Travel Guide

By

Anamika Das

ID: 161-35-1440

A project report 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 –2020

Approval

This project titled on "**Travel Guide**", submitted by **Anamika Das** (**ID: 161-35-1440**) to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfilment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS

---------- **Chairman**

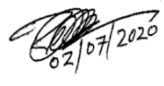
Prof. Dr. Touhid Bhuiyan
Professor and Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

---------- **Internal Examiner 1**

Dr. Md. Asraf Ali
Associate Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

---------- **Internal Examiner 2**

Asif khan Shakir
Lecturer (Senior Scale)
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University



---------- **External Examiner 2**

Prof. Dr. Mohammod Abul Kaashem
Professor
Department of Computer Science and Engineering
Dhaka University of Engineering and Technology, Gazipur

Declaration

It hereby declares that this project has been completed by me under the supervision of **Dr. Md Mostafijur Rahman, Associate Professor, Department of Software Engineering, and Daffodil International University**. It is also declared that neither this work nor any part of this has been submitted elsewhere for award of any degree or diploma by me.

Submitted by:

Anamika Das

Student ID: 161-35-1440

Batch: 19th

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

Certified by:

Dr. Md Mostafijur Rahman

Associate Professor

Department of Software Engineering

Faculty of Science & Information Technology

Daffodil International University

Acknowledgement

First of all, I am grateful to The Almighty Allah for making me eligible to complete this work. After that I would like to thank my supervisor Dr. Md Mostafijur Rahman, Associate Professor, Department of Software Engineering. I am extremely thankful to his skill, sincere and valuable guidance and encouragement extended to me. I would like to express my heartiest thanks to Dr. Touhid Bhuiyan, professor & head of Software Engineering department for his continuous encouragement. I also would like to thank all the faculty members of the Department of Software Engineering for their assistance & encouragement. Last but not least, I would like to thank to my parents for their unconditional support and love.

Abstract

"Travel Guide" this is an online based travel guide taking system. It helps traveler to get their travel guide through internet by using website. This project is helpful for user who doesn't waste their time to get their desire jobs as a travel guide. It is also useful for tour and tourism event organizer or company to get their client. All types of users can easily login from this site, and travelers can send request to hire travel guide by providing their own requirements. Travel guides can to response a request at a time in according with their choice. Tour organizers can create tour events or packages without trouble. Admin can monitor registered users as needed, approve events which are created by organizer or company for showing on traveler screen. This website another benefit is traveler who books package not only one organizer or company same packages but other organizer or company packages also see by searching. All of users are able to update their profile information.

Table of Contents

Chapter Three

List of Figures

Chapter One: Introduction

Chapter Three: Software Requirement Analysis

Chapter Six: User Manual

List of Tables

Chapter One: Introduction

Chapter Two: Software Requirement Specification

Chapter Three: Software Requirement Analysis

Chapter Seven: System Testing

Chapter One

Introduction

1.1 Overview

This system is fully online based. Online based system traveler hires travel guide to smooth and secure his travel. This online system traveler can see that all categories packages of various event organizer or company and booking easily. Normally any online system we can found only one company tour packages but this system traveler search any type of packages and they can see same package of various company. So traveler can easily find out their suitable packages.

1.2 Purpose

- Provide tour guide to serve the correct info, local points of view and to give abusive, relevant, arranged heritage, transportation, and lodging info to tourists.
- Make a platform to provide job for people as a travel guide.
- Make an easy way to event Organizer or Tour & Tourism Company to get tourist in online.
- Create a successful tour event without wasting time that is easy to manage various events over the internet.
- Make the process traveler choose tour packages not only one organizer or company, they can also see others company packages and booking those.

1.3 Background

Before this system generate tourists were going to in different places without any travel guide. As a result they didn't get plan for a perfect trip and it was costly and unsecure for unknown places. Many local people are jobless where has lots of visited places but they well-known about culture, background history, the easiest way to reach of those places. The arrival and the departure time at each tourist site is restricted by its travel agency which is not available tour time for someone. When they don’t choice package,

waste time to go for adventure that is long and more hassle. So to remove these problems and make it more efficient for both tourist, travel guide and tour organizer, I take this project.

1.4 Objectives

- To make the system automatic and digital.
- To reduce the manual works.
- To make the process time efficient.
- To secure and arrange the data efficiently.
- To make an easy way to find an authentic travel guide.
- To found secure tour packages easily.
- Easy to use.

1.5 Stakeholder

There are many members are associate with this project. They have helped to develop the system directly or indirectly.

1.5.1 Internal Stakeholder

1. Admin
2. Tourist
3. Travel Guide
4. Tour Event Organizer

1.5.2 External Stakeholder

1. Visitors

1.6 Proposed System

To develop this system, I proposed a model for this online Travel Guide.

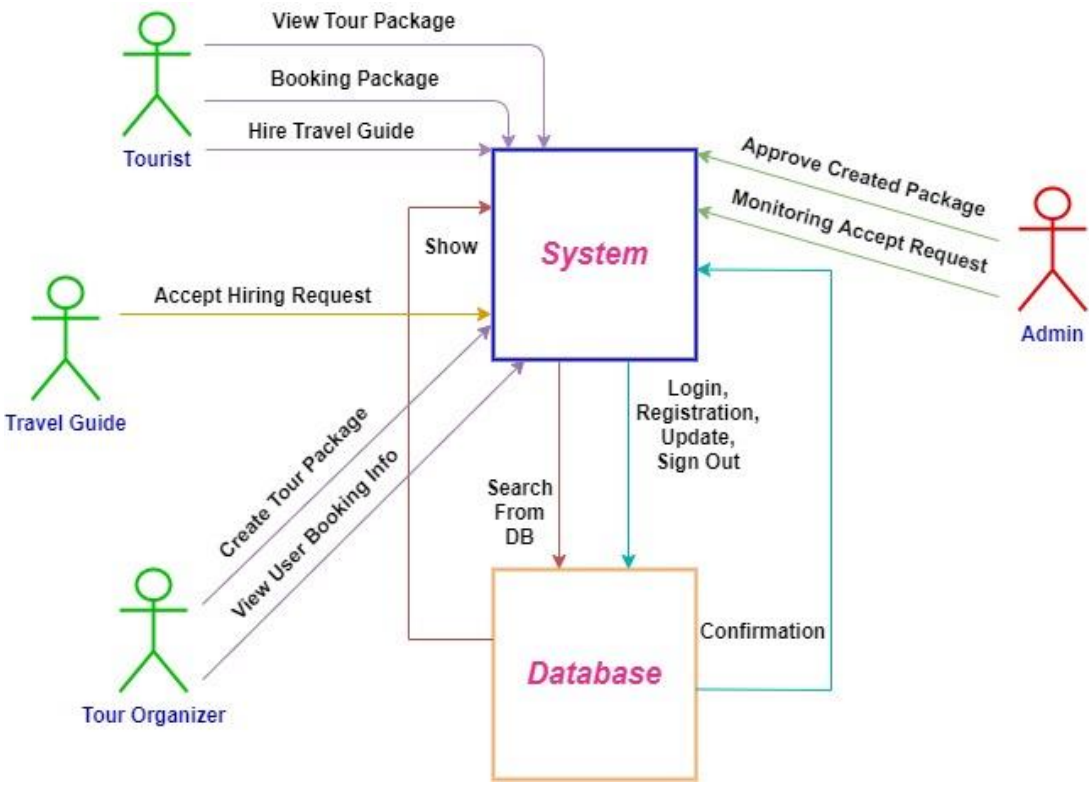


Figure 1.1: Propose System Model

In this system user should register for using the system. To registration a user, first he/she give his/her information. After register, a user can use the system through login using email, NID and password. Than the user can hire travel guide as needed. According to the user choice, he can select tour package and booking package. After confirmation by the user, system booked the package for that user.

1.7 Project Plan

Properly full fill the requirements and complete the project at the right time, project schedule helps for proper planning. I also make a project schedule to complete my project properly.

Table 1.1: Project Plan

Task Name	Resource Name	Start	Finish	Duration(Days)
Planning	PM,BA,	01-06-2019	12-06-2019	12
Requirement	RE, RA	13-06-2019	18-07-2019	36
System Design	UI Designer,	19-07-2019	16-09-2019	60
Database Design	DB Designer	17-09-2019	16-10-2019	30
Development	Developer	17-10-2019	29-02-2020	136
Testing	Tester	01-03-2020	25-03-2020	25
Implementation	Implementation Eng.	25-03-2020	12-04-2020	18
Delivery	PM, Implementation Eng.	13-04-2020	25-04-2020	13
			Total	330 Days

1.8 Gantt chart

In project planning, I use Gantt chart to manage my project properly. To use these tools, I can track the entire task which is not done or not. Also track which one is schedule for the next task. I control my project duration by these tools.

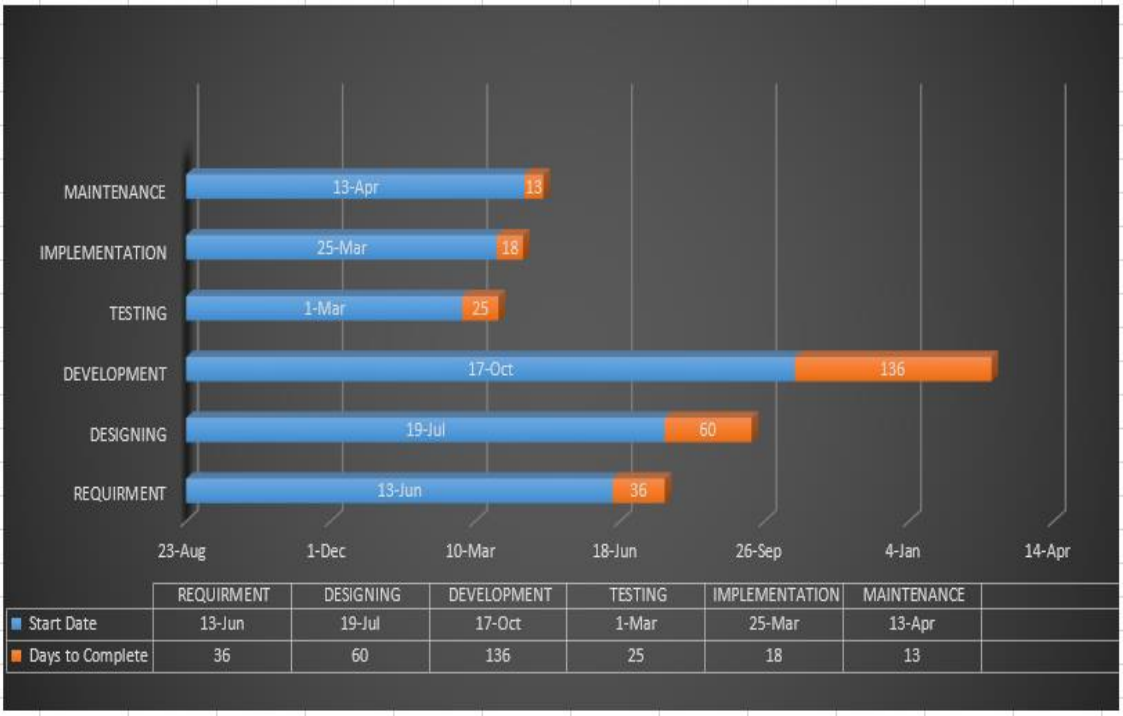


Figure 1.2: Gantt chart

1.9 Risk

In Gantt chart tools, I show the project duration. I took days for developing, so that I could mitigate my risk. In developing page, I find out a risk that risk is developer. Who may sick or leave.

1.10 Milestone

Milestones, a time frame of a project, will define the task. These project milestones are as follows:

Table 1.2: Project Milestone

Task No	Task Name	Duration
01	Planning	12 Days
02	Requirement gathering & Analysis	36 Days
03	System Design	60 Days
04	Data Base Design	30 Days
05	Development	136 Days
06	Testing	25 Days
07	Implementation	18 Days
08	Relies	13 Days
	Total	330 Days

Chapter Two

Software Requirement Specification

2.1 Requirement satisfaction

Requirement is the process of identifying the user satisfaction form the System. So, Requirement is an important part of project management.
When I selected this project I thought about some specific Software requirement, like as

- Who is the stakeholder of this system?
- Is it helpful for them or not?
- Functional & Non- functional requirements?
- Is it efficient for using?

2.2 Functional requirement

The functional requirements of the system are like below –

2.2.1 User Registration

Table 2.1: User Registration

FR-01	**User Registration**
Description	In this system there are many users like Tourist, Travel guide, Tour Event Organization, Admin, visitor etc. But registration is must for those User who want to hire Travel guide, get tour package and create tour event. This page is required some information like as username, email, phone number, NID number, user type etc. and store the information.
Stakeholder	**Traveler, Travel Guide, Tour Organizer, Admin**

2.2.2 User Login

Table 2.2: Discuss About User Login

FR-02	**User Login**
Description	Here, login is must for those User who want to hire Travel guide, get tour package and create tour event. This page is required some information like as email, NID number, and password to login in this system after registration.
Stakeholder	**Traveler, Travel Guide, Tour Organizer, Admin**

2.2.3 Hire Travel Guide

Table 2.3: Discuss About Hire Travel Guide

FR-03	**Hire Travel Guide**
Description	Only registered user can hire travel guide as his tour helper. This module will help the user to hire travel guide through taking some information.
Stakeholder	**Traveler**

2.2.4 Accept Hiring Request

Table 2.4: Discuss About Accept Request

FR-04	**Accept Hiring Request**
Description	Only registered Travel Guide can accept request which is send from traveler. This module will help the Travel Guide to accept hiring request and sent confirmation to traveler.
Stakeholder	**Travel Guide**

2.2.5 Create Tour Event

Table 2.5: Discuss About Create Tour Event

FR-05	**Create Tour Event**
Description	Only registered Tour Organizer can create tour event for getting tourist. This module will help the user to create tour events through taking some information by the system.
Stakeholder	**Tour Organizer or Company**

2.2.6 Approve Tour Event

Table 2.6: Discuss About Approve Event

FR-06	**Approve Tour Event**
Description	Only Admin can approve created tour event for showing on traveler wall. This module will help the user to approve created tour events.
Stakeholder	**Admin**

2.2.7 Show and Booking Tour Package

Table 2.7: Discuss About Booking Tour Package

FR-07	**Show and Booking Tour Package**
Description	Only registered Traveler can show tour packages which are approved by Admin and booking those from his wall. This module will help the traveler to show and booking package.
Stakeholder	**Traveler**

2.2.8 Monitoring System

Table 2.8: Discuss About Monitoring System

FR-08	**Monitoring System**
Description	Without Admin there is nobody monitor the whole system. Only Admin can track all users' information.
Stakeholder	**Admin**

2.3 Performance Requirement

It's very necessary to sustain the performance of the project. To assure the better performance, this project has to meet some requirements which will provide the better performance.

2.3.1 Speed and Latency Requirement

While inserting or viewing the system in the browser, system need a minimum amount of speed to perform the task.

Table 2.9: Discuss About Speed and Latency Requirement

SLR-01	**The system will be faster**
Description	When the user browsing, it depends on their internet speed. It also depends on server bandwidth speed.
Stakeholders	**User, Admin, Visitor**

2.3.2 Legibility and accuracy requirements

System has to confirm the Legibility and Accuracy of the data.

Table 2.10: Discuss About Legibility and Accuracy

LAR-01	**Data accuracy**
Description	The input data should be correct and right pattern data, otherwise the input information never save. Like email, username, NID, user type etc. the input information is not valid, the data never save. Or the input data pattern is not match; the system never saves or accepts the data.
Stakeholders	**User, Admin, Visitor**

2.3.3 Capacity requirements

The system should maintain the all inserting data.

Table 2.11: Explain on Capacity Requirements

CR-01	**Manage the all data in database system.**
Description	All registration data like Traveler, Travel Guide, Tour Organizer, Admin data and other information are store in the database in right format.
Stakeholders	**User, Admin, Visitor.**

2.3.4 Dependability requirements

Dependability means, it measures of a system availability, reliability, security etc. Here, dependability means the run time of this project.

2.3.5 Reliability and availability

Table 2.12: Explain on Reliability and Availability

RA-01	The system must be available 24x7
Description	➢ It's available 24 hours in a day ➢ The system must be updated regularly
Stakeholders	**Traveler, Travel Guide, Tour Organizer, Admin, Visitor.**

2.3.6 Safety critical requirements:

There are no specific safety critical requirements.

