Local Tourist

By

Ashraful Islam Sheiblu
152-35-1129

A thesis submitted in partial fulfillment of the requirement for the degree of

Bachelor of Science in Software Engineering

Department of Software Engineering
DAFFODIL INTERNATIONAL UNIVERSITY

Spring - 2019

Letter of Approval

This **Project** titled "Local Tourist", submitted by **Ashraful Islam Sheiblu**, **152-35-1129** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc in Software Engineering and approved as to its style and contents.

BOARD OF EXAMINERS

Dr. Touhid Bhuiyan
Professor and Head   **Chairman**
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

K. M. Imtiaz-Ud-Din
Assistant Professor   **Internal Examiner 1**
Department of Software Engineering
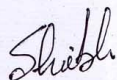Faculty of Science and Information Technology
Daffodil International University

Md. Shohel Arman
Lecturer   **Internal Examiner 2**
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Mr. Shafiul Hasan
Managing Director   **External Examiner**
Vivacom Solution, Dhaka

DECLARATION

We hereby declare that we have taken this thesis under the supervision of **Mr. Iftekharul Alam Efat**, Senior Lecturer, **Department of Software Engineering, Daffodil International University**. We also declare that neither this thesis/project nor any part of this has been submitted elsewhere for award of any degree.

Ashraful Islam Sheiblu
ID: 152-35-1129
Batch : 17th
Department of Software Engineering
Faculty of Science & Information Technology
Daffodil International University

Certified by:

Mr. Iftekharul Alam Efat

Senior Lecturer

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

©Daffodil International University

Acknowledgement

After all, I was successful as I successfully replaced the final semester. And so I'm pleased with my Almighty. From the beginning of my university career, I learned a lot about the knowledge about software engineering and core computer science about my course teachers. Moreover, they teach our ethics, ethics, and modesty.

As well, I am very grateful to my parents that I was always supported by them. They always inspire me to my opinion priorities and as well.

My supervisor to allow me to work with this project. **Mr. Iftekharul Alam Efat** too much appreciated. He always supports me to make this project successful. Especially I am very grateful to my brother, who always admires my professionalism.

Executive Summary

January was the when we started the development phase. We all know that, people have to put hard work, dedication, and full concentration to complete a project successfully.

Here comes to requirements analysis to help the developments phase. Requirements analysis helps every step of development phase.

Database it's the important part of any system. That's why, in the initial step of development phase I have focused on database design, make the table and their attribute.

Complete a project in not easy at all. Also it's not the end, you have to ensure that you project works properly and for that you have to test your project. And here comes the testing part. Testing ensure that your system quality.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Project Overview

Traveling teaches you more than any university. Travelling make your life better and happier in several ways. You will meet the different people during traveling. If you want to be more open and more accepting then you should travel. You gain experience through traveling that you never get at home. There will be some people in traveling that will change your life. If you want more independent and confident go to travel.

But do you realize when we travel any place, do we visit it properly? No, we don't visit any placed every inch. Think about the general traveling system, people go only main focused place say like Cox Bazar. And Cox Bazar to another focused placed. And they never reach any placed deeply. Because of the lack of communication, security reason, etc.

For the reason, we are going to build an application called "Local Tourist". We build this application because of helping the traveler to travel the hole around of any placed.

In our Application, there is a traveler part. Suppose you are a traveler, you want to travel any place so you can search into the system for knowing upcoming tour details of any specific placed. And you can see all details about the tour like tour date, cost, etc. If you want to enroll in the tour you have to ensure that you are an authentic user, by login into the system. Then you can send enroll request. In our system, the only guide can create a tour.

Traveler enrollment request comes to the Guide. In this system, guide play a vital role that's why when admin accepts the guide profile they make sure that the guide information is solid and accurate.

When an enrollment request come to guide then the guide check that request and after seeing the traveler given information, the guide can cancel or accept the enrollment request. After cancel or accept the enrollment request, a notification goes to the traveler that his enrollment request is canceled or successful.

A traveler also can book a local guide, to guide him. Suppose, any traveler needs to visit a district so he can search for a guide of that district and sending booking request.

So that's why we decided to make a system to help the traveler to travel everywhere in a placed. When any people travel any place with a local guide of that area then there is no security issue and the traveler can visit all around the placed.

1.2 Project Purpose

The ultimate purpose of this project named "Local Tour" is to make a tradition for a traveler to visit anywhere in a place properly. And build a communication system between traveler and local guide. As if, a traveler can find a tour easily and also can book a guide. The people called guide is make money using this system.

1.2.1 Background

We recognize that, people travel only a focused place but a focused placed have some other tourism sport and a traveler missing those tourism sport. People know a place but not everything on that place because the lack of availability of information. And they also don't know that there have a tourist placed. So that's why we have thought that here we can help the people.

1.2.2 Benefits & Beneficiaries

Our application is going to be beneficial in various ways. Some of them are below

- It helps a traveler to find and see upcoming tour on a specific area
- It's more easy for a traveler to enroll the tour
- Also a traveler easily can find a "Guide"
- Traveler can send guide booking request more easily
- After complete the tour, the traveler can give rating on a guide
- Traveler can update their own profile
- This app helps the guide for create and arrange a tour
- Guide can accept or reject traveler enrollment request
- Guide also can delete or cancel the tour that is created by him
- After complete the tour, guide also can giving rating on a traveler
- Guide can update their own profile also
- The guide information is trustable because an admin is completed the guide registration process

1.2.3 Goals

The final goal of this project is to build an application based on web. Our aim is to help the traveler to travel any place properly and make their traveling path easy and more secure. We want build a platform where a traveler can find his maximum needs. And the guide side we will try to make this application useful for them as they can get their return values.

1.3 Stakeholders

There are three stakeholders in our "Local Tour" System. And they are:

- ❖ Traveler
- ❖ Guide
- ❖ Admin

Traveler: Traveler is the primary stakeholders in this system. The main user in this system is a traveler. Traveler searches a tour on a specific placed and sees the tour details if they like the tour they can send an enrollment request. Also, a traveler can book a guide to giving him company on any tour. After the tour, they can give a rating on that guide.

Guide: The Guide is another primary stakeholder in this system. A guide can create a tour based on any specific area. And on that tour, if any enrollment request comes from any traveler then the guide can accept or reject that request. For any reason, the guide can delete or cancel the tour. After the tour, guide also can give a rating on a traveler.

Admin: There is a small roll in admin part of this system. When a guide registration request come to admin they check the guide information if the guide provide solid information then admin accept there registration.

1.4 Proposed System Model

Before going to develop a system it is very important to have a system model. We have already Prepared a system model. This model will clarify our proposed system in brief.
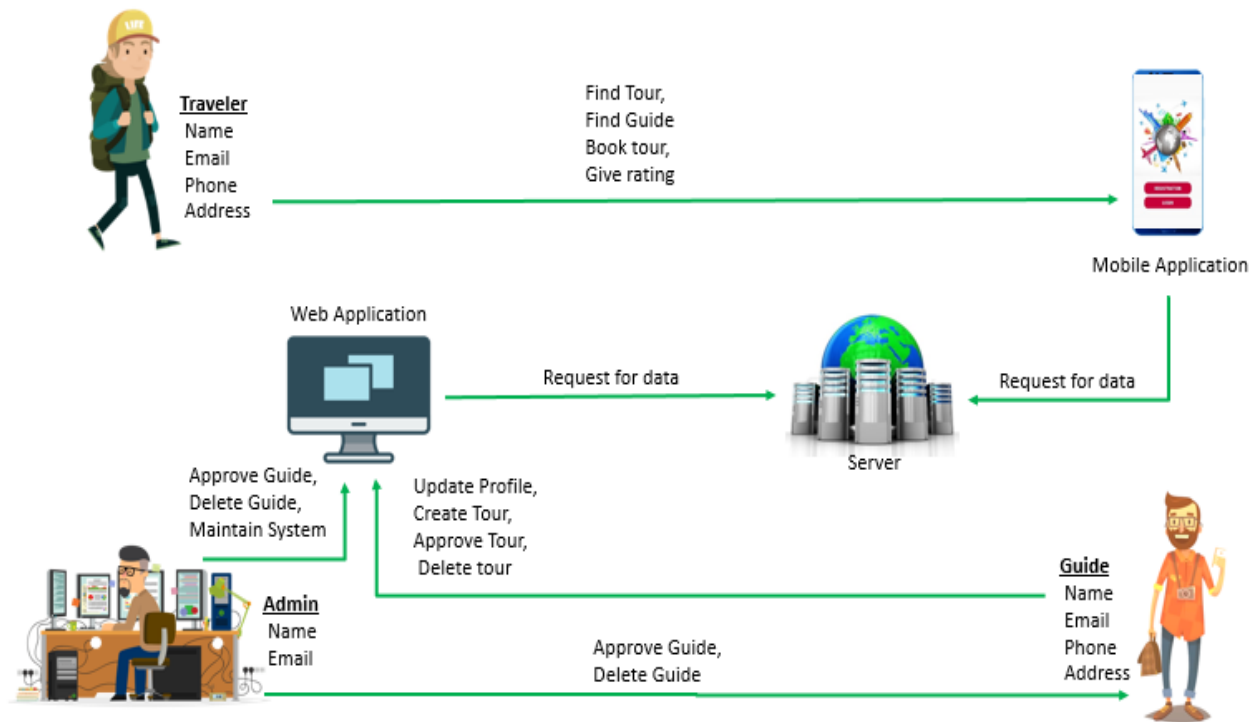
Figure 1.1: Proposed system model

Our project named "Local Tourist" is going to be developed based on web and mobile, this application reserve different types of modules as well as user privileges. And each module architecture is different from another one. Before describing project overview, I need to expose about our system module architecture

Such as:
 Traveler Module
 Guide Module
 Admin Module

Traveler: Traveler is the primary stakeholders in this system. The main user in this system is a traveler. Traveler searches a tour on a specific placed and sees the tour details if they like the tour they can send an enrollment request. Also, a traveler can book a guide to giving him company on any tour. After the tour, they can give a rating on that guide.

Guide: The Guide is another primary stakeholder in this system. A guide can create a tour based on any specific area. And on that tour, if any enrollment request comes from any traveler then the guide can accept or reject that request. For any reason, the guide can delete or cancel the tour. After the tour, guide also can give a rating on a traveler.

Admin: There is a small roll in admin part of this system. When a guide registration request come to admin they check the guide information if the guide provide solid information then admin accept there registration.

1.5 Project Schedule

We need to prepare a scheduling plan to complete the project on time. It also refers to make Communication with what task need to get done within timeframe.

1.5.1 Gantt Chart

Gantt chart is mainly a production control tools. It remained us to complete our assigned tasks Within a certain period of time. For developing software, it is mostly used. Now I will show a Gantt chart for our project.

Activities		M1 H1	M1 H2	M2 H1	M2 H2	M3 H1	M3 H2	M4 H1	M4 H2	M5 H1	M5 H2	M6 H1	M6 H2	M7 H1	M7 H2
Planning	Ideas	🟩													
	Problem definition	🟩	🟩												
	Proposal planning		🟩	🟧											
Requirements	Requirement specification			🟩	🟩										
	Requirement analysis					🟩	🟩								
QA-1	Quality assurance							🟩	🟩						
System design	Sketching							🟩	🟧						
	Design specification							🟩	🟩						
	Database design							🟩	🟥						
Implementation	Guide Module (Web)									🟩	🟩		🟩	🟧	
	Admin Module (Web)									🟩	🟩		🟩	🟧	
	Traveler Module (Mobile app)								🟩	🟩				🟧	
QA-2	Test cases								🟩	🟩					
Testing	Unit testing									🟩		🟩		🟩	
	Black box testing										🟩	🟩		🟩	
Delivery	Software release														🟩
Scheduled Time		🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
Buffered Time									🟧	🟧				🟧	
M	Month	H1				First Half				H2				Second Half	

Figure 1.2: Gantt chart

1.5.2 Release Plan or Milestone

Activities	Duration in Half Month (15 days)	Total Month
Brainstorming	Month-1 Half-1,	.5
Problem identification	Month-1 Half-1, Month-1 Half-2	1
Requirement specification	Month-2 Half-1, Month-2 Half-2	1
Requirement analysis	Month-3 Half-1, Month-3 Half-2	1
Sketching	Month-4 Half-1	.5
Design specification	Month-4 Half-1, Month-4 Half-2	1
Database design	Month-4 Half-2	.5
Guide Module (Web)	Month-4 Half-2, Month-5 Half-1, Month-6 Half-2,	1.5
Admin Module (Web)	Month-5 Half-1, Month-5 Half-2, Month-6 Half-2,	1.5
Traveler Module (Mobile app)	Month-4 Half-2, Month-5 Half-1, Month-5 Half-2, Month-6 Half-2,	2
Test cases	Month-4 Half-2, Month-5 Half-1,	1
Unit testing	Month-4 Half-2, Month-5 Half-2, Month-6 Half-1, Month-7 Half-1	2
Black box testing	Month-5 Half-1, Month-5 Half-2, Month-6 Half-1, Month-7 Half-1	2
Software release	Month-7 Half-2	.5

Chapter 2

Software Requirement Specification

2.1 Functional Requirements

Functional requirements refer to the function which is must be belong to the system. Functional requirements are mandatory to perform the software system. There is no system without functional requirements. Now, we are going to discuss functional requirements for our project.

