



Second Year Group Project 2020 Proposal

Travel Management System

Details of Project Supervisor, Co-supervisor

Proposed Project Supervisor (Academic Staff of UCSC):

Name of the supervisor : Ms. Kulani Sumanasekara
Signature of the supervisor : Kulani Sumanasekara
Date : 25.05.2020

Proposed Project Co-Supervisor (Assigned by Course Coordinator):

Name of the co-supervisor : Ms. Gayani Rupasinghe
Signature of the co-supervisor: Gayani Rupasinghe
Date : 25.05.2020

Degree Program: Information Systems Group Number: 07

Group members:

- Asini Pathmila Silva (18020798)
- Ruwanthi Hemachandra (18020291)
- Hansaka Sandaruwan (18020704)
- Sachini Maleesha (18020356)

Table of the content

Details of Project Supervisor, Co-supervisor	1
Table of the content	2
List of the figures.....	4
List of tables	6
1. Problem Statement	7
2. Project Goals.....	11
3. Project Scope	11
3.1 Identify the project scope and clear boundaries	11
4. Project Objectives	12
5. Feasibility Study	13
5.1 Technical Feasibility.....	13
5.2 Economic feasibility	14
5.3 Legal and Ethical feasibility.....	14
5.4 Schedule feasibility.....	14
5.5 Operational feasibility.....	15
6. Deliverables of the project.....	16
7. Methodology.....	16
7.1 Development Model	16
7.2 Reasons to select this development model.....	17
8. Project constraints and assumptions identification	18
8.1 Project constraints	18
8.1.1 Cost.....	18
8.1.2 Scope	18
8.1.3 Quality	18
8.1.4 Customer Satisfaction.....	18
8.1.5 Risk.....	18
8.1.6 Resources.....	18
8.1.7 Time	18
8.2 Assumptions	19
9. Functional Requirement Identification.....	20
9.1 Component diagram	22

9.2 Use case diagram.....	23
9.2.1 Actor wise use cases.....	24
9.3 Use case Descriptions	27
9.3.1 Use case descriptions for Admin's functionalities.....	27
9.3.2 Use case descriptions for User's functionalities.	32
9.3.3 Use case descriptions for Guide's functionalities.	36
9.3.4 Use case descriptions for the hotel's functionalities.....	38
9.3.5 Use case descriptions for vehicle's functionalities.	40
9.4 Activity Diagrams	42
9.4.1 Activity diagrams for User	42
9.4.2 Activity diagrams for Admin	54
9.4.3 Activity diagrams for Hotel	69
9.4.4 Activity diagrams for Guide	74
9.4.5 Activity diagrams for Vehicle.....	79
10. Non-Functional Requirements.....	84
10.1 Identification of Quality Attributes	84
10.2 Suggestion to achieve Quality Attributes	84
11. Technologies to be used.	86
12. Work Breakdown Structure	87
13. Project Timeline	88
13.1 Block Timeline	88
13.2 Project Milestone Timeline	88
14. Gantt Chart	89
15. Declaration.....	92
16. References	93

List of the figures

<i>Figure 01: Survey Results</i>	08
<i>Figure 02: Schedule feasibility graph</i>	15
<i>Figure 03: Waterfall Model</i>	16
<i>Figure 04: Component Diagram</i>	22
<i>Figure 05: Use case Diagram</i>	23
<i>Figure 06: Use case Diagram for the user</i>	24
<i>Figure 07: Use case Diagram for the hotel</i>	25
<i>Figure 08: Use case Diagram for the guide</i>	25
<i>Figure 09: Use case Diagram for the vehicle</i>	26
<i>Figure 10: Use case Diagram for the admin</i>	26
<i>Figure 11: User login to the system</i>	42
<i>Figure 12: User creates an account</i>	43
<i>Figure 13: User adds comments</i>	44
<i>Figure 14: User books travel package</i>	45
<i>Figure 15: User cancels the booking</i>	46
<i>Figure 16: User deletes the account</i>	47
<i>Figure 17: User does the payment</i>	48
<i>Figure 18: User search</i>	49
<i>Figure 19: User updates the account</i>	50
<i>Figure 20: User view [not registered user]</i>	51
<i>Figure 21: User view [registered user]</i>	52
<i>Figure 22: User sends message</i>	53
<i>Figure 23: Admin login to the system</i>	54
<i>Figure 24: Admin views accounts</i>	55
<i>Figure 25: Admin views bookings</i>	56
<i>Figure 26: Admin views reports</i>	57
<i>Figure 27: Admin adds travel package</i>	58
<i>Figure 28: Admin adds category</i>	59
<i>Figure 29: Admin adds subcategory</i>	60
<i>Figure 30: Admin adds destinations</i>	61
<i>Figure 31: Admin checks availability</i>	62
<i>Figure 32: Admin deletes category</i>	63
<i>Figure 33: Admin deletes subcategory</i>	64
<i>Figure 34: Admin deletes travel package</i>	65
<i>Figure 35: Admin updates category</i>	66
<i>Figure 36: Admin updates subcategory</i>	67
<i>Figure 37: Admin updates travel package</i>	68
<i>Figure 38: Hotel login to the system</i>	69
<i>Figure 39: Hotel creates an account</i>	70
<i>Figure 40: Hotel updates the account</i>	71
<i>Figure 41: Hotel deletes the account</i>	72
<i>Figure 42: Hotel changes the availability</i>	73
<i>Figure 43: Guide login to the system</i>	74
<i>Figure 44: Guide creates an account</i>	75
<i>Figure 45: Guide updates the account</i>	76

<i>Figure 46: Guide deletes the account</i>	77
<i>Figure 47: Guide changes the availability</i>	78
<i>Figure 48: Vehicle login to the system</i>	79
<i>Figure 49: Vehicle creates an account</i>	80
<i>Figure 50: Vehicle updates the account</i>	81
<i>Figure 51: Vehicle deletes the account</i>	82
<i>Figure 52: Vehicle changes availability</i>	83
<i>Figure 53: Work Breakdown Structure</i>	87
<i>Figure 54: Block Timeline-1</i>	88
<i>Figure 55: Block Timeline-2</i>	88
<i>Figure 56: Project Milestone Timeline</i>	88
<i>Figure 57: Gantt chart</i>	89