2.4 Maintainability and supportability

To maintain the system and support the system, some people are connected with the project.

2.4.1 Supportability requirements specification

- **SRS-1.** Understand the system's behavior on a technical support is required by the system operator. The reason for reading them might be
- **SRS-2.** System malfunction has occurred and the system operator has to find the exact point of time when this happened
- **SRS-3.** System produces wrong results and the developers must be able to reproduce the data flow through the system
- **SRS-4**. Anyone tried to breach the system's security mechanisms and the system operator must understand what he did.

2.5 Adaptability requirements

There is no specific adaptability Requirements.

2.5.1 Security requirements

- **SR-1.** Log in as a Traveler
- **SR-2.** Log in as Travel Guide.
- **SR-3.** Log in as a Tour Organization.
- **SR-4.** Log in as an Admin.

To get access to this system or a specific module the system must provide an authentication mechanism. To prevent anyone to exploit stolen Data all user’s password must be encrypted in hash process.

2.5.2 Access requirements

This system provides accesses the different module, by access the authentication way the authentic user.

2.5.3 Integrity requirements

To prevent credentials information of user from being stolen, all passwords are stored in encrypted form. The Requirements significantly reduces the value of stolen user credentials, it’s not easy to decrypt the password.

2.6 Usability and human integrity requirements

This system easy to use and all the person who wants to get a travel guide or tour package and who manage it.

2.7 Data Validation

In this stage I have try to validate almost all input field.

2.8 User Interface Design

It is important to consult the system users and their necessities while designing the user interface.

Chapter Three

Software Requirement Analysis

3.1 Use Case Diagram:

In this system a user what things he/she can do, is describe in this picture that provide in below. A user can login in the system. But before login he/she must registration in this system as a user which has three types like Traveler, Travel Guide and Tour Organizer. Then they can access the login option. After login Traveler can access on his/her portal to take travel guide or booking tour package for his/her easy travel. Tour Guide can access on his own portal to response traveler request. Tour Organizer can create tour event or package to get tourist for his/her business. Admin should login to approve created tour package and he/she can manage all the event, task of the system.

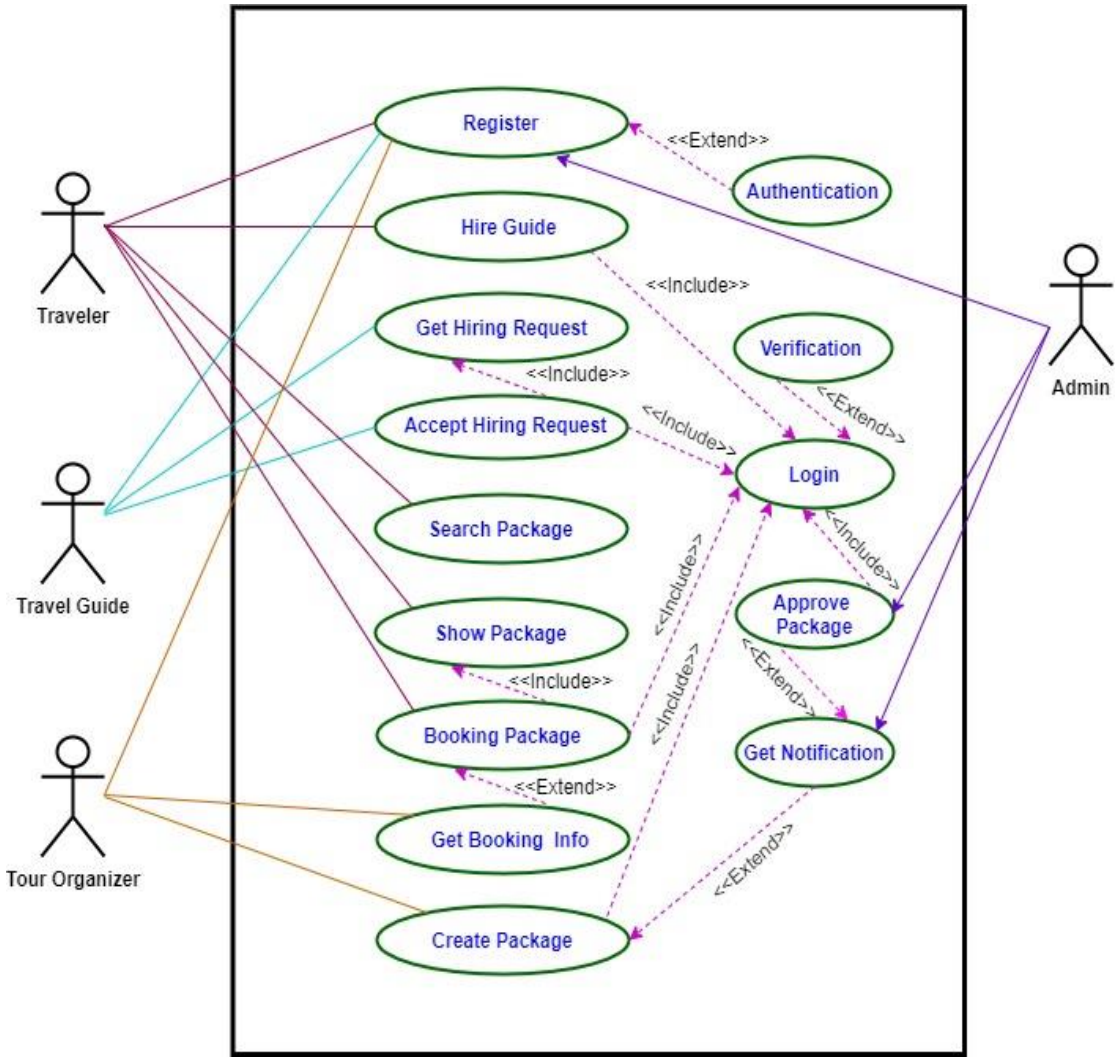


Figure 3.1: Use Case Diagram

3.1.1 Registration

Table 3.1: Use Case Description of Registration

Use Case	Registration	
Use case no	01	
Goal	Register to the system	
Precondition	Need to fill up with correct information	
Primary actor Secondary actor	Traveler, Travel Guide, Tour Organizer, Admin None	
Trigger	Button	
Description/Main success scenario	Step	Action
	1.	Browse into the system
	2.	Select the registration button if new in this system
	3.	Select submit button
	4.	Registration complete
Post Condition	User can login with email, NID & password	
Alternative flow	Not applicable	

3.1.2 Login

Table 3.2 : Use Case Description of Login

Use case	Login	
Use case no	02	
Goal	Login into the system to change anything into the system	
Precondition	Must be registered	
Primary actor Secondary actor	Traveler, Travel Guide, Tour Organizer, Admin None	
Trigger	Button	
Description/Main success scenario	Step	Action
	1.	Visit the Travel Guide system home page
	2.	Select login page
	3.	Submit email, NID and password
	4.	If yes, than login successful
Post Condition	System response with a login successful message	
Alternative flow	Not applicable	

3.1.3 Hire Guide

Table 3.3 : Use Case Description of Hire Guide

Use case	Hire Guide	
Use case no	03	
Goal	To take a Travel Guide as a tour assistant from this system	
Precondition	Must be Login into the system	
Primary actor Secondary actor	Traveler None	
Trigger	Button	
Description/Main success scenario	Step	Action
	1.	Login into the system
	2.	Go to Hire guide page
	3.	Provide required information about tour plan
	4.	Click Hire Guide button to send request
Post Condition	System response with a successful message	
Alternative flow	Not applicable	

3.1.4 Get Hiring Request

Table 3.4 : Use Case Description of Get Hiring Request

Use case	Get Hiring Request	
Use case no	04	
Goal	Get hiring request from traveler	
Precondition	Must be Login into the system	
Primary actor Secondary actor	Travel Guide None	
Trigger	Button	
Description/Main success scenario	Step	Action
	1.	Login into the system
	2.	Check notification option from profile page
Post Condition	Show request details	
Alternative flow	Not applicable	

3.1.5 Accept Hiring Request

Table 3.5: Use Case Description of Accept Hiring Request

Use case	Accept Hiring Request	
Use case no	05	
Goal	Accept hiring request to perform as a travel guide	
Precondition	Must be Login into the system and get request	
Primary actor Secondary actor	Travel Guide None	
Trigger	Button	
Description/Main success scenario	Step	Action
	1.	Login into the system
	2.	View notification from profile page
	3.	Check that it is available or not
	4.	Click on Accept request button if it is available
Post Condition	System response with a successful message	
Alternative flow	Not applicable	

3.1.6 Search package

Table 3.6: Use Case Description of Search Package

Use case	Search package	
Use case no	06	
Goal	Search preferable package with tourist place name	
Precondition	Must be Login into the system and go to search box	
Primary actor Secondary actor	Traveler None	
Trigger	Button	
Description/Main success scenario	Step	Action
	1.	Login into the system
	2.	Go to search box from profile page
	3.	Write tourist place name
	4.	Press enter or click for searching
Post Condition	System response with available package	
Alternative flow	Not applicable	

3.1.7 Show package

Table 3.7: Use Case Description of Show package

Use case	Show package	
Use case no	07	
Goal	Show all category of tour packages	
Precondition	Must be Login into the system	
Primary actor Secondary actor	Traveler None	
Trigger	Button	
Description/Main success scenario	Step	Action
	1.	Login into the system
	2.	Go to Tour Package page from profile page
	3.	View all post of tour packages
Post Condition	System show available package	
Alternative flow	Not applicable	

3.1.8 Booking package

Table 3.8: Use Case Description of Booking Package

Use case	Booking package	
Use case no	08	
Goal	Booking specific tour package from all package	
Precondition	Must be Login into the system	
Primary actor Secondary actor	Traveler None	
Trigger	Button	
Description/Main success scenario	Step	Action
	1.	Login into the system
	2.	Go to Tour Package page from profile page
	3.	View all post of tour packages
	4.	Click on Booking button which is choice
Post Condition	System response with a successful message	
Alternative flow	Not applicable	

3.1.9 Get Booking Information

Table 3.9: Use Case Description of Get Booking Information

Use case	Get Booking Information	
Use case no	09	
Goal	Get booking information which are booked	
Precondition	Must be Login into the system	
Primary actor Secondary actor	Tour Organizer None	
Trigger	Button	
Description/Main success scenario	Step	Action
	1.	Login into the system
	2.	Go to notification option
	3.	View booking information who is booked which package
Post Condition	System show booking information	
Alternative flow	Not applicable	

3.1.10 Create package

Table 3.10: Use Case Description of Create Package

Use case	Create package	
Use case no	10	
Goal	To create tour package or event	
Precondition	Must be Login into the system	
Primary actor Secondary actor	Tour organizer None	
Trigger	Button	
Description/Main success scenario	Step	Action
	1.	Login into the system
	2.	Go to Create event page from profile page
	3.	Fill up all required option to create an event or package
	4.	Click on Create button
Post Condition	System response with a successful message	
Alternative flow	Not applicable	

3.1.11 Approve package

Table 3.11: Use Case Description of Approve Package

Use case	Approve package	
Use case no	11	
Goal	To approve tour package or event which are created	
Precondition	Must be Login into the system	
Primary actor Secondary actor	Admin None	
Trigger	Button	
Description/Main success scenario	Step	Action
	1.	Login into the system
	2.	Go to notification option from profile page
	3.	View created package which is unapproved
	4.	Click on Approve button to show new created package on Tour Package page of Traveler's profile.
Post Condition	System response with showing approved which was unapproved	
Alternative flow	Not applicable	

3.2 Activity Diagram

Following activity diagrams are exactly describing the flow of the different state of the project.

3.2.1 Registration

By this figure I explain my system. If anyone enter the system, he/she see the all the option. And who are registered user and he/she can login in the system. According to the rule he/she can access different potion.