2.1.1: Guide Registration

FR-1	Guide Registration		
Description	Guide must registration the system using unique email, password, name, address and location. In future, Guide can't change her register email address. Without registration, they can't enter the guide panel and can't use guide facilities.		
Stakeholders	Guide	**Priority**	High

2.1.2: Guide Login

FR-2	Guide Login		
Description	After successfully login, Guide login the system using email and password.		
Stakeholders	Traveler	**Priority**	High

2.1.3: Guide can update own profile

FR-3	Guide can update own profile		
Description	Guide may need to update their profile. For updating own profile, they need to login the system. They can't change the email address.		
Stakeholders	Guide	**Priority**	Medium

2.1.4: Guide must activate their account

FR-4	Guide must activate their account		
Description	After login, Guide must give additional information for security reason. When they give all information, then Admin will activate (approve) their account. If account is inactive, Guide profile, activity is invisible and guide can't create tour.		
Stakeholders	Guide	Priority	High

2.1.5: Guide can create a tour

FR-5	Guide can create a tour		
Description	After activate their account, Guide can arrange a tour. They must give tour description, location, cost details, hotel and transport information, maximum people attend, and additional offer. Guide can't take extra money without budget.		
Stakeholders	Guide	Priority	High

2.1.6: Guide can edit a tour

FR-6	Guide can create a tour		
Description	Guide may need to edit tour information. Guide can edit all information and offer. If traveler enrolls the tour, Guide can't increase budget. And can't change (decrease) hotel and transport quality.		
Stakeholders	Guide	Priority	High

2.1.7: Guide can delete a tour

FR-7	Guide can delete a tour		
Description	Guide can delete their tour if and only if traveler not enrolls the tour.		
Stakeholders	Guide	Priority	Medium

2.1.8: Guide can rate the traveler

FR-8	Guide can rate the traveler		
Description	Guide can rating the traveler if traveler hired the guide or attended the tour. Rating range must be 1 to 5. Traveler can ratings the guide only one time for every tour or every hire.		
Stakeholders	Guide	**Priority**	low

2.1.9: Accept or Deny traveler request

FR-9	Accept or Deny traveler request		
Description	When Traveler send a tour or hire request, Guide has the power to accept or deny the request. If Guide accept a request		
Stakeholders	Guide	**Priority**	High

2.1.10: Guide previous tour and hire history

FR-10	Guide can see previous tour and hire history		
Description	Guide can see their previous tour and hire history details. They can see additional information like cost, rating, traveler details and information.		
Stakeholders	Guide	**Priority**	Medium

2.1.11: Guide can view upcoming tour and guide event.

FR-11	Guide can view upcoming tour and hiring even.		
Description	When traveler booking a tour or hire a guide and guide approved it, then guide can see the upcoming tour or hiring event. He can see that info, until the date will over.		
Stakeholders	Guide	**Priority**	High

2.1.12: Admin approve guide

FR-12	Admin approve guide.		
Description	When guide update her profile with give her full additional information, then the admin approve guide account. And guide account will activate.		
Stakeholders	Admin	**Priority**	High

2.1.13: Admin can see guide profile.

FR-13	Admin can see guide profile.		
Description	Admin can see any guide profile. But they can't edit guide profile.		
Stakeholders	Admin	**Priority**	High

2.1.14: Admin can see traveler profile.

FR-14	Admin can see traveler profile.		
Description	Admin can see any traveler profile. But they can't edit traveler profile.		
Stakeholders	Admin	**Priority**	Medium

2.2 Data Requirements

To direct data gathering efforts and ensure that meetings with others, on whom you depend for model information, are productive, it may be useful to prepare a specific list of questions that identify the data needed.

- Route data locations
- Capacity and resources of the data requirements
- Data availability schedules
- Quantity of data
- Data source sequence

2.3 Performance Requirements

It is very important to maintain performance of any software system. To ensure performance, we need to maintain some steps. Now, I will explain some perspective by which we are going to enhance the performance of the project.

2.3.1 Speed and Latency Requirements

SLR-01	Search result must be faster.		
Description	When traveler search for a tour or guide, then the search result must show within seconds.		
Stakeholders	Traveler	**Priority**	High

2.3.2 Precision or Accuracy Requirements

PAR-1	Search result must be accurate		
Description	When a traveler search for a tour or guide with location, then system must be show accurate location/area based guide or tour.		
Stakeholders	Traveler	**Priority**	High

2.3.3 Capacity Requirements

The developed system by us must be capable to handle user data, provide accurate information, handling database, manage http request etc.

CR-1	The system will handle hundreds of data.		
Description	The system need to handle data hundreds of data every moment.		
Stakeholders	Traveler	**Priority**	High

2.4 Dependability Requirements

The term dependability is measured based on four dimensions. Such as:

- Availability
- Reliability
- Safety
- Security

If we want to say that our application system is dependable then it must fulfill the four dimensions. But there are other tasks. Like there is no way to make mistakes or our system should have the ability to detect and then remove errors.

2.4.1 Reliability Requirements

Now, I will mention requirements which is related to reliability and availability.

RAR-1	The system must be available on 24 X 7		
Description	- Our system must be available all day long, every day in a week - The system must be updated regularly - System must be malware free		
Stakeholders	Guide, Traveler	**Priority**	Medium

2.4.2 Availability Requirements

To ensure robustness and fault-tolerance facilities to the end users, it is urgent to ensure 0% crush. Moreover, it must show accurate results.

RFT-1	The system handles all user access without system errors		
Description	Hundreds of user may hit our application system at a time. All their requests must be handled without any fault.		
Stakeholders	N/A	**Priority**	High

2.4.3 Robustness or Fault-Tolerance Requirements

It is very important to provide after service or support to the end users.

2.5 Maintainability & Supportability Requirements

It is very important to provide after service or support to the end users.

2.5.1 Maintainability and Supportability Requirements

Supportability requirements may have related to some extends. Like:

- Testability
- Extensibility
- Adaptability
- Maintainability
- Serviceability
- Install ability

2.5.2 Adaptability Requirements

There are no Adaptability Requirements in our system

2.6 Security Requirements

Making software security as a requirement is very important. Software security requirements should be its functional requirement. Software security enforces security of an application system. Functionality related to software security can either be directly tested or observed. Some security related requirements is given below:

- Signing in a traveler and guide
- Get access according to logged in user
- Booking tour a traveler without having any issue
- Signing out as a traveler and guide

While accessing to the system, each and every module must provide a central authentication mechanism. There is also a process to prevent entering into the system by ensuring hashed password for the unauthenticated users

2.6.1 Access Requirements

For accessing to our application system, there remain some authentication and authorization techniques. And every module of our system will provide it. Now I will provide an explanation below.

FR-22	Application provides Login system.		
Description	Every module is designed in such a way that it only give access to the authorized and authenticated users.		
Stakeholders	Guide Traveler	Priority	High

2.6.2 Integrity Requirements

Integrity requirements refer to a security system which ensures an expectation of data quality. It also ensures that all data of the system would never be exposed to the malicious modification or accidental destruction. For that reason, we will store our user passwords as encrypted format which is impossible to decrypt. It is also called hashed password.

2.6.3 Privacy Requirements It

It is very important to ensure privacy of the system users. Privacy requirements enhances to protect stakeholder's privacy. In this way, all data or a partial part of data are going to be disclosed according to system's privacy policy. To ensure privacy, the central database should be protected by the anonymous. Users are permitted to get access to those data which are being associated by them

2.7 Usability and Human-Interaction Requirements

The system can be failed for Usability. User Experience is one of the most important factors to any system. The system must be easy to use, easy to understand and easy to learn.

2.7.1 Personalization and Internationalization Requirements

In our Project there are no personalization and internationalization requirements.

2.7.3 Understandability and Politeness Requirements

We made this system based on a targeted and guides. And there must have a diagram for easy to understand the process. So that, User can understands this system easily.

2.7.4 Accessibility Requirements

The requirements for how easy it should be for people and our system is very much easy to use and understand

2.7.5 User Documentation Requirements

There is no user documentation require in this system.

2.7.6 Training Requirements

There is no user documentation require in this system.

2.8 Look and Feel Requirements

If your systems look garbage then user can fell boring and not going to next step. Look and feel requirements are how the system will look like and how the user interface or graphical user interface of our system will display to the user.

2.8.1 Appearance Requirements

We make it easy understandable so that travelers and guide easy to understand it . Such as, if there any field Time duration (start or end time) field then the word "Date Of Tour (Start Date)" appeared into the field.

AR1	In optional field the word "Date Of Tour (Start Date)" must appear into the field		
Description	If there is no optional word appeared into the field that means this field is mandatory for input.		
Stakeholders	Traveler, Guide	Priority	High

2.8.2 Style Requirements

SR1	All content must be appear within a format		
Description	Input field and other view result show an specific format		
Stakeholders	Traveler, Guide	Priority	Medium

2.9 Operational and Environmental Requirements

There is no Operational and Environmental Requirements in this system

2.10 Legal Requirements

There is no Legal Requirements in this system

Chapter 3

Requirement Analysis

3.1 Use Case Diagram

We have use case diagram. And there are five actors. Each actor plays different role. And those Are already indicated to this use case diagram. This diagram will clarify our system in brief.
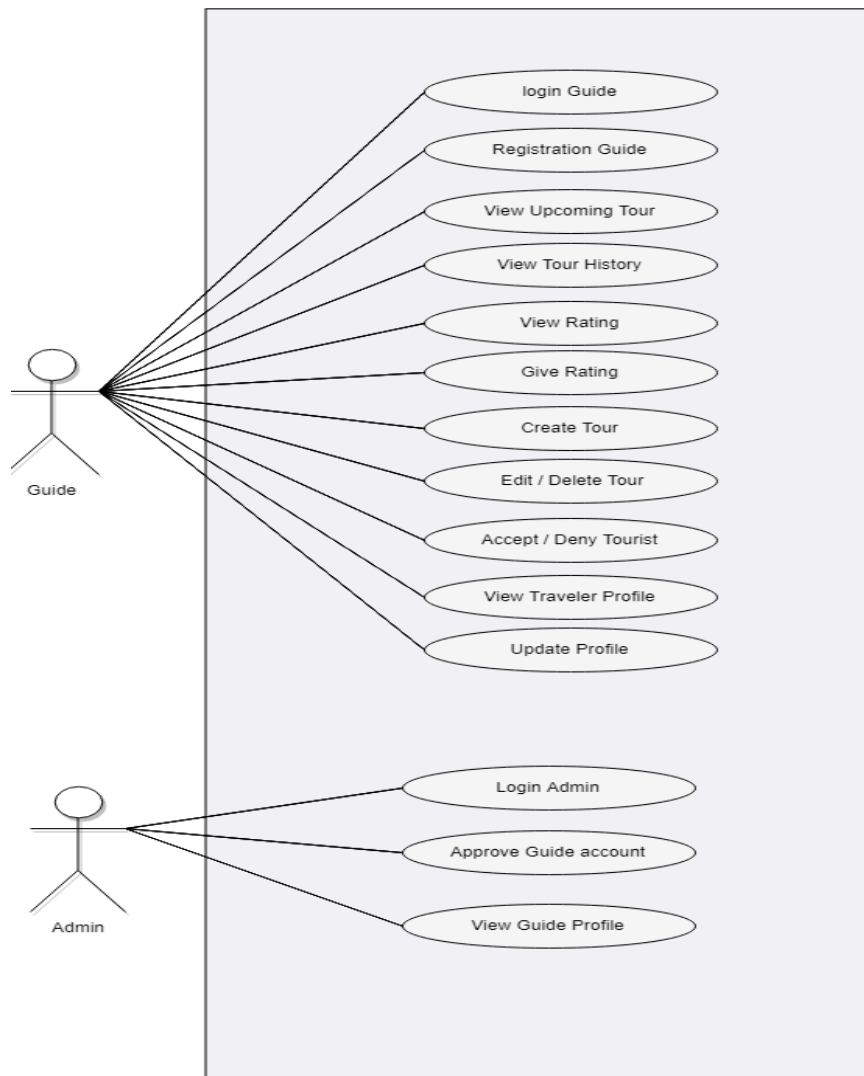
Figure 3.1: Use Case diagram for "Local Tourist"

3.1.1 Registration

Use Case Title	Registration	
Goal	Guide must be registration the system.	
Preconditions	• Registration data have to be genuine	
Success End Condition	Registration done by a traveler or guide	
Failure End Condition	Registration can't done by a traveler or guide	
Primary Actors: Secondary Actors:	Guide System	
Trigger	After click registration button, field will open	
Description **Main Success** **Scenario**	**Step**	**Action**
	1	Registration data have to be solid
	2	Traveler and guide both have to be registered
	3	Illegal data can't accept for registration
Alternative Flows	**Step**	**Branching Action**
	1a	Guide can't registered
	2a	Illegal data can accept for registration
Quality Requirements	**Step**	**Requirements**
		N/A

3.1.2 Login

Use Case Title	Login	
Goal	Guide must be Login the system.	
Preconditions	• Without registration guide can't be login	
Success End Condition	Login done by a guide	
Failure End Condition	Login can't done by a guide	
Primary Actors: Secondary Actors:	Traveler Guide	
Trigger	After click Login button, system will open	
Description Main Success Scenario	**Step**	**Action**
	1	After registration second step will login
	2	After registration guide both will login
	3	Without registration guide can't login the system
Alternative Flows	**Step**	**Branching Action**
	1a	Guide can't Login
	2a	After registration guide can't will login
Quality Requirements	**Step**	**Requirements**
		N/A