List of tables

<i>Table 01: Use case description for admin logins into the system</i>	26
<i>Table 02: Use case description for admin views accounts' profiles.</i>	26
<i>Table 03: Use case description for admin views booking list.</i>	26
<i>Table 04: Use case description for admin views the reports.</i>	27
<i>Table 05: Use case description for admin adds a travel package.</i>	27
<i>Table 06: Use case description for admin adds a category</i>	27
<i>Table 07: Use case description for admin adds a subcategory</i>	28
<i>Table 08: Use case description for admin adds a destination</i>	28
<i>Table 09: Use case description for admin checks availability</i>	28
<i>Table 10: Use case description for admin deletes a category</i>	29
<i>Table 11: Use case description for admin deletes a subcategory</i>	29
<i>Table 12: Use case description for admin deletes a travel package</i>	29
<i>Table 13: Use case description for admin updates a category</i>	30
<i>Table 14: Use case description for admin updates a subcategory</i>	30
<i>Table 15: Use case description for admin updates a travel package</i>	30
<i>Table 16: Use case description for user logins into the system</i>	31
<i>Table 17: Use case description for user creates an account</i>	31
<i>Table 18: Use case description for user adds comments</i>	31
<i>Table 19: Use case description for user books travel package</i>	32
<i>Table 20: Use case description for user cancels the booking</i>	32
<i>Table 21: Use case description for user deletes the account</i>	32
<i>Table 22: Use case description for user does the payment</i>	33
<i>Table 23: Use case description for user searches</i>	33
<i>Table 24: Use case description for user updates the account</i>	33
<i>Table 25: Use case description for user [not registered user] views</i>	34
<i>Table 26: Use case description for user [registered user] views</i>	34
<i>Table 27: Use case description for user sends a message</i>	34
<i>Table 28: Use case description for guide login into the system</i>	35
<i>Table 29: Use case description for guide creates and account</i>	35
<i>Table 30: Use case description for guide updates the profile</i>	35
<i>Table 31: Use case description for guide deletes the account</i>	36
<i>Table 32: Use case description for guide changes availability</i>	36
<i>Table 33: Use case description for hotel logins into the system</i>	37
<i>Table 34: Use case description for hotel creates an account</i>	37
<i>Table 35: Use case description for hotel updates the account</i>	37
<i>Table 36: Use case description for hotel deletes the account</i>	38
<i>Table 37: Use case description for hotel changes availability</i>	38
<i>Table 38: Use case description for vehicle logins into the system</i>	39
<i>Table 39: Use case description for vehicle creates an account</i>	39
<i>Table 40: Use case description for vehicle updates the account</i>	39
<i>Table 41: Use case description for vehicle deletes the account</i>	40
<i>Table 42: Use case description for vehicle changes availability</i>	40

1. Problem Statement

Nowadays, internet users are increasing. So the purpose of the proposed website is established to internet users to organize their journeys. Our main purpose is to facilitate the best and quality service for organizing journeys for users.

Most people use their way to organize a journey. But it is a time-consuming way to do that. Many people have to face the following issues when they are organizing their journeys.

1. The budget of the travel plan.
2. Finding hotel booking options that are relevant to their plan.
3. Finding a vehicle relevant to the number of travelers.
4. Having less knowledge about the places to visit.

It is not easy to find a hotel according to his budget. Sometimes the conditions of the hotel will not be good. But before someone is booking the hotel, he can't get an idea about the conditions of that hotel. This will be a practical problem for travel organizing.

Finding a vehicle also creates a problem, because of the budget, condition of the vehicle, and the number of seats, etc. If someone wants to book a vehicle he has to check the conditions of the vehicle. Because of that he has to waste time and money to search for the best vehicle.

Considering the current systems, there are travel packages, but they are not categorized. Most of the current systems targeted most foreign people. The local people want to travel around Sri Lanka, but they can't think about such packages due to the budget.

From our proposed website, we are providing solutions to these problems. In our website, the user has to choose the travel package selecting the category and the subcategory and just give the personal details and the number of travelers and the number of hotel rooms.

After that, the administrator receives the inquiry and he will find a hotel, vehicle, and guide for the journey. Therefore, it is very easy to go on a journey because the user does not have to worry about the bookings. The administrator of the system will arrange everything.

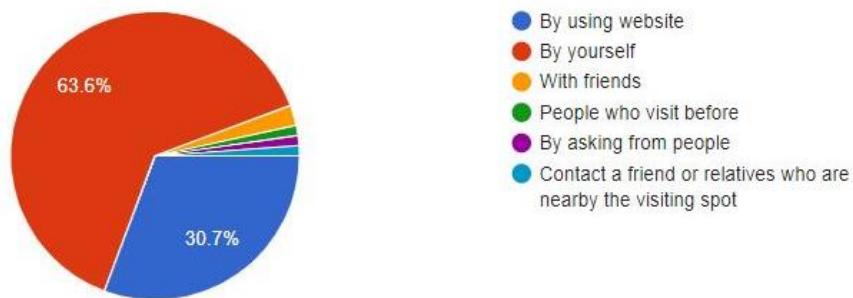
Another functionality that we provide is the registration of the hotels, guides, and vehicles on our website. From those registered accounts, the administrator will select the hotel, vehicle, and guide according to the package price.

Hotels have to change their availability by entering the available rooms regularly. The guide has to change his availability whether he is free or not and the vehicle owner has to change the availability of the vehicle. From that, they will find customers.

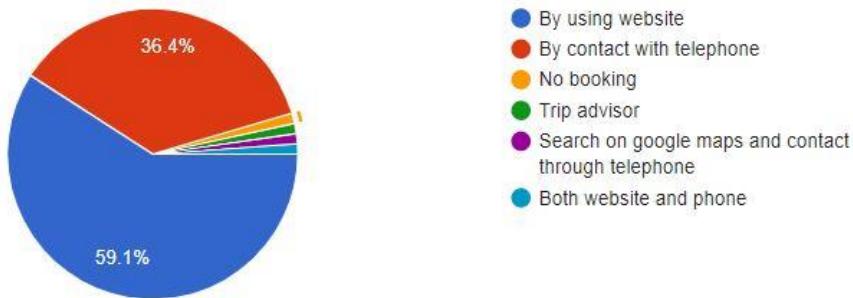
Users can give ratings to the hotel, guide, and vehicle which they had in the journey. So that the quality of the service of the hotel, guide, and the vehicle will be increased. From the feedbacks the proposed system provides easy and quality service to users.

We surveyed to find the problems that people are facing when they are arranging a journey. 89 people responded to our survey. All of them wanted to travel to different places in Sri Lanka.

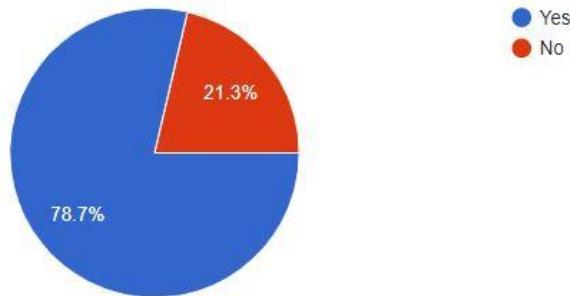
How you plan your journey?



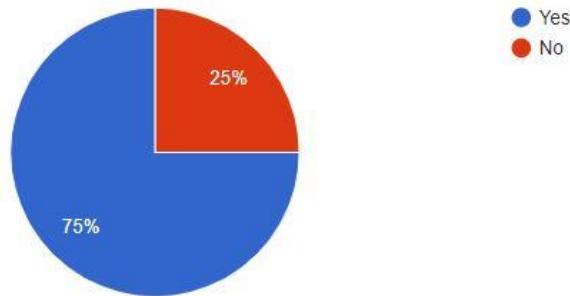
How do you book a hotel for the journey?



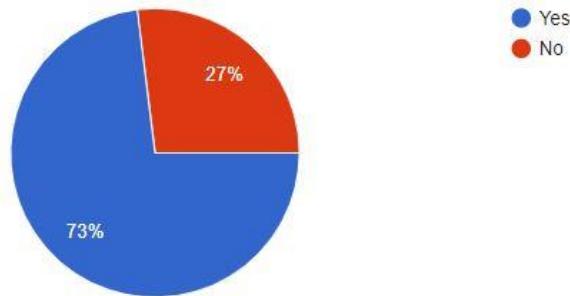
Do you face some problems to find a vehicle for the journey?



Do you think it is better to have a guide for your journey?