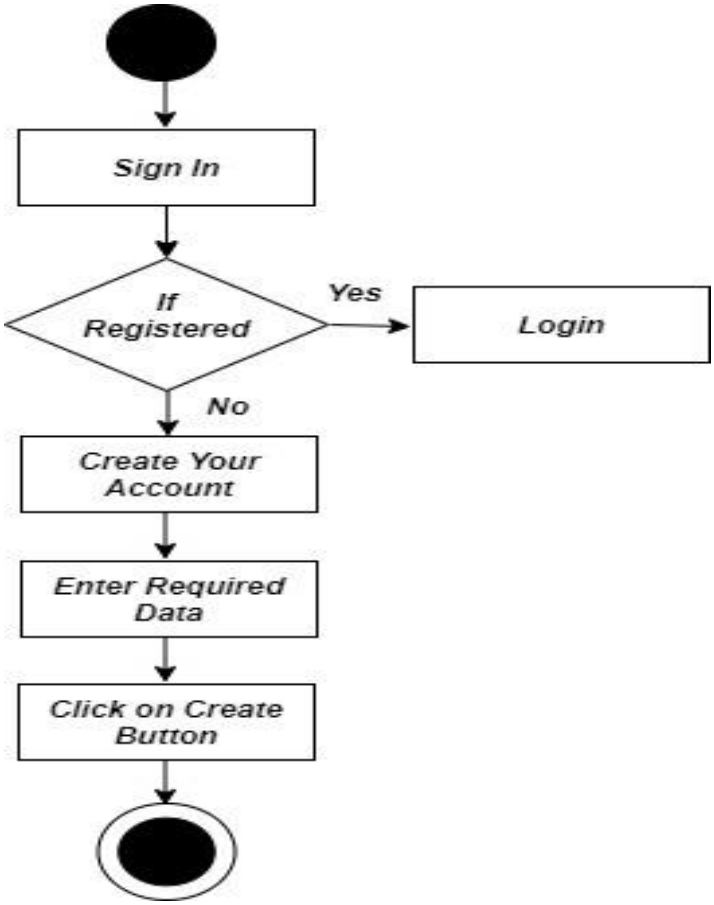


Figure 3.2: Activity Diagram of Registration

3.2.2 Login

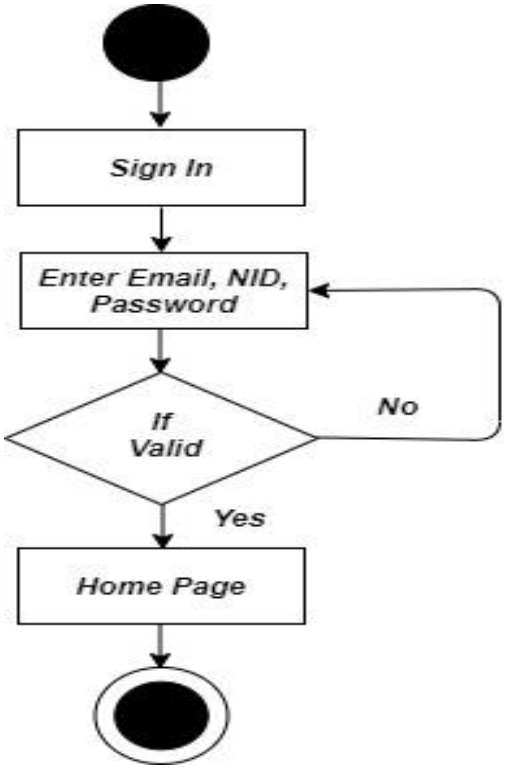


Figure 3.3: Activity Diagram of Login

3.2.3 Hire Travel Guide

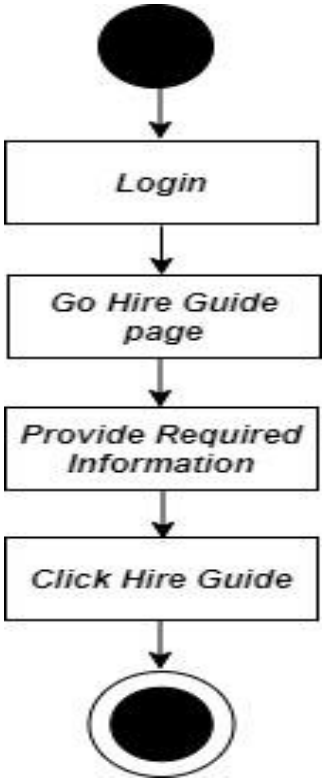


Figure 3.4: Activity Diagram of Hire Travel Guide

3.2.4 Get and Accept Hiring Request

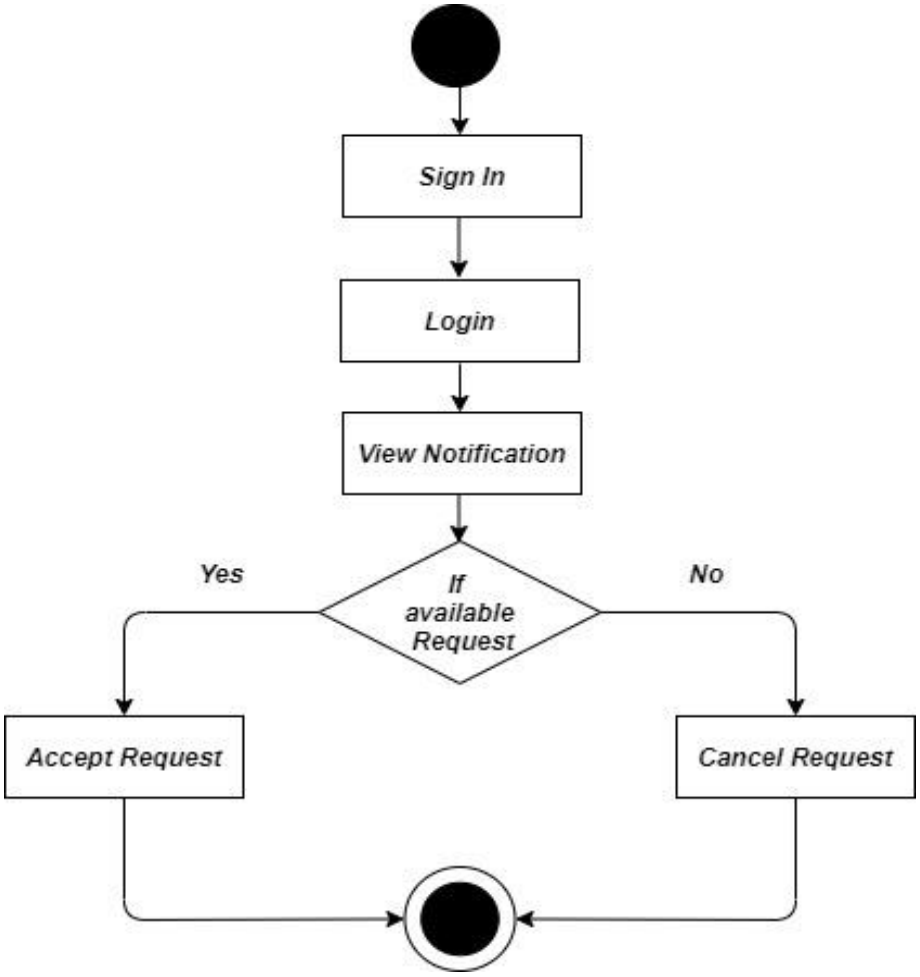


Figure 3.5: Activity Diagram of Accept Hiring Request

3.2.5 Search and Show Package

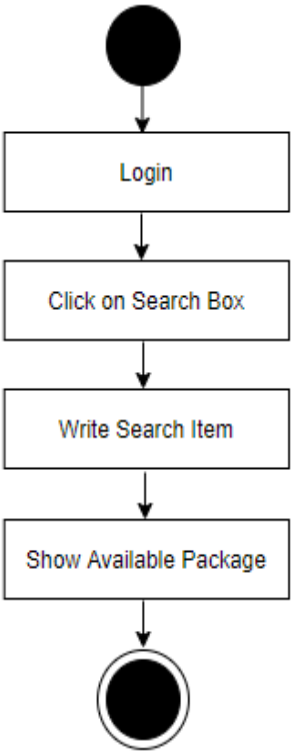


Figure 3.6: Activity Diagram of Search and Show Package

3.2.6 Booking Package

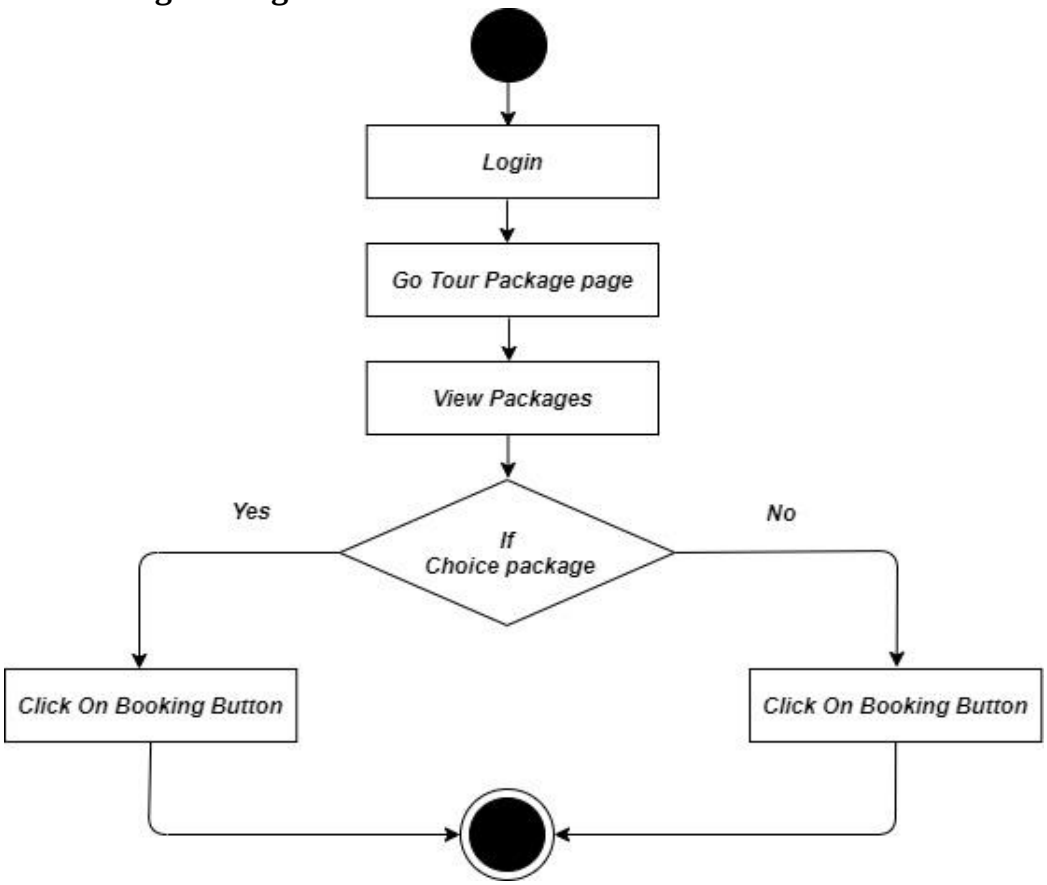


Figure 3.7: Activity Diagram of Booking Package

3.2.7 Get Booking Information

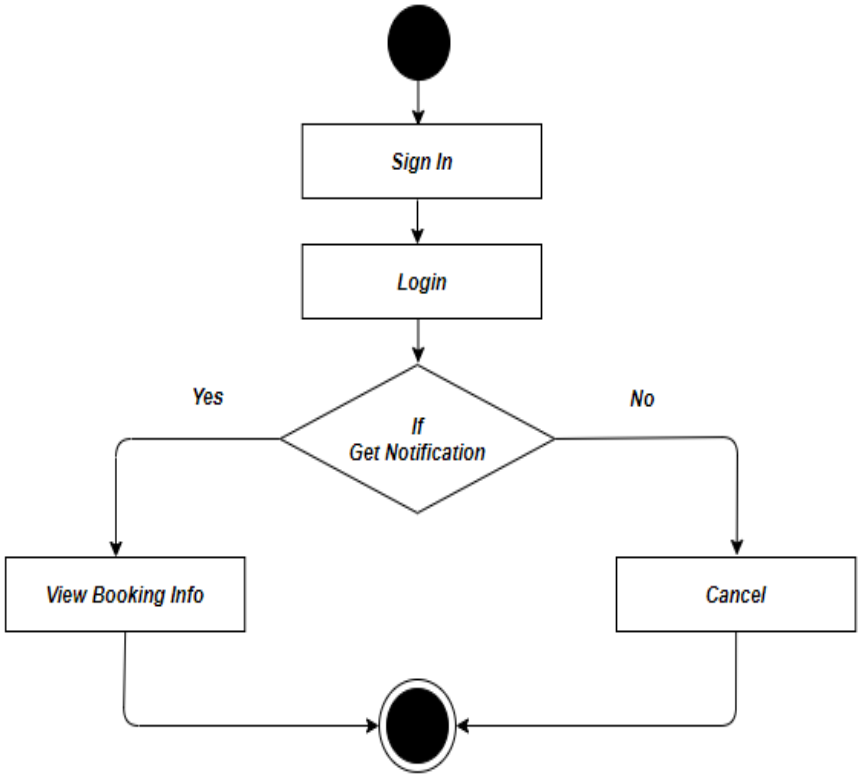


Figure 3.8: Activity Diagram of get Booking Information

3.2.8 Create Package

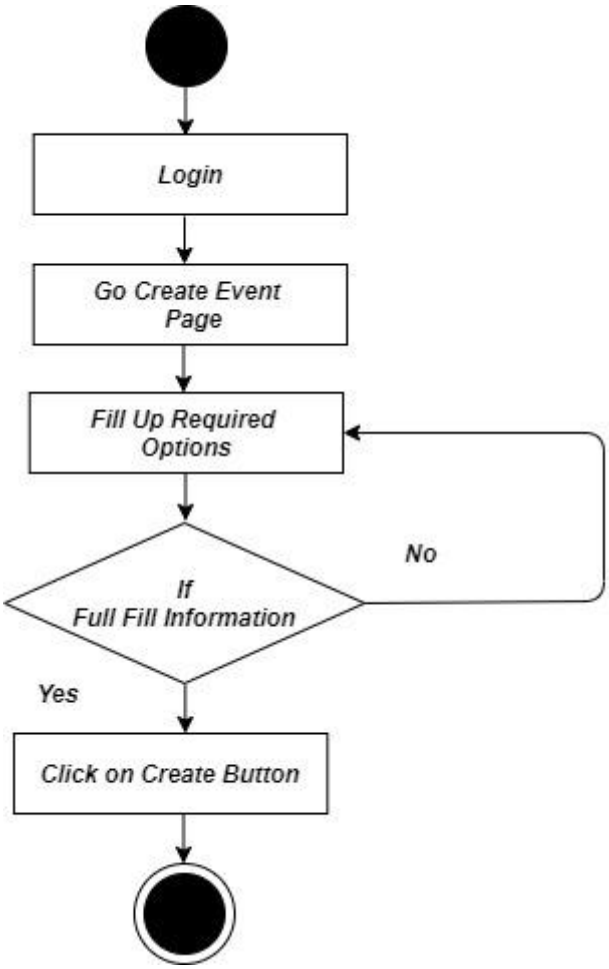


Figure 3.9: Activity Diagram of Create package

3.2.9 Approve Created Package

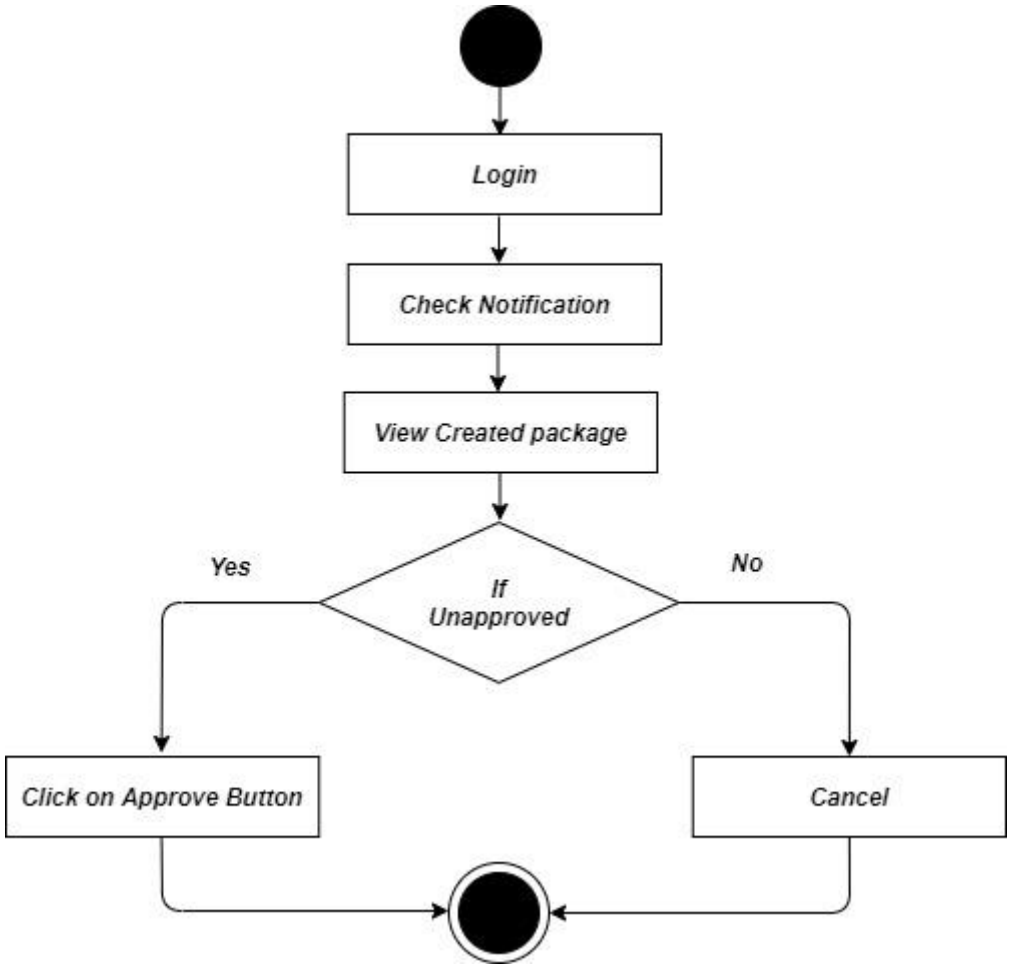


Figure 3.10: Activity Diagram of Approve Package

3.3 Sequence Diagram:

Sequence Diagram show the process in sequential way that it's actor done. In this section describe the sequence system to database.