3.1.3 View Upcoming Tour

Use Case Title	View Upcoming Tour	
Goal	Guide can see upcoming tour	
Preconditions	• Guide must login the system	
Success End Condition	After login, guide can see upcoming tour.	
Failure End Condition	Guide can't see upcoming tour.	
Primary Actors: Secondary Actors:	Guide Traveler	
Trigger	After click, firstly Guide can see upcoming tour.	
Description Main Success Scenario	**Step**	**Action**
	1	Guide can login the system
	2	Guide can view upcoming tour
	3	Guide can book the tour
Alternative Flows	**Step**	**Branching Action**
	1a	Guide can't enter the system
	2a	Guide can't view a tour
	3a	Guide create tour
Quality Requirements	**Step**	**Requirements**
		N/A

3.1.4 View Tour History

Use Case Title	View Tour History	
Goal	Guide can see their tour history	
Preconditions	• Guide must login the system	
Success End Condition	Before tour complete guide can see the tour history.	
Failure End Condition	Guide can't see their tour history.	
Primary Actors: Secondary Actors:	Guide	
Trigger	After press tour history button, guide can see the tour history	
Description Main Success Scenario	**Step**	**Action**
	1	Guide can login the system
	2	Guide can see guide tour history.
Alternative Flows	**Step**	**Branching Action**
	1a	Guide can't enter the system
	2a	Guide can't view their tour history.
Quality Requirements	**Step**	**Requirements**
		N/A

3.1.5 Give Rating

Use Case Title	Give Rating	
Goal	Guide can give rating for traveler tour.	
Preconditions	• Guide must login the system.	
Success End Condition	After tour guide can rate the traveler	
Failure End Condition	Guide can't rate the traveler.	
Primary Actors: Secondary Actors:	Guide Traveler	
Trigger	After finish tour and click tour history details, traveler can give rating.	
Description Main Success Scenario	**Step**	**Action**
	1	Traveler login the system
	2	Traveler book a tour
	3	Traveler give rating.
Alternative Flows	**Step**	**Branching Action**
	1a	Traveler can't rating.
	2a	Traveler can view guide previous rating.
Quality Requirements	**Step**	**Requirements**
		N/A

3.1.6 Update Profile

Use Case Title	Update Profile	
Goal	Guide have to be update their profile	
Preconditions	• Guide must login the system	
Success End Condition	After login guide can update profile.	
Failure End Condition	Guide can't update profile.	
Primary Actors: Secondary Actors:	Guide	
Trigger	After click update profile, it shows guide profile details.	
Description **Main Success** **Scenario**	**Step**	**Action**
	1	Guide can request for access
	2	Guide can update profile
	3	Guide can create tour
Alternative Flows	**Step**	**Branching Action**
	1a	Guide can't update profile
	2a	Guide book by a traveler
Quality Requirements	**Step**	**Requirements**
		N/A

3.1.7 Create Tour

Use Case Title	Create Tour	
Goal	Guide can create tour	
Preconditions	• Guide must login the system	
Success End Condition	After login guide can create tour.	
Failure End Condition	Guide can't create tour.	
Primary Actors: Secondary Actors:	Guide	
Trigger	After insert data, guide can be create a tour.	
Description Main Success Scenario	**Step**	**Action**
	1	Guide can request for access
	2	Guide can update profile
	3	Guide can create tour
Alternative Flows	**Step**	**Branching Action**
	1a	Guide can't create tour
	2a	Tour book by a traveler
Quality Requirements	**Step**	**Requirements**
		N/A

3.1.8 Edit Tour

Use Case Title	Edit Tour	
Goal	Guide can edit tour	
Preconditions	• Guide must login the system	
Success End Condition	After create tour, guide can edit tour.	
Failure End Condition	Guide can't edit tour.	
Primary Actors: Secondary Actors:	Guide N/A	
Trigger	After insert data, guide can edit tour.	
Description Main Success Scenario	Step	Action
	1	Guide can update profile
	2	Guide create tour
	3	Guide edit tour
Alternative Flows	Step	Branching Action
	1a	Guide can't edit tour
	2a	Tour book by a traveler
Quality Requirements	Step	Requirements
		N/A

3.1.9 Delete Tour

Use Case Title	Delete Tour	
Goal	Guide can delete tour	
Preconditions	• Guide must login the system	
Success End Condition	After create tour, guide can delete tour. And If no one enroll this system.	
Failure End Condition	Guide can't delete tour.	
Primary Actors: Secondary Actors:	Guide	
Trigger	After create tour, guide can delete tour.	
Description Main Success Scenario	**Step**	**Action**
	1	Guide can update profile
	2	Guide create tour
	3	Guide delete tour
Alternative Flows	**Step**	**Branching Action**
	1a	Guide can't delete tour
	2a	Guide can edit the tour
Quality Requirements	**Step**	**Requirements**
		N/A

3.1.10 Approve Guide

Use Case Title	Approve Guide	
Goal	Admin Approve Guide	
Preconditions	• Admin must login the system	
Success End Condition	After request guide, admin can approve guide.	
Failure End Condition	Admin can't approve guide.	
Primary Actors: Secondary Actors:	Admin Guide	
Trigger	After guide login, admin can approve guide.	
Description Main Success Scenario	**Step**	**Action**
	1	Admin login the system
	2	Admin approve the guide
	3	Guide create tour
Alternative Flows	**Step**	**Branching Action**
	1a	Admin can't approve guide
	2a	Guide can't create a tour
Quality Requirements	**Step**	**Requirements**
		N/A

3.1.11 View Guide Profile

Use Case Title	View Guide Profile	
Goal	Admin can view guide details	
Preconditions	• Admin must login the system	
Success End Condition	After successfully login admin view the guide details.	
Failure End Condition	Admin can't view guide details.	
Primary Actors: Secondary Actors:	Admin Guide	
Trigger	After click guide details button, it shows guide details.	
Description Main Success Scenario	**Step**	**Action**
	1	Admin login the system
	2	Admin can view guide details
Alternative Flows	**Step**	**Branching Action**
	1a	Admin can't see guide profile.
Quality Requirements	**Step**	**Requirements**
		N/A

3.2 Activity Diagram

We have prepared some activity diagram according to our use case. These activity diagrams Are properly referring the flow of the individual conditions of our project.

3.2.1 Login Guide
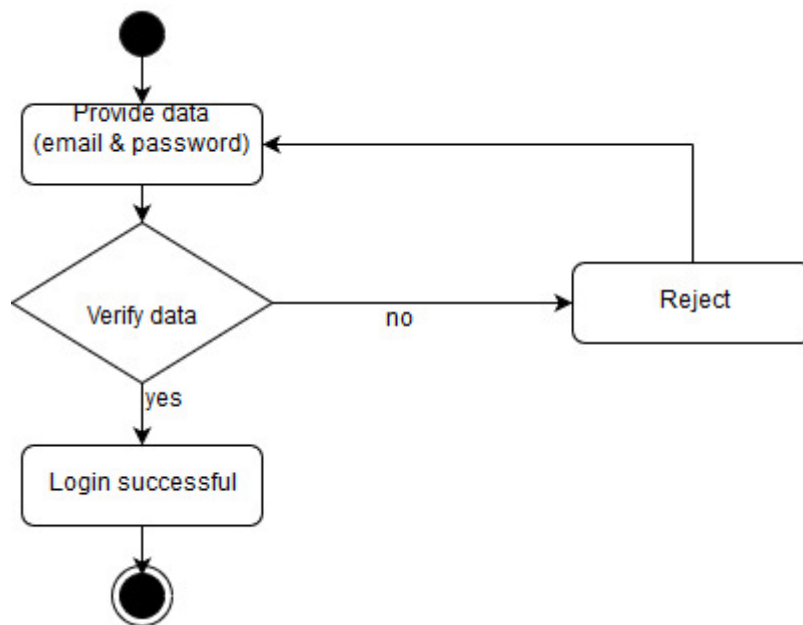
Figure 3.2.1: Activity diagram for "login Guide"

3.2.2 View Upcoming Tour
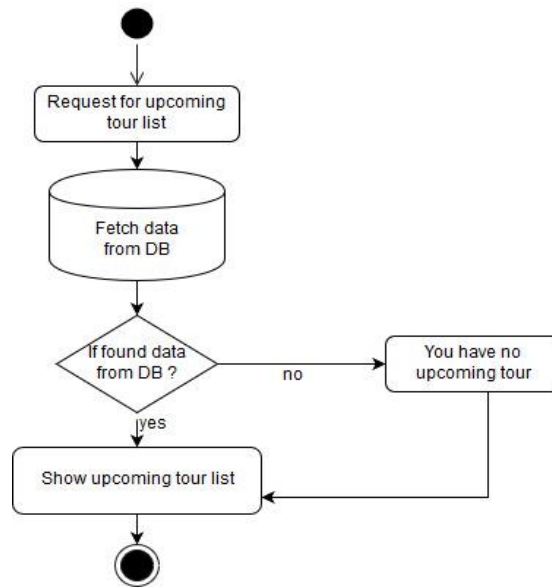
Figure 3.2.2: Activity diagram for "View Upcoming Tour"

3.2.3 View previous tour history
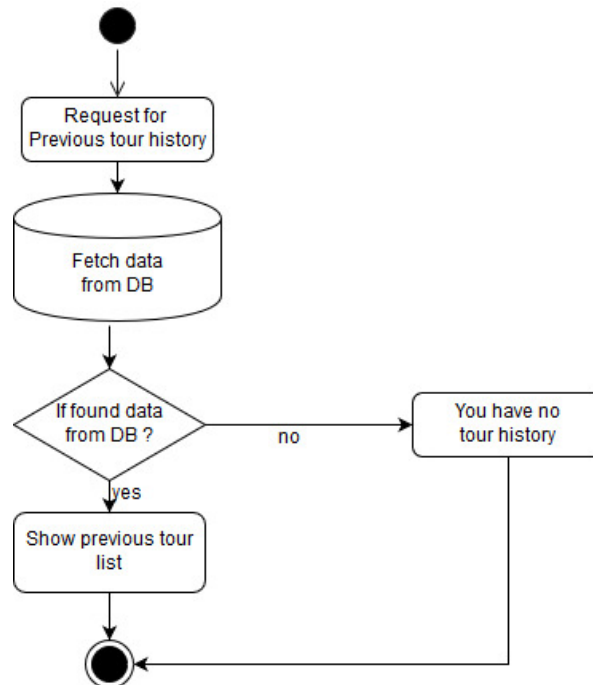
Figure 3.2.3: Activity diagram for "View Previous Tour History"