Do you like to book a travel package which include hotel, guide and vehicle other than your option?



Is it easy to have a website like this to plan your journey?



Figure 01: Survey Results

- From this little survey, we get an idea about how people are facing problems in booking a vehicle or a hotel. They have to struggle with booking a hotel according to the price and the availability. As well they have to find a vehicle that is suitable for the number of travelers of the journey. They will use different methods to find their needs.
- From the proposed website user can find a better way to manage their journeys. Because the proposed website will fulfill the local peoples' needs for traveling.

2. Project Goals

- Design and implement a comprehensive travel management system within 52 weeks.
- Finish the project with the expected quality of the system.
- Provide a specific, and relevant travel management system.

3. Project Scope

3.1 Identify the project scope and clear boundaries

- Main actors
 - User (travelers)
 - Administrator
 - Hotel
 - Guide
 - Vehicle
- Users can view the website but if they want to book a package he has to create an account on the website. After that the user has to select the travel package by selecting the category and the subcategory. Then there will be travel packages.
- From our proposed website, the user has to select the travel package which he likes according to the number of dates, prices, and locations. Users are not allowed for arranging a travel package. That means he has to say ok with the predefined travel packages.
- After the user books a travel package, the user has to pay for the booking. We will use a payment gateway, so that will be an out of boundary activity.
- As well, the user doesn't have an option to select the hotel for accommodation. That will be chosen by the administrator according to the price of the travel package and the number of booked rooms by the user.
- As well the user isn't allowed to choose a vehicle. That is the responsibility of the administrator. He will book the vehicle according to the number of travelers.
- A guide will be chosen by the administrator according to the travel location. There is no option to select a guide by a user.
- Users will be able to cancel the booking before two days of the booking date. If there is no two days gap, the user will be unable to cancel the booking.
- Users will be able to put comments, send messages, search travel packages, and view destinations.
- The administrator has the authority to add, update, delete the travel packages' categories, subcategories, and travel packages.

- The administrator views reports including the list of messages which are sent by users. But the reaction of the admin will be an out of boundary activity.
- User, Hotel, Vehicle, and Guide accounts have the authority to update or delete their accounts.
- Hotel, Vehicle, and Guide accounts have an option to change their availability. Considering the Hotel account, he will update the available number of rooms considering the room's type. The Vehicle and Guide accounts have the option to change whether they are available or not.
- The administrator organizes the journey by contacting available hotels, guides, and vehicles by checking the availabilities of those accounts. It will be an out of boundary task.

4. Project Objectives

- Providing quality service to a user who wants to plan a journey. This will be covered by the professional team and well-designed travel packages.
- Providing an easy way to plan a journey. The user has to select a travel package and book it using the number of travelers and the types of rooms for accommodation. The administrator will find the hotel, vehicle, and guide for the journey.
- Using marketing we are going to promote our website in various market segments. From that organization will get advantages in a competitive market.
- We provide facilities like accommodation, transportation and also there will be a guide according to the user's needs.
- We are providing facilities to register hotels, vehicles, and guides to the website. From that those registered once will get the opportunity to find customers to their business.
- Establish short-term and long-term profitability by increasing users and other accounts (hotels, guides, and vehicles).

5. Feasibility Study

From this feasibility study, we are analyzing whether our software meets the requirements. As well we used this feasibility study to determine whether our software can be implemented using current technology and within a specified budget and schedule. And also we are doing this feasibility study to find whether our software can be integrated with other existing software.

We make this feasibility study covering five areas.

1. Technical Feasibility
2. Economic Feasibility
3. Legal and Ethical Feasibility
4. Operational Feasibility
5. Schedule Feasibility

5.1 Technical Feasibility

- Considering our project it is possible with current technologies. When we consider our technical side we are using technologies and tools like, Programming languages (HTML, CSS, JavaScript, PHP, MySQL), WAMP server [Microsoft Windows o/s, Apache MySQL PHP], phpMyAdmin, diagram drawing tools (draw.io), etc.
- Each of these technologies is freely available and technical skills required are manageable.
- Front-end tools like HTML, CSS, JavaScript are easily compatible with our current hardware configuration.
- Back-end language, PHP is stable as compared to other programming languages and platform-independent. PHP is easily connected with the database and makes a secure connection with the database. And also the PHP unit is used to perform the unit testing quickly and easily.
- Considering the webserver technology, Apache is a proven, and reliable tool.
- Considering the database, MySQL is easily accessible. According to web technology, Apache we can use the WAMP web development stack.
- If we want to expand our system with more functionalities and features.
- The front-end and back-end technologies provide the accuracy, reliability, and ease of access and data security.

5.2 Economic feasibility

- According to our project, our development cost will be zero because we use open source software and freely available tools.
- Now we will use the test mode of a payment gateway but when the website hosts we have to pay for the payment gateway.
- As well the administrator should know to maintain the system. If it is not the training cost will be added.
- The proposed website provides benefits to the administrator. He will save his time and money because when he arranges the journey he will find everything in the same place such as finding a hotel, guide, and vehicle.
- And also considering the value of the working project, it will provide economic benefits to the administrator because he will receive a commission from the hotel, guide, and the vehicle owner who is engaging with travel packages.
- When comparing the existing websites they are not providing facilities like ours. Because of that many users will engage with our website. From that administrator will get money.

5.3 Legal and Ethical feasibility

- When considering the project like ours' legal aspects have a minor effect on the feasibility study. Because those legal aspects include software licenses.
- When we consider our project we are using freely available tools and open-source software.
- Our system will not save the users' sensitive data like credit card information.
- In ethically we have to consider how our group members think about the project.
- And also those ethical points should include how we work together and how we interact with other members of the group.

5.4 Schedule feasibility

- The project was started on the 20th of April 2020.
- We have to fulfill the requirements and implement our project within one year.
- We have four members in the group, so we can separate our works and implement our project on time. Everyone will be a developer as well as a tester. So that will reduce the time that we are taking to implement the project
- As well as we can include alternative timelines like holidays to do those tasks.
- We are planning our works according to the following steps.