3.3.1 Registration:

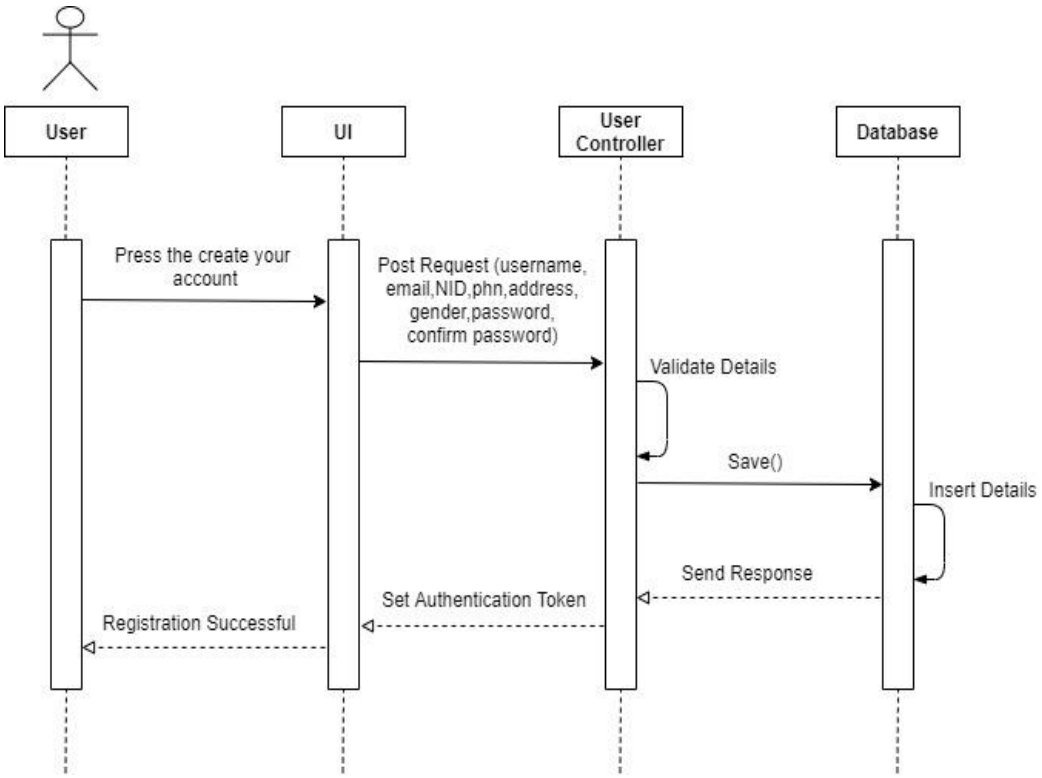


Figure 3.3.1: Registration of Sequence Diagram

3.3.2 Login:

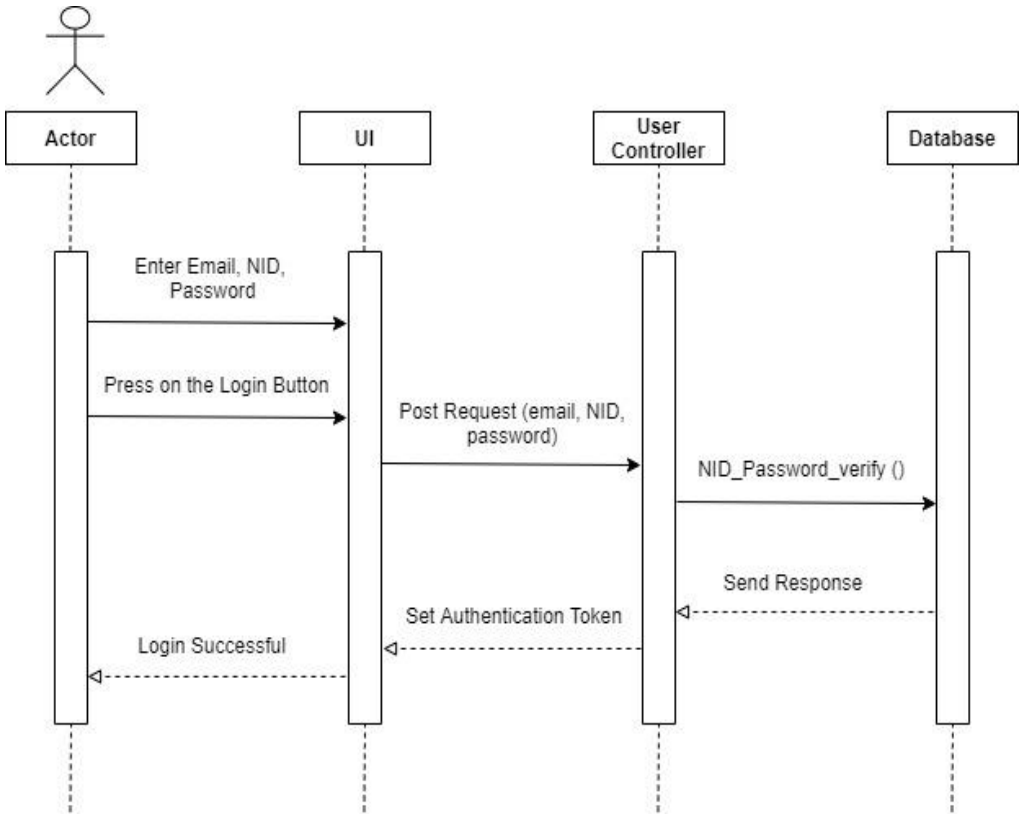


Figure 3.3.2: Login of Sequence Diagram

3.3.3 Hire Travel Guide:

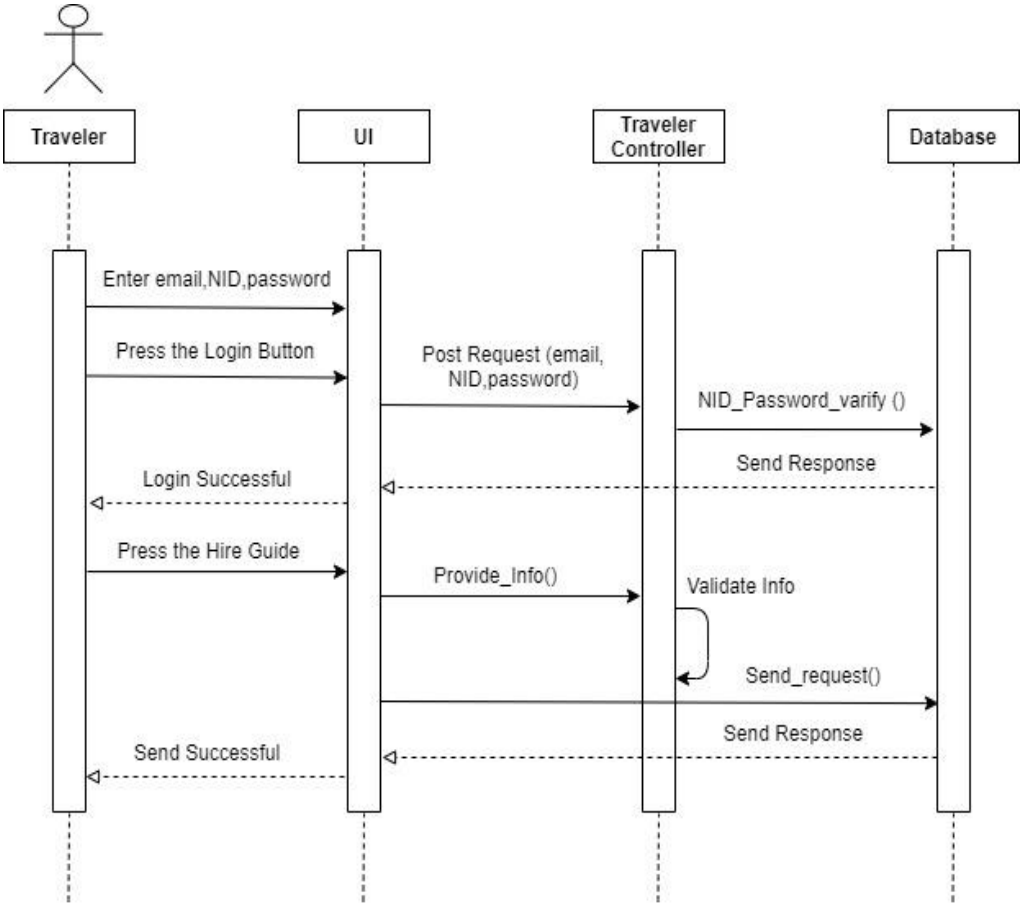


Figure 3.3.3: Sequence Diagram of Hire Travel-Guide

3.3.4 Booking Package:

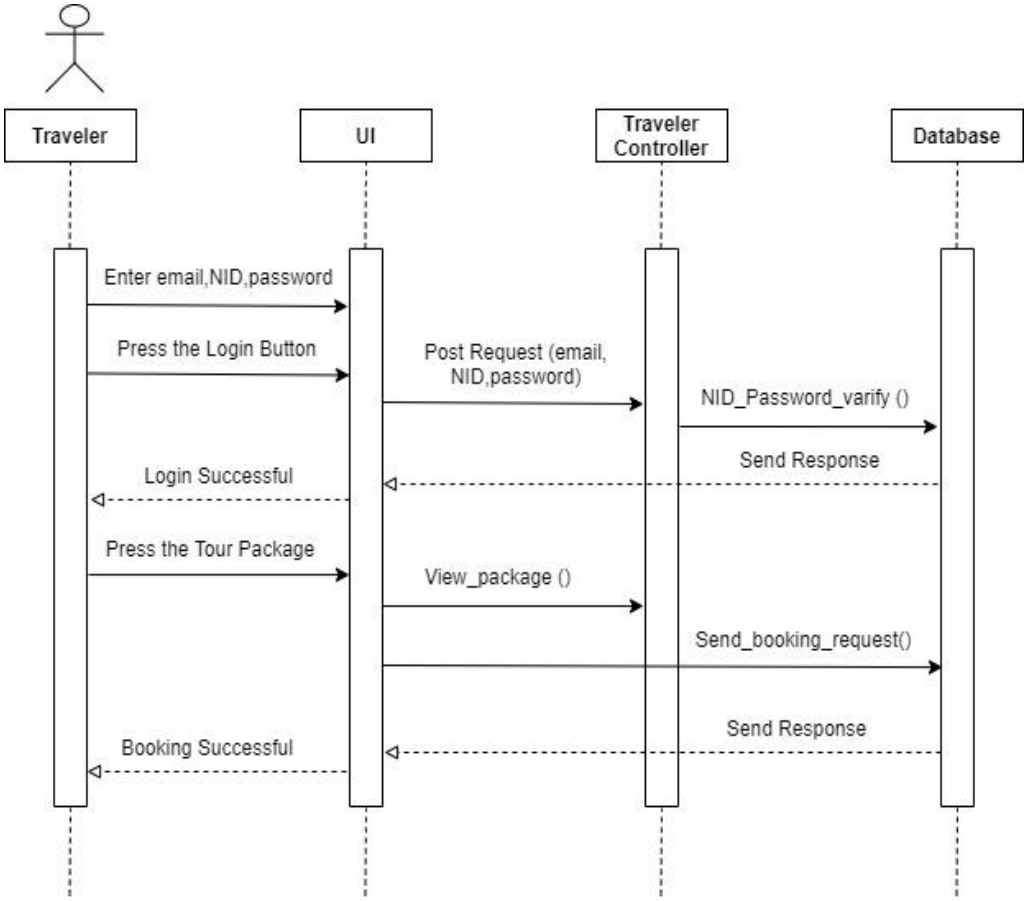


Figure 3.3.4: Sequence Diagram of Booking Package

3.3.5 Get and Accept Hiring Request:

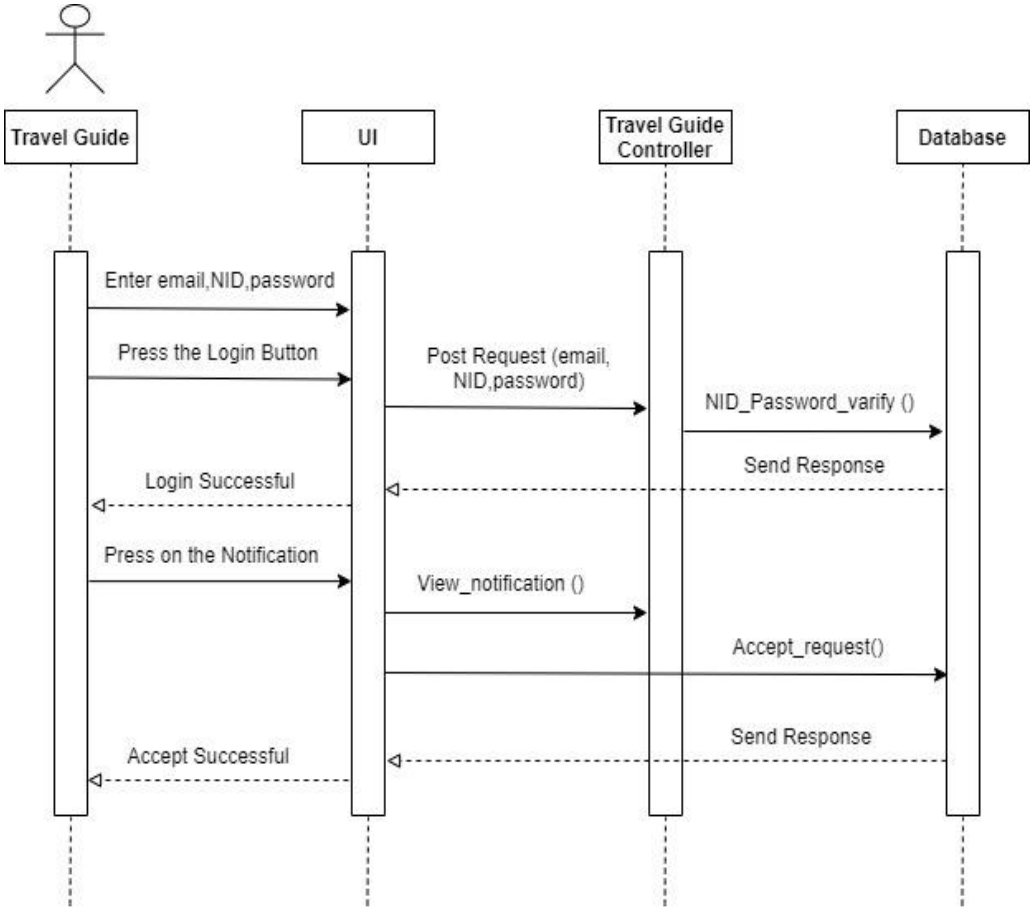


Figure 3.3.5: Get and Accept Hiring Request

3.3.6 Get Booking Information:

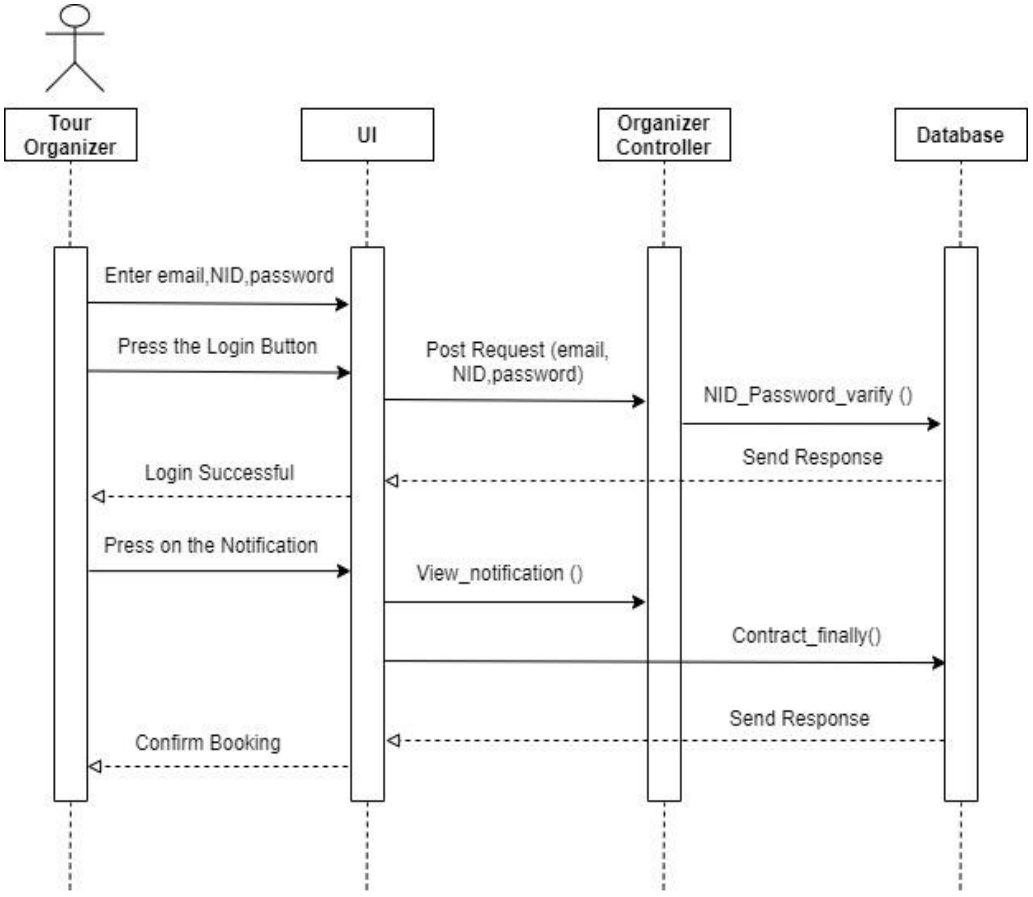


Figure 3.3.6: Get Booking Information of Sequence Diagram

3.3.7 Create Tour Package:

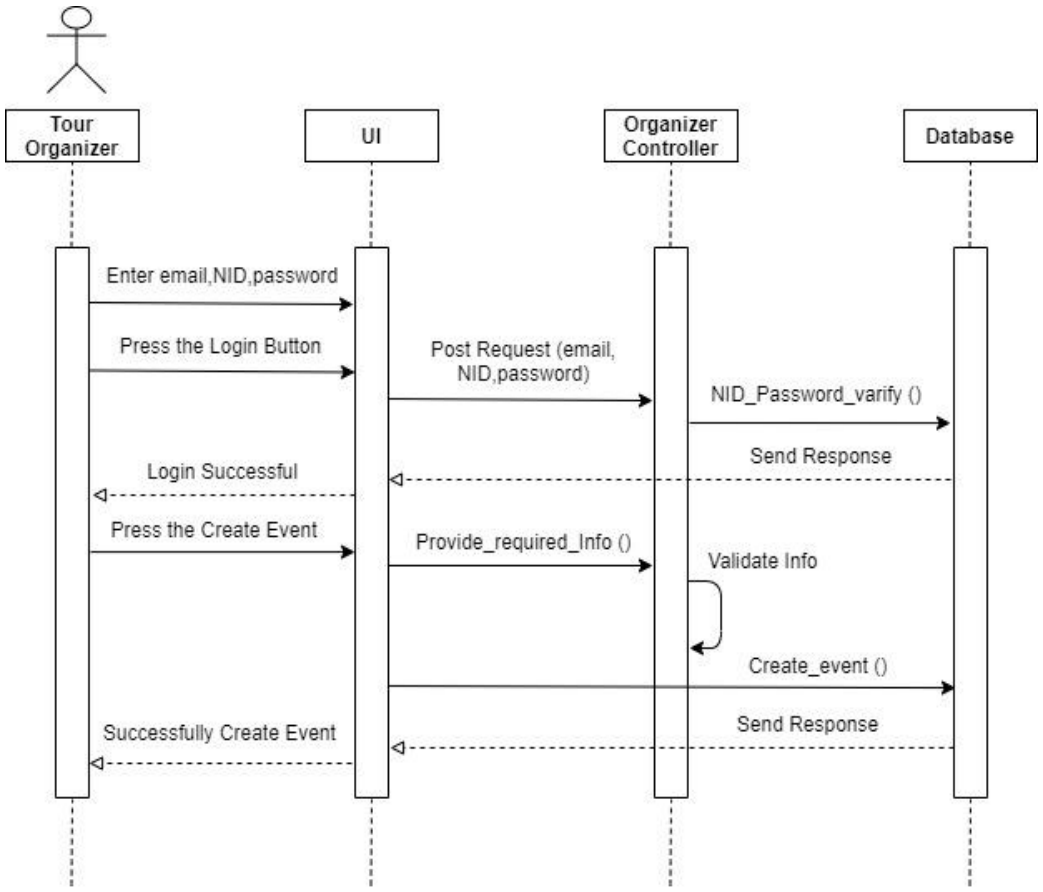


Figure 3.3.7: Create Tour Package of Sequence Diagram

3.3.8 Approve Created Package:

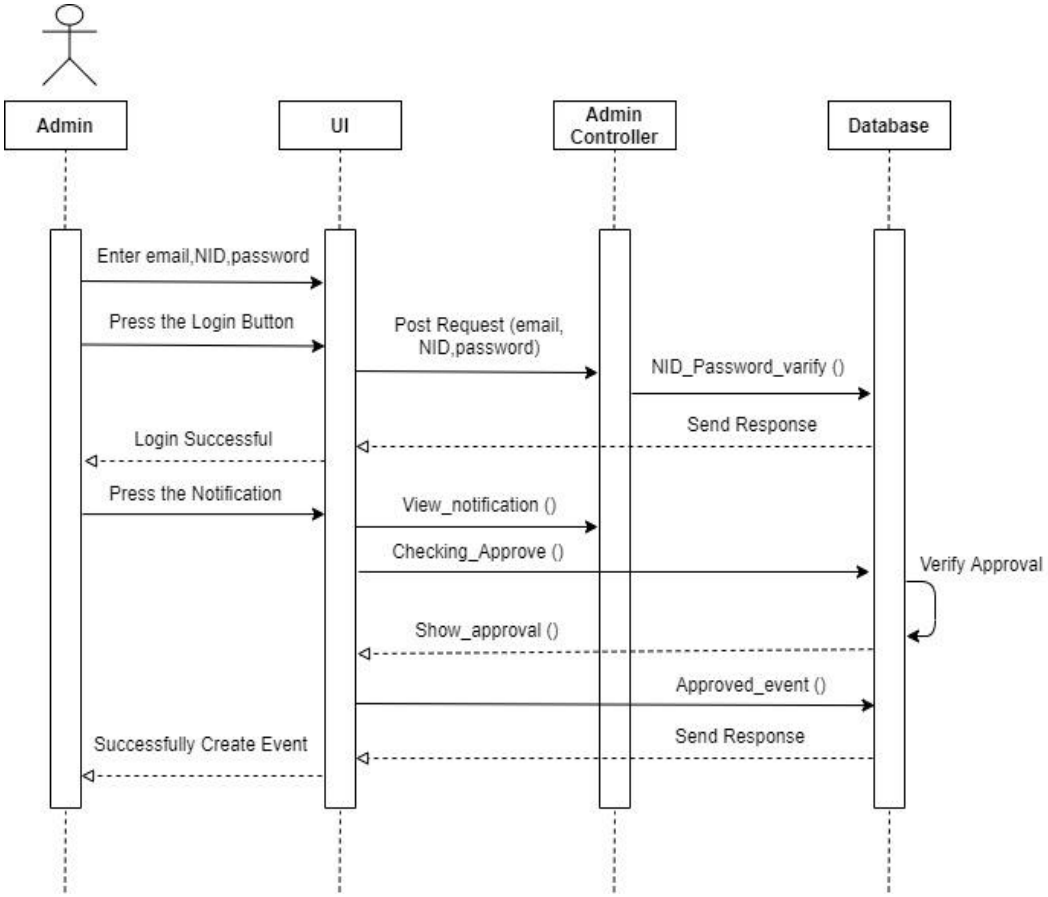


Figure 3.3.8: Sequence Diagram of Approve Created Package

3.3.9 Monitor User Information:

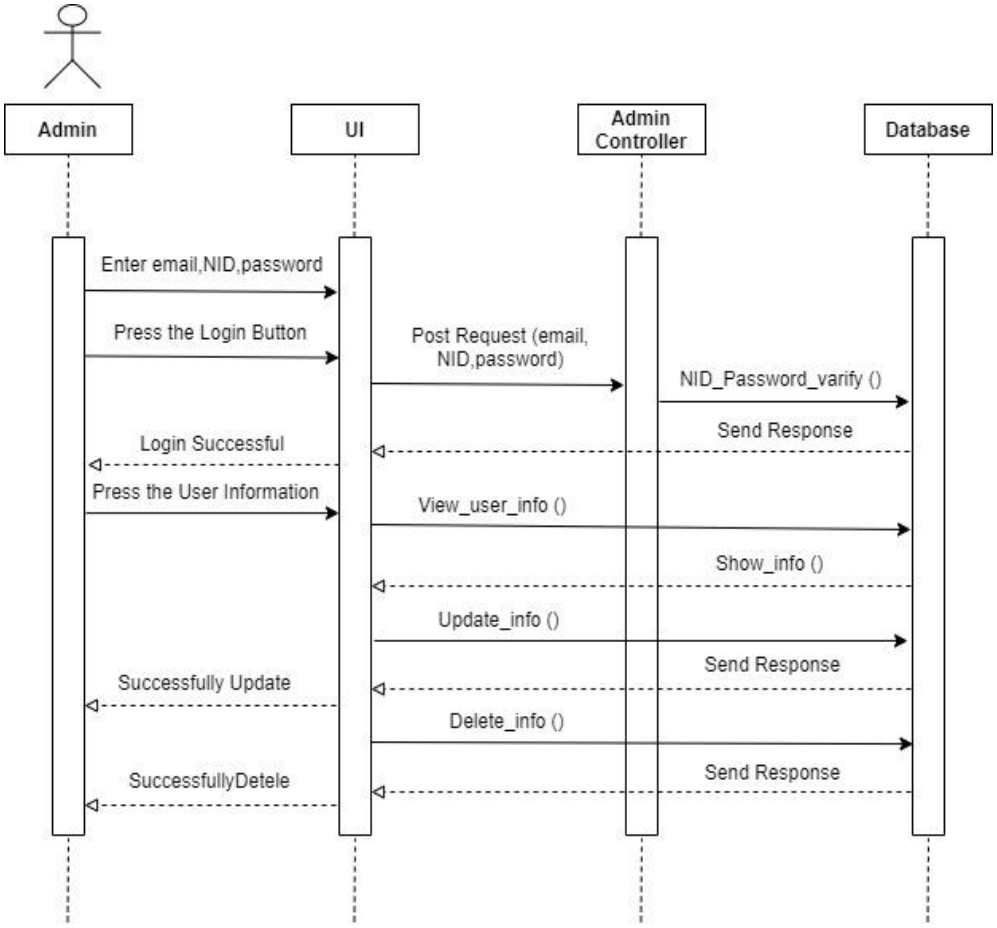


Figure 3.3.9: Sequence Diagram of Monitor User Information

3.4 Class Diagram:

To Describe the system properly and its process with attributes and methods.

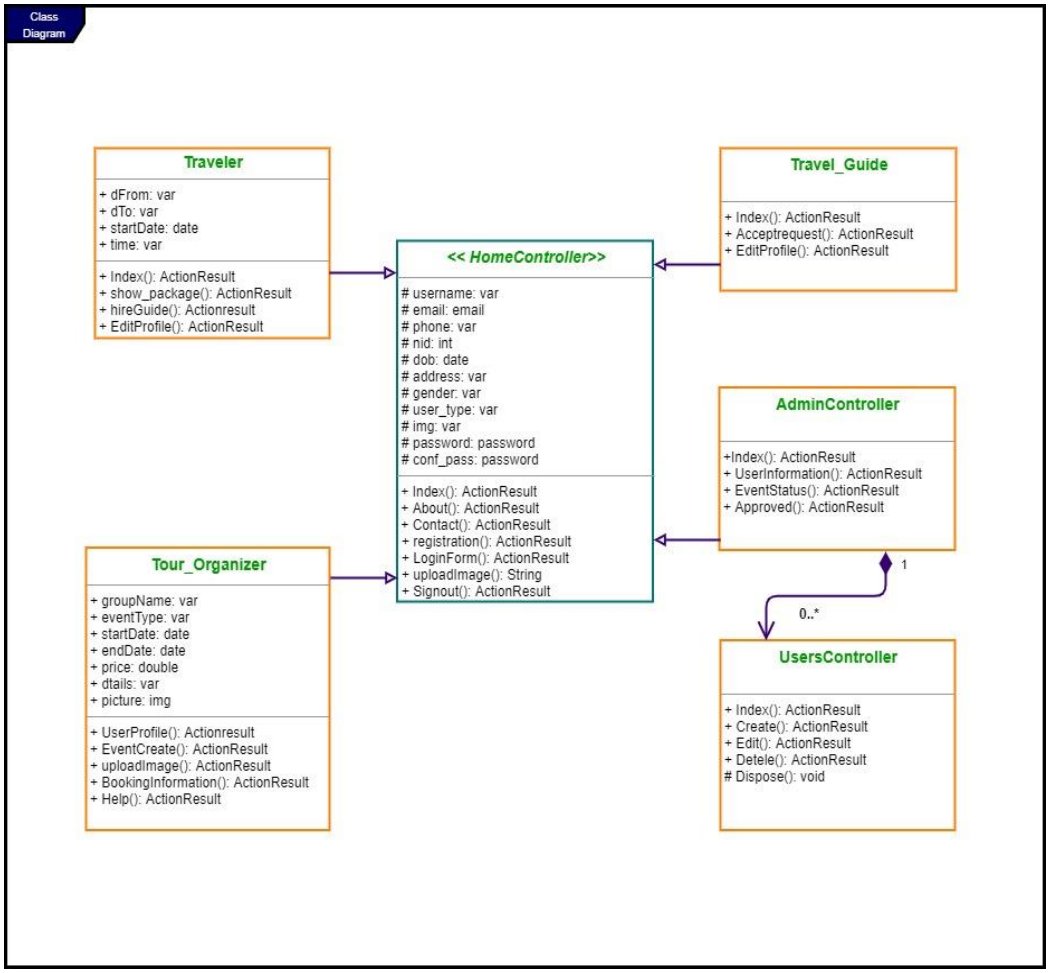


Figure 3.4.1 : Class Diagram

3.5 Entity Relationship Diagram:

To describe the system database with its table, primary key, foreign key and the relation between the tables.

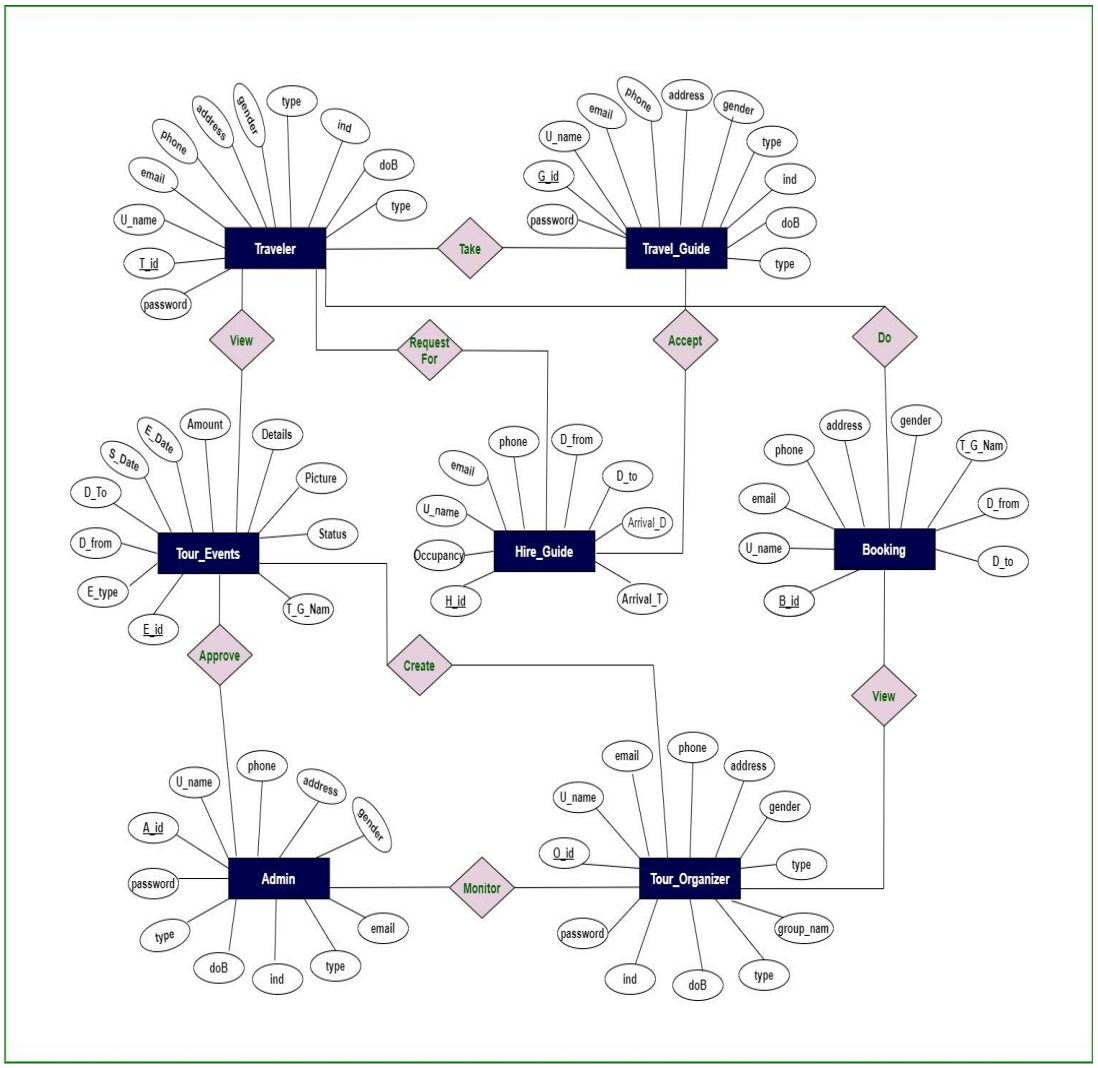


Figure 3.5.1: Entity Relationship Diagram

Chapter Four

Technology and Tools

4.1 User Interface Technology

User interface (UI) is everything designed into a system view that a person's associates with this system may like the interface of this system.