3.2.4 Update Profile
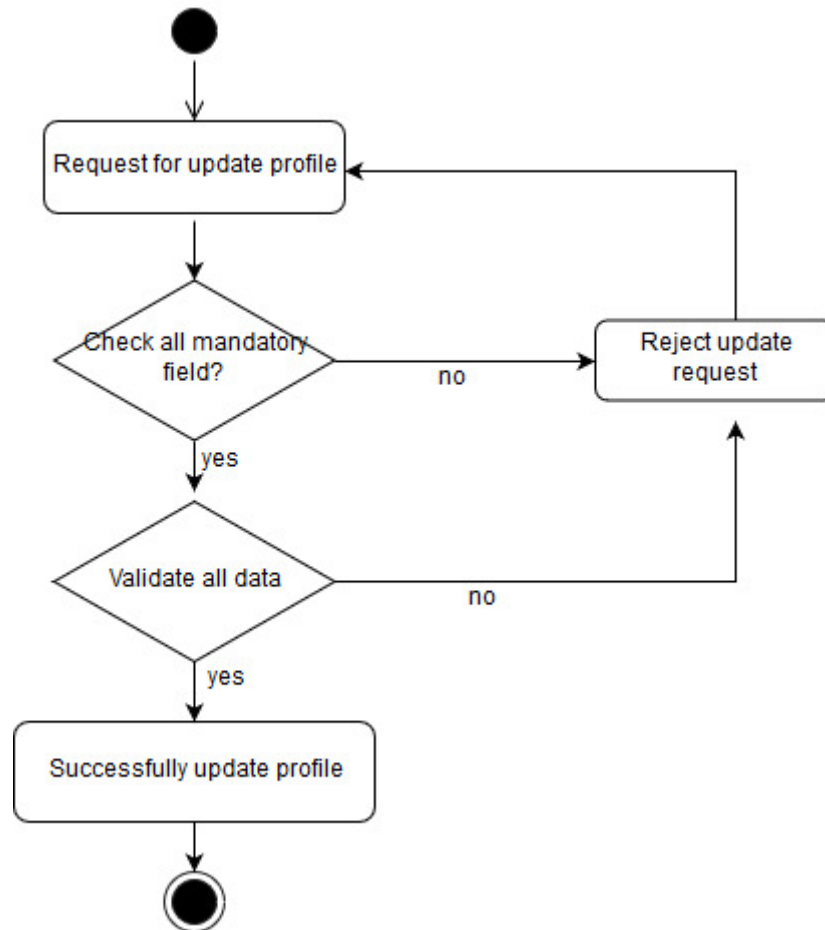
Figure 3.2.1: Activity diagram for "Update Profile"

3.2.8 Create tour
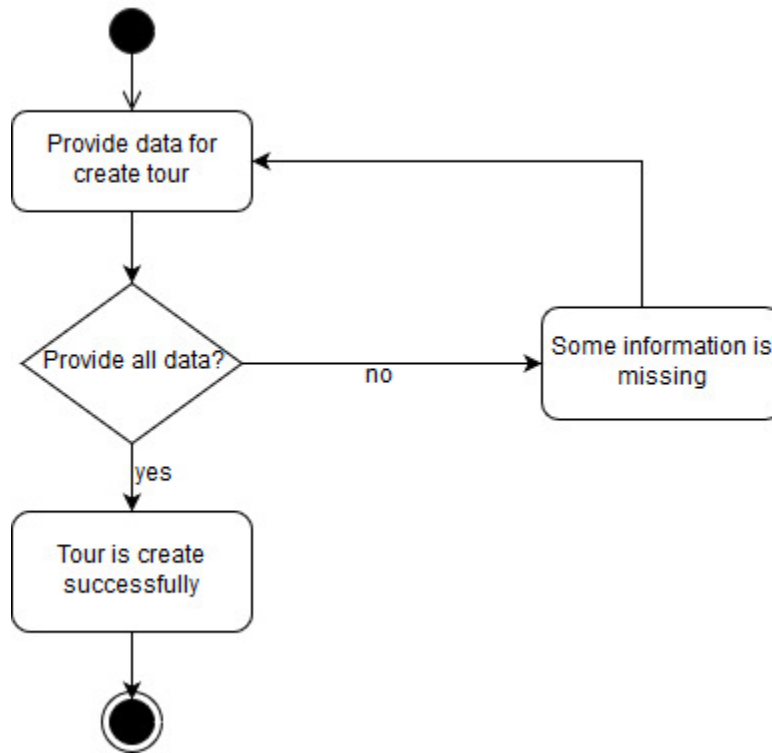
Figure 3.2.1: Activity diagram for "Create Tour"

3.2.9 View Guide details
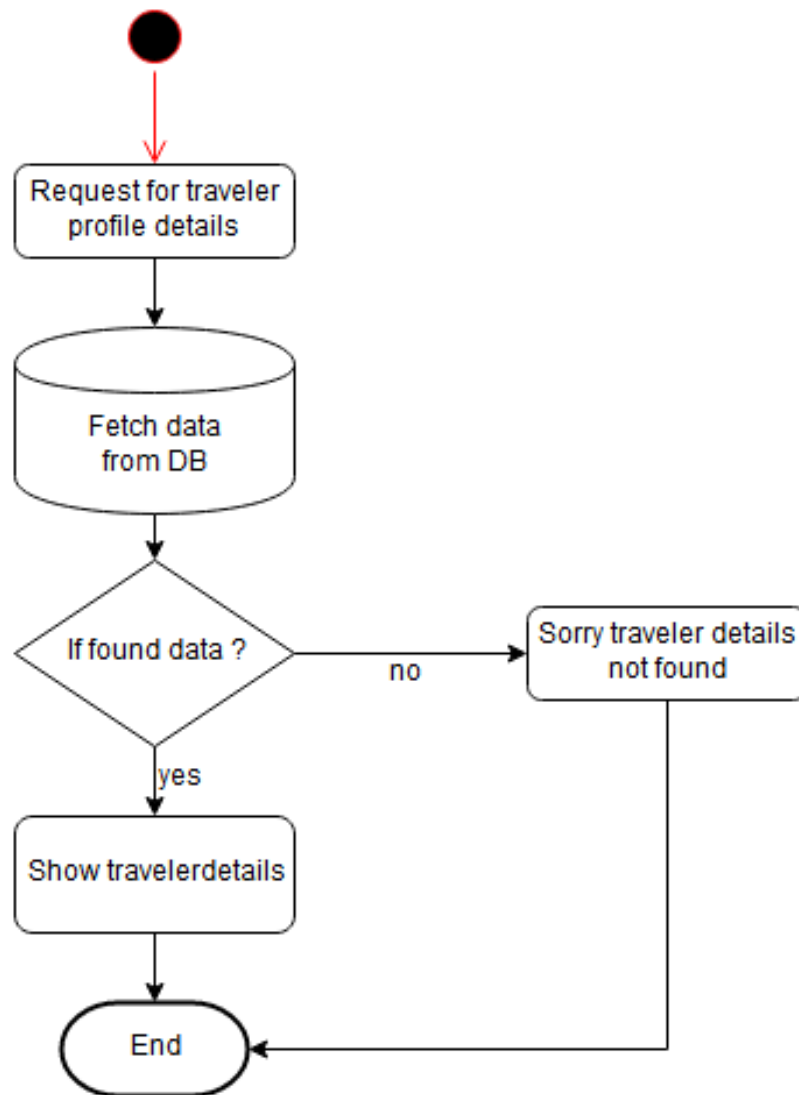
Figure 3.2.1: Activity diagram for "View Guide Details"

3.2.10 Update profile
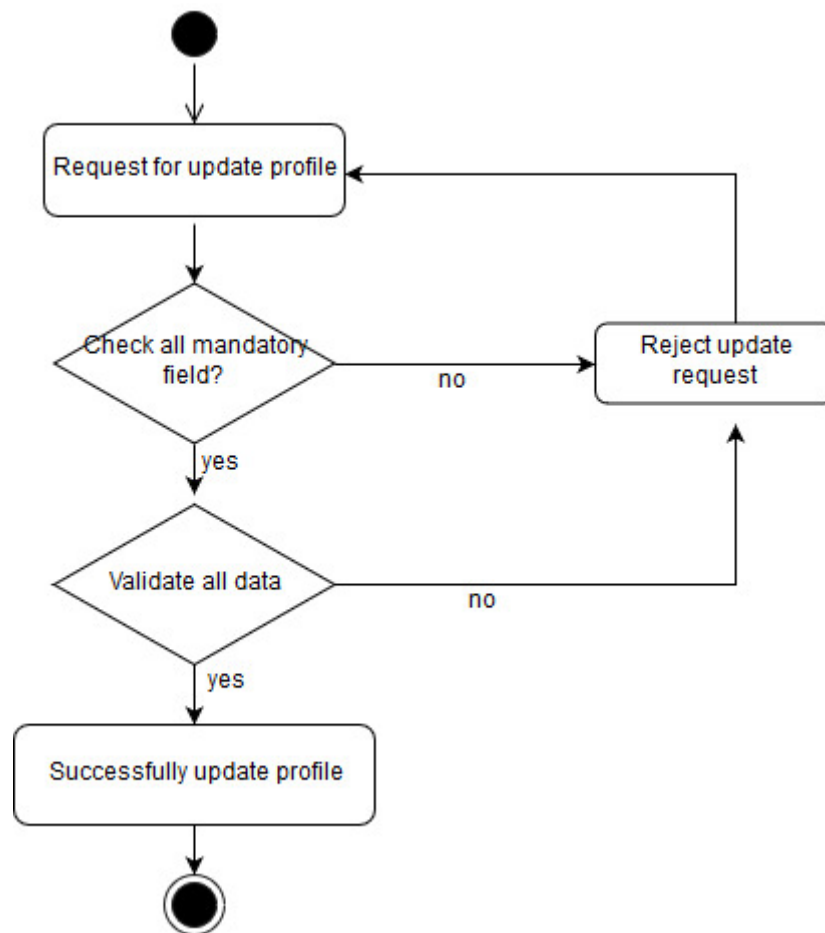
Figure 3.2.1: Activity diagram for "Update Profile"

3.2.11 View tour Details
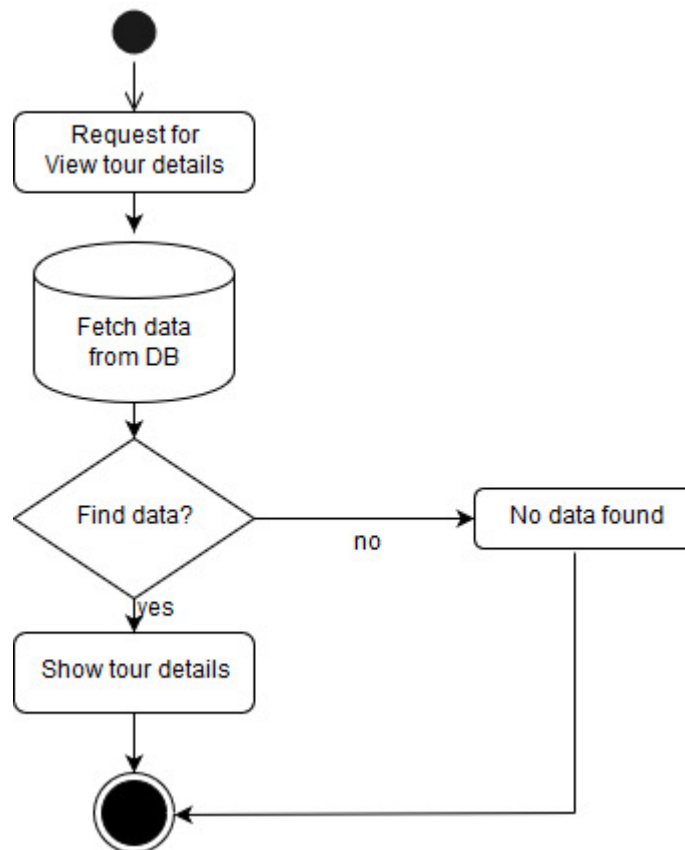
Figure 3.2.1: Activity diagram for "view tour details"

3.3 Sequence Diagrams

Mainly sequence diagrams understand us how the data will be followed in any application. Now we are going to show some sequence diagrams.

3.3.1 Traveler Login

Use must be login to the system for guide booking or searching tour.
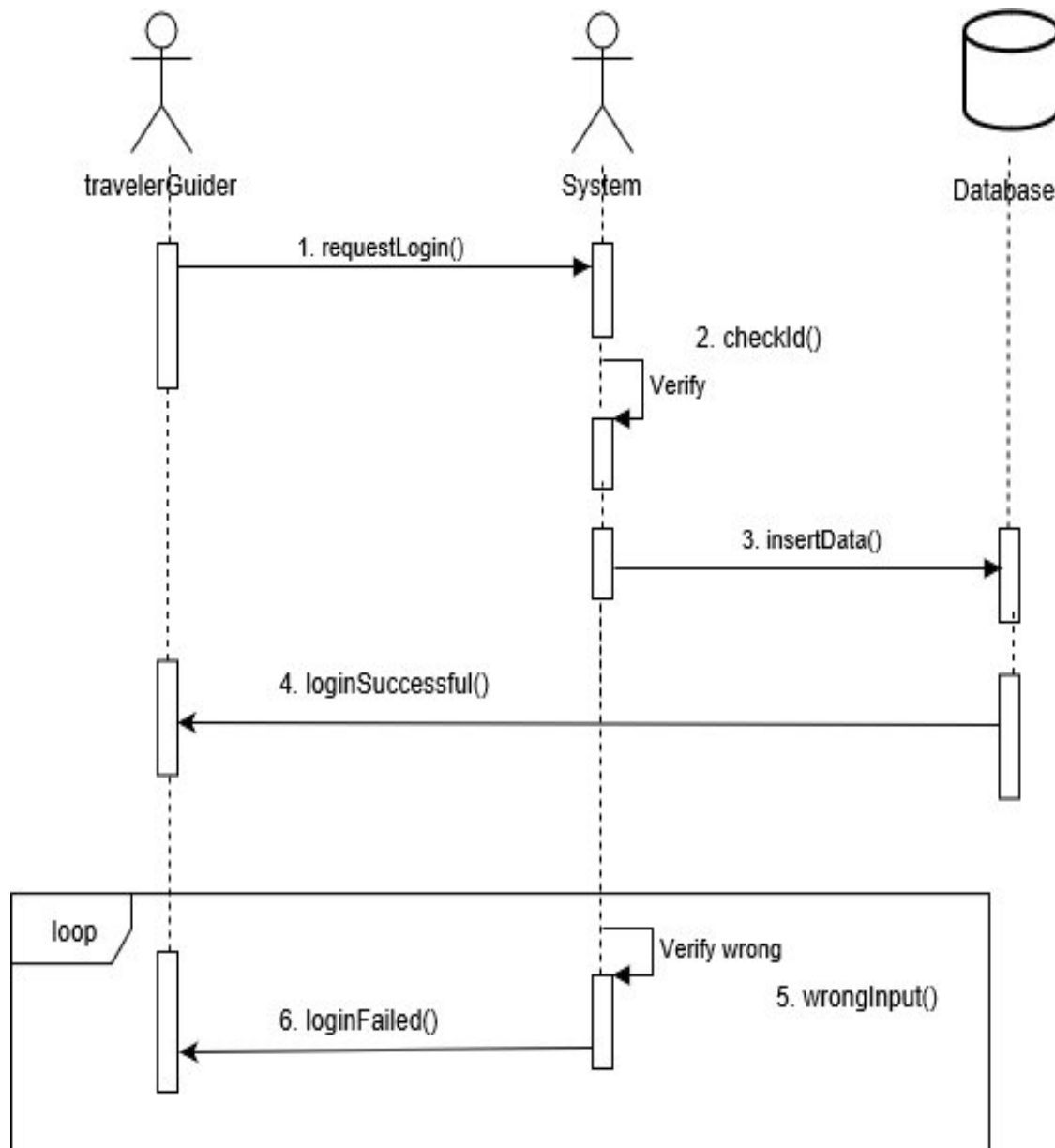
Figure 3.3.1: Sequence diagram for "Traveler login"

