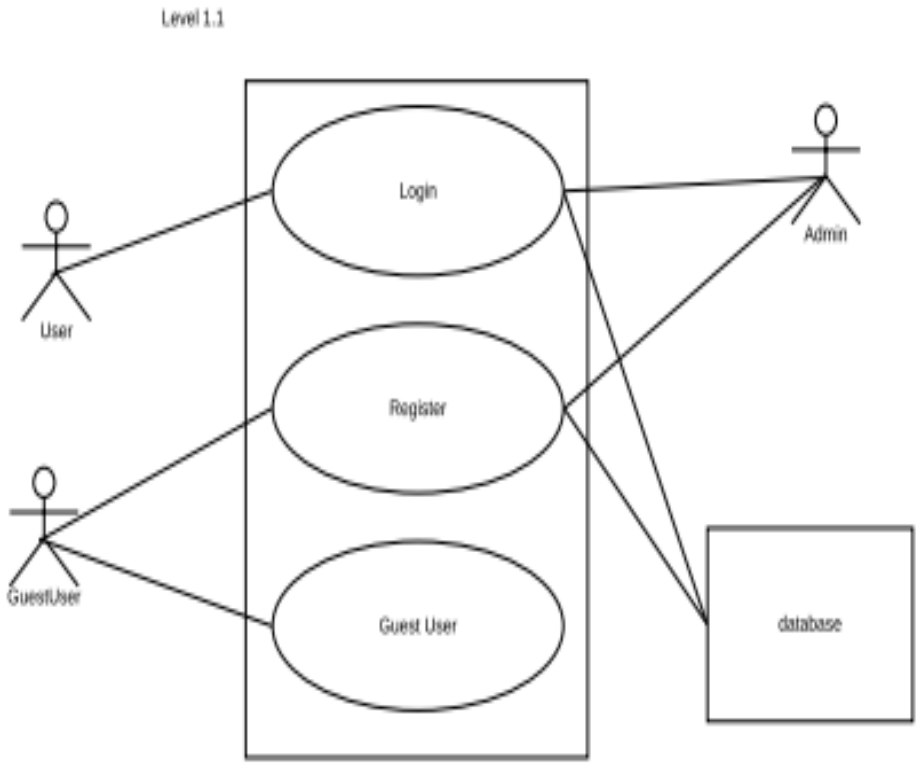


Figure 3: Level 1.1 Use Case Diagram of Authentication Subsystem

4.2.1.1.1 Reset Password

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

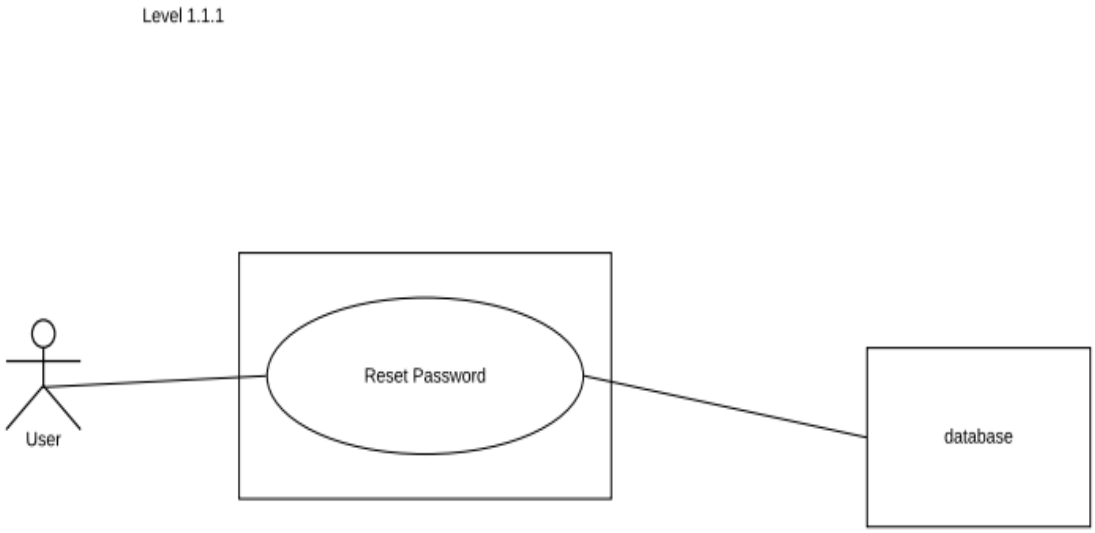
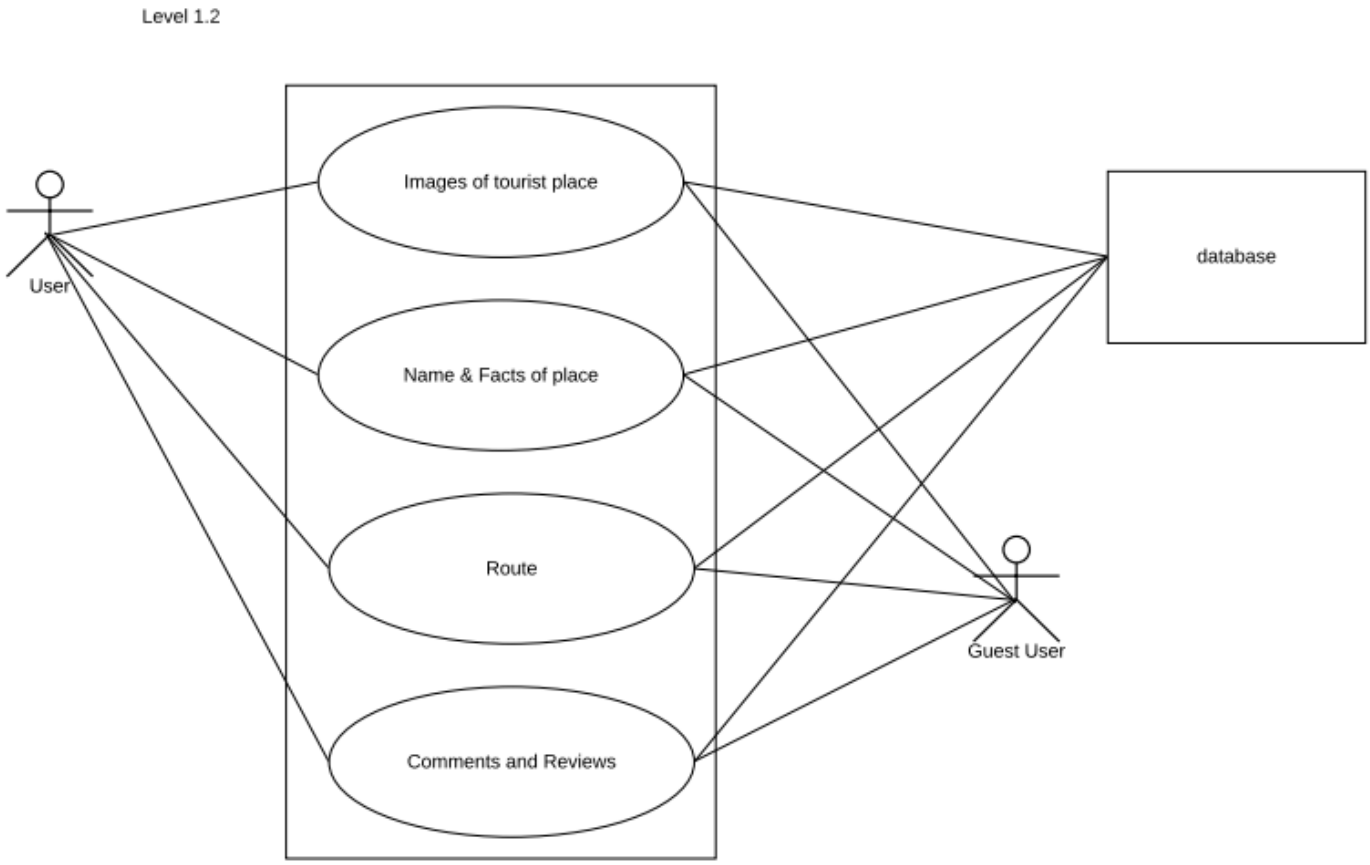


Figure 4: Level 1.1.1 Use Case Diagram of Authentication Subsystem

4.2.1.2 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



4.2.1.3 Update:

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

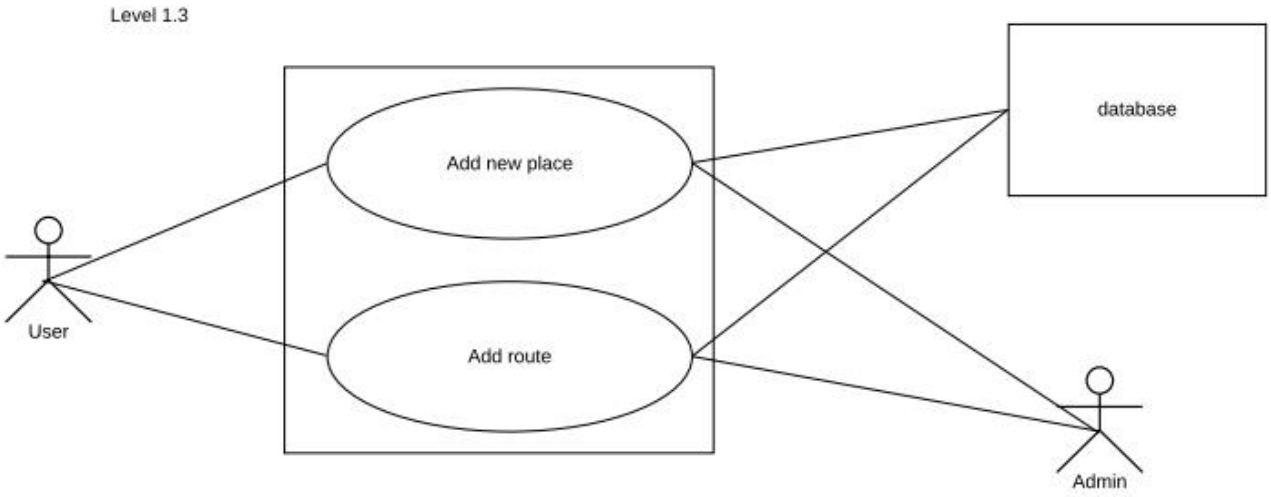


Figure 6: Level 1.3 Use Case Diagram of Update Subsystem

4.2.1.4 Search:

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.

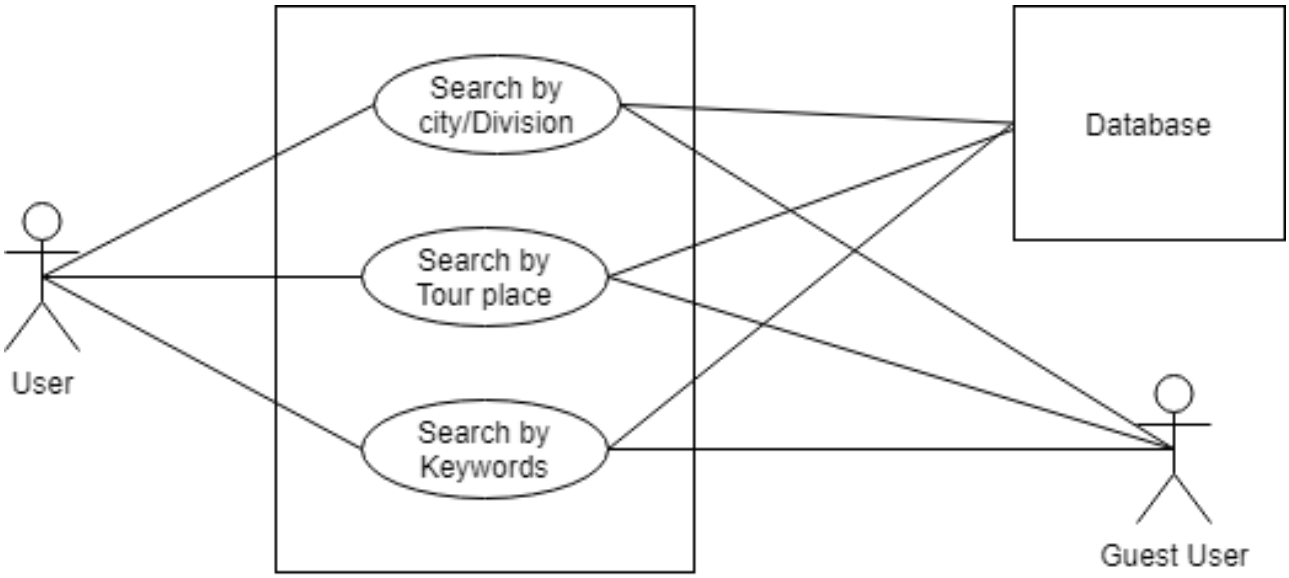


Figure 7: Level 1.4 Use Case Diagram of Search Subsystem

4.2.1.5 Setting:

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.

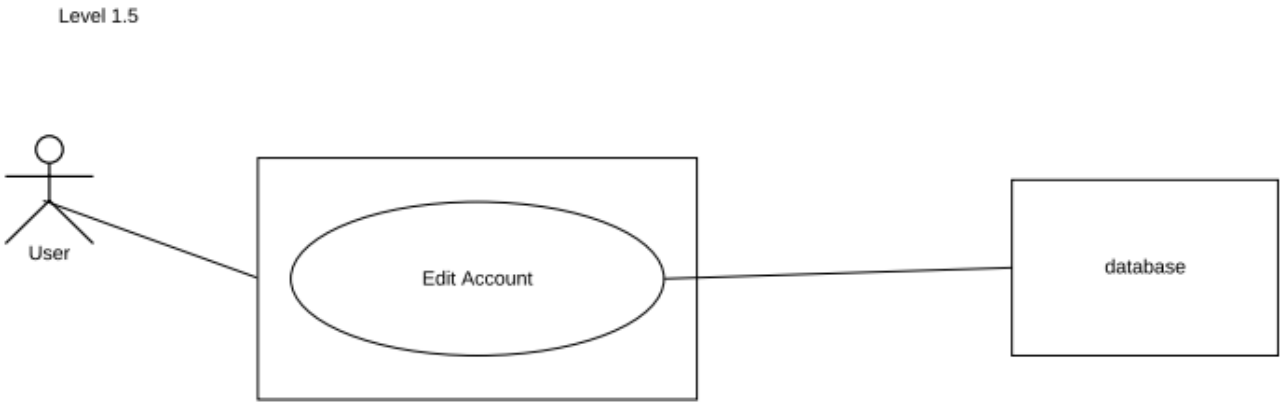


Figure 8: Level 1.5 Use Case Diagram of Setting Subsystem

4.2.1.5.1 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.