4.1.1 Technology

- Programming language: C#
- Web server: IIS Express
- Design: html, CSS, bootstrap, JavaScript, JQuery
- Framework: ASP.NET MVC

4.1.2 Tools

- Microsoft Visual Studio
- Sublime Text 3

4.2 Database Technology

Database technology take information and store, organize, and process it in a way that enables users to easily and intuitively go back and find details they are searching for.

4.2.1 Technology

- Database server: SQL Server Management Studio

4.2.2 Tool

- SQL Server Management Studio

Chapter Five

Implementation

5.1 Hardware & Software Specifications

In this stage I want to describe what's needed to build this application.

5.1.1 Hardware Requirements

- PROCESSOR: Dual Core or above
- RAM: 2GB or above
- Cache Memory: 2MB or above
- HDD: 20GB or above

5.1.2 Software Requirements

- **IDE**: Visual Studio
- **Database:** SQL Server Management Studio
- **Web-Server:** IIS express

Chapter Six

User Manual

6.1 Home Page

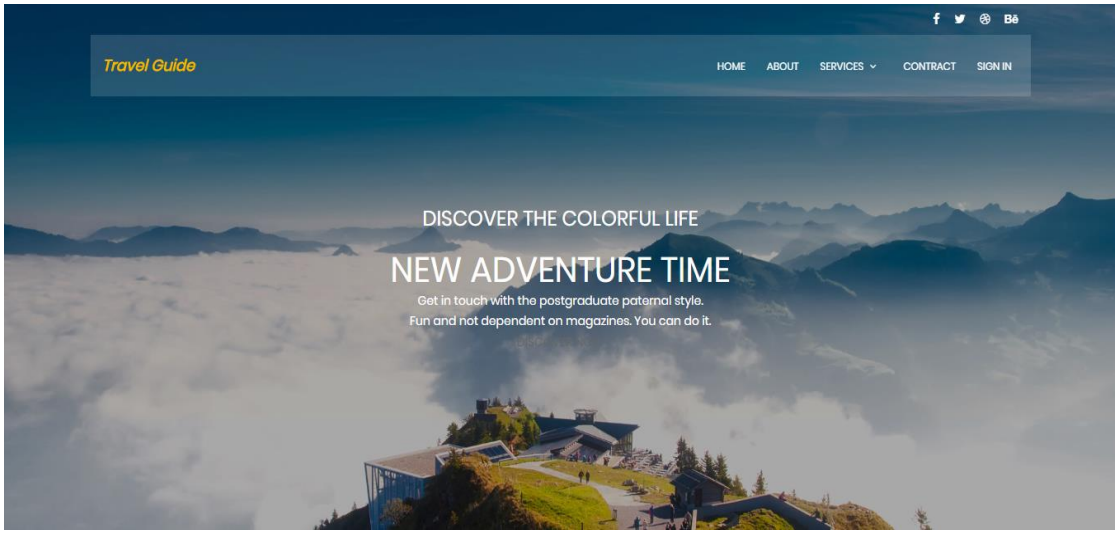


Figure 6.1: UI Design of Home Page

6.2 Registration

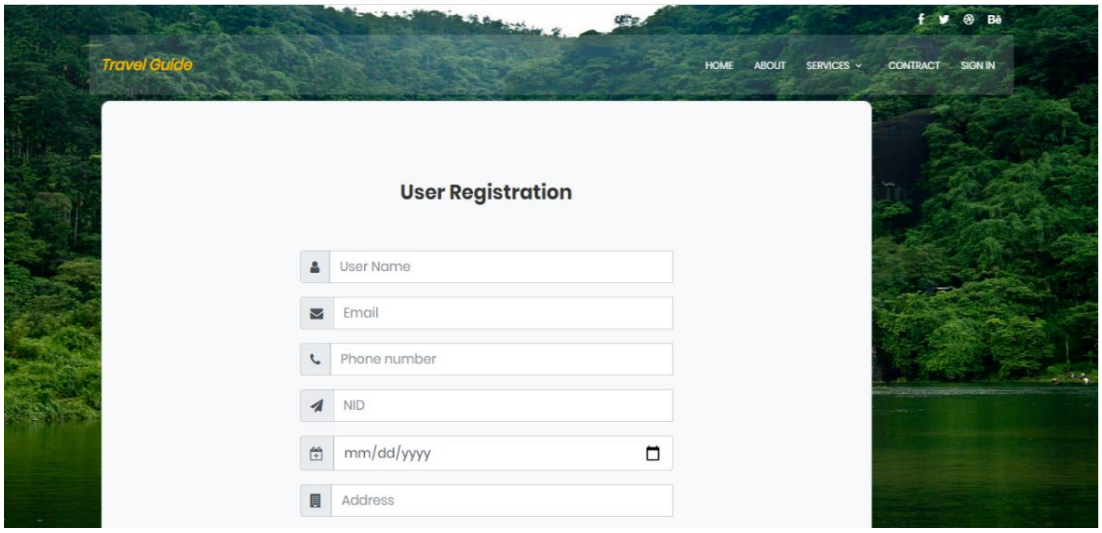


Figure 6.2: UI Design of Registration Page

6.3 Login

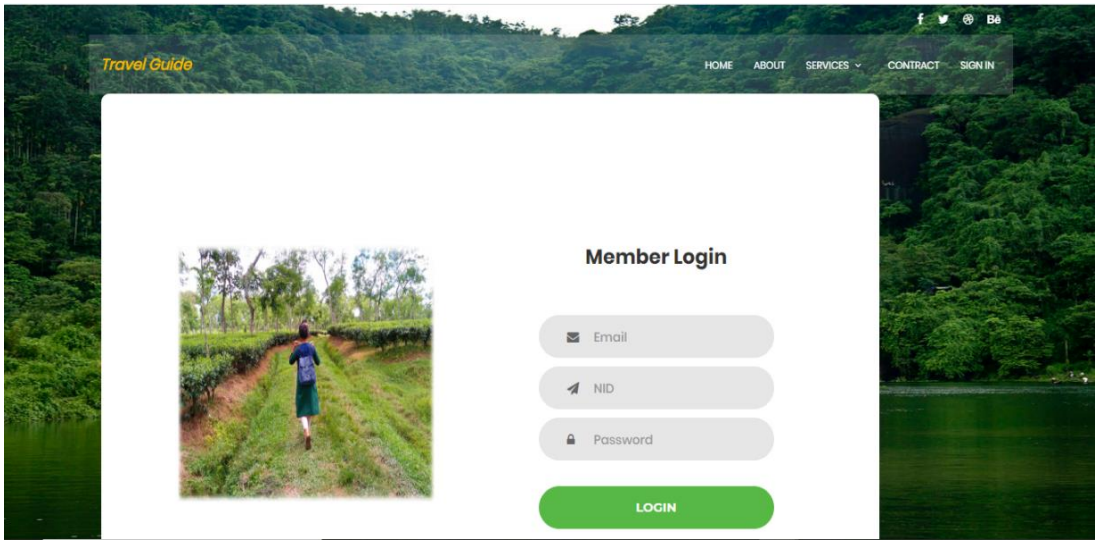


Figure 6.3: UI Design of Login Page

6.4 About

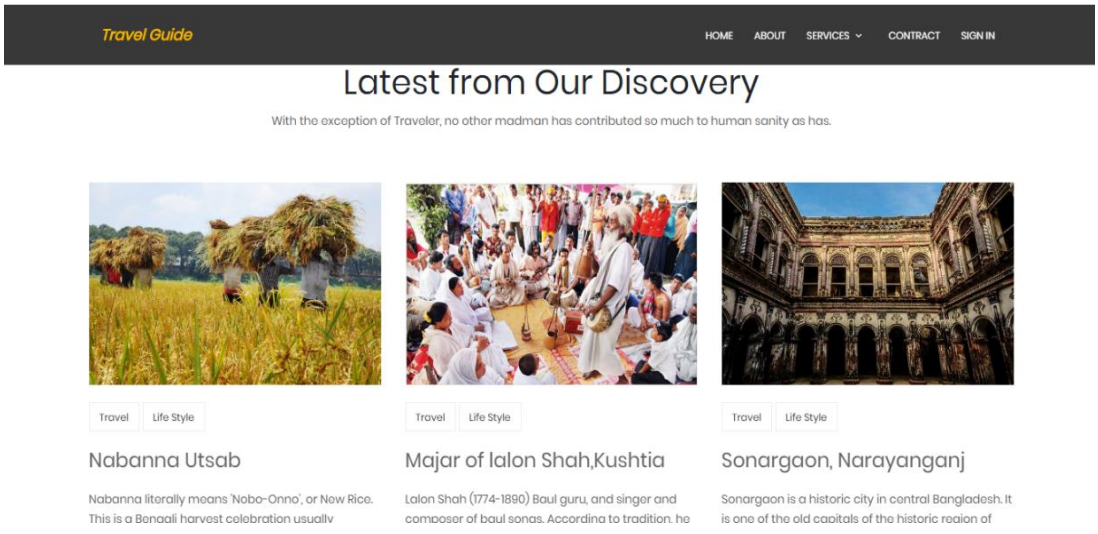


Figure 6.4: UI / UX Design of About Page

6.5 Contract

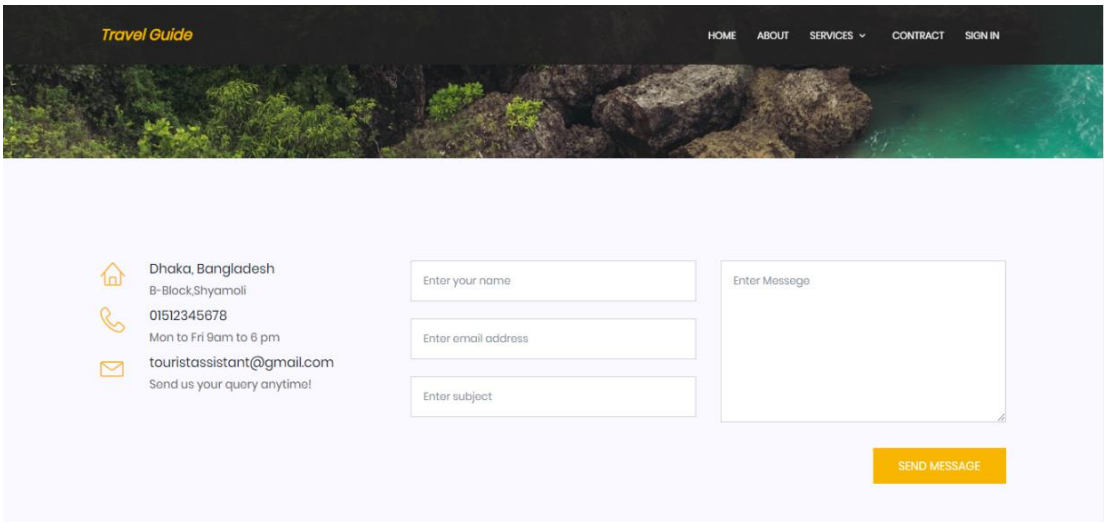


Figure 6.5: UI / UX Design of Contract Page

6.6 Hire Guide

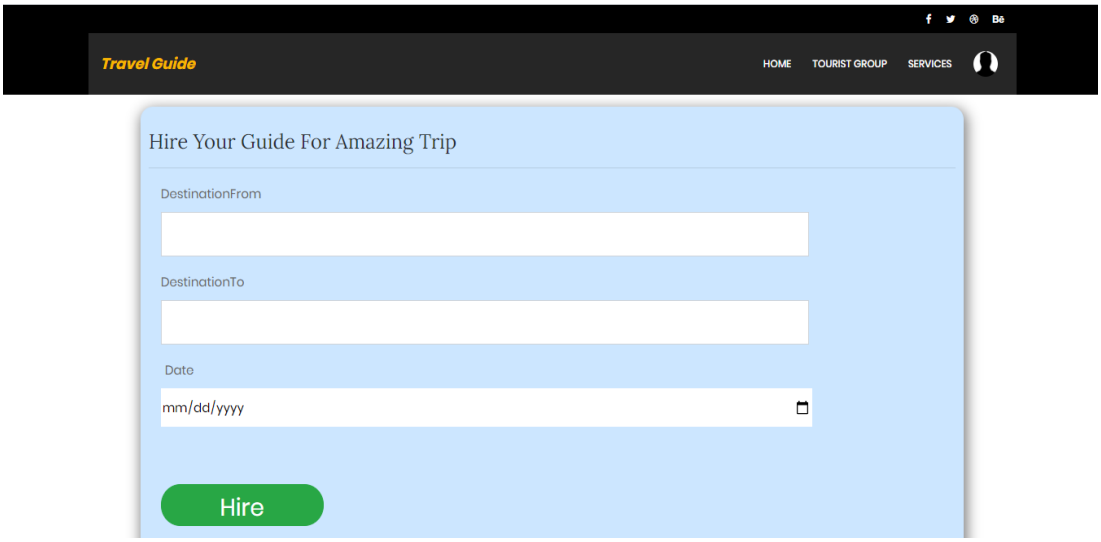


Figure 6.6: UI / UX Design of Hire Guide Page

6.7 Accept Hire Request

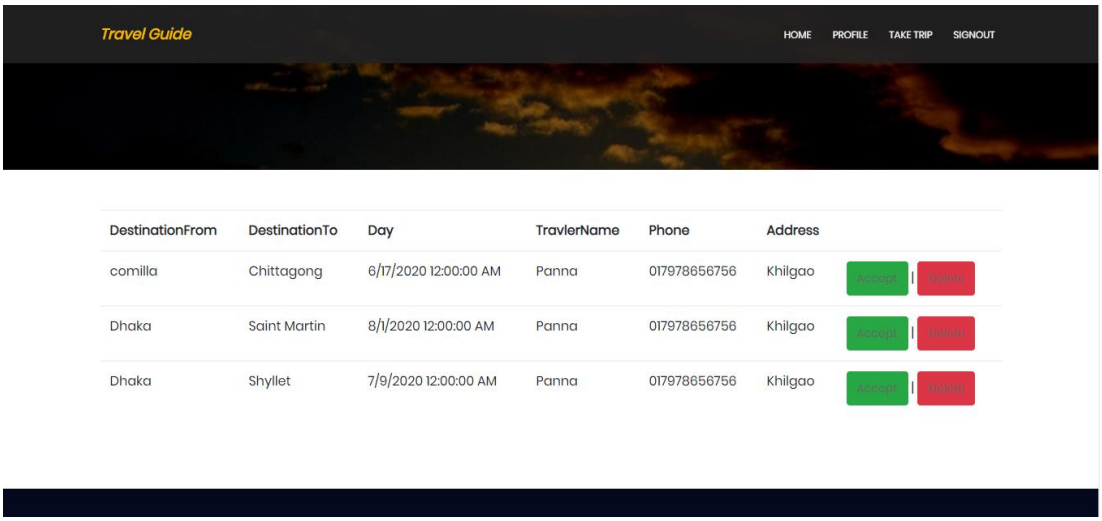


Figure 6.7: UI / UX Design of Accept Hire Request Page

6.8 Create Tour Event

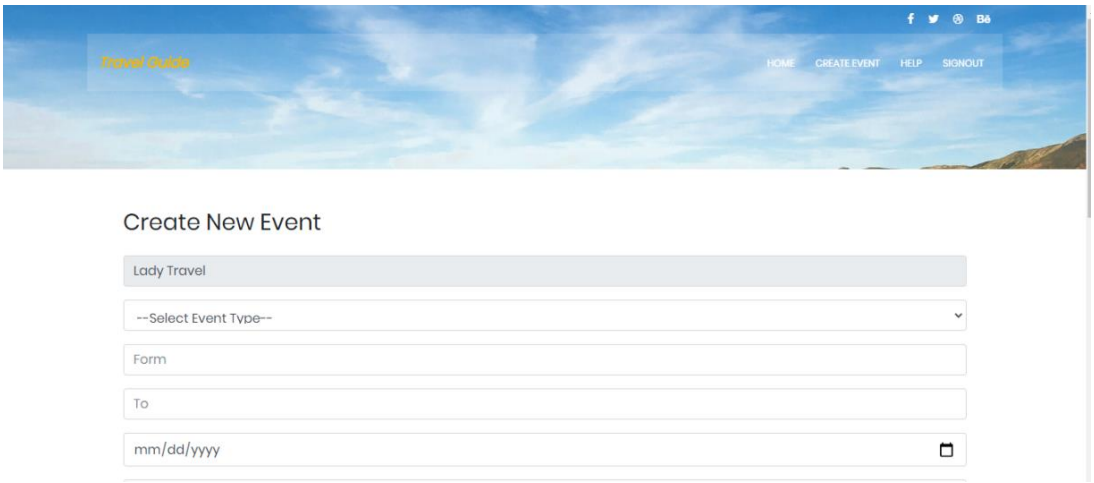


Figure 6.8: UI / UX Design of Create Tour Event Page

6.9 Approve Created Event

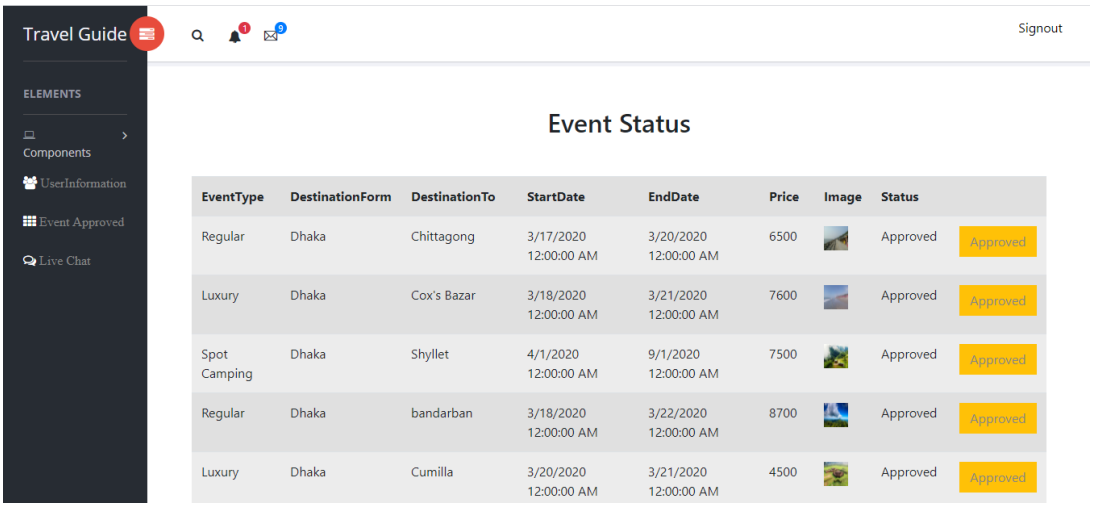


Figure 6.9: UI / UX Design of Approved Event Page

6.10 Show and Booking Package

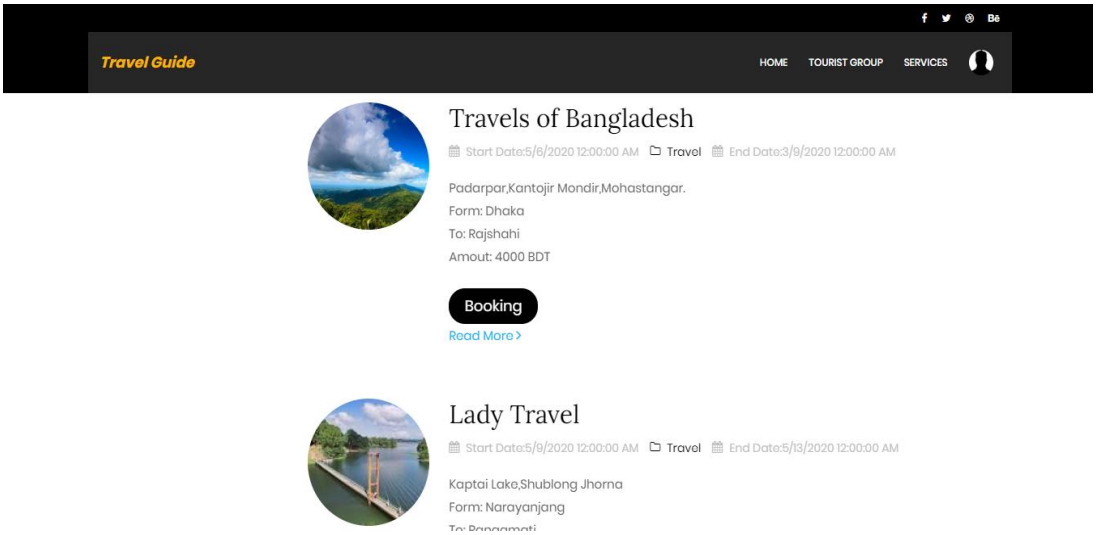


Figure 6.10: UI / UX Design of Show and Booking Page

6.11 User Booing Information

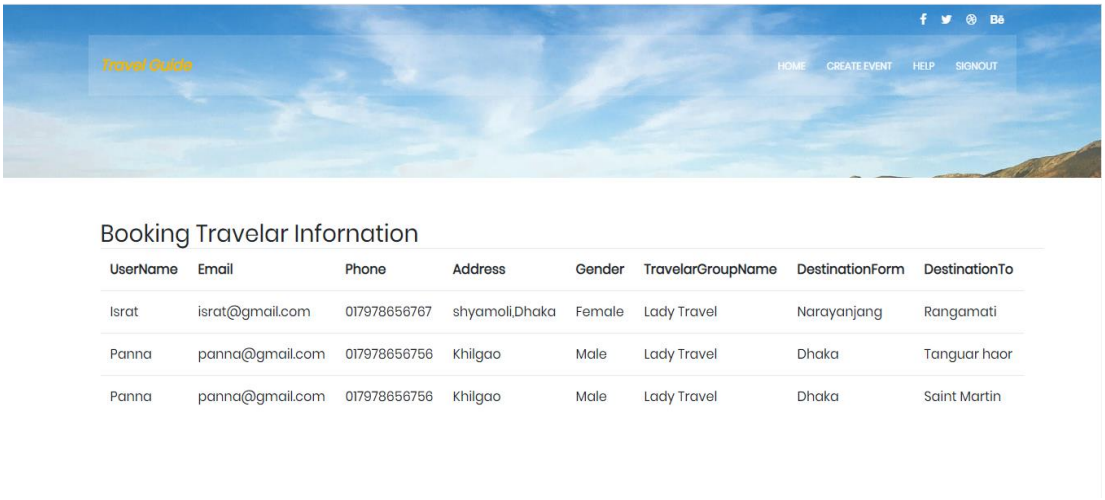


Figure 6.11: UI / UX Design of User Booking Information Page

6.12 Help

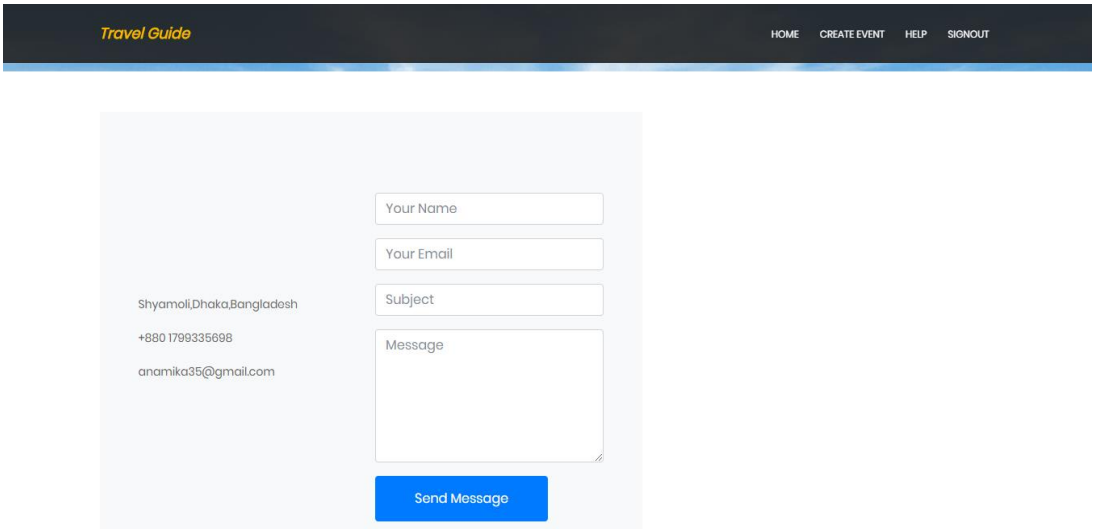


Figure 6.12: UI / UX Design of Help Page

6.13 User Information

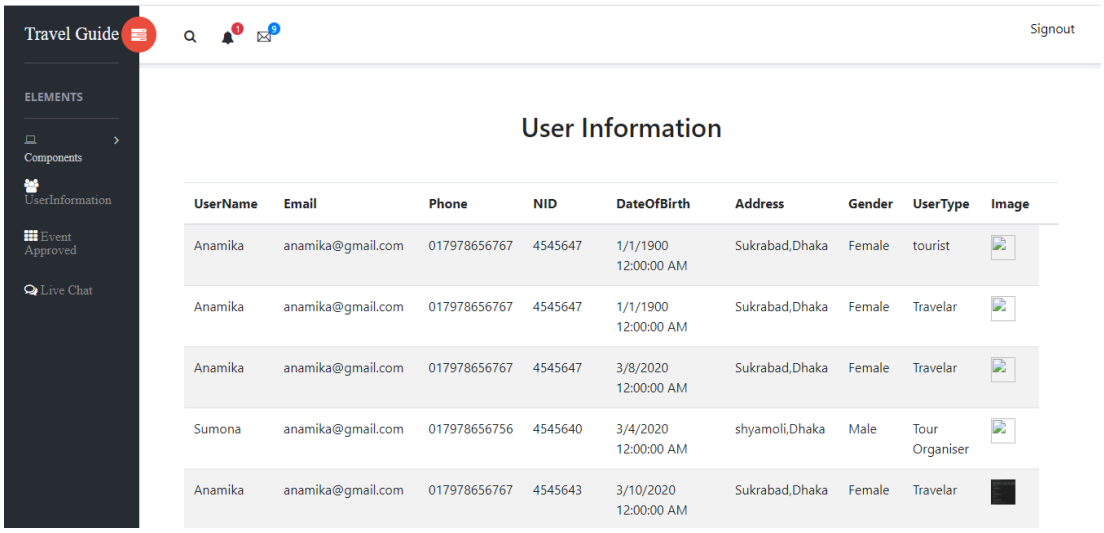


Figure 6.13: UI / UX Design of User Information Page

Chapter Seven

System Testing

7.1 Testing

The testing of the software was done in full manual end user data flow testing style. The testing approach is narrated here omitting the technical details.

Two common type of testing is black box testing and white box testing. Black box testing is also called functional testing. In this stage we test only functionality, input, and output. White box testing is structure level testing. For this project I have used black box testing method.

7.2 Testing Strategy

A testing strategy is a general approach to the testing process rather than a method of devising particular system or component tests. Different testing strategies may be adopted depending on the type of system to be tested and the development process used.

In this project I have used Black Box Testing method.

7.3 Test approach

A test approach is the test strategy implementation of a project, defines how testing would be carried out. Test approach has two techniques:

- **Proactive -** An approach in which the test design process is initiated as early as possible in order to find and fix the defects before the build is created.
- **Reactive -** An approach in which the testing is not started until after design and coding are completed.

7.3.1 Black Box Testing

Black box trying out additionally referred to as purposeful checking out that ignores the inner mechanism of a device or element and specializes in the outputs generated in response to selected inputs and execution conditions. We've decided to perform equivalence partitioning and Boundary value evaluation for this gadget.

7.3.2 White Box Testing

White box checking out is a software trying out method wherein the inner structure or implementation of the item being examined is understood to the tester. The tester chooses inputs to workout paths thru the code and determines the proper outputs. Programming understand how and the implementation know-how is crucial.

7.4 Pass / Fail Criteria

The entrance criteria for each phase of testing must be met before the next phase can commence. Now the criteria for pass and fail are given below.