3.3.2 View Tour Details

After find tour, traveler can be see the tour details.
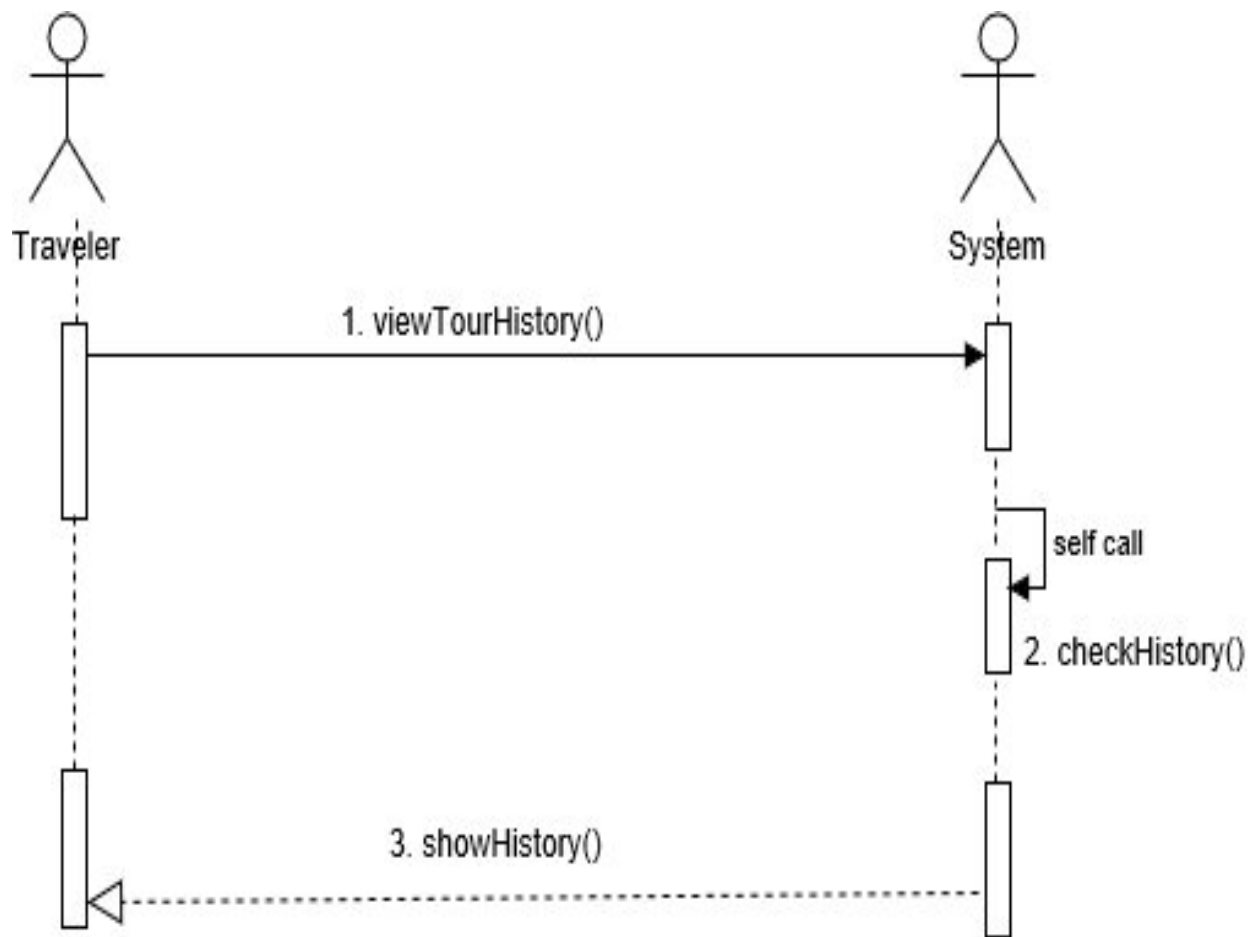
Figure 3.3.2: Sequence diagram for "View Tour Details"

3.3.3 View Upcoming Tour

After login in the system, traveler can view upcoming tour and the tour create by admin.
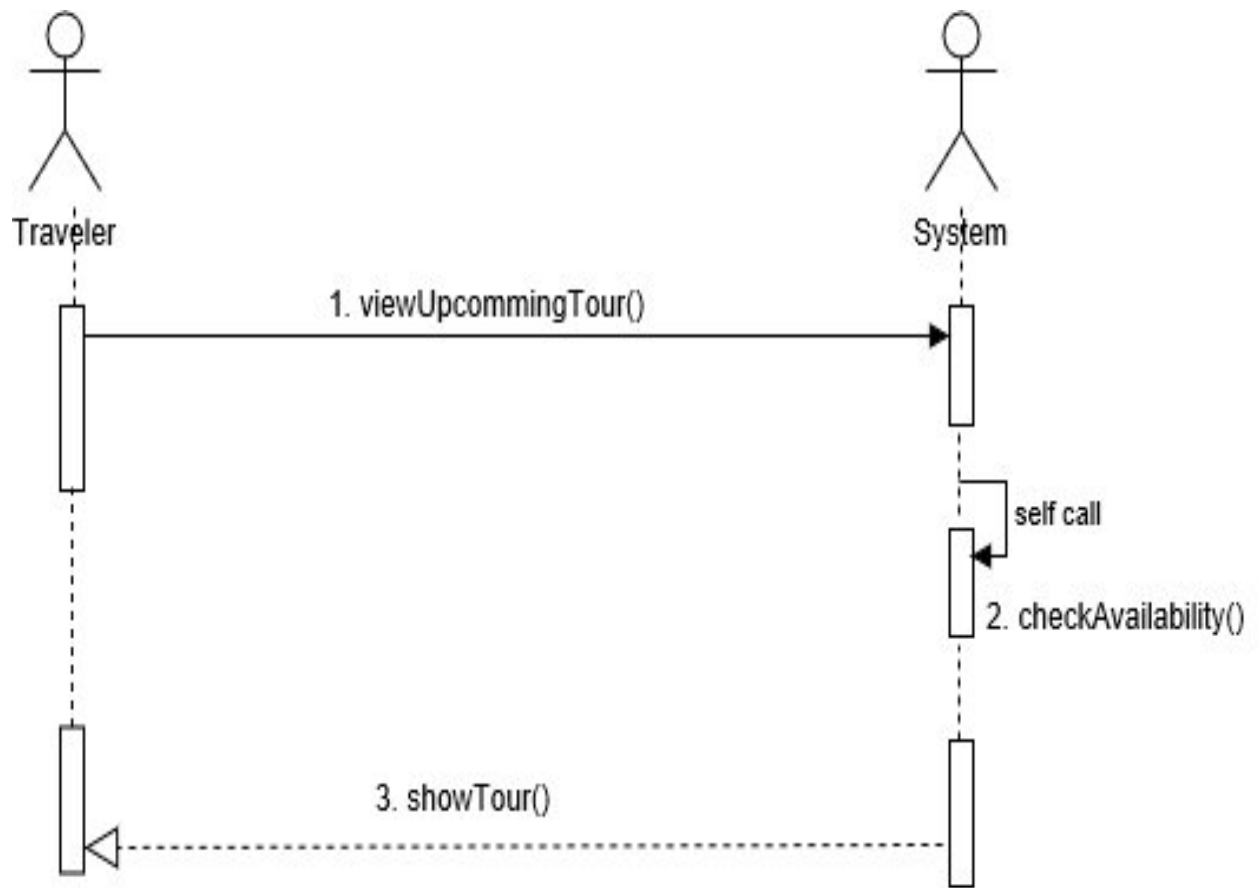
Figure 3.3.3: Sequence diagram for "View Upcoming Tour"

3.3.4 View Previous History

Traveler can view their tour history, which they have already done.
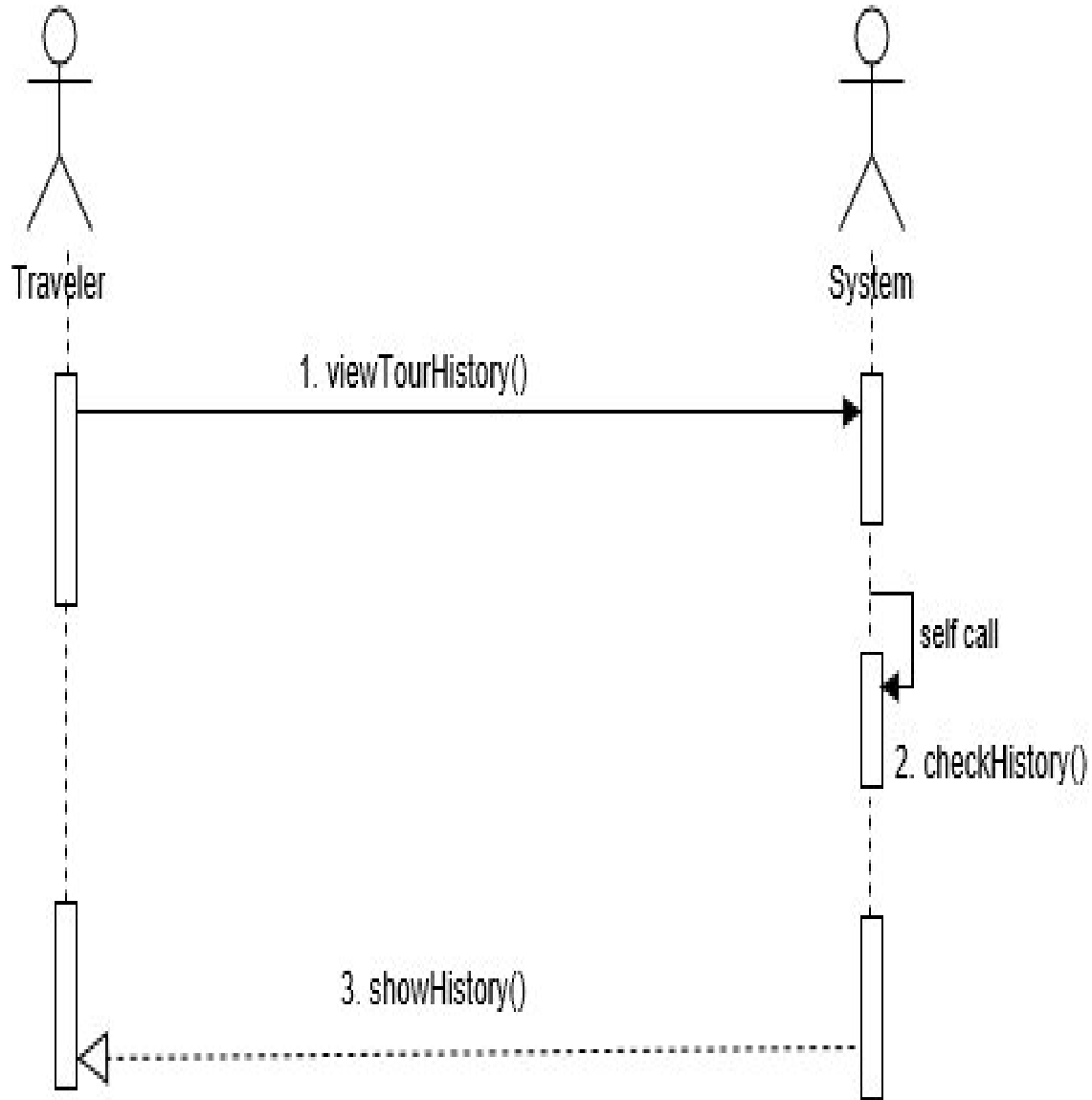
Figure 3.3.4: Sequence diagram for "View Previous History"

3.3.5 Give Rating

After finish a tour, traveler can rating the guide for guiding ability.
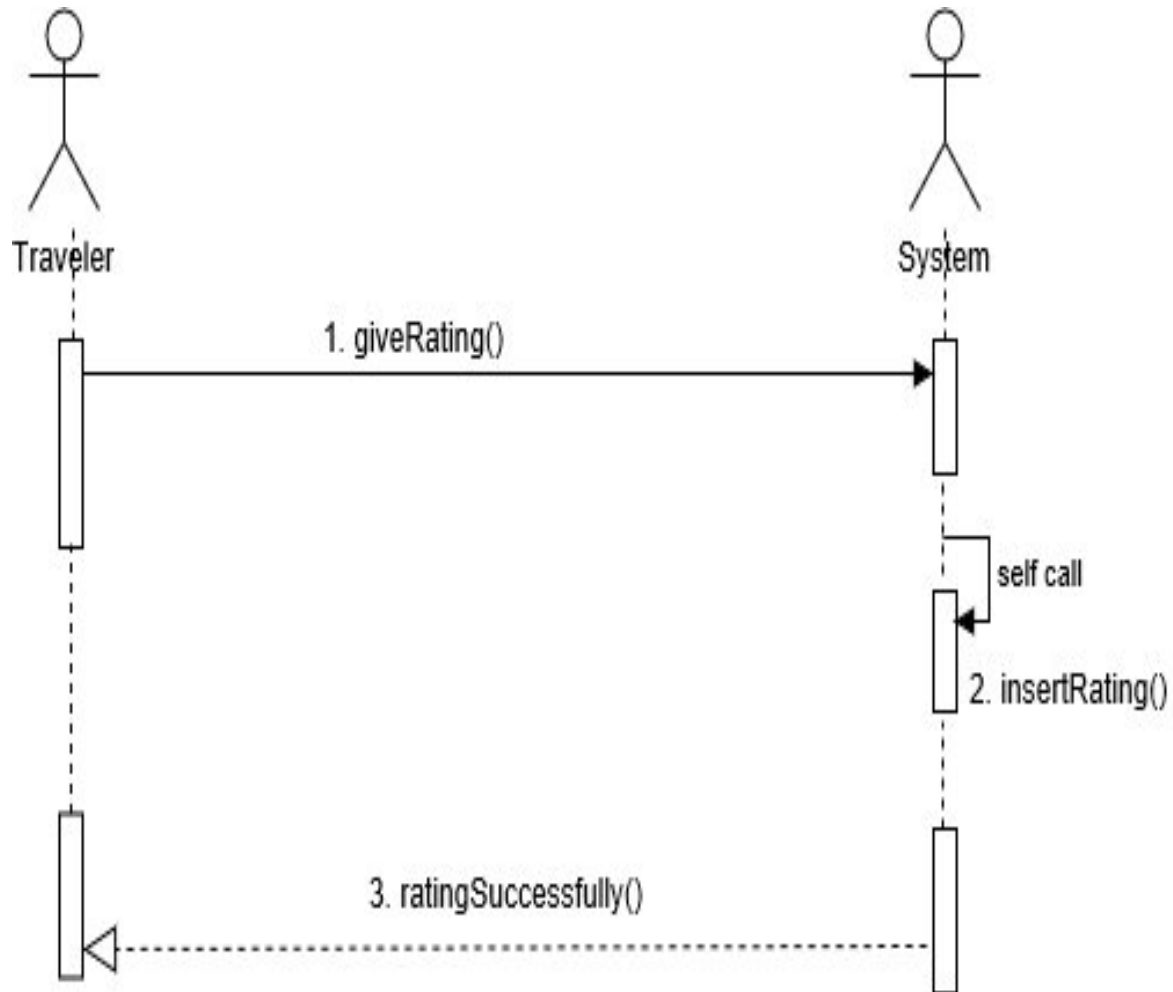
Figure 3.3.5: Sequence diagram for "Give Rating"