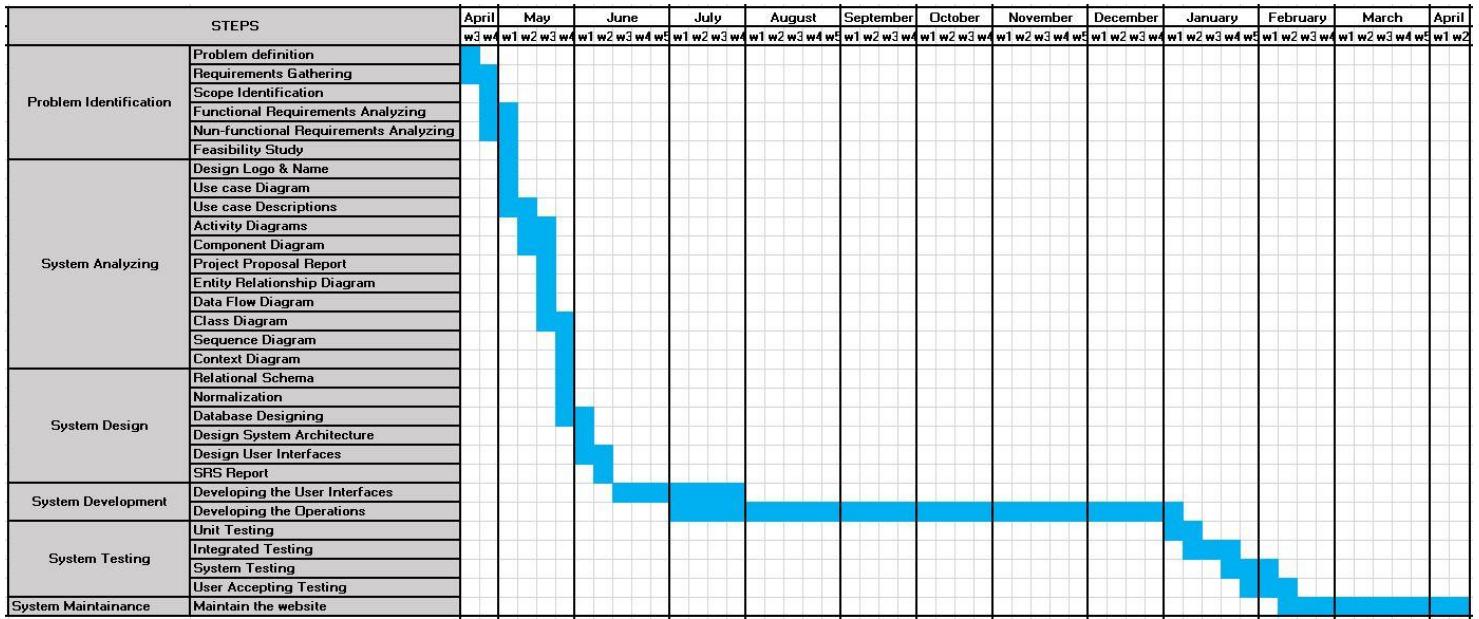


Figure 02: Schedule feasibility graph

5.5 Operational feasibility

- When we consider our project it is a really good solution for current requirements.
- When we consider the survey that we did, it is crystal clear that our solution for travel management will fulfill the requirements of the user.
- The user has to book a hotel and a vehicle for their journey. According to the survey, it is not easy to find those needs. So from our system, they have to only spend a few minutes to find a travel package and enter their details. The administrator will find all the other needs of them.
- When considering the proposed system, it will provide more benefits. Because the user has to pay the cost separately. But in our scenario, the cost will be based on the travel package. So it is economically benefited for the user.
- Not only the user, the other accounts hotel, guide, and the vehicle will also get benefits. If they are hiring themselves that will be a big cost more than they registered on this website. Because they have to have a method like a website to get hired. So the only thing they have to do is change their availability on this website. So they will get advantages from our website.

6. Deliverables of the project.

- In the System Analyzing phase, we are providing the proposal of the system. It will include the project scope, feasibility study, functional and non-functional requirements, etc.
- In the System Designing phase, we are going to provide the Software Requirement Specification (SRS) Document including database structure, wireframes, and the prototype.
- In the final stage, we are providing a working system with user documentation.

7. Methodology

7.1 Development Model

- We are going to use the waterfall model as our development model.

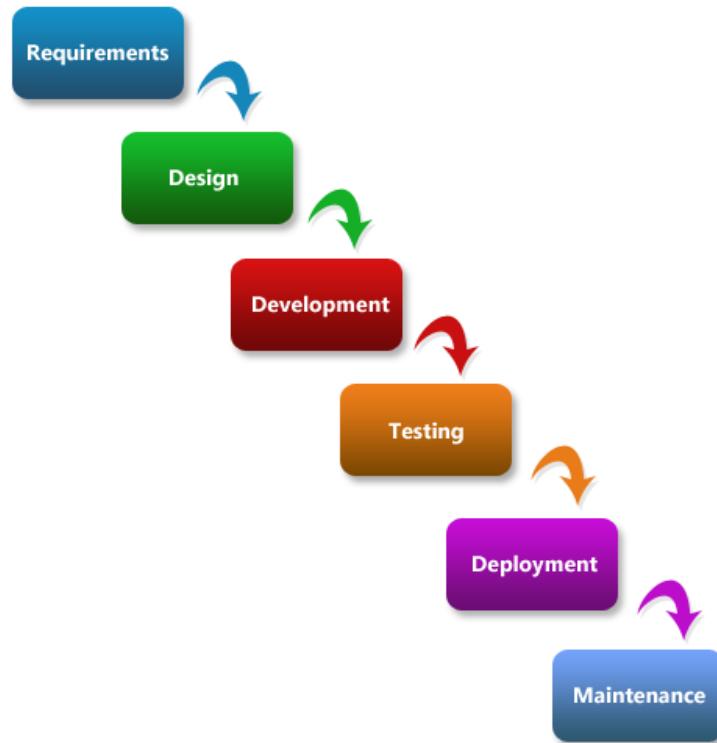


Figure 02: Waterfall Model

- Step 1: We gather requirements according to the problem that we identified. Here we have surveyed to identify the problem of society. Then we decided to have solutions for those issues.
- Step 2: Here we have to design our interfaces, database rationalities. As well we have to design the database with tables which are interacting with our system.
- Step 3: In this phase, we are going to implement our system. We have to implement the system that we designed in the designing phase.
- Step 4: In this phase, we are going to test our system to find bugs and errors and fix them.
- Step 5: We are planning to give our project for selected users and get their ideas about our project.
- Step 6: This phase will be used to further improve and maintain the system.

7.2 Reasons to select this development model

Considering the Second Year Group Project, we have the following reasons to select the Waterfall model as our development model.

1. It is milestone-focused development. The deadlines and guidelines for the project are analyzed. In the first milestone, we have to provide all the requirements of the project. In the interim presentation which is the third milestone, we have to submit the finalize user interface and the finalize database. So there is no opportunity to change the user interfaces or the database which are developed in the design phase. Because of that, there is no opportunity to go back to the above phases from the current phase and change the functionalities of the system. So because of the workflow which is created according to the milestones, it will follow the Waterfall model.
2. We have to define the scope at the beginning of the project. So we have found clear requirements.
3. And also, we have identified the technologies for the project at the beginning. So those technologies can't change during the development of time.

8. Project constraints and assumptions identification

8.1 Project constraints

8.1.1 Cost

- According to our proposed system, the development cost will be zero. But if we want to host the website we have to pay for the domain.
- As well now we are using a test model of a payment gateway but if we want to host the website we have to pay for the payment gateway.

8.1.2 Scope

- Under section 1.4.1 we provide the scope of our proposed website.

8.1.3 Quality

- From our proposed website users can give ratings for the various sections. Through that, the administrator can get an idea about the quality of the service which is given by the website.

8.1.4 Customer Satisfaction

- When the development of the project we have to implement a user-friendly environment.
- When considering the working website, the administrator has a responsibility to provide good service to the user.

8.1.5 Risk

- When the development of the project we have to submit the project in the given period of the time.
- When considering the working website, the administrator will face some risks of the conditions of the hotel and the vehicle
- As well the weather condition will be an issue sometimes.

8.1.6 Resources

- For the development of the proposed system, we are using our laptops, freely available tools, and open-source software.
- There will be four members to develop the proposed system.

8.1.7 Time

- We have only one year to do the proposed system. In that period we have to identify the problem, gather requirements, system designing, system implementation, system testing, and the maintenance of the system.