- According to the given scenario the expected result need to take place then the scenario will be considered as pass otherwise that criteria should be failed.
- If an item tested 10 times, 9 times perfectly worked and single time do not work properly then it will consider as fail case.
- System crash will be considered as fail case.
- After submitting a query in the system, if expected page won’t appear then it will be considered as fail case.

7.5 Testing Environment

Testing environment is a setup of software and hardware for the testing teams to execute test cases. In other words, it supports test execution with hardware, software and network configured.

For test environment, key area to set up includes.

- System and applications
- Test data
- Database server
- Front end running environment
- Client operating system
- Browser, Network
- Hardware includes Server Operating system
- Documentation require like reference documents/configuration guides/installation guides/ user manuals.

7.6 Use Case Testing

Use Case Testing is a functional Black Box Testing technique that helps to identify test scenarios that exercise the whole system on each transaction basis from start to finish.

7.7 Test Case

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

7.7.1 Test case for User Registration

Table 7.7.1: Test case for User Registration (Pass)

Test Case ID: TC 01			**Module Name: Registration**			
Sub Module: User Registration			**Test Designed by:** Anamika			
Test Priority (Low/Medium/High): High			**Test Designed date:** 20/04/20			
Test Title: Student Registration with valid information			**Test Executed by:** Anamika			
Description: Test the system's on registration page			**Test Execution date:** 20/04/20			
Pre-conditions: The user navigate to registration page and input the required filled. And click on the Register button.						
Step	**Test Steps**	**Test Data**	**Code Module**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	**Navigate to Registration Page**	Click on Registration Tab		User can be registered successfully	Redirect to home page with User access part.	Pass
2	**Name**	Panna				
3	**NID**	09876				
4	**E-mail**	panna@gmail.com				
5	**Phone**	017978656756				
6	**Password**	panna				
7	**Confirm Password**	panna				
Post-conditions: If the user information's are valid then the information will saved in the database otherwise show the invalid fields.						

Table 7.7.2: Test case for User Registration (Fail)

Test Case ID: TC 02			**Module Name: Registration**			
Sub Module: User Registration			**Test Designed by:** Anamika			
Test Priority (Low/Medium/High): High			**Test Designed date:** 20/04/20			
Test Title: Student Registration with valid information			**Test Executed by:** Anamika			
Description: Test the system's on registration page			**Test Execution date:** 20/04/20			
Pre-conditions: The user navigate to registration page and input the required filled. And click on the Register button.						
Step	**Test Steps**	**Test Data**	**Code Module**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	**Navigate to Registration Page**	Click on Registration Tab		User cannot be registered successfully	Redirect to home page with User access part error.	Fail
2	**Name**	Panna				
3	**NID**	09876				
4	**E-mail**	panna@gmail.com				
5	**Phone**	Null				
6	**Password**	panna				
7	**Confirm Password**	panna				
Post-conditions: If the user information's are valid then the information will saved in the database otherwise show the invalid fields.						

7.7.2 Test Case for User Login

Table 7.7.3: Test Case for User Login (pass)

Test Case ID: TC 03				**Module Name: Login**		
Sub Module: User Login				**Test Designed by:** Anamika		
Test Priority (Low/Medium/High): High				**Test Designed date:** 21/04/20		
Test Title: User Login with valid email.				**Test Executed by:** Anamika		
Description: Test the system's Login page				**Test Execution date:** 21/04/20		
Pre-conditions: The user has valid email and password. The current email is **panna**@gmail.com **and password panna** The system navigates to Login page. And click on the Login button.						
Step	**Test Steps**	**Test Data**	**Code Module**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	**Navigate to Login Page**	Click on Login Tab		User login successfully	Home Page successfully.	Pass
2	**Input Email and it must be Unique**	**panna@gmail.com**				
3	**Input Password**	panna				
4	**Click on Signup Button**					
Post-conditions: Student is validated with database and successfully login to account. The account session details are logged in database.						