3.3.6 View Guide Details

After hiring a guide, traveler can view guide details.
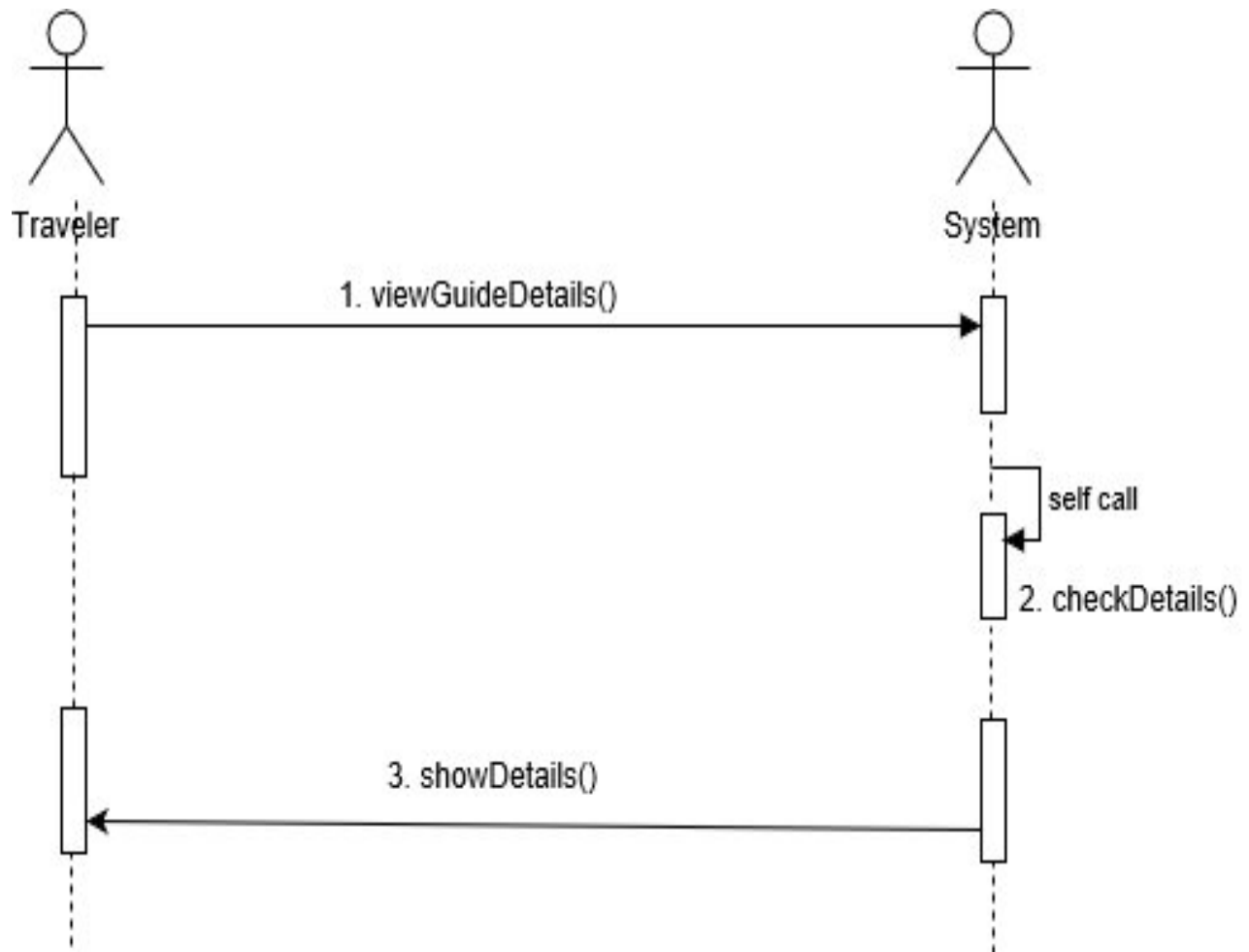
Figure 3.3.6: Sequence diagram for "View Guide Details"

3.3.7 Update Profile

Traveler or Guide can update their profile.
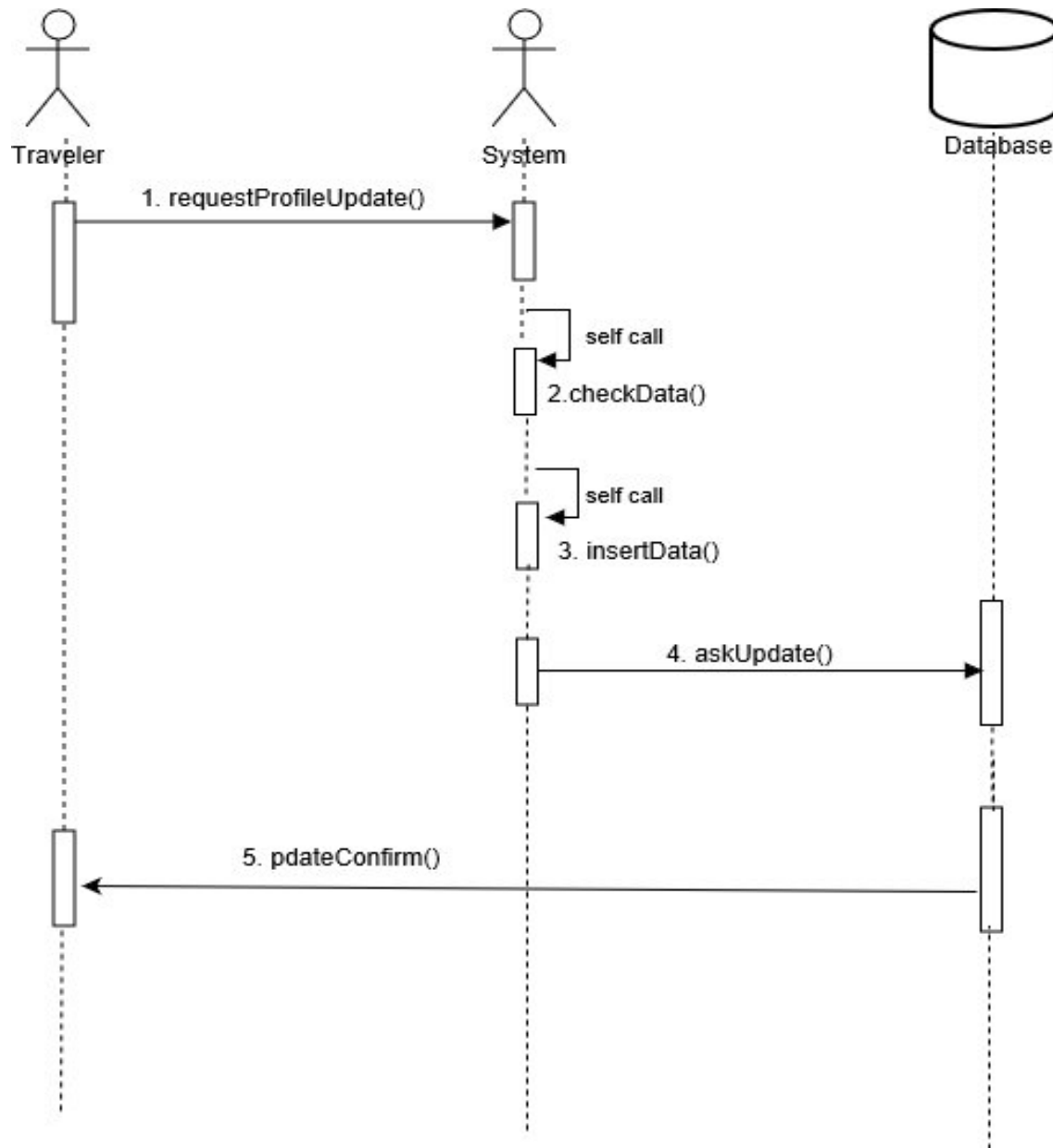
Figure 3.3.7: Sequence diagram for "Update profile"

3.3.8 Create Tour

After successful registration and login, guide can be create tour.
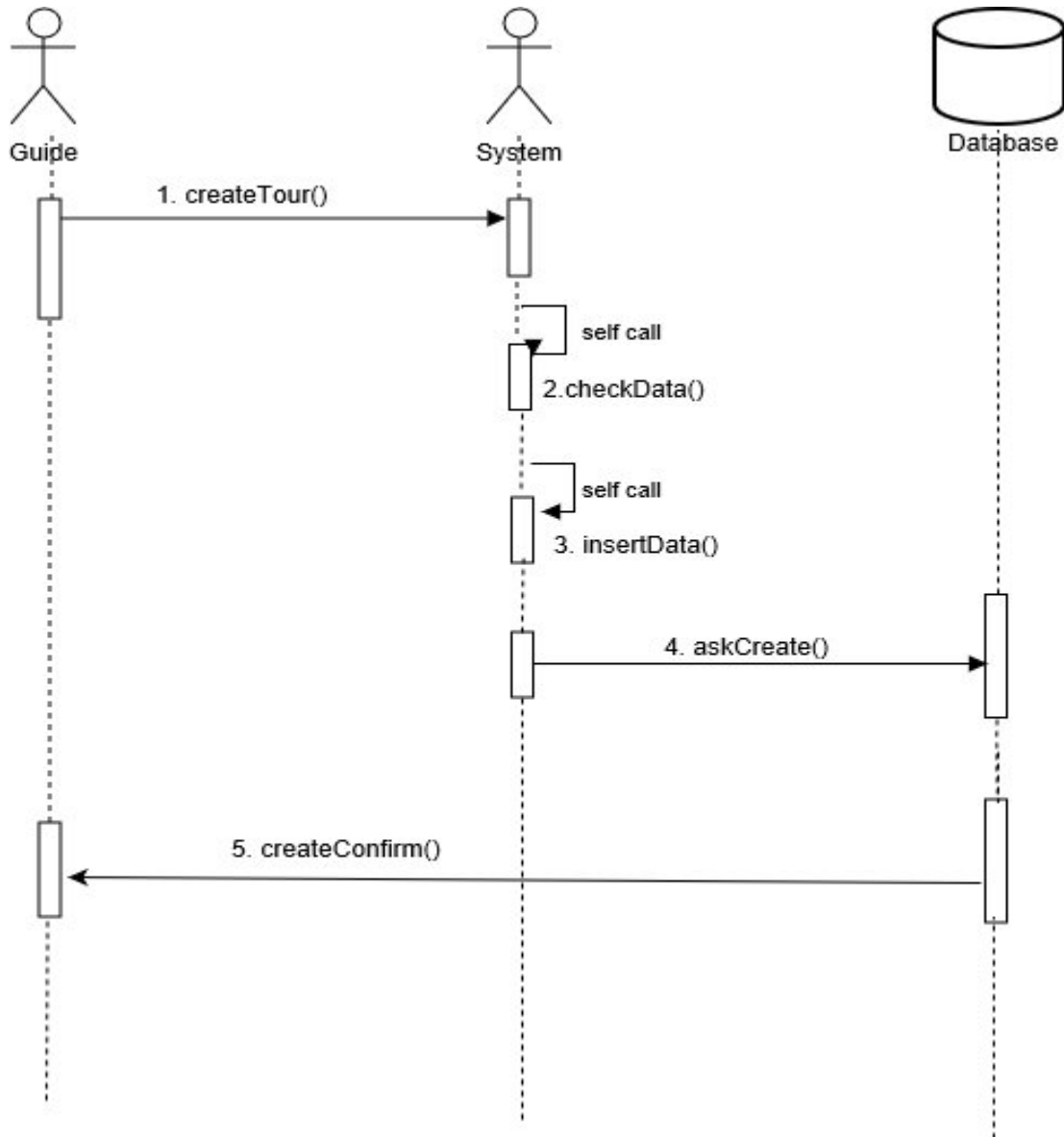
Figure 3.3.8: Sequence diagram for "Create Tour"