8.2 Assumptions

- During the development of the proposed system,
 - We have all the resources that we need to develop the proposed website.
 - All our equipment is in good condition.
 - Our team members have all the required skills.
- Considering the working website administrator will assume,
 - All the vehicles are in good condition.
 - All the hotels have good service and facilities.
 - The weather condition will not be an issue.
- Considering the technology,
 - We have all the required technology.
 - All the members learned about those technologies. If someone has a problem with some technologies, he will learn those within this period.
- Considering the finance of the proposed system,
 - The development cost will be zero.

9. Functional Requirement Identification

According to our proposed system, we can find the following requirements as functional requirements.

1. User Requirements

- User
 - Users must register to the system by creating an account.
 - The user will log in to the website by using his username and password.
 - Users can search for information about travel packages by entering the number of days, price of the package or location name.
 - Users can view travel packages with separating categories and subcategories.
 - Users can view the destinations.
 - Users can view his profile.
 - The user can update his profile.
 - Users can delete his profile.
 - Users can add comments.
 - Users can book travel packages with entering details.
 - Users can pay for the booked travel package.
 - Users can cancel the bookings within two days.
 - Users can send messages to admin, regarding issues.
- Hotel
 - Hotel users must register to the system by creating an account.
 - Hotel users will log in to the website by using his username and password.
 - Hotel users can view their profile.
 - The hotel user can update his profile.
 - Hotel users can delete his profile.
 - Hotel user can change the availability
- Guide
 - Guide users must register to the system by creating an account.
 - Guide user will log in to the website by using his username and the password.
 - Guide users can view their profile.
 - Guide users can update their profile.
 - Guide users can delete their profile.
 - Guide user can change their availability

- Vehicle
 - Vehicle users must register to the system by creating an account.
 - Vehicle users will log in to the website by using his username and password.
 - The vehicle user can view his profile.
 - The vehicle user can update his profile.
 - The vehicle user can delete his profile.
 - Vehicle user can change the availability

2. System Requirements

- The system can be able to store the details of bookings.
- The system should have different home pages for different user account types.
- The system will be able to generate reports.
- The system should be able to add, update, and delete travel packages.
- The system should be able to add, update, and delete categories and subcategories.
- The system should be able to add a destination.
- The system should be able to search for accounts and view accounts.