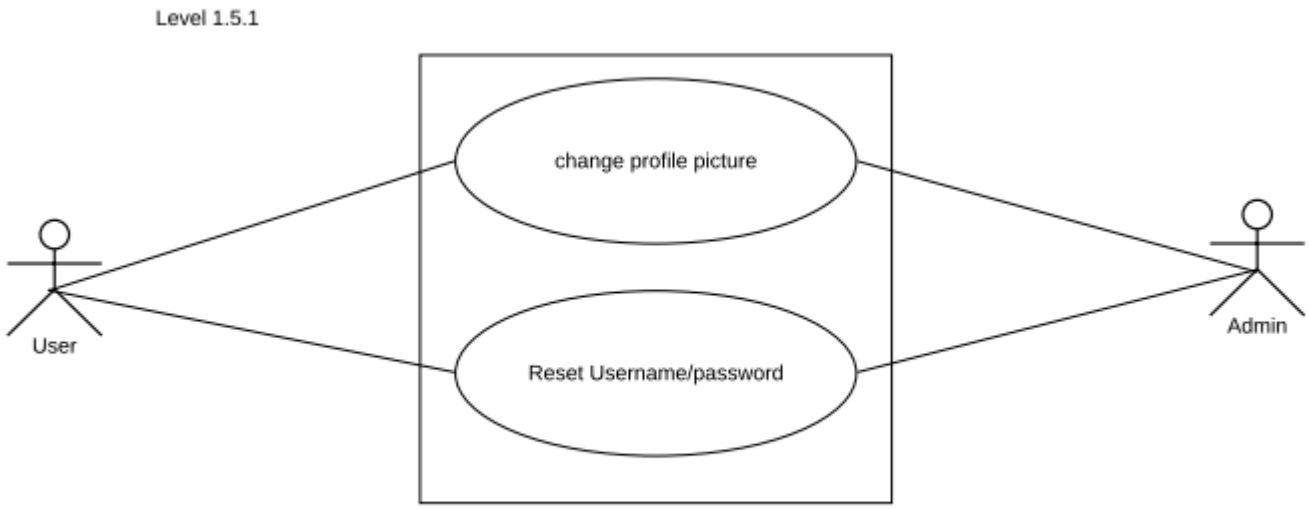


Figure 9: Level 1.5.1 Use Case Diagram of Subsystem's task edit account

4.2.1.6 Emergency Contact:

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 –

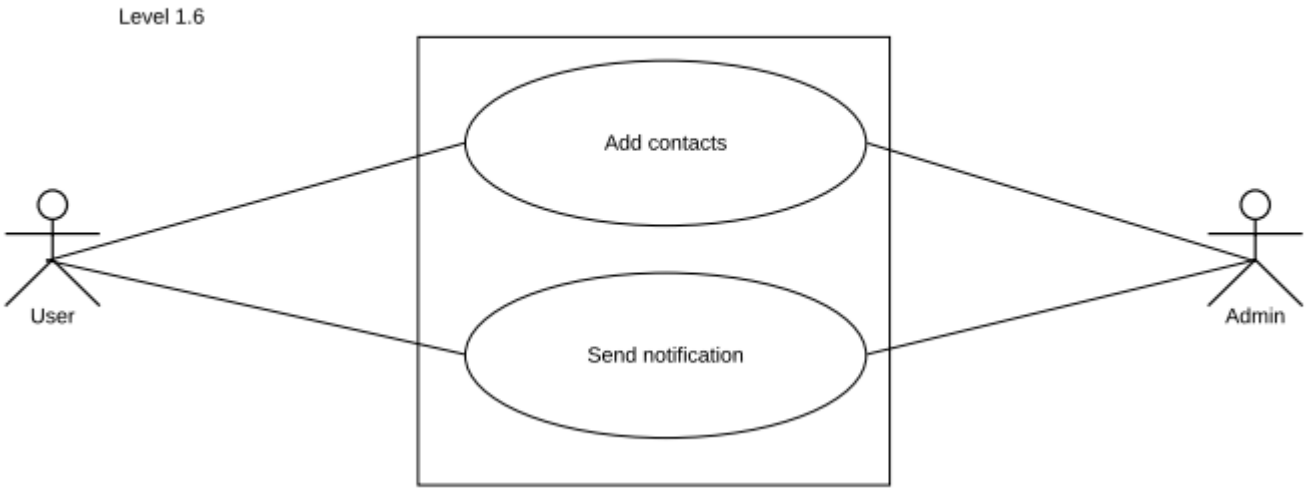


Figure 10: Level 1.6 Use Case Diagram of Emergency Contact Subsystem

4.3 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:

4.3.1. Activity diagram of Register:

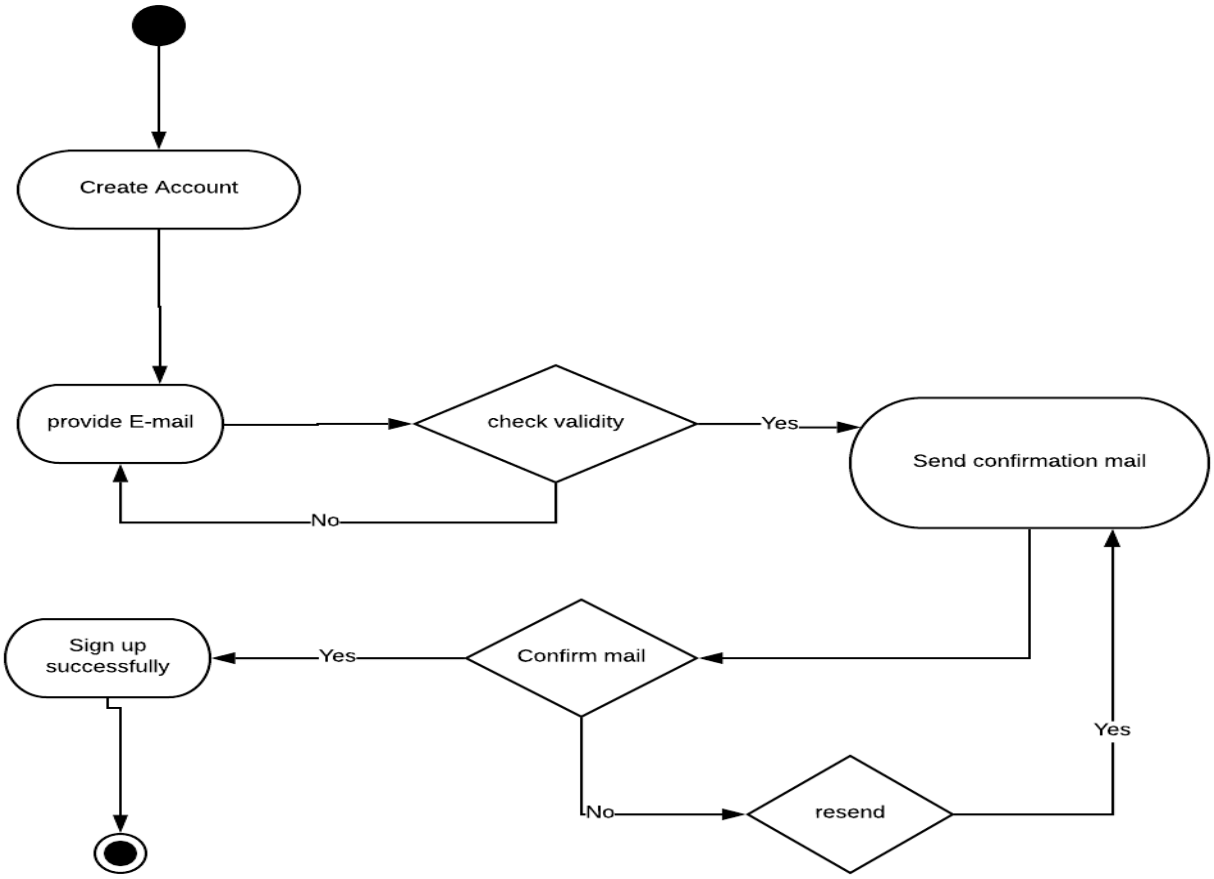


Figure 11: Level 4.3.1 Activity Diagram of Sign Up

4.3.2 Activity diagram of Log In:

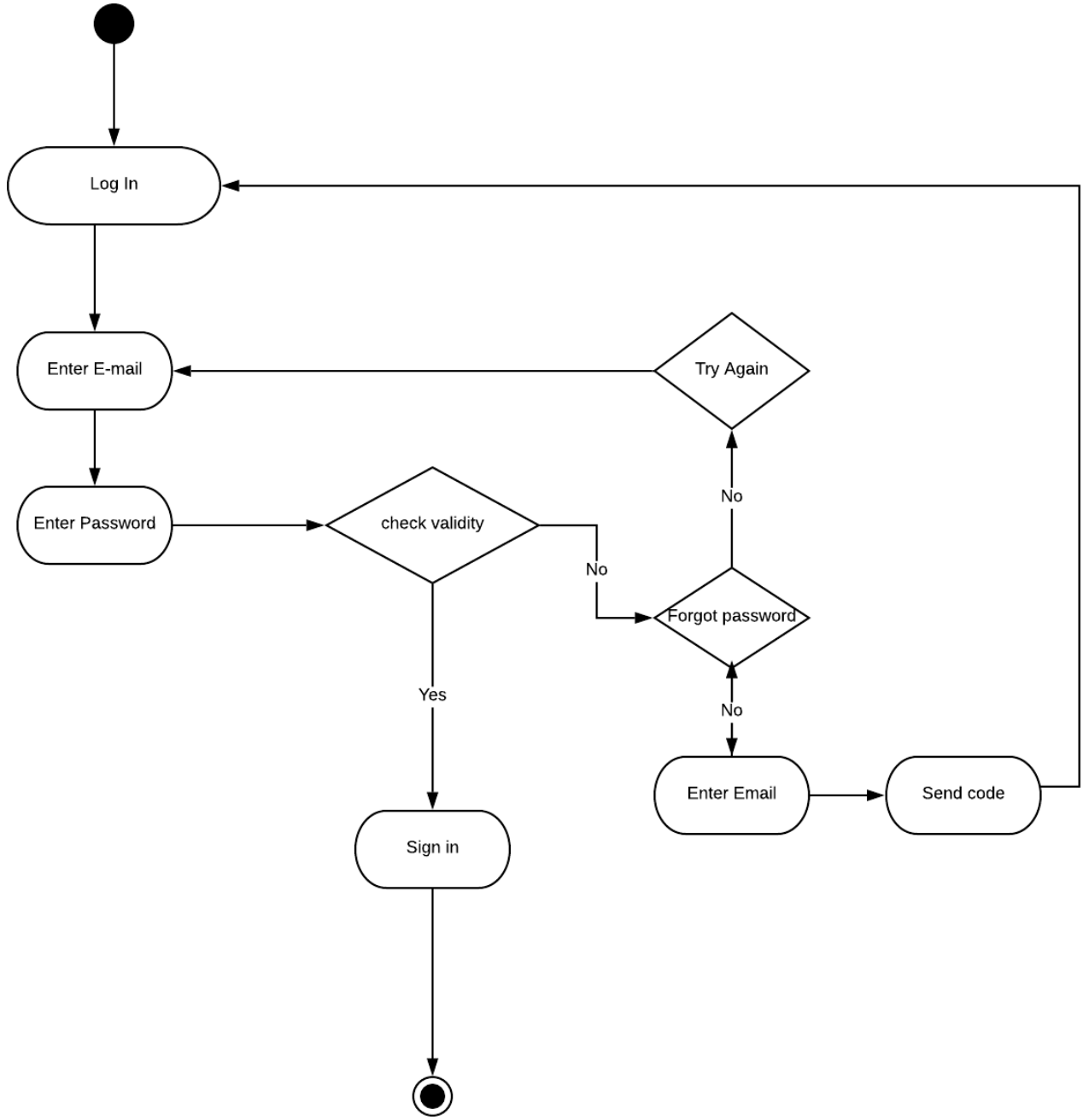


Figure 12: Level 4.3.2 Activity Diagram of Log In

4.3.3 Activity diagram of Guest user:

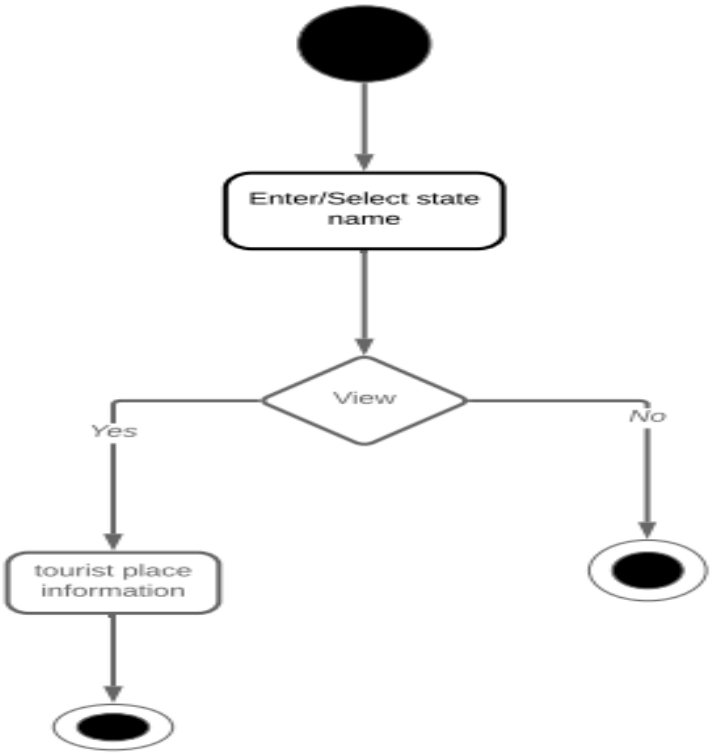


Figure 13: Level 4.3.2 Activity Diagram of Guest user

4.3.4 Activity diagram of Browse:

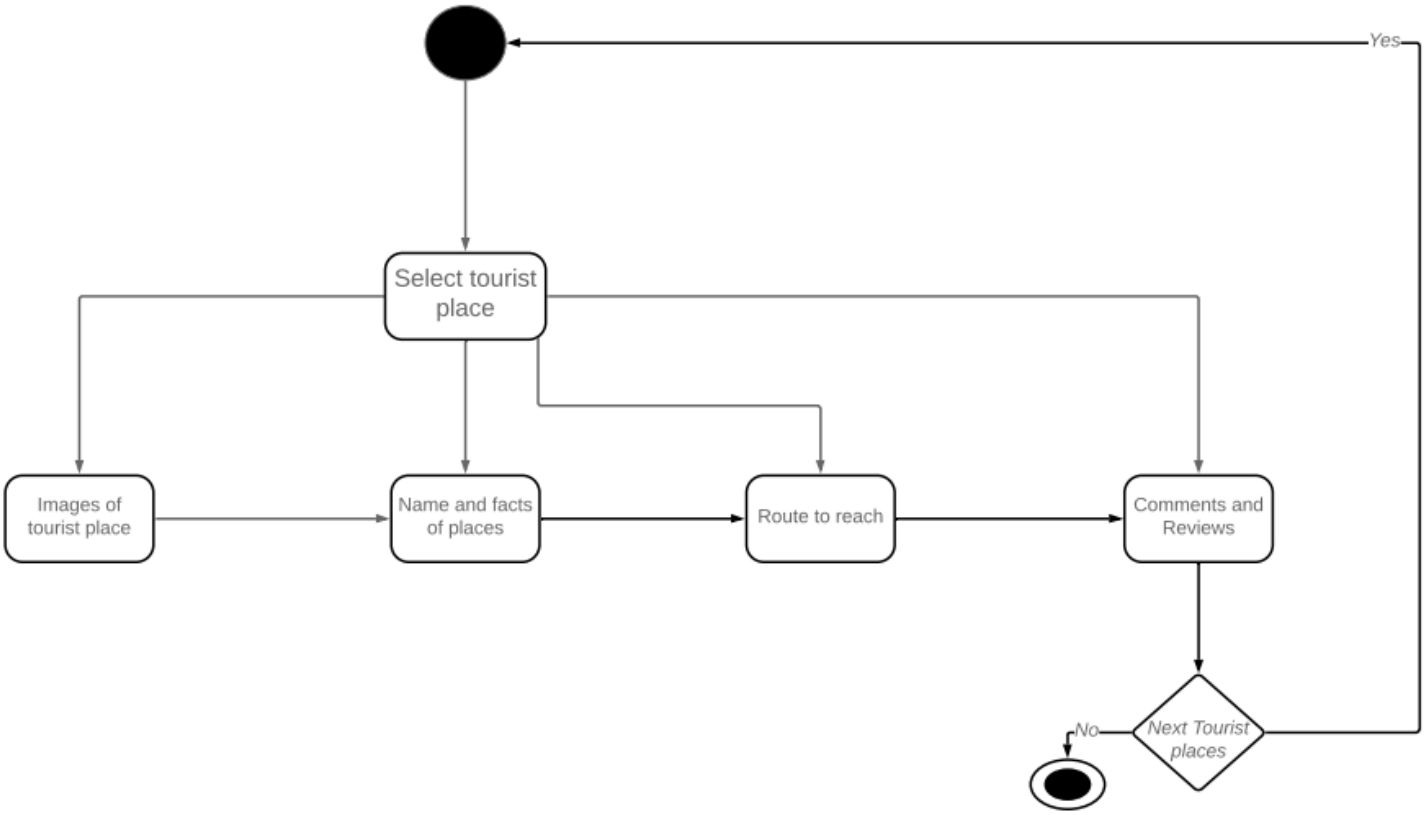


Figure 14: Level 4.3.3 Activity Diagram of Browse

4.3.5 Activity diagram of Update:

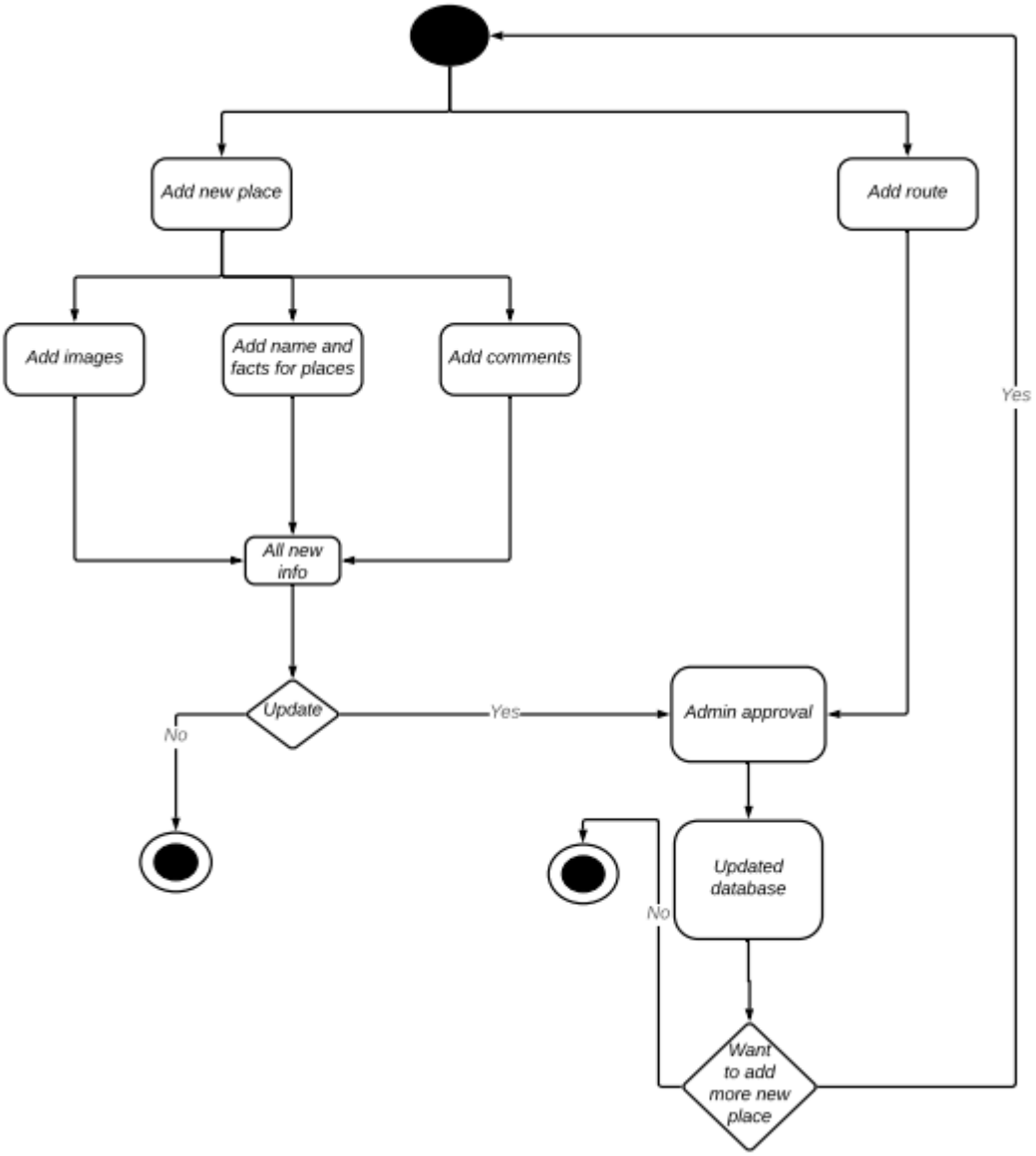


Figure 13: Level 4.3.4 Activity Diagram of Update

4.3.6 Activity diagram of edit contact:

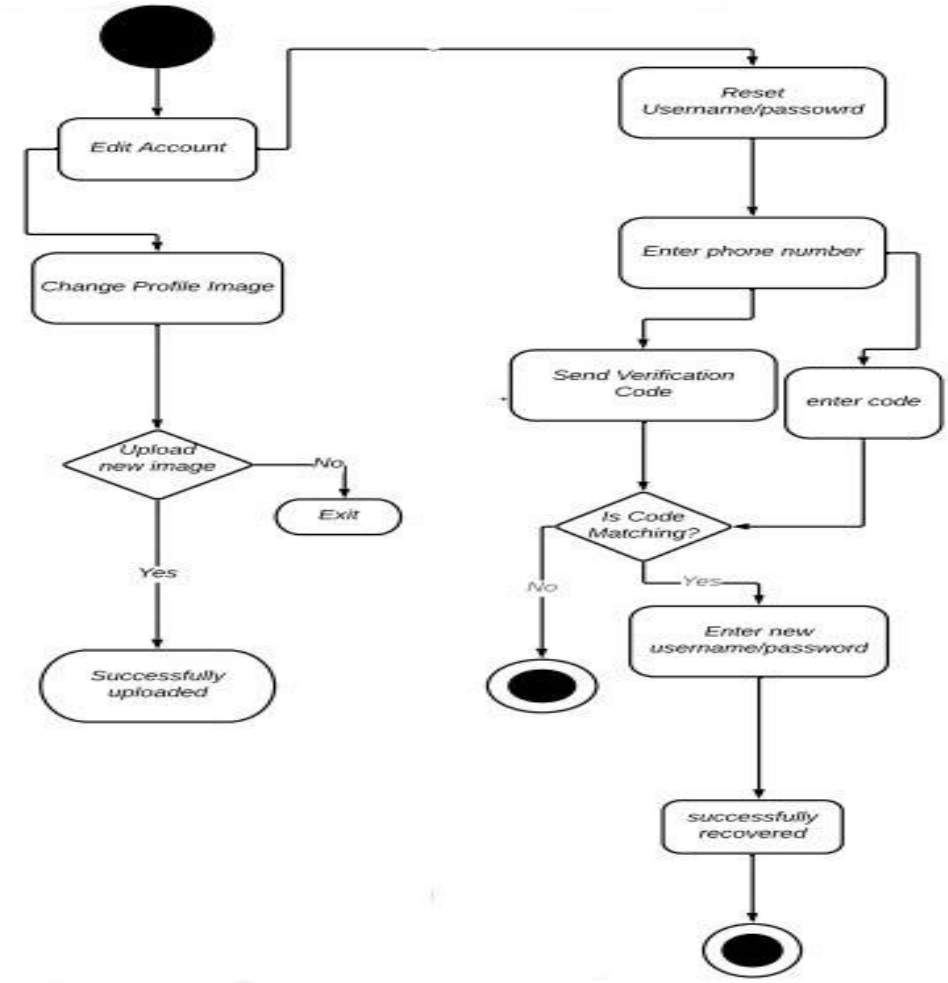


Figure 14: Level 4.3.5 Activity Diagram of edit contact

4.3.7 Activity diagram of Emergency Contact:

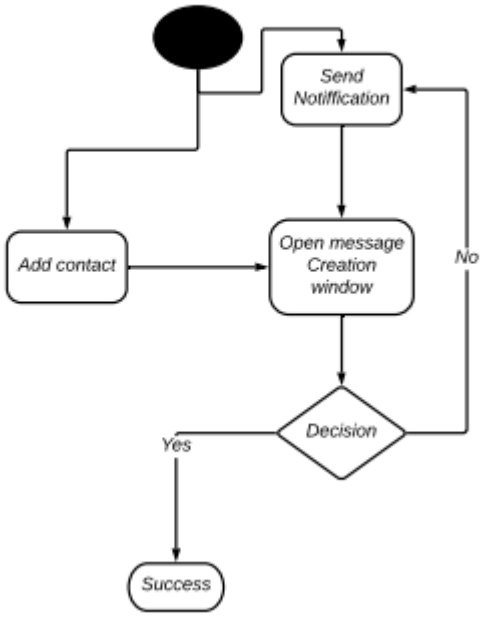


Figure 15: Level 4.3.4 Activity Diagram of emergency contact

4.4 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:

4.4.1 Swim-lane diagram of Register

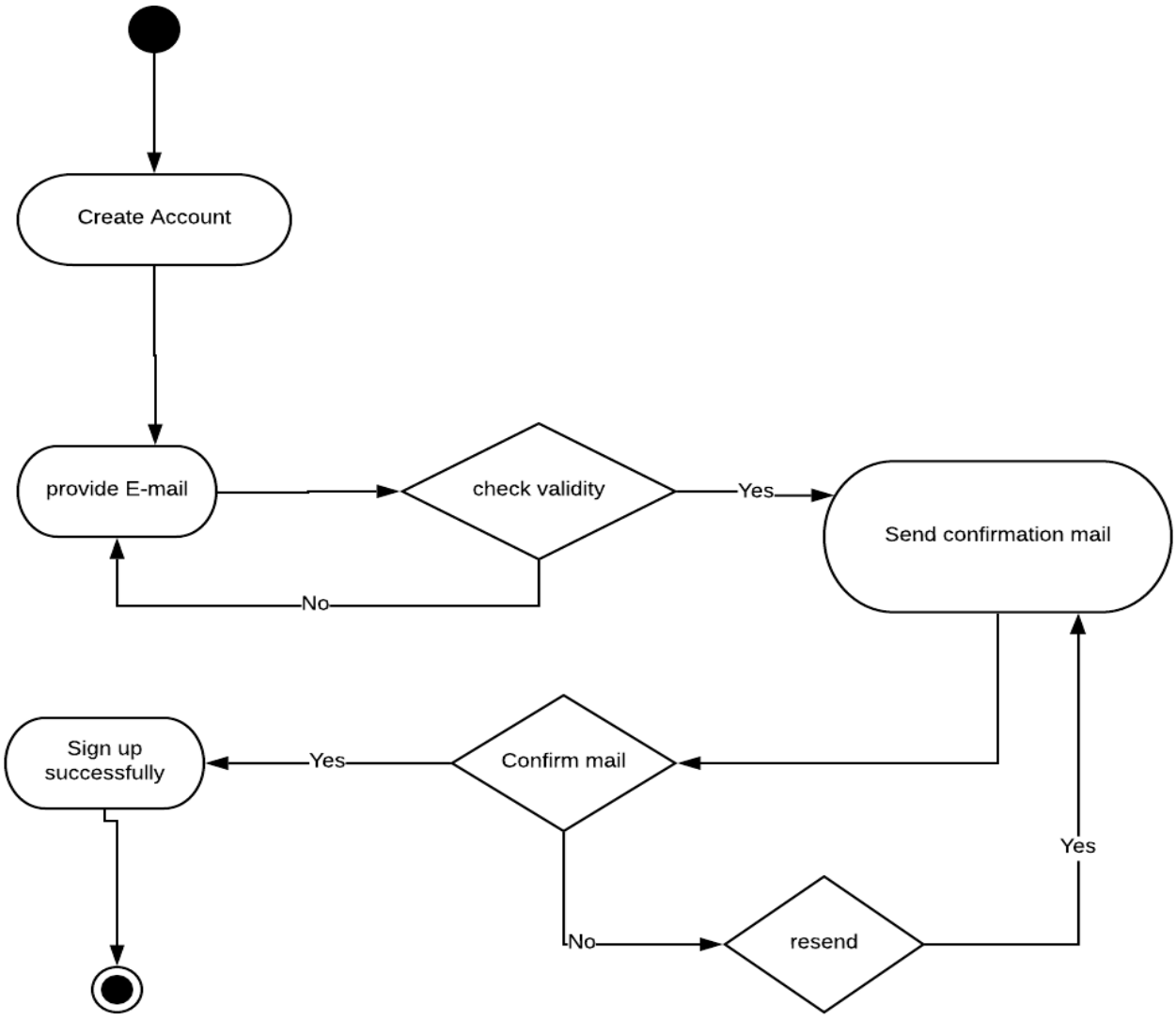


Figure 16: Level 4.4.1 Swim-lane Diagram of Register

4.4.2 Swim-lane diagram of Log In

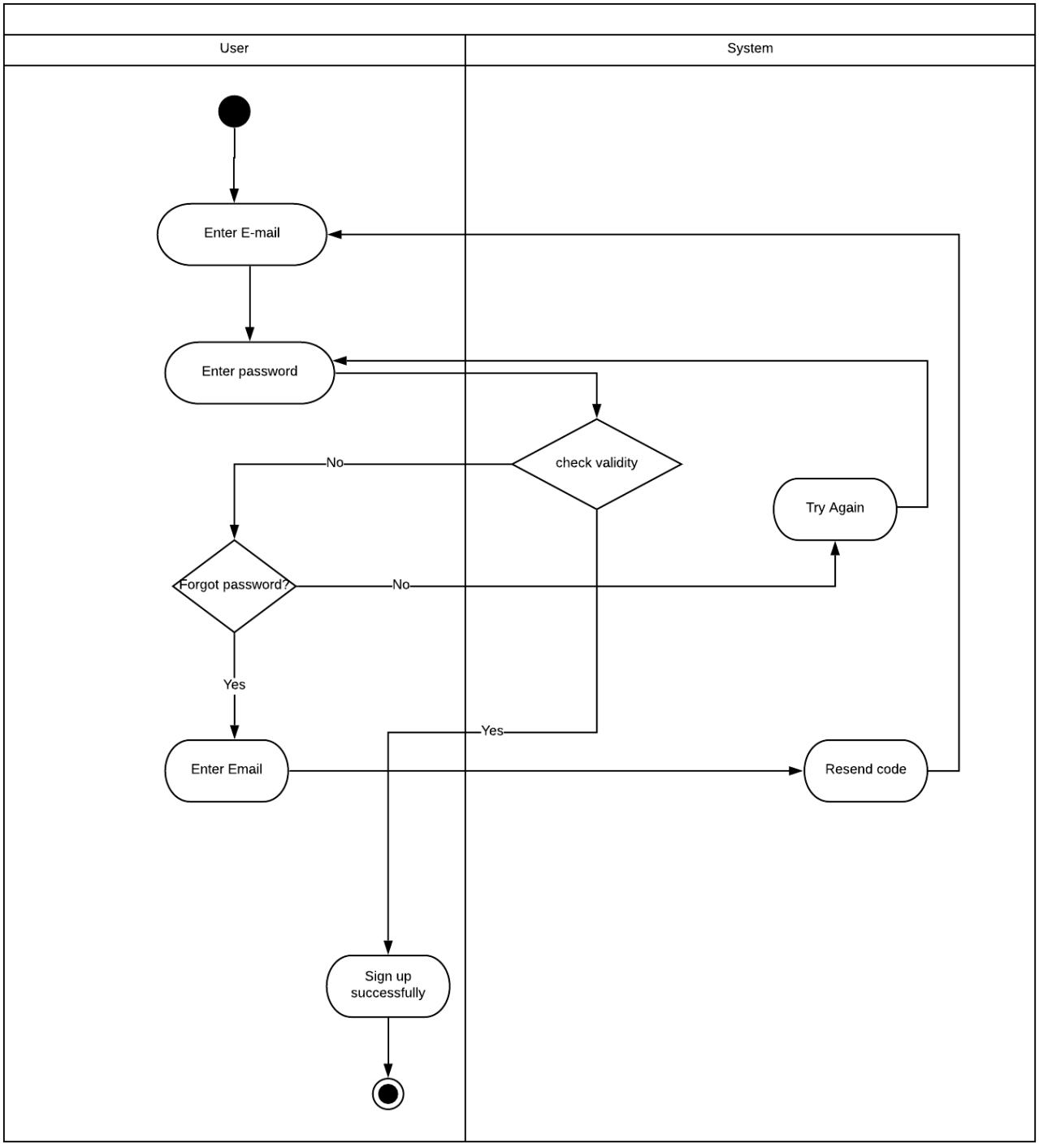


Figure 17: Level 4.4.2 Swim-lane Diagram of Log In

4.4.3 Swim-lane diagram of Guest user

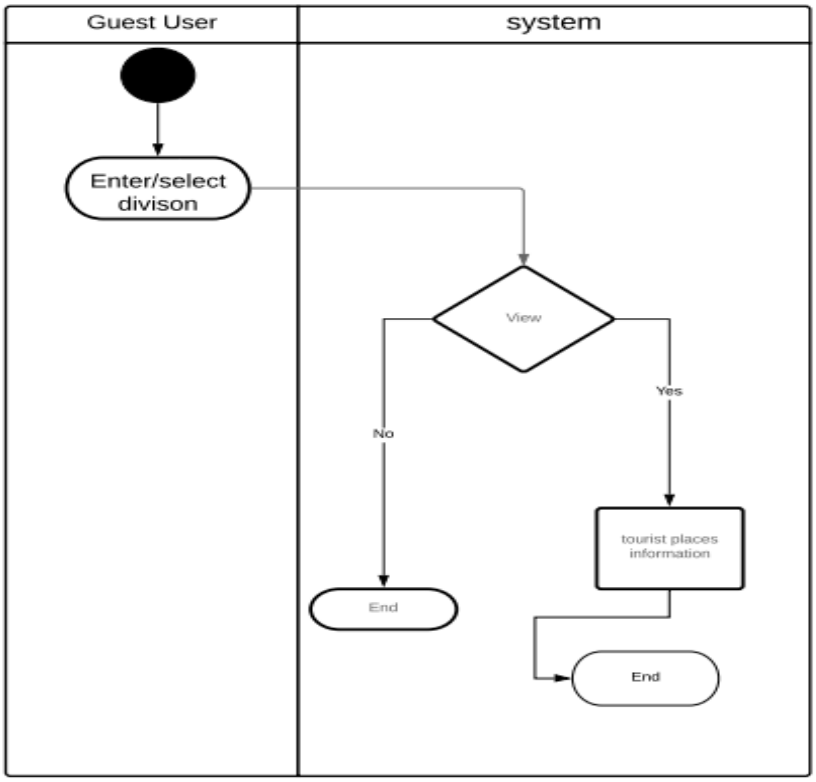


Figure 18: Level 4.4.3 Swim-lane Diagram of Guest user

4.4.4 Swim-lane diagram of Browse

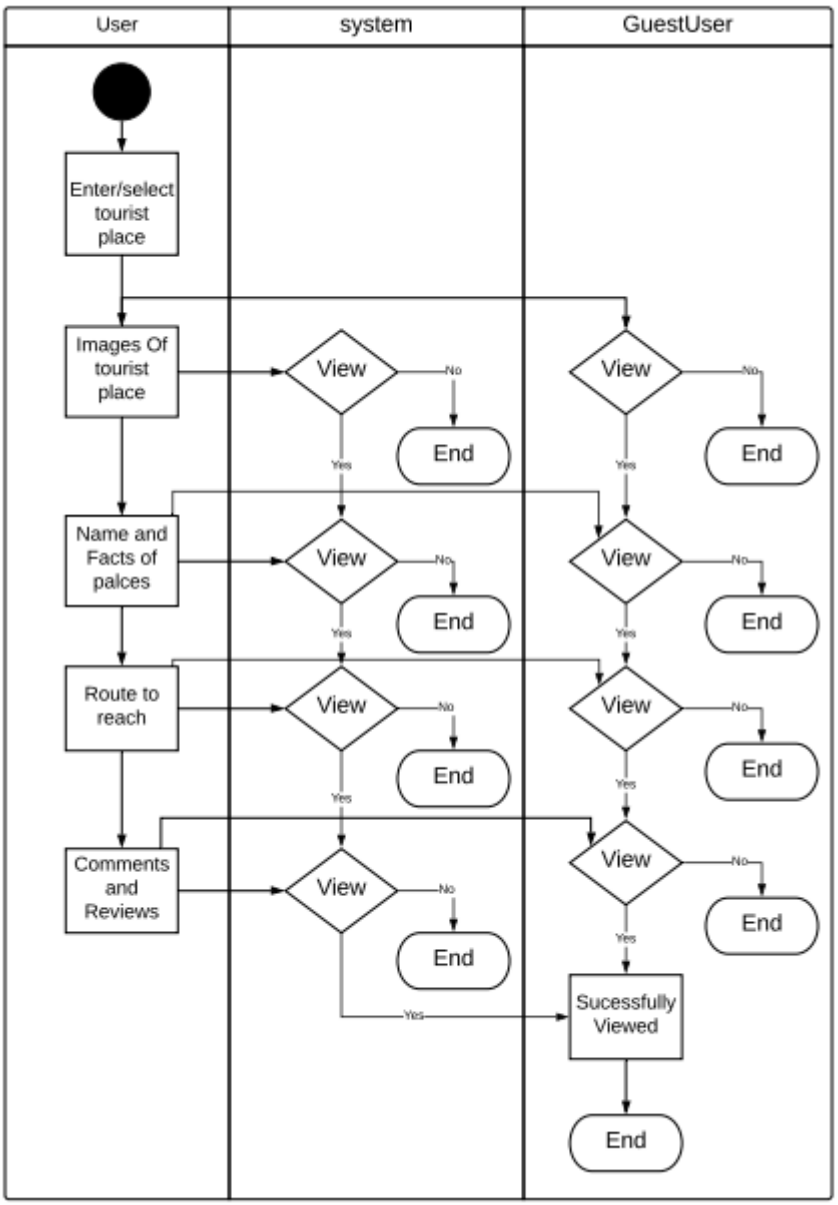


Figure 19: Level 4.4.4 Swim-lane Diagram of Browse

4.4.5 Swim-lane diagram of Update

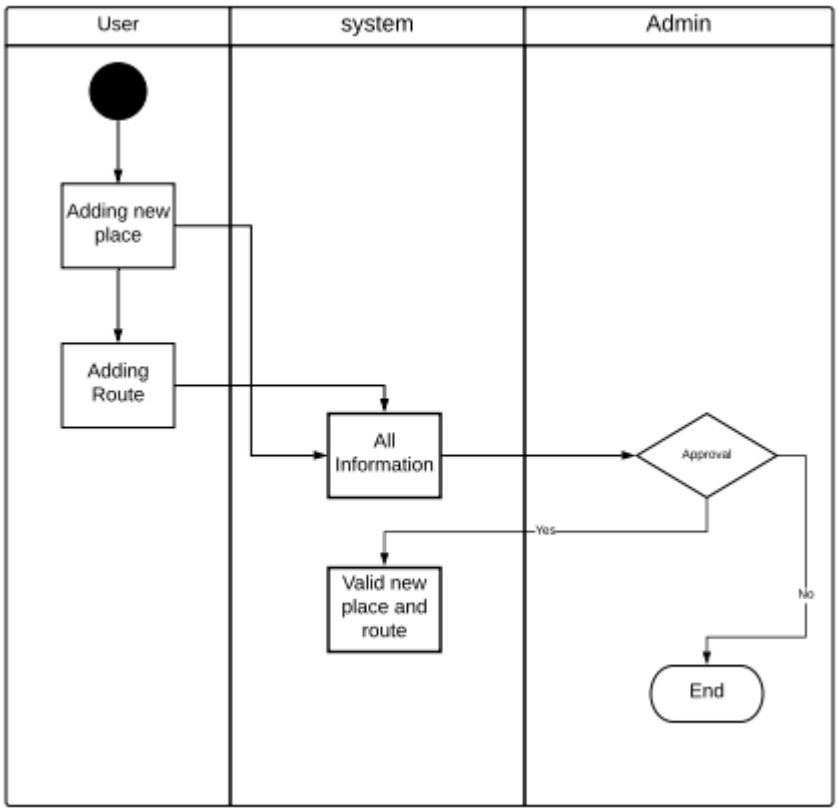


Figure 20: Level 4.4.5 Swim-lane Diagram of Update

4.4.7 Swim-lane diagram of Edit contact

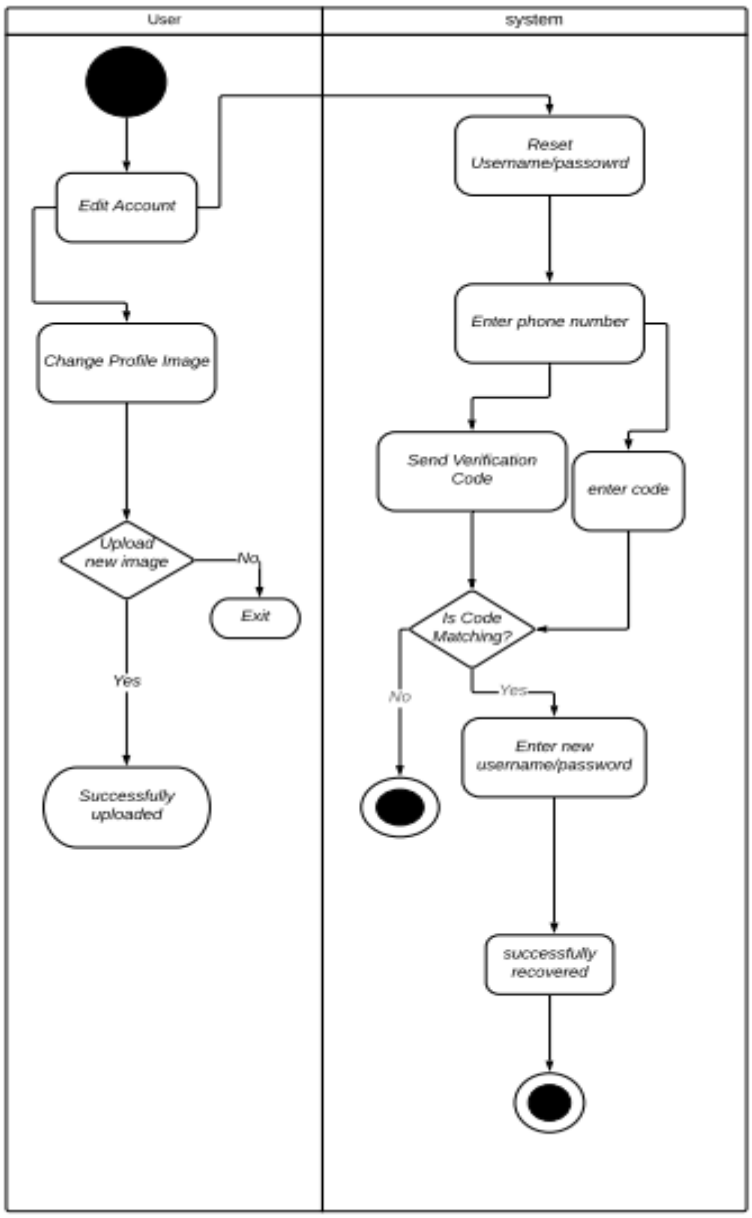


Figure 21: Level 4.4.5 Swim-lane Diagram of Edit contact

4.4.8 Swim-lane diagram of Emergency contact

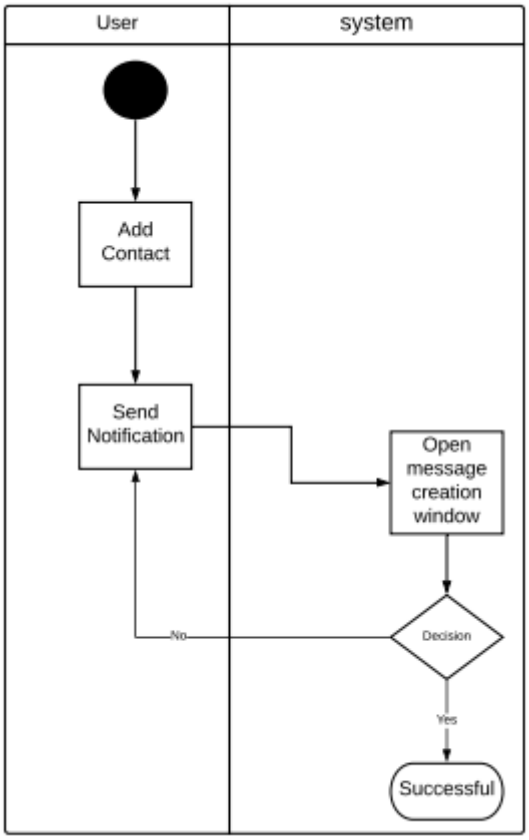


Figure 22: Level 4.4.6 Swim-lane Diagram of emergency contact

Chapter Five

Data Model

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.

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:

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	
12.	Sign out	S	
13.	Place Image	P	
14.	Place Name	P	
15.	Travel Agency	S	36
16.	Route	P	
17.	Profile	P	4
18.	Search	P	
19.	Location	S	36,38
20.	Division	P	
21.	Address	S	36,38,37
22.	Comment	S	
23.	Emergency Contract	S	
24.	Edit Account	P	
25.	Send Notification	S	
26.	Profile photo	S	4

27.	Settings	S	
28.	Database	P	
29.	Edit Profile	P	
30.	Remove	S	
31.	Email	S	
32.	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	

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

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.

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

5.2.5 Relationship among Data Objects

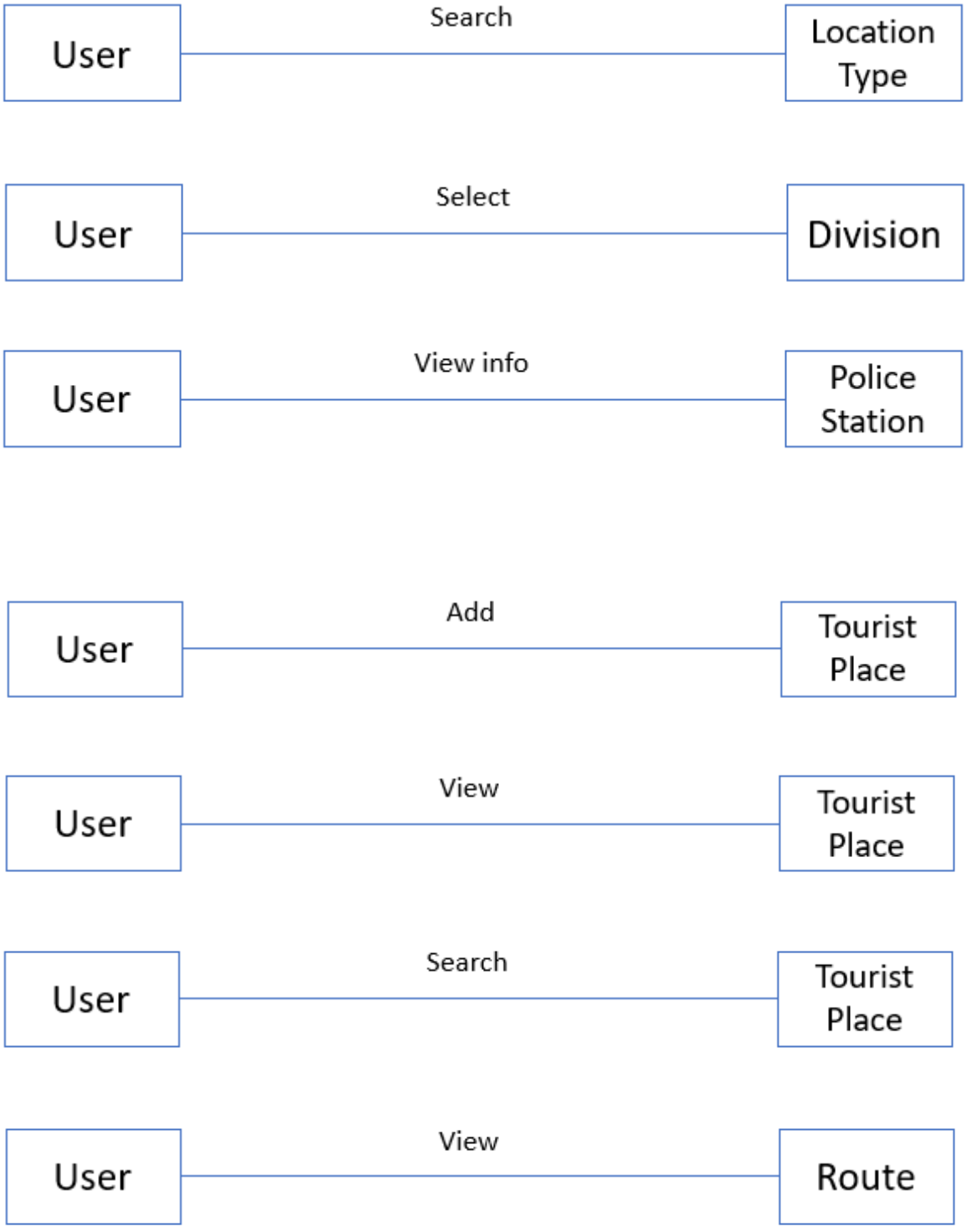


Figure : Relationship among Data Objects

5.3 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.

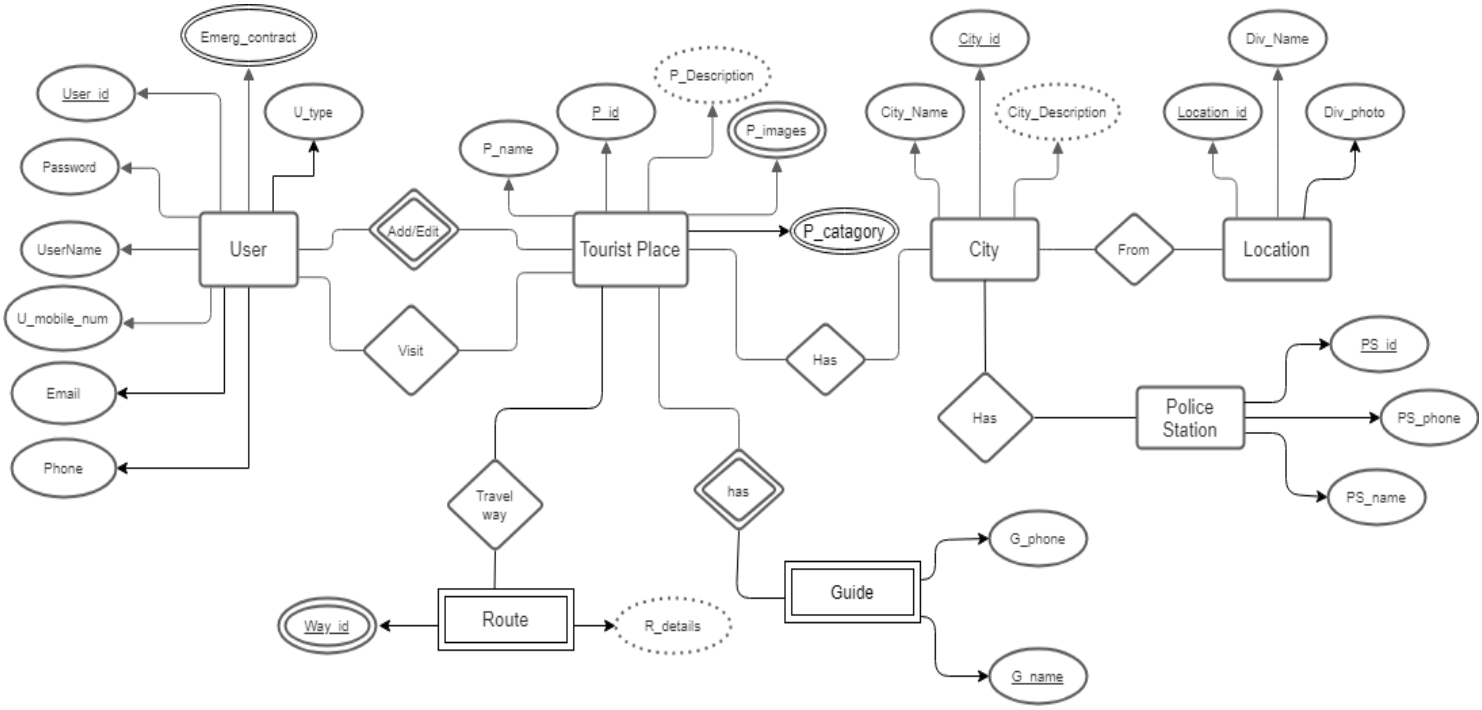


Figure : Entity Relationship Diagram

5.4 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

Chapter Six

Class Based Model Of Visit Bangladesh Android Application

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.