3.3.9 Edit or Delete Tour

After create tour, guide can be edit or delete the tour.
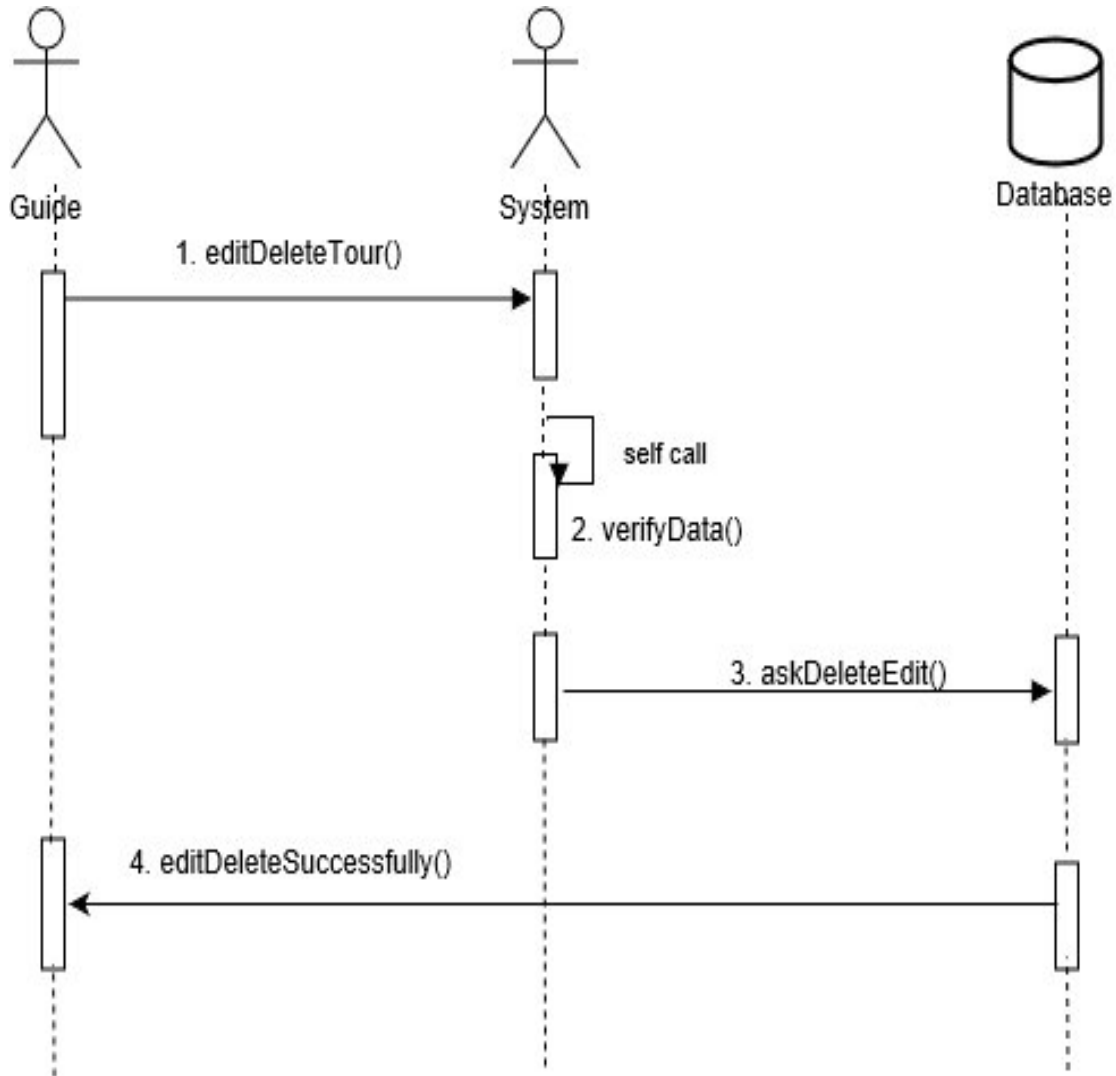
Figure 3.3.8: Sequence diagram for "Edit and Delete tour"

3.3.10 Approve Guide.

After guide registration, admin can approve or reject the guide registration request.
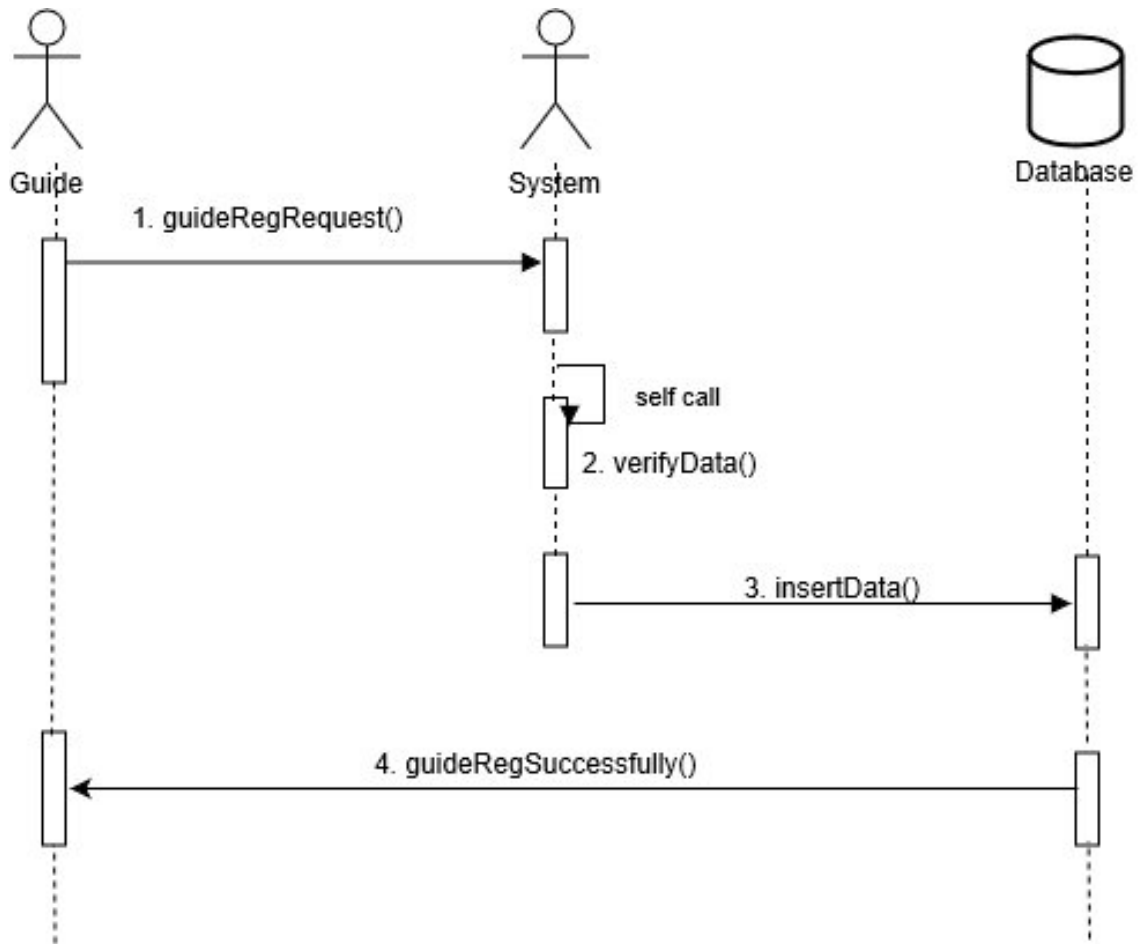
Figure 3.3.10: Sequence diagram for "Approve Guide"

Chapter 4
System Design Specification

4.1 Development tools and technology

Development tools can be programming tools, software development tool, debugging tool, etc. Software development tools are developed by experts and it's used to develop software. There are too many different types of development tools and technology can found to make the process of software development very easy and smooth. We are going to use some tools and technology to build our system and that is below:

4.1.1 User Interface Technology

In software development, user interface refer to visual part of the software. User interface is designed into an information device and a person may interact with it. The ultimate goal of user interface is engaged the user into the system. A better user interface makes an application   easy, efficient, and enjoyable. So it's important to makes the UI better for better user engaged.

4.1.1.1 CSS

CSS is stands for **"**Cascading Style Sheets". CSS is a style sheet language that is described, how HTML elements will be displayed. You may thin CSS with a human body, where human bone is HTML and when you add HTML and CSS together then it's become actual human body. Basically there are three types of CSS developer used that are Inline CSS, Internal CSS, and External CSS. Using external CSS is good practice for a developer. Sometimes in a single portion inline CSS are used.

4.1.1.2 Bootstrap

Bootstrap is a free and open-source CSS framework. For responsive web site and web application design, it's a superb framework. It contains CSS and JavaScript-based design template. Bootstrap is browser friendly framework and it's supports almost every well-known browser. The responsive layout of devices with a variety of screen size can be getting into the bootstrap.

4.1.1.3 Programming Language

Programming language is an essential part of building any application. A programming language is a formal language. It's would be better to choose the programming language which is appropriate with your application. Choosing the right language can make the processing part smooth and give better performance. In our project, there are two part one is web another is mobile based. And for the web part, we use PHP for the back end server side. PHP stands for "Hypertext Preprocessor". PHP is a general-purpose programming language originally designed for development. For the mobile part, we used Java in the back end side. Java is also a general-purpose programming language that is class-based. Both of them are open source language.

4.1.2 Implemented tools and platform

Beside the choosing programming language there is also need to select some tools for developing software. You have to analyzing which tools and platform is appropriate with your system because there are so many tools and technology into the planate. So finding the best tools that are good fit with your system is a task.

4.1.2.1 Code Editor

A code editor or any IDE is must for developed any application system. A source-code editor is a text editor program that is designed specifically for editing source code. Editor is a fundamental programming tool. For our project we are going to use the code editor that is called "Sublime Text". Sublime text is a cross-platform source code editor and it can be used for free. It has some grate feature such as multi-select editing, Auto completion, Syntax highlight, etc.

4.1.2.2 Web Server

A Web server is a program that uses Hypertext Transfer Protocol to serve the files that form Web pages to users. For our project we are used apache HTTP Server. It's free and open-source cross-platform web server software. Apache is developed and maintained by an open community of developers

4.1.2.3 Database Server

Database server is refer to the back-end system of a database application using client/server architecture. We are decided to using Relational Database Management System. There are many Relation Database but we are choosing the MySQL Relation database. MySQL have some great feature. MySQL is free and open-source software. MySQL is used by many database-driven web applications.
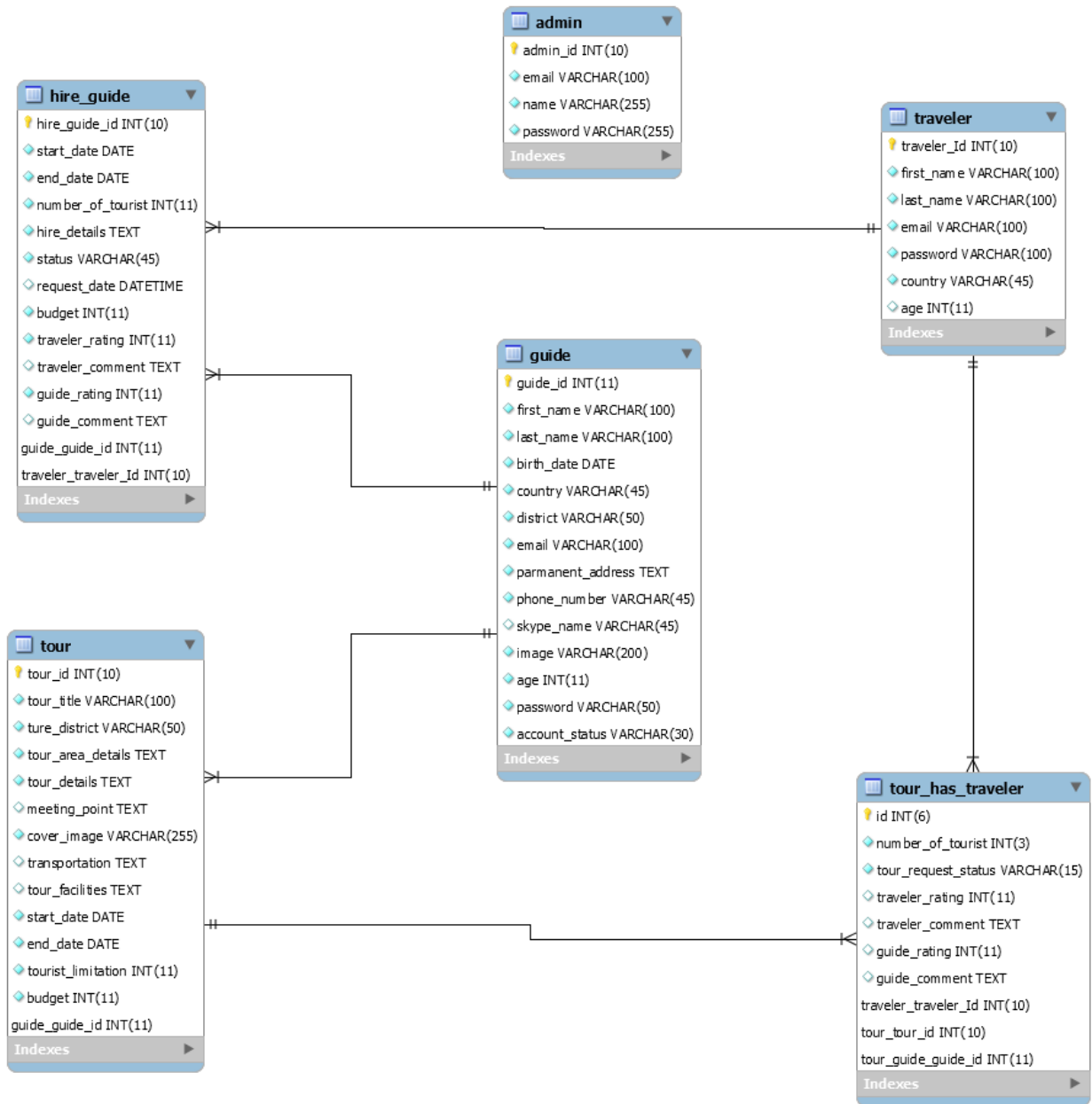
4.2 Class Diagram

4.3 Database Design Diagram

Figure 4.2: Database Diagram

Chapter 5

System Test

5.1 Testing Features

Feature testing can be considered as making change to add or modify the new functionality to The existing project. To test the features and functionality, a new test set is to be written for Testing purpose. Almost every feature and functionality have different characteristics. Those Are designed to make the application more useful, intuitive, reliable, secured, scalable, effective And efficient.