9.1 Component diagram

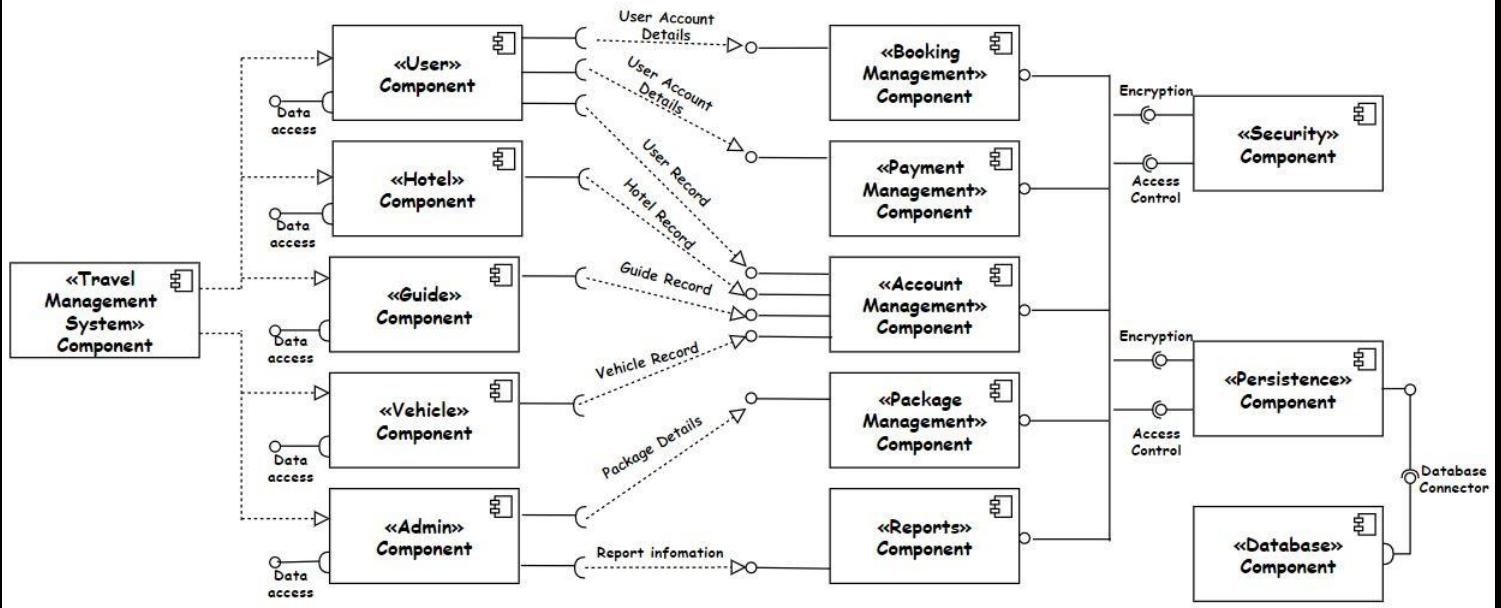


Figure 04: Component Diagram

9.2 Use case diagram



Figure 05: Use case Diagram

9.2.1 Actor wise use cases

1. User

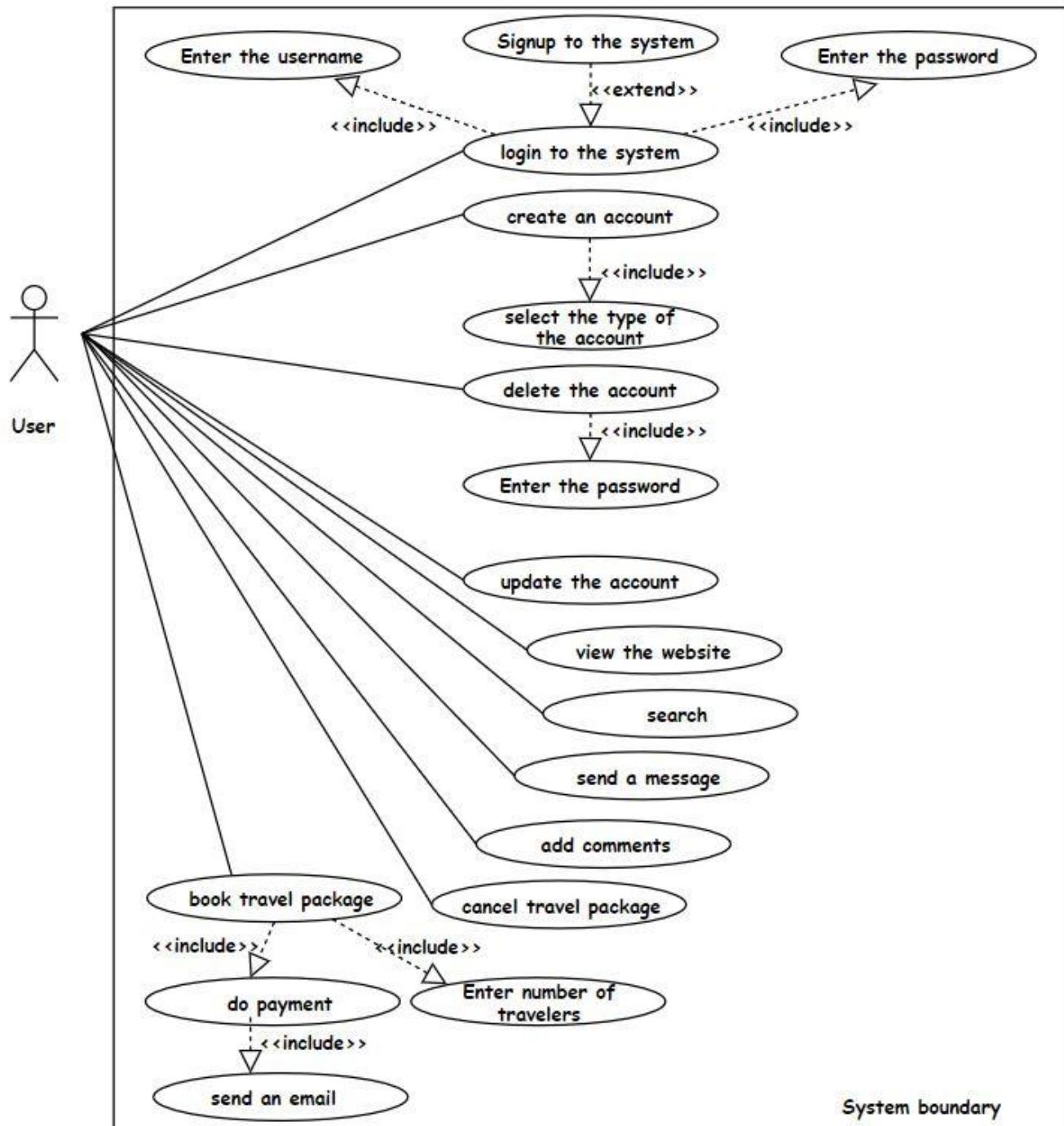


Figure 06: Use case Diagram for the user

2. Hotel

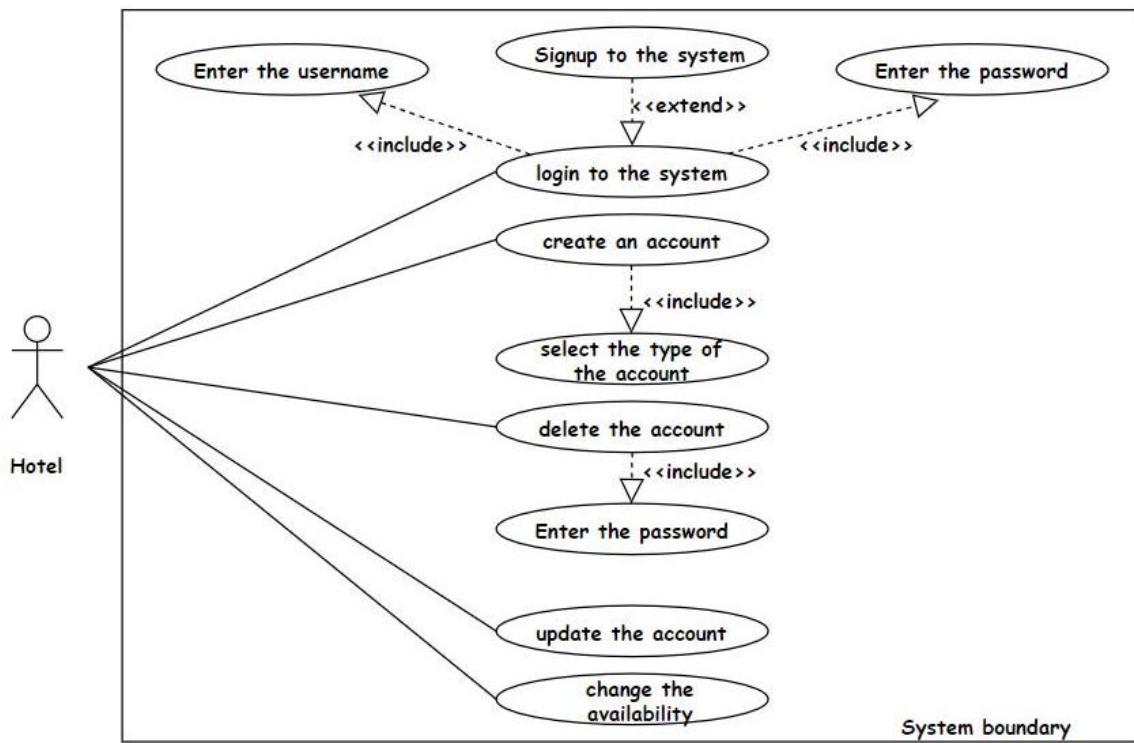


Figure 07: Use case Diagram for the hotel

3. Guide

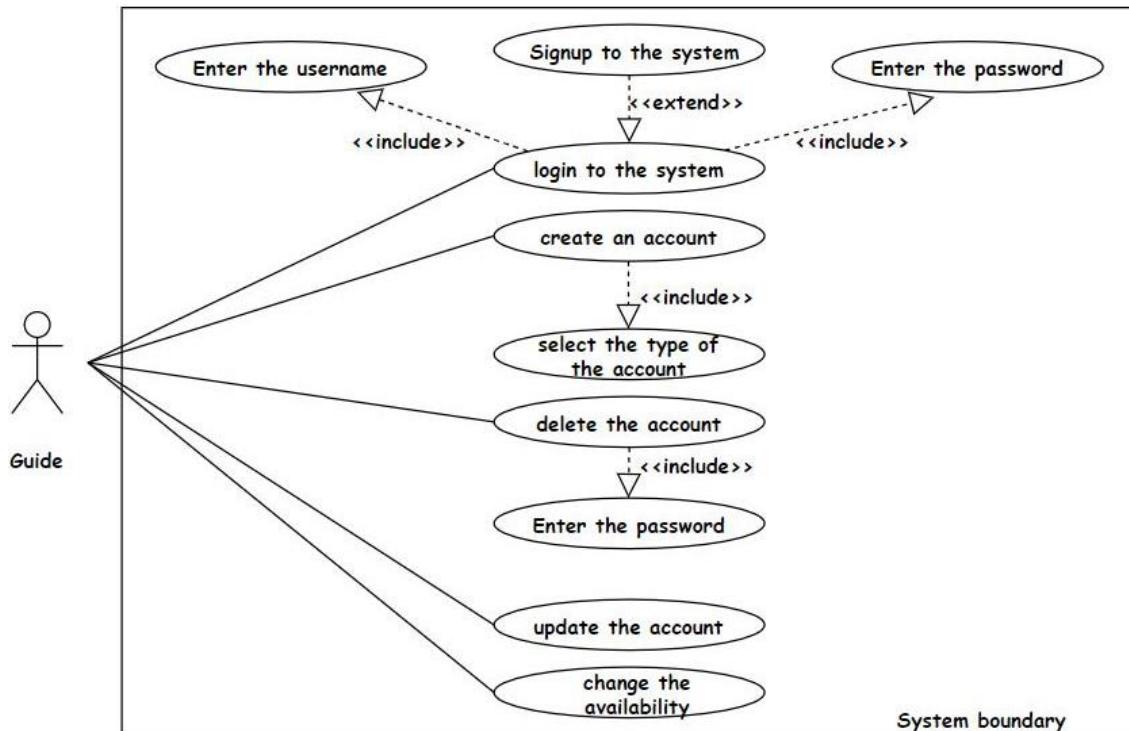


Figure 08: Use case Diagram for the guide

4. Vehicle

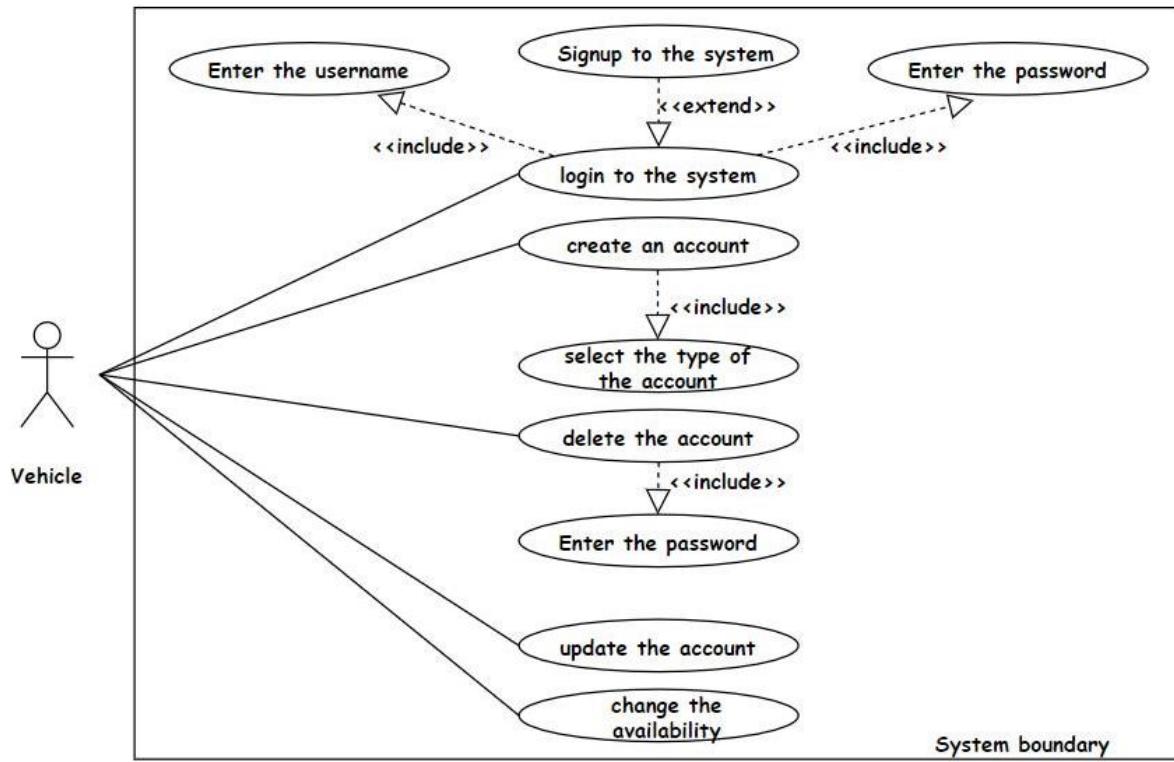


Figure 09: Use case Diagram for the vehicle

5. Admin

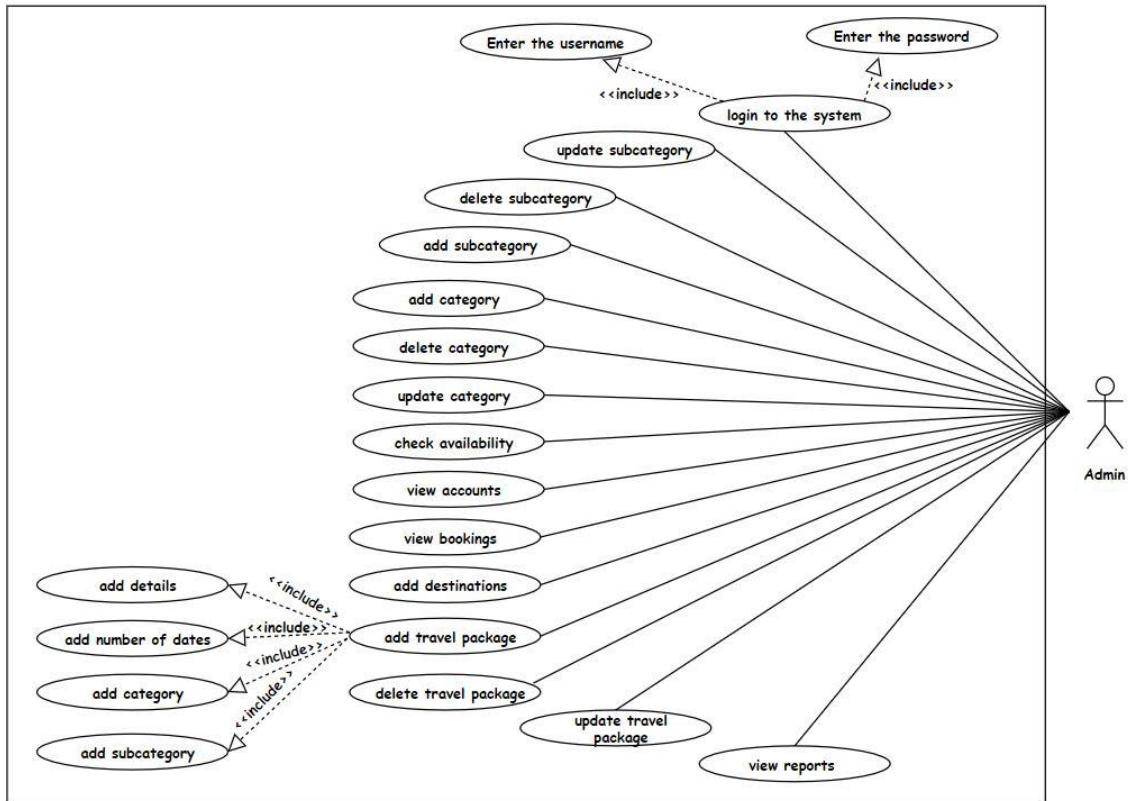


Figure 10: Use case Diagram for the admin

9.3 Use case Descriptions

9.3.1 Use case descriptions for Admin's functionalities.

Use case	Login to the system
Summary	Admin logins into the system.
Actors	Admin
Preconditions	Admin should not be logged in to the system.
Description	Admin should enter the username and the password to login to the system.
Exceptions	Admin's username and the password not in the system. Admin already logs in to the system.
Post conditions	Admin logged in to the system. Admin views the home page

Table 01: Use case description for admin logins into the system

Use case	View accounts
Summary	Admin views accounts' profiles.
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select the view profile and select the type of the accounts.
Description	Admin will see a list of accounts. Then admin can view the accounts' profiles using their ids or names. From selecting the name of the account, the admin will be able to see more details of the corresponding account.
Exceptions	The selected profile does not exist in the system.
Post conditions	Admin will view the selected profile.

Table 02: Use case description for admin views accounts' profiles.

Use case	View bookings
Summary	Admin views booking list.
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select the view bookings.
Description	Admin can view the booking list. Then admin can enter booking id or date to find the booking. From that admin will get more details about the user and he will get the number of days and the number of people who will join the journey.
Exceptions	There is no booking available.
Post conditions	Admin will view the booking list and the more details of the selected booking.

Table 03: Use case description for admin views booking list.