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.

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
5.	Username	S	2
6.	Id	S	2
7.	Password	S	2
8.	Account	P	4,5
9.	Nearest Place	S	4
10.	Information	S	2
11.	Sign in	S	3
12.	Sign out	S	3
13.	Place Image	P	
14.	Place Name	P	
15.	Travel Agency	S	5
16.	Route	P	

17.	Profile	P	2
18.	Search	P	
19.	Location	S	2
20.	Division	P	6
21.	Address	S	6
22.	Comment	S	2
23.	Emergency Contract	S	2
24.	Edit Account	P	
25.	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

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
18.	Search	1,2,3
19.	Location	2,3,4,5
20.	Division	3,4,2,5
21.	Address	
22.	Comment	
23.	Emergency Contract	

24.	Edit Account	
25.	Notification	2,3,4,6
26.	Profile photo	
27.	Settings	2
28.	Database	1,2,6
29.	Edit Profile	
30.	Remove	
31.	Email	
32.	Phone	
33.	Date	
34.	Update Location	2,
35.	Guide	
36.	Tourist place	
37.	City	
38.	Police Station	
39.	Admin	1,2,3,4,5
40.	Add	

6.2.1 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

6.2.3 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

6.2.3 Method Identification:

Table : Methods of Class

No.	Class	Methods
1	Account	• signUp() • login() • signOut() • lockAccount() • sendVerificationCode() • recoverPassword() • verifyEmail() • verifyUser() • manageForgotPassword()
2	User	• setUserID() • getUserID () • setName() • getName() • setEmail() • getEmail() • setPhoneNo() • getPhoneNo() • setEmergencyContractNo() • getEmergencyContractNo () • addPlace() • Emergency()
3	Admin	• addPlace() • removePlace() • editPlace() • editProfile()

4	System	• viewOptions() • showSearchedItem() • generateSMS() • sendSMS() • setConnection()
5	Notification	• setNotificationID() • getNotificationID() • getSender() • setSender() • setReceiver() • getReceiver() • getDescription() • setDescription() • sendNotification() • generateNotification()
6	Search	• searchbyCity() • searchbyPlace() • searchbykeywords() • searchPopular() • searchTypewise() • searchRecommended()
7	Activity	• storeAuthenticationInfo() • storeLocationModificationInfo() • storeNewLocationInfo()
8	Database	• insertInfo() • updateInfo() • deleteInfo() • retrieveInfo()
9	Location	• getLocationID() • setLocationID()

		• getDivisionName() • setDivisionName () • getDivisionPhoto() • setDivisionPhoto ()
10	City	• getCityID() • setCityID() • getCityName() • setCityName() • getCityDetails() • setCityDetails() • getPoliceStationID() • setPoliceStationID() • getPoliceStationName() • setPoliceStationName() • getPoliceStationDetails() • setPoliceStationDetails()
11	Tour Place	• getTourPlaceID() • setTourPlaceID() • getTourPlaceName() • setTourPlaceName() • getTourPlaceImages() • setTourPlaceImages() • getTourPlaceDescription() • setTourPlaceDescription() • getRoute() • setRoute() • getRouteDetails() • setRouteDetails () • getGuideName() • setGuideName() • getGuideNumber() • setGuideNumber()

6.2.4 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 alluser 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 alluser class.

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

6.3 Class Responsibility Collaborator modeling

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

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

Responsibility	Collaborators
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

Responsibility	Collaborators
Authentication	Account
Add Place Info	Database
Request Emergency	System

Table 22: Admin

Responsibility	Collaborators
Authentication	Account
Add Place Info	Location
Edit Place Info	Location

Table 23: System

Responsibility	Collaborators
Get searched Items	Search
Generate Notifications	Notification
Generate Emergency Message	Notification

Table : Notification

Responsibility	Collaborators
Gets Notification	User
Generate Notifications	System

Table : Search Card

Responsibility	Collaborators
Generating Search	System
Search for Tour Place	Location
Search for City	Location

Table : Database Card

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

Table : Location Card

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

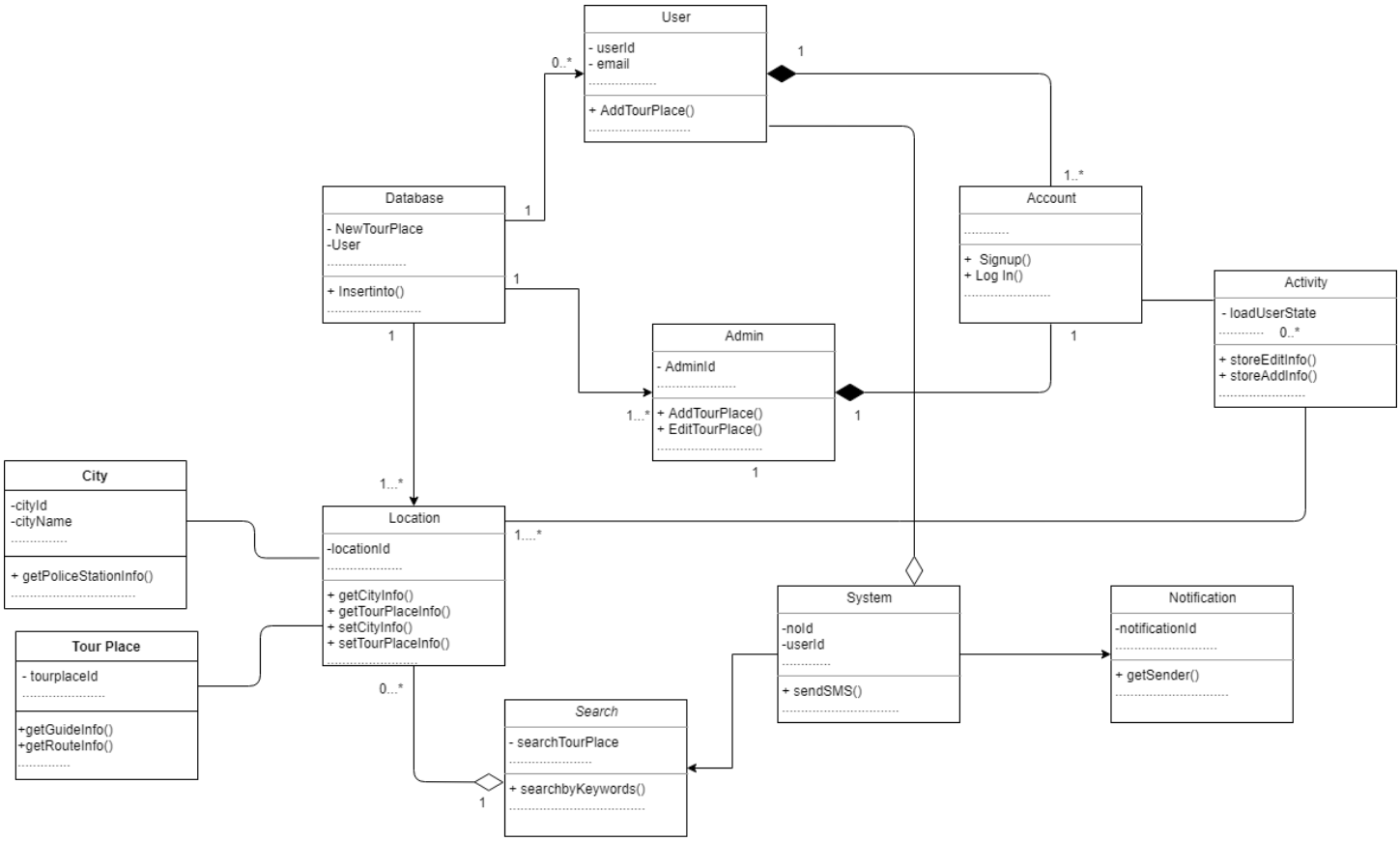
Table : City Card

Responsibility	Collaborators
Update/View city Information	Location

Table : Tour Place Card

Responsibility	Collaborators
Update/View Tour Place Information	Location

6.3.2 UML Diagram



CHAPTER 7

DATA FLOW MODEL OF ‘Visit Bangladesh’ A MOBILE APPLICATION

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:

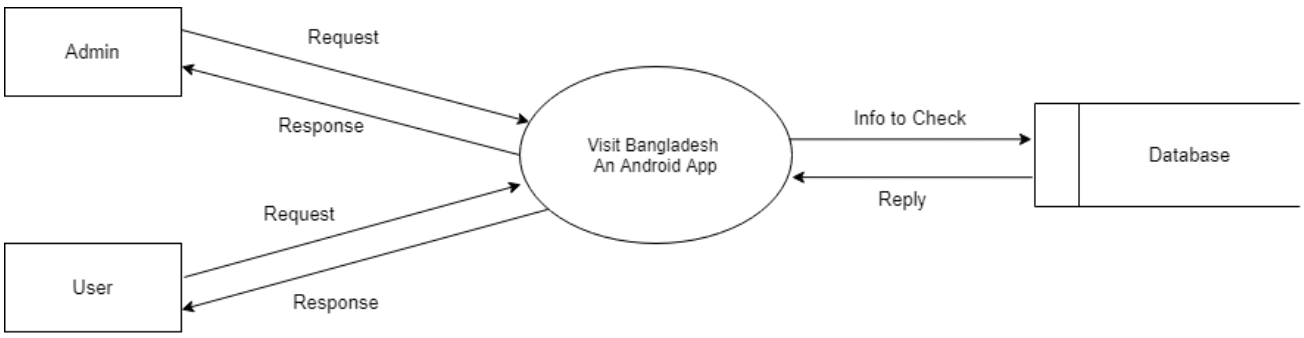


Figure 1: level 0 for Visit Bangladesh

7.2.1: Level 1.1

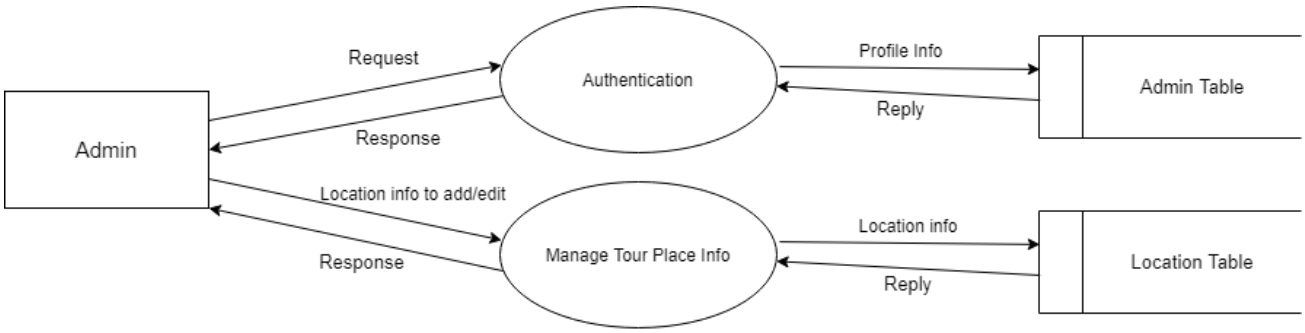


Figure 2: level 1.1 for Visit Bangladesh

7.2.2. Level 1.2

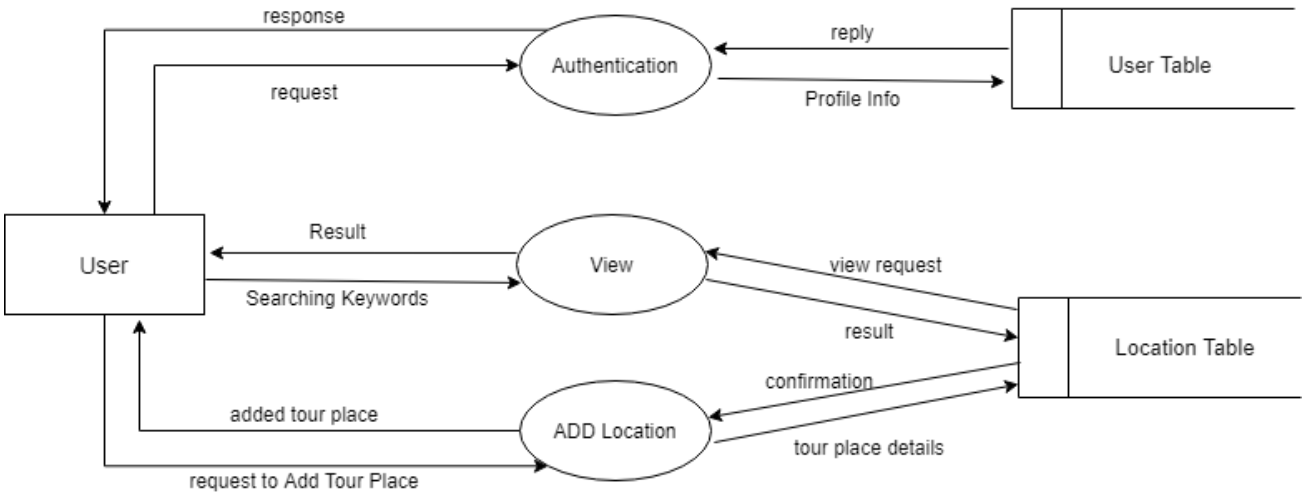


Figure 3: level 1.2 for Visit Bangladesh

7.3.1 Level 2.1

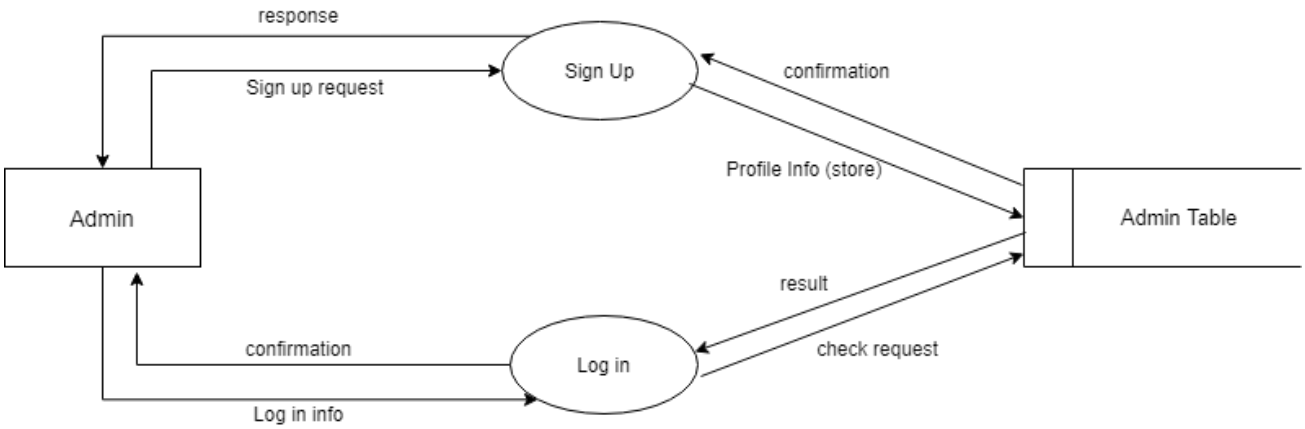


Figure 4: level 2.1. for Visit Bangladesh

7.3.2 Level 2.2

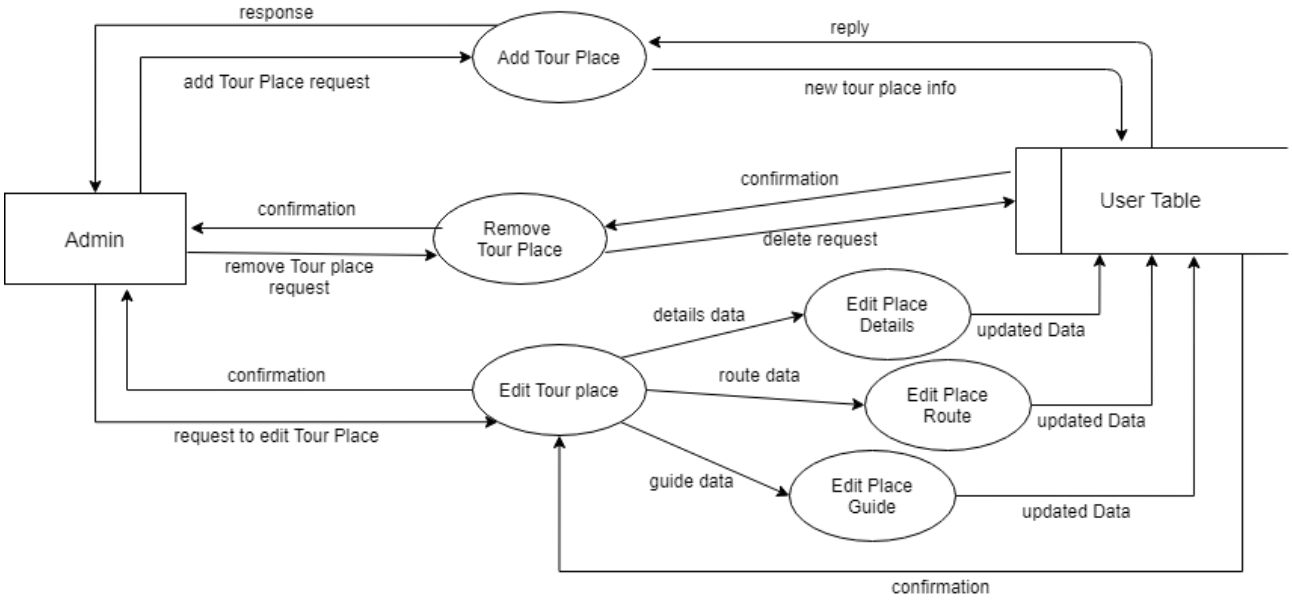


Figure 5: level 2.2 for Visit Bangladesh

7.3.3 Level 2.3

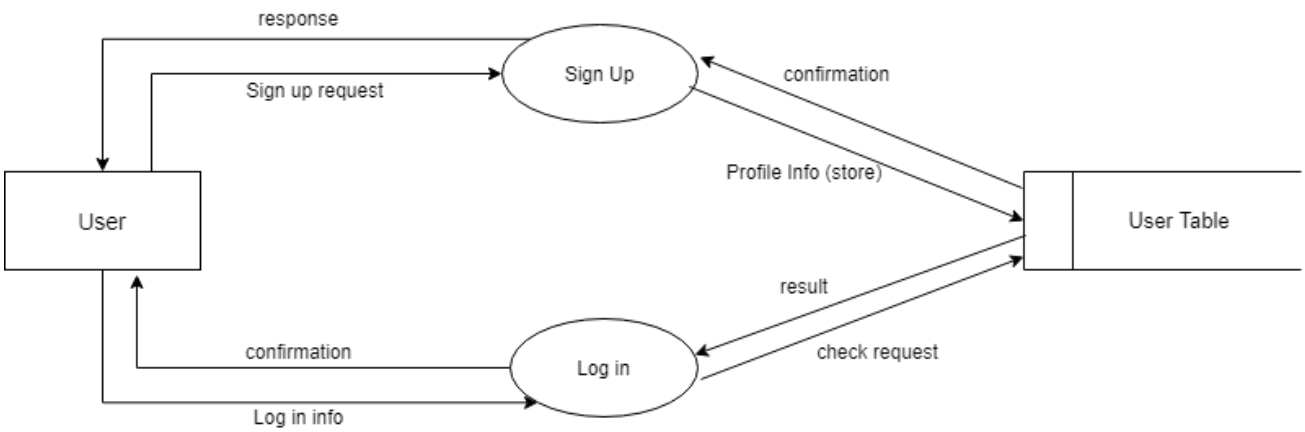


Figure 6: level 2.3 for Visit Bangladesh

7.3.4 Level 2.4

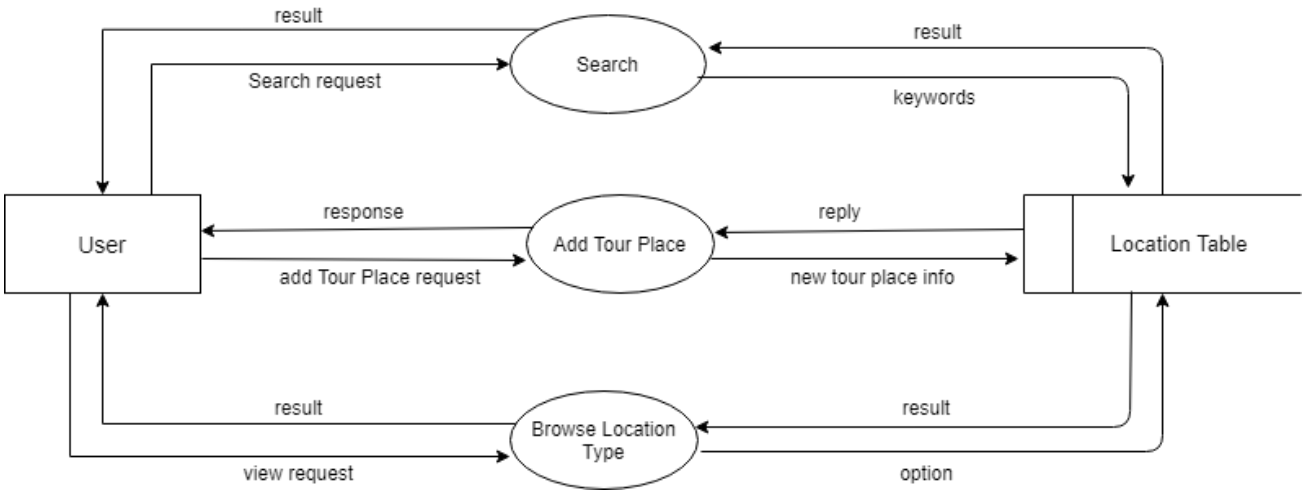


Figure 7: level 2.4 for Visit Bangladesh

7.3.5 Level 2.5

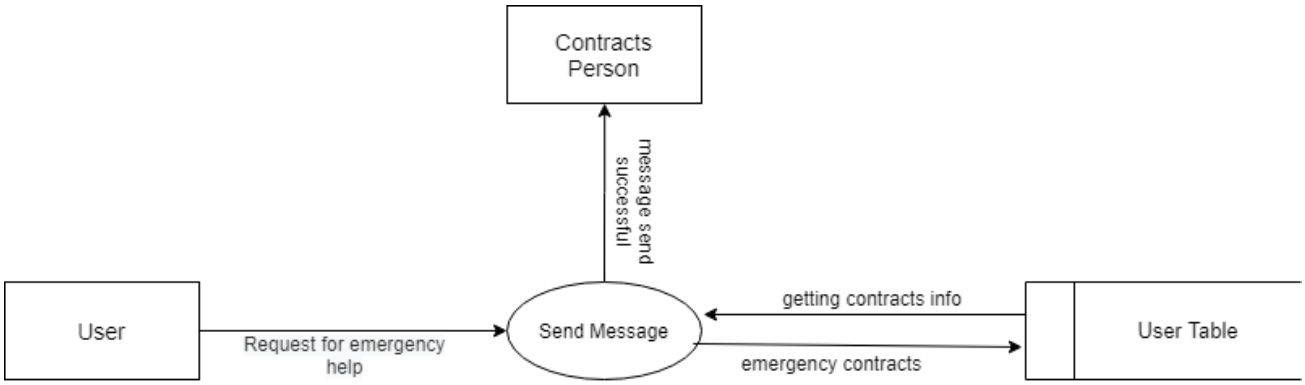


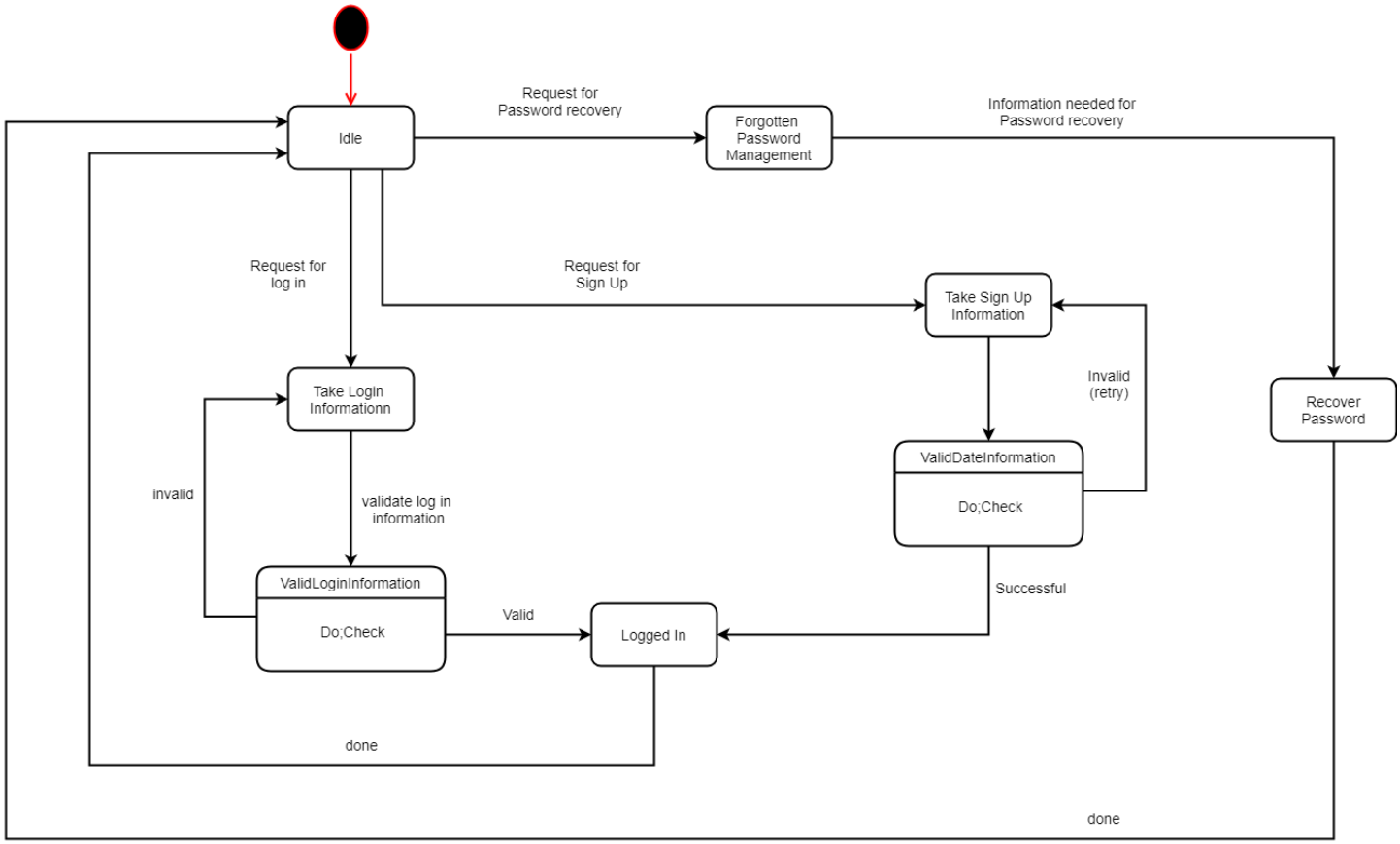
Figure 8: level 2.5 for Visit Bangladesh

CHAPTER 8

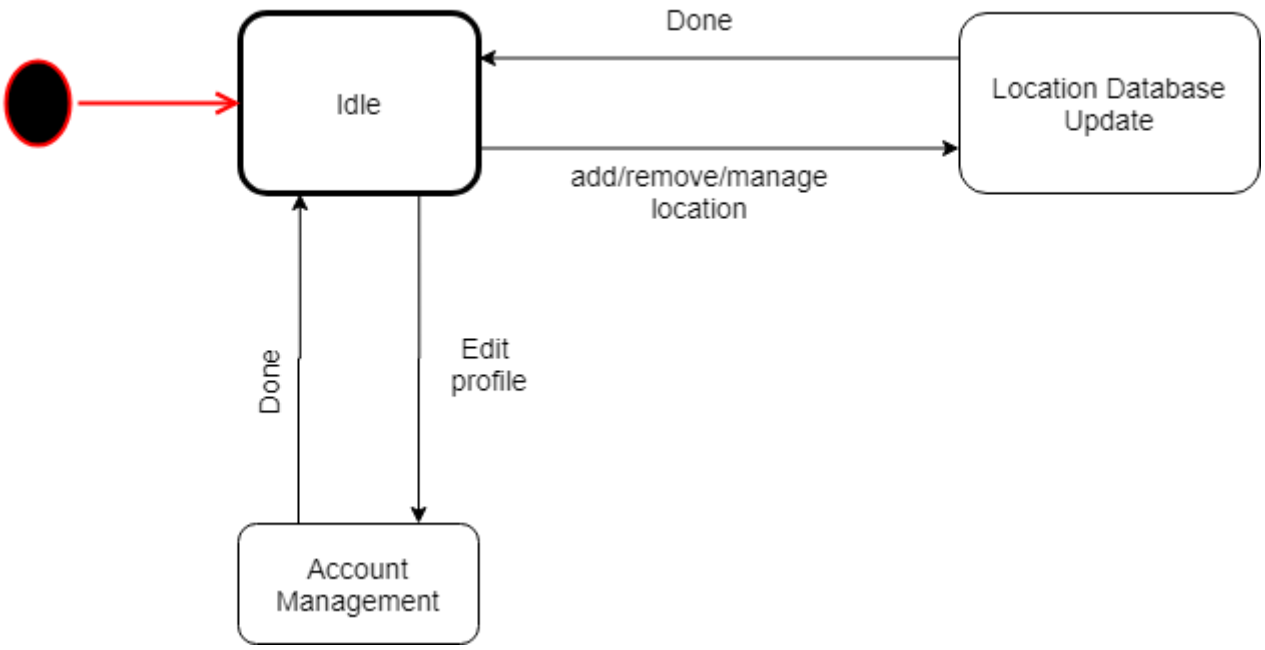
BEHAVIORAL MODEL OF 'Visit Bangladesh' A MOBILE APPLICATION

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 an user. State diagram of this project are following:

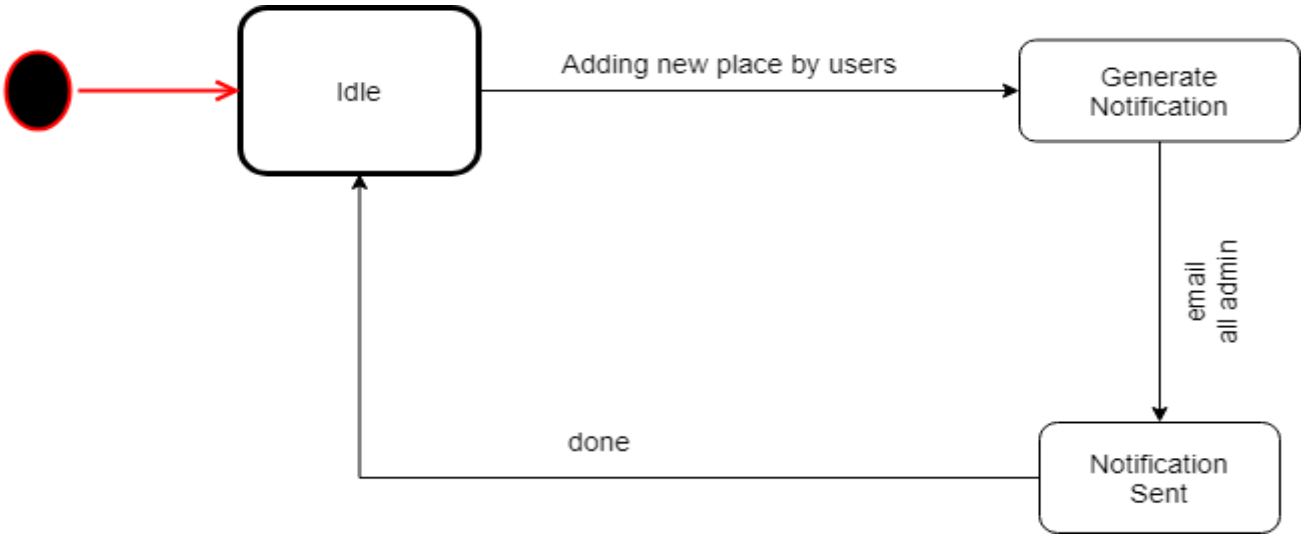
8.1.1 State diagram of Account :