5.1.1 Features to be tested

Features	Priority	Description
Login		User must be authenticated by login
Logout		Session must be destroyed after logout
Registration		
Create tour		
Update profile		
Hiring guide		
Enroll Request		
Find Tour		
Find Guide		
Approve Guide profile		

Here, 1 = Low Priority, 2 = Medium Priority, 3 = High Priority

5.2 Testing Strategy

Test strategy is a set of guidelines that explains test design and determines how testing needs to be done. Components of Test strategy includes- objectives and scope, documentation formats, test processes, team reporting structure, client communication strategy, a test strategy is carried out by the project manager. It says what type of technique to follow and which module to test. Test strategy narrates about the general approaches, Test strategy cannot be changed, and it is a long-term plan of action. In smaller project, test strategy is often found as a section of a test plan.

5.2.1 Test approach

To complete the test process, testers must take some approach. There are mainly two test Approach.

Automation testing: Manual testing is testing of the software where tests are executed manually by a QA Analysts. It is performed to discover bugs in software under development. In Manual testing, the tester checks all the essential features of the given application or software. In this process, the software testers execute the test cases and generate the test reports without the help of any automation software testing tools.

Manual testing: In Automated Software Testing, testers write code/test scripts to automate test execution. Testers use appropriate automation tools to develop the test scripts and validate the software. The goal is to complete test execution in a less amount of time. Automated testing entirely relies on the pre-scripted test which runs automatically to compare actual result with the expected results. This helps the tester to determine whether or not an application performs as expected.

5.2.1.1 Black Box Testing

Black box testing is a software testing method by which test engineers test the software without Having knowledge of the internal architecture of the application that need to be tested. It is also Known as behavioral testing. Black box testing can be both functional and non-functional. It Ignores the internal mechanism of a system. We have decided to perform the equivalent class Partitioning and Boundary value analysis techniques to implement.

5.2.1.2 White Box Testing

White box testing is a testing technique that examines the program structure and derives test data from the program logic/code. The other names of glass box testing are clear box testing, open box testing, logic driven testing or path driven testing or structural testing.

White Box Testing method is applicable to the following levels of software testing:

- **Unit Testing:** For testing paths within a unit.
- **Integration Testing:** For testing paths between units.
- **System Testing:** For testing paths between subsystems.
-

5.2.2 Pass / Fail Criteria

Pass or fail criteria will be set by the test engineers. They will prepare the pass / fail criteria on The basis of which input data are worked and which are not works. Those data that are Worked well will considered as pass criteria. And rest of input data will be considered as fail Criteria.

- Now I will give the pass / fail criteria below.
- System crash will not be considered as pass case.
- If any criteria pass 100% times, then it will be considered as pass criteria only.
- If data can't be displayed to the application properly, then it is also to be considered as fail criteria.

5.3 Testing Schedule

Test Phase Time	Time
Testing plan create	1 week
Test specification	1 week
Unit testing During development time	During development time
Component test	1 week
Test Phase Time	Time
Integration testing	2 week
Validating use cases	2 week
Testing user interfaces	1 week
Performance testing	2 week
Release to production	1 week

5.4 Trace Ability Matrix

5.5 Testing Environment

A test environment acts as a resource for the execution of the test cases on a software product. It fulfils all the pre-requisite needed in the test execution such as hardware and software requirement, tools and frameworks, system and devices, etc.

For making the environment for testing, some key area need to setup. Those are:
- System and applications
- Test data
- Database server
- Front-end running environment
-
- Client operating system
- Browser

5.6 Test Cases

A test case is a document, which has a set of test data, preconditions, expected results and post conditions, developed for a particular test scenario in order to verify compliance against a specific requirement. Test Case acts as the starting point for the test execution, and after applying a set of input values, the application has a definitive outcome and leaves the system at some end point or also known as execution post condition.

5.6.1 Login

Test Case #1		Test Case Name: Login	
System: Tour by local		**Sub-System:**	
Designed by: Ashraful Islam Sheiblu		**Designed date:**	
Executed By:		**Executed date:**	

Short Description: The registered Guide, Admin login to the System. And before login, System will check the authentication and authorization.

Pre-Condition:
- Admin, guide login the system using registered email and password, and redirect to dashboard. But if wrong can't enter the system.
- Assume that, email ' sheiblu@yahoo.com ' and password is '123456'.

Step	Action	Response	Pass / Fail	Comment
1	Unregistered email and password. (email: aim@gmail.com Password: 1234)	User not found.		
2	Using wrong password. (email: sheiblu@gmail.com Password: 1234)	Wrong password.		
3	Using registered email and password (email: sheiblu@gmail.com Password: 123456)	Successfully login the system and redirect to dashboard.		

Post-Condition: Admin, guide will successfully login to the system.

5.6.2 Create tour

Test Case #3	Test Case Name: Create tour
System: Tour by local	Sub-System:
Designed by: Ashraful Islam Sheiblu	Designed date:
Executed By:	Executed date:

Short Description: Guide must fill up all required fields (like tour area, location, max people, budget ext.) for create tour.

Pre-Condition:
- Guide must provide all mandatory fields.
- Without active account, guide can't access create tour feature.

Step	Action	Response	Pass / Fail	Comment
1	All require fields are not filled up.	Tour creates fails, Please fill up all fields.		
2	All require fields are filled up.	Tours successfully create.		

Post-Condition: Every tour information is saved to the database.

5.6.3 Update profile

Test Case #3	Test Case Name: Update profile
System: Tour by local	Sub-System:
Designed by: Ashraful Islam Sheiblu	Designed date:
Executed By:	Executed date:

Short Description: Providing all mandatory fields, guide can update their profile. Without active profile they can't access.

Pre-Condition:
- Guide must provide all mandatory fields.
- Guide can't change email address.

Step	Action	Response	Pass / Fail	Comment
1	Fill up all mandatory fields.	Profile update successfully.		
2	All require fields are not filled up.	Please fill up all field.		

Post-Condition: Every update data must update and save to database.

5.6.4 View Upcoming Tour

Test Case # 4		Test Case Name: View Upcoming Tour	
System: Tour by local		**Sub-System:**	
Designed by: Ashraful Islam Sheiblu		**Designed date:**	
Executed By:		**Executed date:**	

Short Description: Guide may need to see their upcoming tour. When the tour start date is less than 7 days, guide and traveler see the upcoming tour.

Pre-Condition:
- User must be login the system.
- Tour start date is less than 7 days.

Step	Action	Response	Pass / Fail	Comment
1	Check upcoming tour list when your tour start date is greater than 7 days.	List is empty.		
2	Check upcoming tour list when your tour start date is less than 7 days.	Display upcoming tour list.		
3	Tour start day is already over.	List is empty.		

Post-Condition: List will be shown until tour start.

5.6.5 Approve Guide profile

Test Case #5		Test Case Name: Approve Guide profile		
System: Tour by local		**Sub-System:**		
Designed by: Ashraful Islam Sheiblu		**Designed date:**		
Executed By:		**Executed date:**		
Short Description: Admin approve guide profile, those are pending after validation their information as well.				
Pre-Condition: • Guide must be authenticated by admin. • Only admin can approved or reject the guide.				
Step	**Action**	**Response**	**Pass / Fail**	**Comment**
1	Guide is not approved yet.	Guide can't see any feature like create tour, traveler profile etc. only update their profile.		
2	Guide is approved by admin.	Guide can use all facilities like create tour and another all feature.		
Post-Condition: Before access the guide feature, Guide must be approved by Admin.				

Chapter 6

User Manual

6.1 Create tour

Figure 6.1 create tour

6.2 Guide hiring request
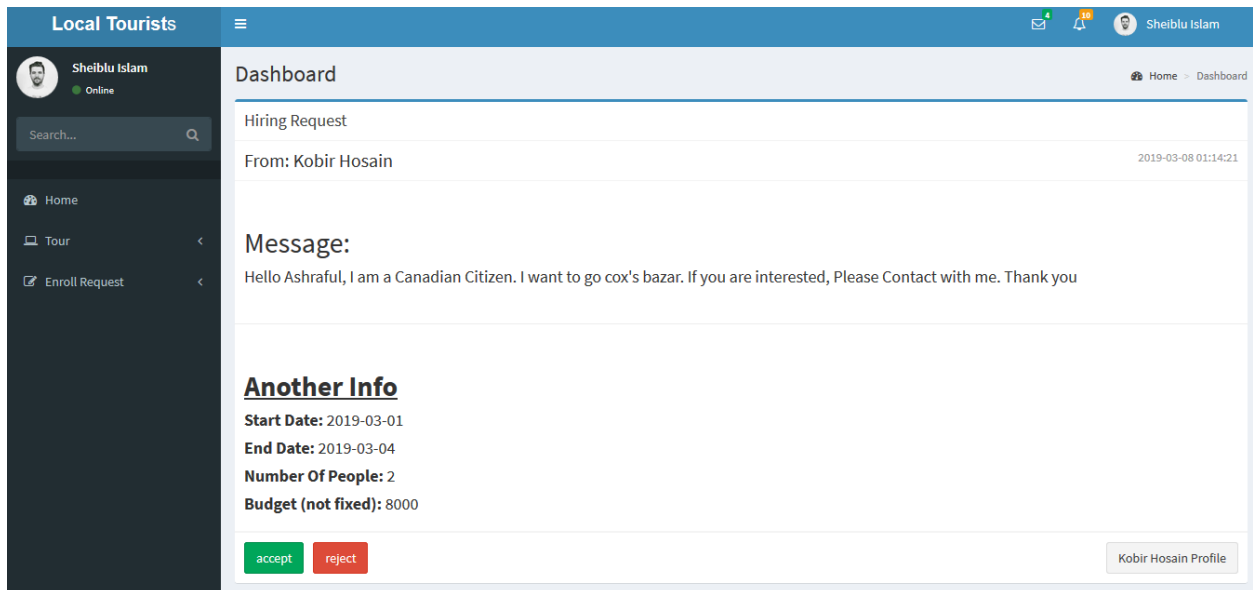
Figure 6.2 guide hiring

6.3 Guide profile for hiring

Figure 6.3: guide profile for hiring

6.4 Guide profile

Figure 6.4 Guide Profile

6.5 Guide request

6.6 Previous tour

Figure 6.6 Previous tour

6.7 tour list

Figure 6.7 tour list

6.8 traveler profile

Figute 6.8 traveler profile

Chapter 7

Conclusion

7.1 GitHub Link

https://github.com/Sheiblu/local_tourist

7.2 Project Summary

The time when we started the development part it's in January. Like other projects, it is important to work hard, endurance, dedication, and concentration to complete the project. There are many reasons for the capitalization of the project development, such as the requirements of the stakeholder to be properly filled.

7.3 Limitations

For developing this project, I have faced some limitations. Now I will describe those in brief.

- **Payment Method:** We do not have any payment method directly in this application. So users can not make any payment transactions through our system. The user pays for him to use our system.

- **No Mobile Version:** In this system Guide and Admin has no mobile version.

7.4 Obstacles and Achievements

By doing this project we learn how to collect and analyze requirements. Brainstorming is a must for any project you're doing to make sure it's done. System analysis and database design that we learn about this project correctly. Our supervisor helps us at every step of this project Also there are some obstacles and achievement in our project and that is below:

- Requirements getting from stakeholders: There are three types of stakeholders in this system and everyone have different functionality. It would be better, if we could gather requirements from them directly.

- Scope Change: When we check the requirements then some features has been added and a few of feature get cutting. When it was needed we did brainstorming for that.

7.5 Future Scope

I learned a lot while finishing this project. To build and improve this project. I have met some young entrepreneurs and enthusiasts. I thank all of them very much.

Besides, I am grateful to them because I have been able to complete my project by adopting their important opinions and discussions. It will help me in the future to work with similar projects

7.6 References

- www.google.com
- www.youtube.com
- www.github.com
- www.wikipedia.com
- www.php.net
- www.mysql.com