Table 7.7.4: Test Case for User Login (Fail)

<table>
<tr><td colspan="4">Test Case ID: TC 04</td><td colspan="3">Module Name: Login</td></tr>
<tr><td colspan="4">Sub Module: User Login</td><td colspan="3">Test Designed by: Anamika</td></tr>
<tr><td colspan="4">Test Priority (Low/Medium/High): High</td><td colspan="3">Test Designed date: 21/04/20</td></tr>
<tr><td colspan="4">Test Title: User Login with valid email/user name and password</td><td colspan="3">Test Executed by: Anamika</td></tr>
<tr><td colspan="4">Description: Test the system’s Login page</td><td colspan="3">Test Execution date: 21/04/20</td></tr>
<tr><td colspan="7">Pre-conditions: The user has valid email and password.
The current email is panna@gmail.com and password panna.
The system navigates to Login page. And click on the Login button.</td></tr>
<tr><td>Step</td><td>Test Steps</td><td>Test Data</td><td>Code Module</td><td>Expected Result</td><td>Actual Result</td><td>Pass/Fail</td></tr>
<tr><td>1</td><td>Navigate to Login Page</td><td>Click on Login Tab</td><td rowspan="4"></td><td rowspan="4">User should not be able to login successfully.</td><td rowspan="4">Redirect to the login page with error message</td><td rowspan="4">Fail</td></tr>
<tr><td>2</td><td>Input Email and it must be Unique</td><td>Panna.gmail.com</td></tr>
<tr><td>3</td><td>Input Password</td><td>panna</td></tr>
<tr><td>4</td><td>Click on Signup</td><td></td></tr>
<tr><td colspan="7">Post-conditions: User is validated with database and successfully login to account.
The account session details are logged in database.</td></tr>
</table>

7.7.3 Test Case for Hire Travel Guide

Table 7.7.5: Test Case for Hire Travel Guide (Pass)

<table>
<tr><td colspan="4">Test Case ID: TC 05</td><td colspan="3">Module Name: Hire Guide</td></tr>
<tr><td colspan="4">Sub Module: Hire Travel Guide</td><td colspan="3">Test Designed by: Anamika</td></tr>
<tr><td colspan="4">Test Priority (Low/Medium/High): High</td><td colspan="3">Test Designed date: 22/04/20</td></tr>
<tr><td colspan="4">Test Title: Hire guide with valid information.</td><td colspan="3">Test Executed by: Anamika</td></tr>
<tr><td colspan="4">Description: Test the system’s travel guide hire or booking.</td><td colspan="3">Test Execution date: 22/04/20</td></tr>
<tr><td colspan="7">Pre-conditions: The user has valid email and password.
The current email is panna@gmail.com and password panna.
The system navigates to Hire Guide page. Fill the all information. And click on the Hire button.</td></tr>
<tr><td>Step</td><td>Test Steps</td><td>Test Data</td><td>Code Module</td><td>Expected Result</td><td>Actual Result</td><td>Pass/Fail</td></tr>
<tr><td>1</td><td>Navigate to Hire Guide Page</td><td>Click on Hire Button</td><td rowspan="2"></td><td rowspan="2">Hire successfully.</td><td rowspan="2">Successfully Hire Travel Guide</td><td rowspan="2">Pass</td></tr>
<tr><td>2</td><td>Input information and it must be filled.</td><td></td></tr>
<tr><td>3</td><td>Click on Signup</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="7">Post-conditions: The information validated and stored in database and successfully hire travel guide.</td></tr>
</table>

Table 7.7.6: Test Case for Hire Travel Guide (Fail)

Test Case ID: TC 06	**Module Name: Hire Guide**
Sub Module: Hire Travel Guide	**Test Designed by:** Anamika
Test Priority (Low/Medium/High): High	**Test Designed date:** 22/04/20
Test Title: Required information did not store.	**Test Executed by:** Anamika
Description: Test the system's **travel guide hire or booking.**	**Test Execution date:** 22/04/20

Pre-conditions: The user has valid email and password.
The current email is **panna@gmail.com and password panna**.
The system navigates to **Hire Guide** page. Fill the all information. And click on the **Hire** button.

Step	**Test Steps**	**Test Data**	**Code Module**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	**Navigate to Hire Guide Page**	Click on Hire Button		Did not provide required information.	Hiring is deny.	Fail
2	**Input information and it must be filled.**					
3	**Click on Signup**					

Post-conditions: The information validated and did not stored in database.

7.7.4 Test Case for Booking Package

Table 7.7.7: Test Case for Booking Package (Pass)

Test Case ID: TC 07				**Module Name: Booking Package**		
Sub Module: Booking Travel Package				**Test Designed by:** Anamika		
Test Priority (Low/Medium/High): High				**Test Designed date:** 23/04/20		
Test Title: Booking with select package.				**Test Executed by:** Anamika		
Description: Test the system's **booking package.**				**Test Execution date:** 23/04/20		
Pre-conditions: The user has valid email and password. The current email is **panna@gmail.com** and password **panna**. The system navigates to **Tour Package** page. Select package and click on the **Booking** button.						
Step	**Test Steps**	**Test Data**	**Code Module**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	**Navigate to Tour Package Page**	Click on Booking Button		Booking successfully.	Successfully complete booking	Pass
2	**Input information and it must be filled.**					
Post-conditions: The information stored in database and show successfully **Booking Tour Package**						

Table 7.7.8: Test Case for Booking Package (Fail)

Test Case ID: TC 08	**Module Name: Booking Package**
Sub Module: Booking Travel Package	**Test Designed by:** Anamika
Test Priority (Low/Medium/High): High	**Test Designed date:** 23/04/20
Test Title: Did not select tour package for Booking.	**Test Executed by:** Anamika
Description: Test the system's **booking package.**	**Test Execution date:** 23/04/20

Pre-conditions: The user has valid email and password.
The current email is **panna@gmail.com** and password **panna**.
The system navigates to **Tour Package** page. And click on the **Booking** button.

Step	**Test Steps**	**Test Data**	**Code Module**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	**Navigate to Tour Package Page**	Click on Booking Button		Tour Package did not selected.	Booking is deny.	Fail
2	**Input information and it must be filled.**					
3	**Click on Signup**					

Post-conditions: The information did not stored in database and show **Error massage.**

7.7.5 Test Case for Accept Request

Table 7.7.9: Test Case for Accept Hiring Request (Pass)

Test Case ID: TC 09				**Module Name: Accept Hiring Request**		
Sub Module: Accept Request				**Test Designed by:** Anamika		
Test Priority (Low/Medium/High): High				**Test Designed date:** 24/04/20		
Test Title: Accept Request with valid information.				**Test Executed by:** Anamika		
Description: Test the system's **Accepting Travel Guide Hire Request**				**Test Execution date:** 24/04/20		
Pre-conditions: The user has valid email, NID and password. The current email is **sourav@gmail.com** and password **sourav**. The system navigates to **Travel Guide** page. And click on the **Accept** button.						
Step	**Test Steps**	**Test Data**	**Code Module**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	**Navigate to Travel Guide Page**	Click on Accept Button		Accept request successfully.	Successfully Accept Hiring Request	Pass
2	**Input information and it must be filled.**					
3	**Click on Signup**					
Post-conditions: The information validated and stored in database and successfully **accept hiring request**.						

Table 7.7.10: Test Case for Accept Hiring Request (Fail)

Test Case ID: TC 10			**Module Name: Accept Hiring Request**			
Sub Module: Accept Request			**Test Designed by:** Anamika			
Test Priority (Low/Medium/High): High			**Test Designed date:** 24/04/20			
Test Title: Required information did not store.			**Test Executed by:** Anamika			
Description: Test the system's **Accepting Travel Guide Hire Request.**			**Test Execution date:** 24/04/20			
Pre-conditions: The user has valid email, NID and password. The current email is **sourav@gmail.com and password sourav**. The system navigates to **Travel Guide** page. And click on the **Accept** button.						
Step	**Test Steps**	**Test Data**	**Code Module**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	**Navigate to Travel Guide Page**	Click on Accept Button		Did not accept request.	Accepting is deny.	Fail
2	**Input information and it must be filled.**					
3	**Click on Signup**					
Post-conditions: Did not accept request, store in database.						

7.7.6 Test Case for Create Tour Package

Table 7.7.11: Test Case for Create Tour Package (Pass)

<table>
<tr><td colspan="4">Test Case ID: TC 11</td><td colspan="3">Module Name: Create Tour Package</td></tr>
<tr><td colspan="4">Sub Module: Create Package</td><td colspan="3">Test Designed by: Anamika</td></tr>
<tr><td colspan="4">Test Priority (Low/Medium/High): High</td><td colspan="3">Test Designed date: 25/04/20</td></tr>
<tr><td colspan="4">Test Title: Creating package with the information.</td><td colspan="3">Test Executed by: Anamika</td></tr>
<tr><td colspan="4">Description: Test the system’s creating tour package.</td><td colspan="3">Test Execution date: 25/04/20</td></tr>
<tr><td colspan="7">Pre-conditions: The user has valid email, NID and password.
The current email is sumona@gmail.com and password sumona.
The system navigates to Create Event page. Fill in the information and click on the Create button.</td></tr>
<tr><td>Step</td><td>Test Steps</td><td>Test Data</td><td>Code Module</td><td>Expected Result</td><td>Actual Result</td><td>Pass/Fail</td></tr>
<tr><td>1</td><td>Navigate to Create Event Page</td><td>Click on Create Event Button</td><td rowspan="2"></td><td rowspan="2">Create Package successfully</td><td rowspan="2">Successfully Create package.</td><td rowspan="2">Pass</td></tr>
<tr><td>2</td><td>Input information and it must be filled.</td><td></td></tr>
<tr><td>3</td><td>Click on Signup</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="7">Post-conditions: The information stored in database and show successfully create tour package</td></tr>
</table>

Table 7.7.12: Test Case for Create Tour Package (Fail)

<table>
<tr><td colspan="4">Test Case ID: TC 12</td><td colspan="3">Module Name: Create Tour Package</td></tr>
<tr><td colspan="4">Sub Module: Create Package</td><td colspan="3">Test Designed by: Anamika</td></tr>
<tr><td colspan="4">Test Priority (Low/Medium/High): High</td><td colspan="3">Test Designed date: 25/04/20</td></tr>
<tr><td colspan="4">Test Title: Did not provide required information for creating event.</td><td colspan="3">Test Executed by: Anamika</td></tr>
<tr><td colspan="4">Description: Test the system’s creating package.</td><td colspan="3">Test Execution date: 25/04/20</td></tr>
<tr><td colspan="7">Pre-conditions: The user has valid email, NID and password.
The current email is sumona@gmail.com and password sumona.
The system navigates to Create Event page. And click on the Create button.</td></tr>
<tr><td>Step</td><td>Test Steps</td><td>Test Data</td><td>Code Module</td><td>Expected Result</td><td>Actual Result</td><td>Pass/Fail</td></tr>
<tr><td>1</td><td>Navigate to Create Event Page</td><td>Click on Create Button</td><td rowspan="2"></td><td rowspan="2">Did not provide required information.</td><td rowspan="2">Creating is deny.</td><td rowspan="2">Fail</td></tr>
<tr><td>2</td><td>Input information and it must be filled.</td><td></td></tr>
<tr><td>3</td><td>Click on Signup</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="7">Post-conditions: The information did not stored in database and show Error massage.</td></tr>
</table>

7.7.7 Test Case for Approve Tour Package

Table 7.7.13: Test Case for Approve Tour Package (Pass)

<table>
<tr><td colspan="4">Test Case ID: TC 13</td><td colspan="3">Module Name: Approve Tour Package</td></tr>
<tr><td colspan="4">Sub Module: Approve Package</td><td colspan="3">Test Designed by: Anamika</td></tr>
<tr><td colspan="4">Test Priority (Low/Medium/High): High</td><td colspan="3">Test Designed date: 26/04/20</td></tr>
<tr><td colspan="4">Test Title: Approve package with checking the information.</td><td colspan="3">Test Executed by: Anamika</td></tr>
<tr><td colspan="4">Description: Test the system’s approved tour package.</td><td colspan="3">Test Execution date: 26/04/20</td></tr>
<tr><td colspan="7">Pre-conditions: The user has valid email, NID and password.
The current email is danamika@gmail.com and password danamika.
The system navigates to Approve Event page. Check package and click on the Approved button.</td></tr>
<tr><td>Step</td><td>Test Steps</td><td>Test Data</td><td>Code Module</td><td>Expected Result</td><td>Actual Result</td><td>Pass/Fail</td></tr>
<tr><td>1</td><td>Navigate to Approve Event Page</td><td>Click on Approved Button</td><td rowspan="2"></td><td rowspan="2">Approved Package successfully.</td><td rowspan="2">Successfully Approved.</td><td rowspan="2">Pass</td></tr>
<tr><td>2</td><td>Input information and it must be filled.</td><td></td></tr>
<tr><td>3</td><td>Click on Signup</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="7">Post-conditions: The information stored in database and show successfully approve tour package</td></tr>
</table>

Table 7.7.14: Test Case for Approve Tour Package (Fail)

Test Case ID: TC 14			**Module Name: Approve Tour Package**			
Sub Module: Approve Package			**Test Designed by:** Anamika			
Test Priority (Low/Medium/High): High			**Test Designed date:** 26/04/20			
Test Title: Did not check for approve tour event.			**Test Executed by:** Anamika			
Description: Test the system's **approved tour package.**			**Test Execution date:** 26/04/20			
Pre-conditions: The user has valid email, NID and password. The current email is **danamika@gmail.com** and password **danamika**. The system navigates to **Approve Event** page. And click on the **Approved** button.						
Step	**Test Steps**	**Test Data**	**Code Module**	**Expected Result**	**Actual Result**	**Pass/Fail**
1	**Navigate to Approve Event Page**	Click on Approved Button		Did not check tour package information	Approve is deny.	Fail
2	**Input information and it must be filled.**					
3	**Click on Signup**					
Post-conditions: The information did not stored in database and show **Error massage.**						

7.8 Test Report

Generally, this is a communication to establish transparency to the QA team's activities of the day during the test cycle – includes both defect information and test case run information.

Total unit of test case sample are 14. Some test cases are succeeding in first iteration and some are in second. The succession percent are shown in following table:

Table 7.8.1: Test Report

Number of Unit Test Case	100% Success in first iteration	Less than 100%	Total Succession %
Total : 14	8	3	80%
Total : 14	10	0	100%

Chapter Eight

Conclusion

8.1 Project Summary

This task has been beginning from May. From that starting time I have to work difficult to recognize the requirement. After that I proposed a layout to them via assist of my supervisor. I started out to develop the challenge. From then I gradually expand the venture. I assume storing the statistics in database is very vital. That's why I did this first and made a courting with the tables. Before that I design the UI. This task's UI is very simple and smooth which may be very help for the user's revel in. Then I started coding and executing the assignment. If I did now not check this challenge, there'll stay some bug on this challenge so that you can spoil the entire mission.

8.2 Limitations

- I did not use payment gateway.
- Email transformation.
- System process slow
- Not highly secure

8.3 Future Improvement

This application avoids the manual work and the problems concern with it. Well I have worked hard in order to present the website. Still, I found out that the project can be done in a better way. Due to time and technology concern I am unable to deploy payment gateway other small features. For that reason, I am going to develop new features like following:

- Send Message, call on the Phone
- Send Notification to the mail
- Payment gateway

References

[1] About HTML & CSS methodology, available at < https://www.w3schools.com>; last accessed on 17.08.2019, Time: 6:15pm

[2] About JavaScript methodology, available at < https://www.javascriptmancy.com >; last accessed on 25.08.2019, Time: 8:15pm

[3] Template Choose, available at< https://colorlib.com/wp/templates/>; last accessed on 06.09.2019, Time: 9:30pm

[4] About Bootstrap methodology, available at < https://getbootstrap.com >; last accessed on 25.09.2019, Time: 8:15pm

[5] About database management, Available at< https:// https://www.sqlshack.com/overview-of-microsoft-sql-server-management-studio-ssms/ />; last accessed on 29.10.2019; Time: 1.00am

[6] About c#, available at< https://www.w3schools.com/cs/>; last accessed on 06.11.2019; Time: 8.00pm

[7] About ASP.NET MVC Framework, available at< https://www.tutorialspoint.com/mvc_framework/mvc_framework_architecture.htm>; last access on 05.12.2019; Time: 5.00 pm

[8] Learn about Sequence Diagram, available at< https://www.geeksforgeeks.org/unified-modeling-language-uml-sequence-diagrams/ >; last access on 05.01.2020, Time: 9:40pm

[9] Learn about Test Case, available at< https://www.softwaretestinghelp.com/test-case-template-examples/ />; last accessed on 07.03.2020, Time: 8:00pm