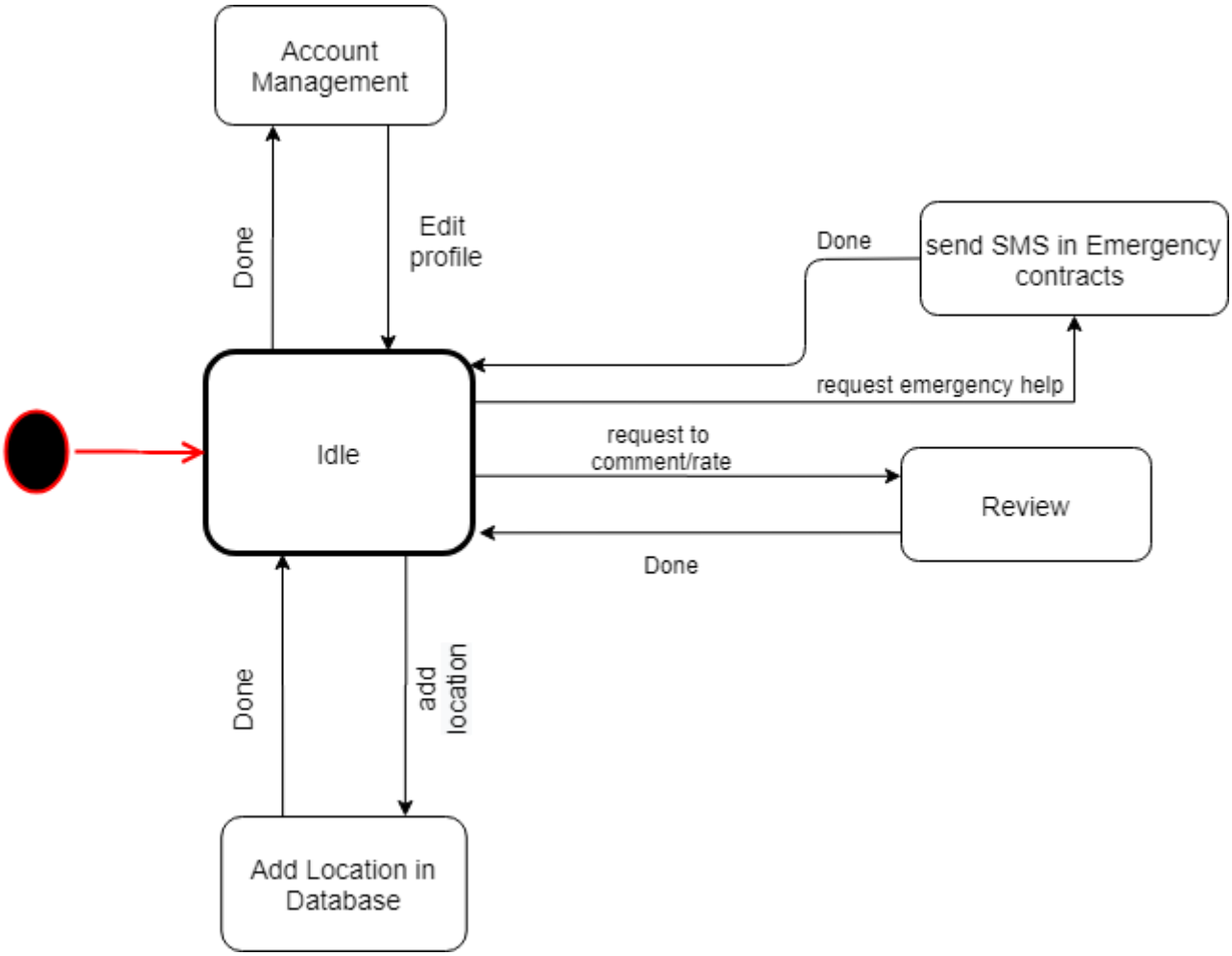
8.1.2 State diagram of Admin:



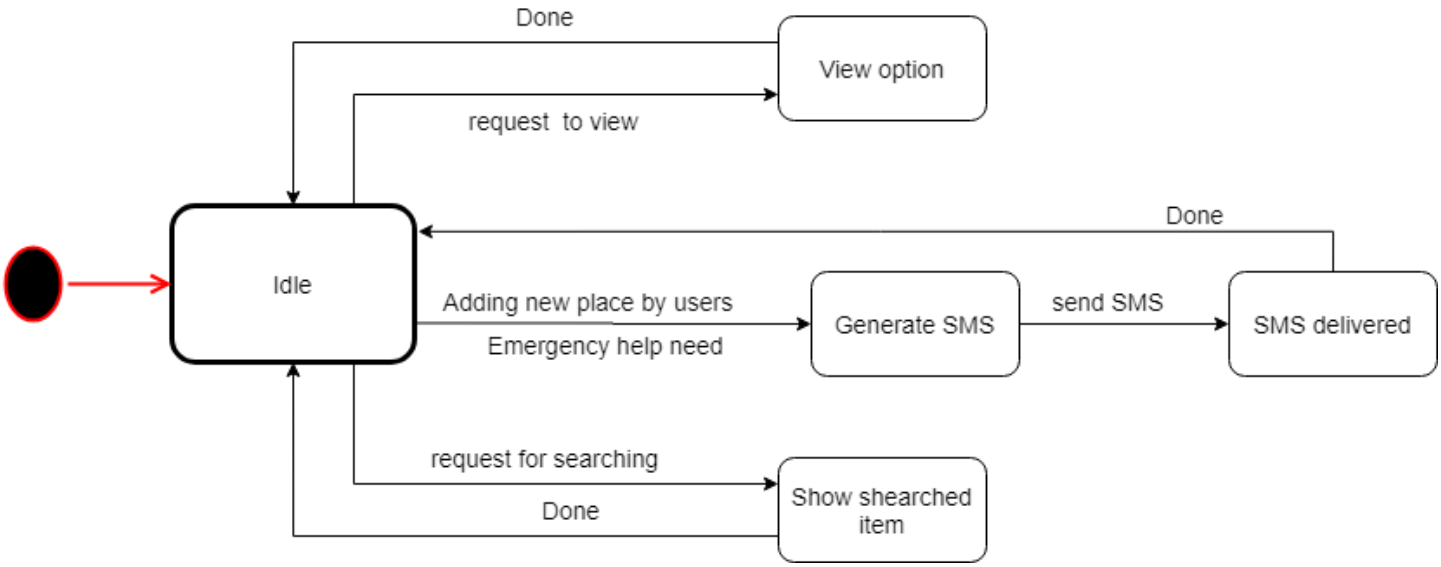
8.1.5 State diagram of Notification:



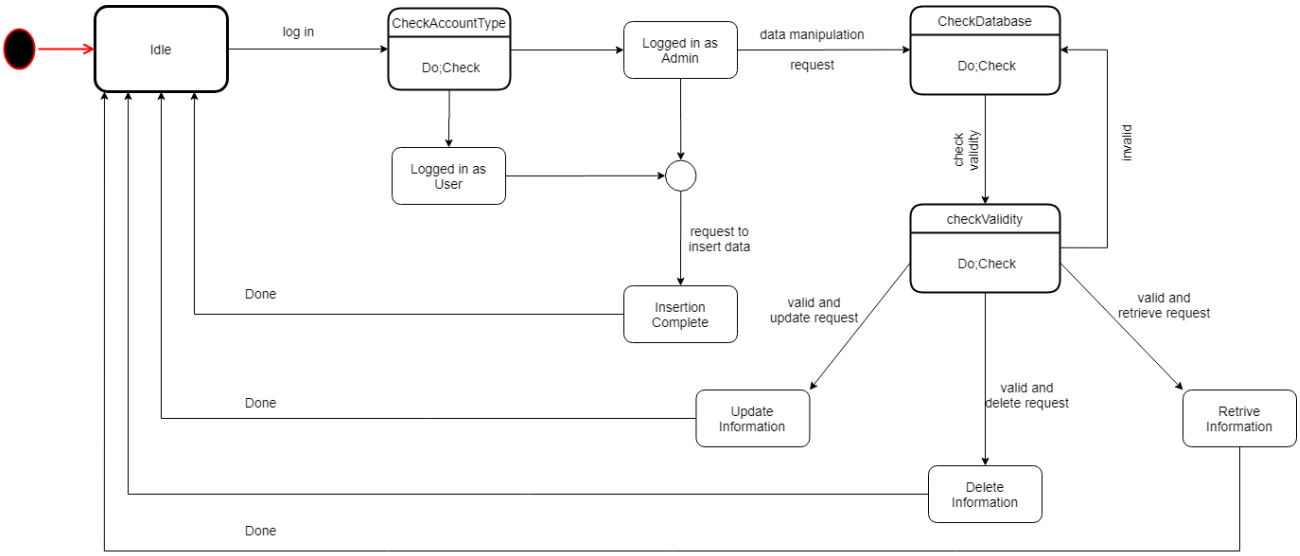
8.1.3 State diagram of User:



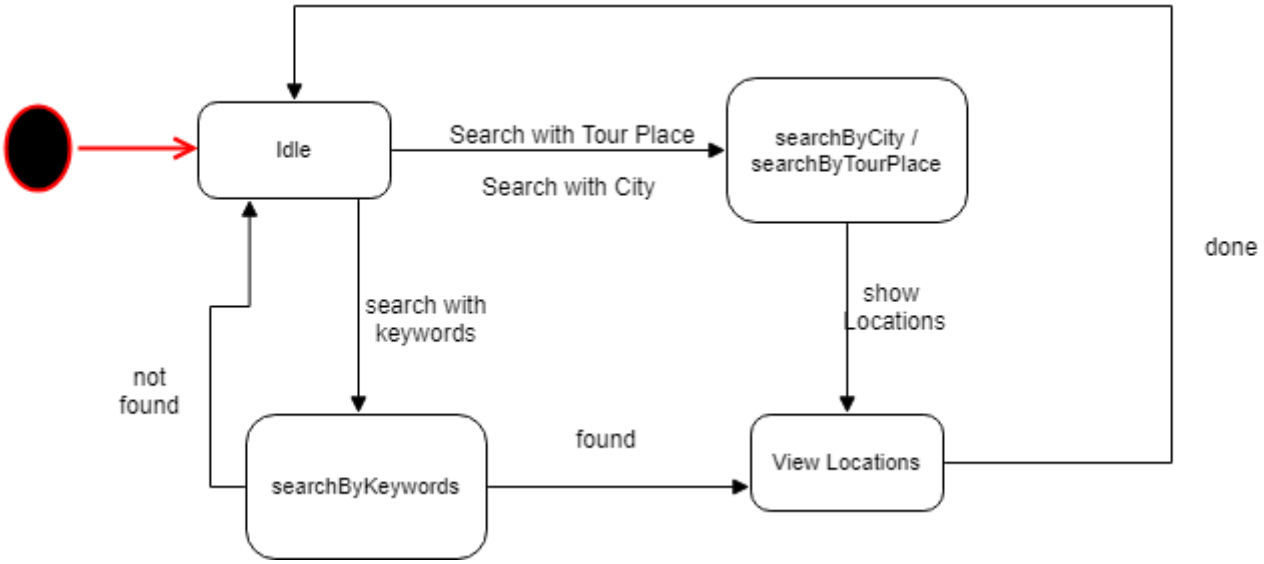
8.1.4 State diagram of System:



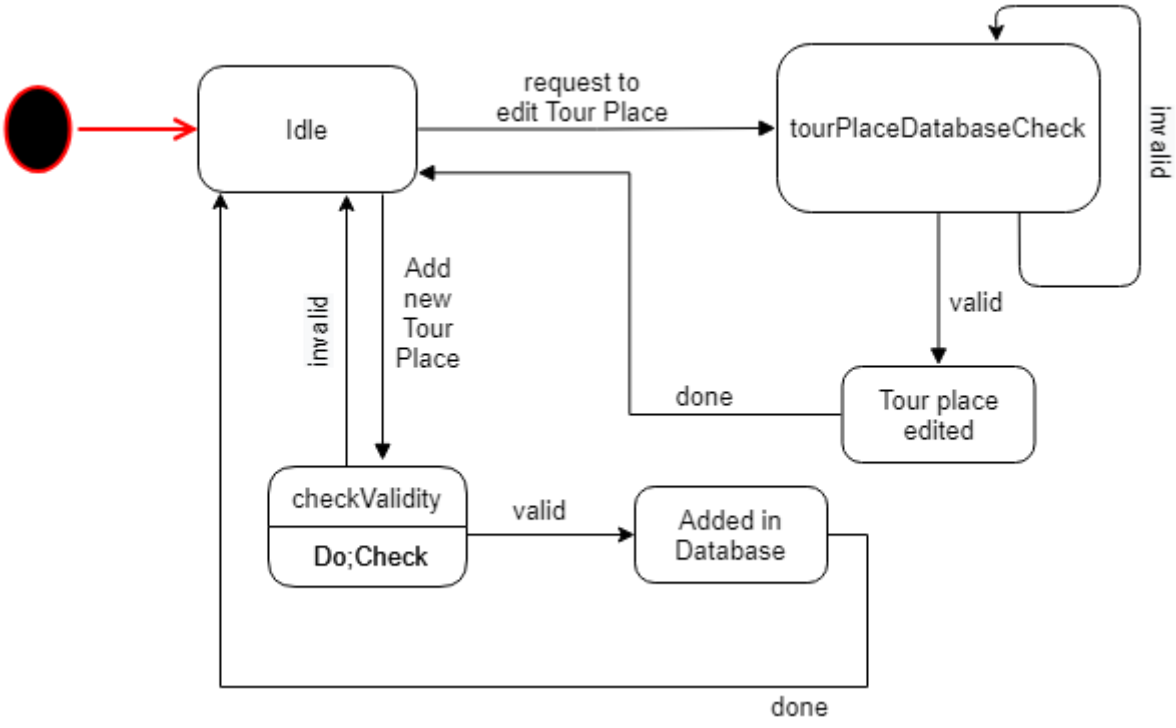
8.1.6 State diagram of Database:



8.1.7 State diagram of Search:



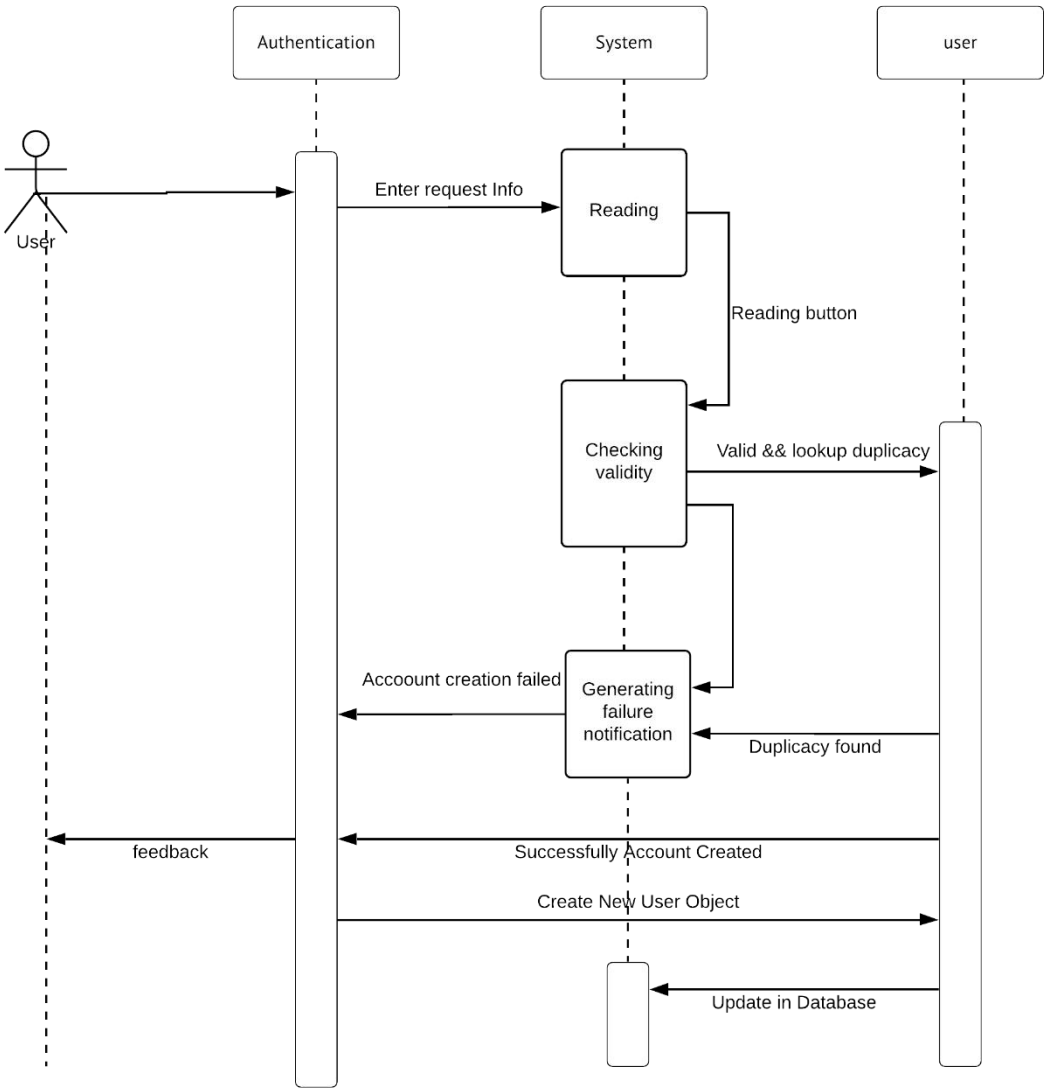
8.1.8 State diagram of Location:



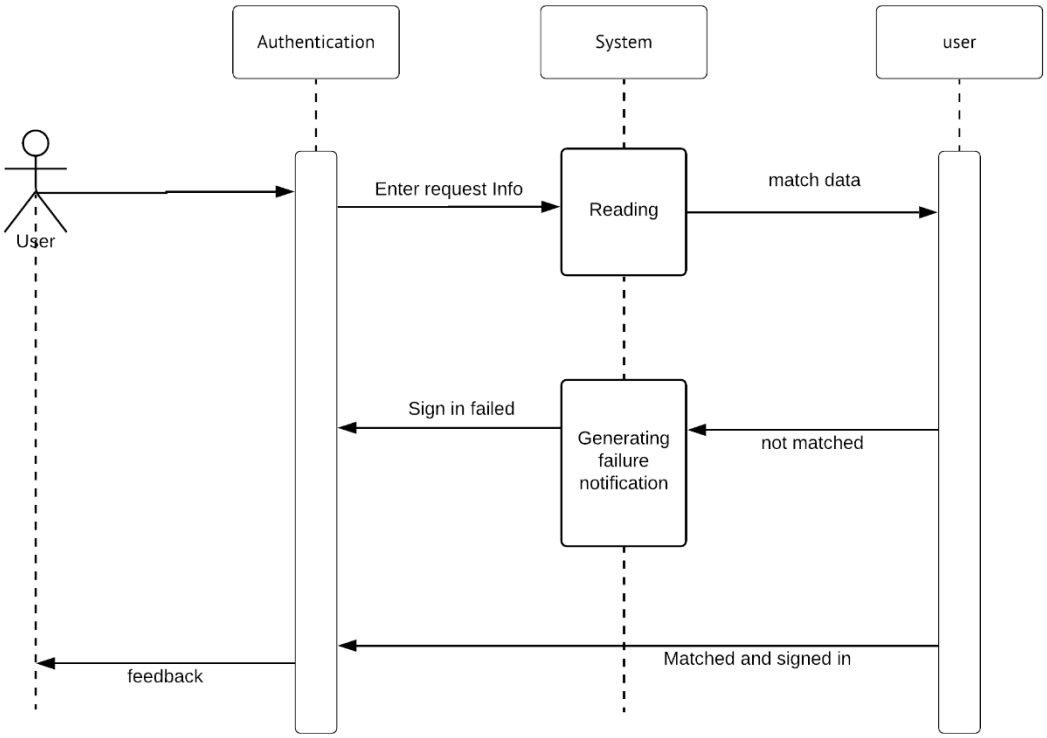
8.2 Sequence diagram

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.

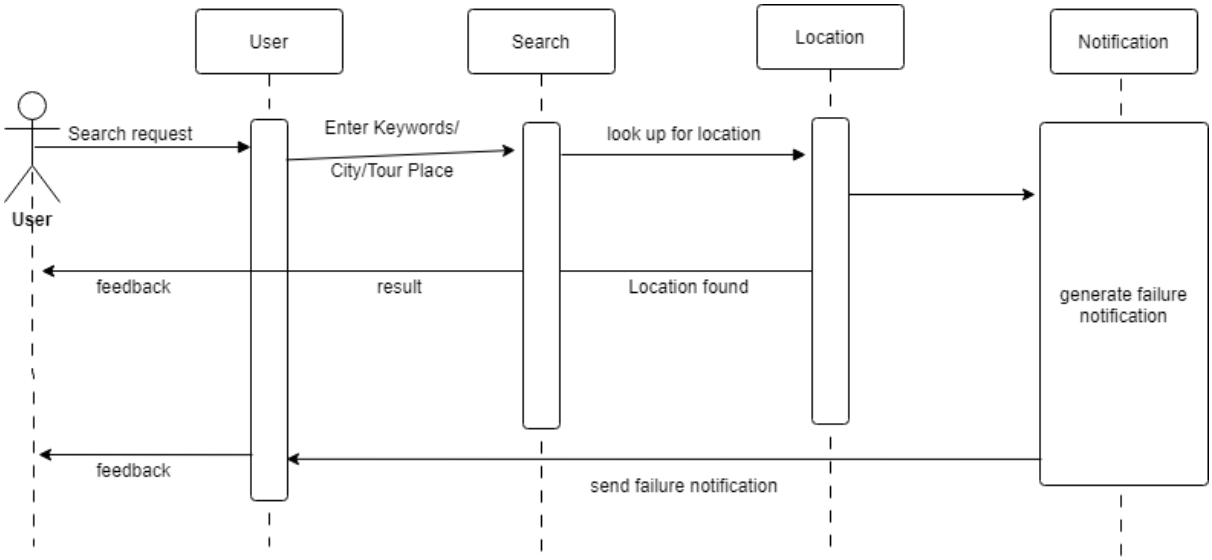
8.2.1 Sequence diagram of Sign Up



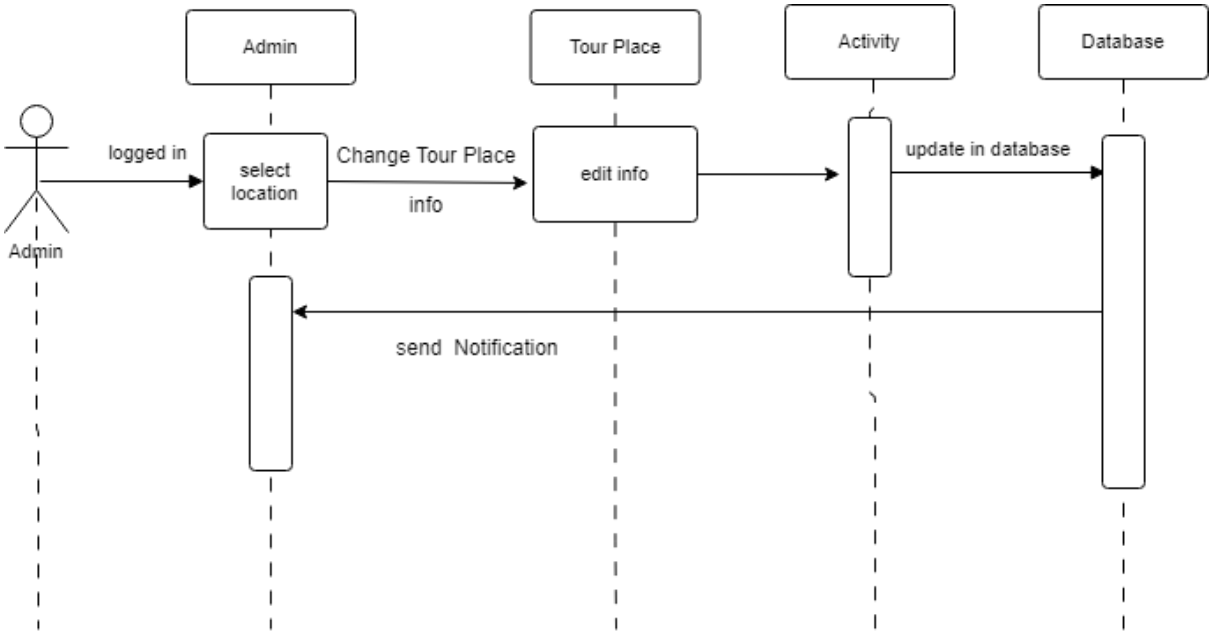
8.2.2 Sequence diagram of Sign In



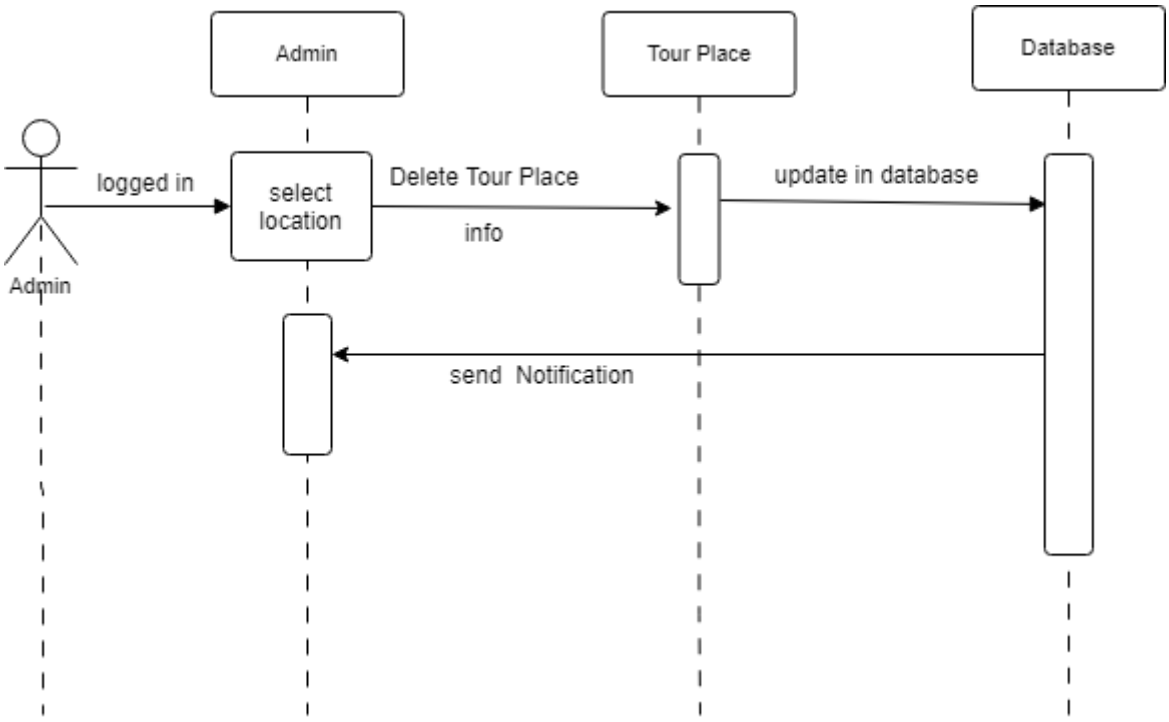
8.2.3 Sequence diagram of Search Location:



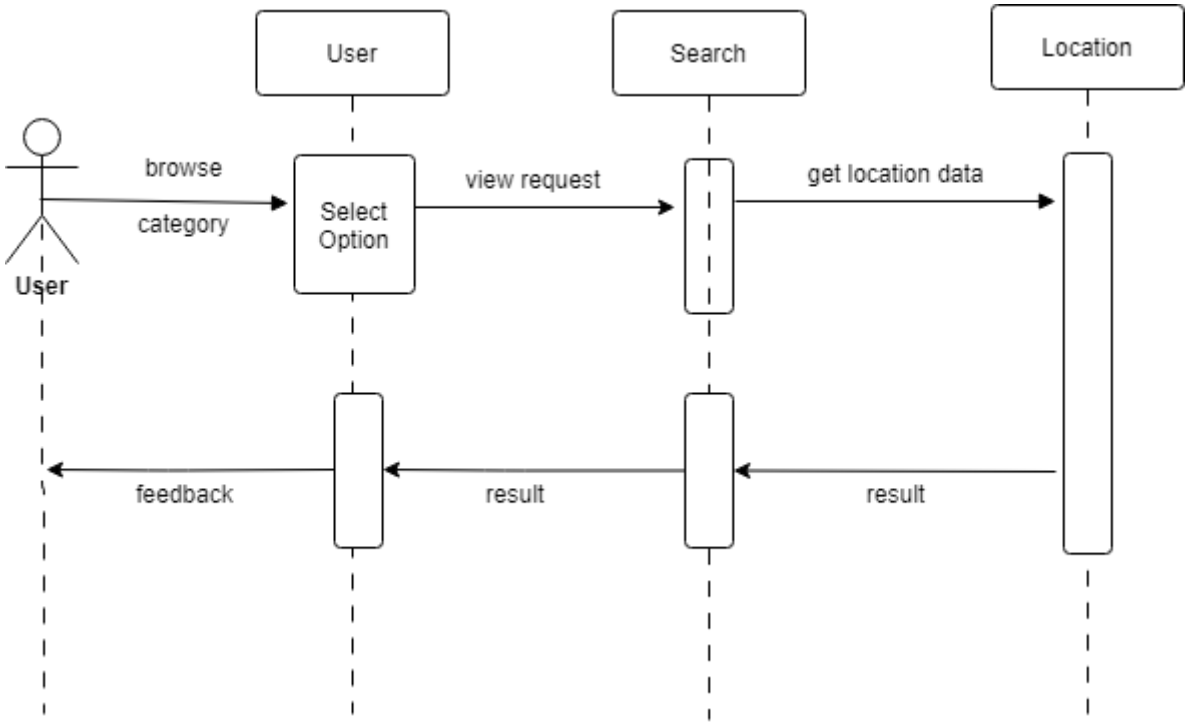
8.2.4 Sequence diagram of Edit Location:



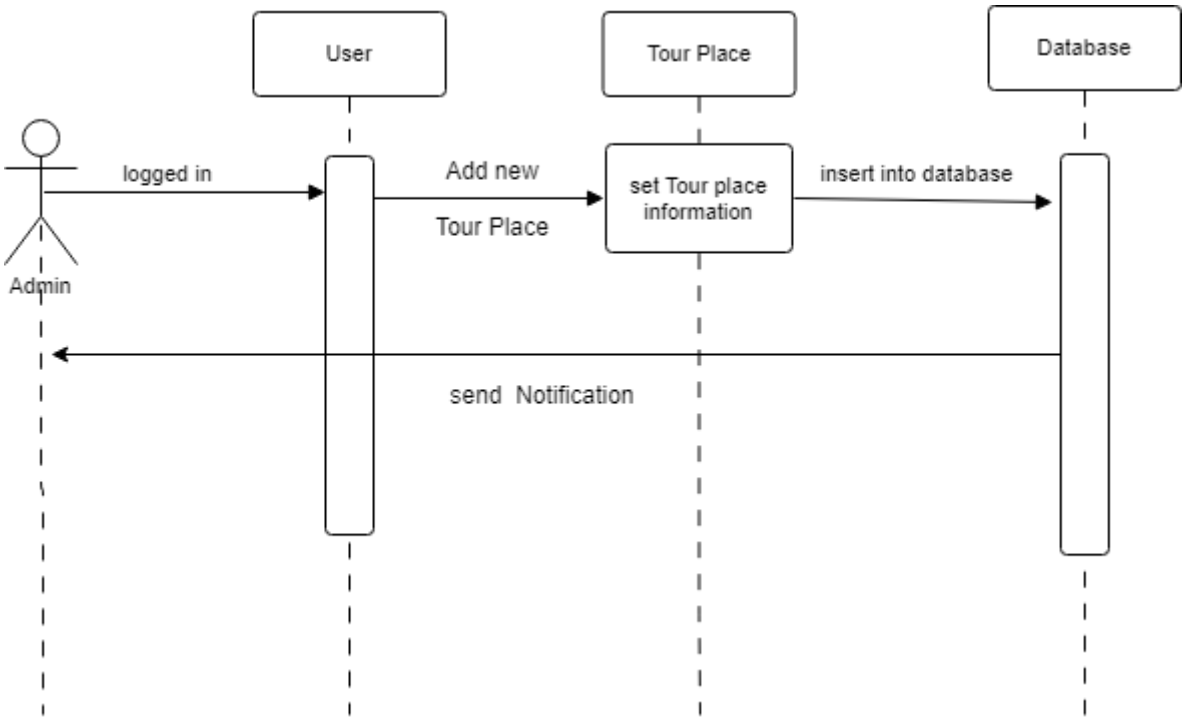
8.2.5 Sequence diagram of Delete Location:



8.2.6 Sequence diagram of Browse Location:



8.2.1 Sequence diagram of Add Location:



Chapter 9: Conclusion

From this SRS report on Visit Bangladesh an android tour guide application for Bangladesh.

We hope our report will be able to convey a clear picture of our system to all stakeholders, and act as a basis throughout full development cycle. We have tried our best to make a complete, practical, consistent and unambiguous specifications; which helped us tremendously in our understanding of the scope and detailed process of software requirement engineering process. We think that this report has been written in an easy-to-read way as well as with full information required to have a good concept over the idea. We sincerely hope this document will be able to satisfy the goals all stakeholders expect from it. We hope that any reader going through this document can easily understand the whole idea behind the Visit Bangladesh Android App. Hopefully, it will be an easy path-showing document for the implementation of the application!

We tried best to remove all dependencies and make an effective and fully designed SRS.