Use case	View reports
Summary	Admin views the reports.
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select the view reports.
Description	Admin can view the reports like the number of travelers for a travel package, best hotels, best guides, the best vehicle, best travel package, etc. First admin has to select the report. After that he will see the report's details.
Exceptions	There are no details to generate reports.
Post conditions	Admin will view the selected report's details.

Table 04: Use case description for admin views the reports.

Use case	Add travel package
Summary	Admin adds a new travel package to the system
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select the add travel package.
Description	Admin can add a new travel package to the system by choosing category and subcategory. Admin has to enter details of the travel plan and the number of days. As well the admin has to enter the price of the travel package for a single person.
Exceptions	Admin is not logged in to the system.
Post conditions	A message of “successfully added” will appear.

Table 05: Use case description for admin adds a travel package.

Use case	Add category
Summary	Admin adds the categories of the travel package to the system
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select the add category.
Description	Admin can add a new category for the travel package by entering its name. Then the system will automatically give an id number to that category.
Exceptions	The added category exists in the system.
Post conditions	A message of “successfully added” will appear.

Table 06: Use case description for admin adds a category

Use case	Add subcategory
Summary	Admin adds the subcategories of the travel package to the system
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select the add subcategory.
Description	First admin should have selected the category of the new subcategory. Then admin can add a new subcategory for the travel package by using its name. Then the system will automatically give an id number to that subcategory.
Exceptions	The added subcategory exists in the system.
Post conditions	A message of “successfully added” will appear.

Table 07: Use case description for admin adds a subcategory

Use case	Add destinations
Summary	Admin adds the destination into the system.
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select the add destinations.
Description	Admin can add a new destination including the details and photos. Then the system will automatically give an id number to that destination.
Exceptions	The added destination name exists in the system.
Post conditions	A message of “successfully added” will appear.

Table 08: Use case description for admin adds a destination

Use case	Check availability
Summary	Admin checks the availability of hotels, guides, and vehicles
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select view availability of the wanted account type.
Description	Admin can view a list of registered accounts according to the selected account type. In this list, the admin can view the available accounts. Then admin can get more details by selecting one of those accounts.
Exceptions	There are no registered accounts of the selected account types.
Post conditions	The list of selected and available accounts will be displayed.

Table 09: Use case description for admin checks availability

Use case	Delete a category
Summary	Admin deletes a category.
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select the delete category.
Description	Admin will delete a category selecting its id number. Then he will see the details of the category. Then there will be a confirmation of the deletion again. After the admin will confirm the deletion the category will be deleted.
Exceptions	The category id is not in the system.
Post conditions	A message of “successfully deleted” will appear.

Table 10: Use case description for admin deletes a category

Use case	Delete a subcategory
Summary	Admin deletes a subcategory.
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select the delete subcategory.
Description	Admin will delete a subcategory selecting its id number. Then he will see the details of the subcategory. Then there will be a confirmation of the deletion again. After admin will confirm the deletion the subcategory will be deleted.
Exceptions	The subcategory id is not in the system.
Post conditions	A message of “successfully deleted” will appear.

Table 11: Use case description for admin deletes a subcategory

Use case	Delete a travel package
Summary	Admin deletes the travel package.
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select the delete travel package.
Description	Admin will delete a travel package selecting its id number. Then he will see the details of the travel package. Then there will be a confirmation of the deletion again. After the admin will confirm the deletion the travel package will be deleted.
Exceptions	The travel package id is not in the system.
Post conditions	A message of “successfully deleted” will appear.

Table 12: Use case description for admin deletes a travel package

Use case	Update category
Summary	Admin updates the category.
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select the update category.
Description	Admin updates the category name by selecting its id.
Exceptions	The updated category does not exist in the system.
Post conditions	A message of “successfully updated” will appear.

Table 13: Use case description for admin updates a category

Use case	Update subcategory
Summary	Admin updates the subcategory.
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should select the update subcategory.
Description	Admin updates the subcategory name by selecting its id.
Exceptions	The update subcategory does not exist in the system.
Post conditions	A message of “successfully updated” will appear.

Table 14: Use case description for admin updates a subcategory

Use case	Update travel package
Summary	Admin updates the travel package.
Actors	Admin
Preconditions	Admin should be logged in to the system by using the username and the password. Admin should be selected the update travel package.
Description	Admin updates the travel package's details selecting its id. Then he can change details about the travel package.
Exceptions	The updated travel package does not exist in the system.
Post conditions	A message of “successfully updated” will appear.

Table 15: Use case description for admin updates a travel package

9.3.2 Use case descriptions for User's functionalities.

Use case	Log in to the system
Summary	User logins into the system
Actors	User
Preconditions	The user should not be logged in to the system by using the username and the password. The user should select the login.
Description	The user will have to enter the username and the password to login into the system.
Exceptions	The user is not registered in the system. The user already logs into the system.
Post-conditions	User logged in to the system

Table 16: Use case description for user logins into the system

Use case	Create an account
Summary	User registers to the system using the account type of user.
Actors	User
Preconditions	The user should not have an account in the system by using the username and the password. The user has to select the account type as the “user”.
Description	User will get a form to fill his information for registration to the website. After he submitted the details he will be a registered user in the system.
Exceptions	The user has an account with the same details like email.
Post-conditions	User will get the home page.

Table 17: Use case description for user creates an account

Use case	Add comments
Summary	The user adds comments about the travel package.
Actors	User
Preconditions	The user should be logged in to the system by using the username and the password. User should select the view profile and add comments.
Description	The user will comment on the travel packages and add photos. There is an option to give ratings for the hotel, vehicle, and the guide as well.
Exceptions	The user is not logged in to the system.
Post-conditions	The user can input the comments to the system.

Table 18: Use case description for user adds comments

Use case	Book travel package
Summary	User books a travel package.
Actors	User
Preconditions	The user should be logged in to the system by using the username and the password.
Description	First user has to select the category and the subcategory of the package. Then he will find travel packages under the subcategory. Pressing the “Book Now” button, the user can view a form to fill. After entering the details he will be able to book a travel package.
Exceptions	The user is not logged in to the system.
Post-conditions	The user will see the payment options for the booked travel package.

Table 19: Use case description for user books travel package

Use case	Cancel the booking
Summary	The user cancels the booking which he made before.
Actors	User
Preconditions	The user should be logged in to the system by using the username and the password. The date of cancelation should be within two dates starting the booking date. The user should select the view profile and view bookings.
Description	User can view the booking which had done by him. Then he will click the name and he cancel the booking by clicking the “cancel” button. After confirming the cancelling the booking will be canceled.
Exceptions	The user is not logged in to the system. User not book a travel package. The cancelation date is after two dates of the booking date.
Post-conditions	The user gets a message for canceling the booking.

Table 20: Use case description for user cancels the booking

Use case	Delete the account
Summary	User can delete his account which includes his details from the system.
Actors	User
Preconditions	The user should be logged in to the system by using the username and the password. The user should select the view profile.
Description	The user can delete his account by clicking the “delete profile” button. After that he has to enter the password of his account. Then there will be a confirmation. After confirming that he will be able to delete the account from the system.
Exceptions	The user is not logged in to the system.
Post-conditions	The user gets a message for deleting his account.

Table 21: Use case description for user deletes the account

Use case	Do payment
Summary	User pay for the booking travel package.
Actors	User
Preconditions	The user should be logged in to the system by using the username and the password. The user should book a travel package before the payment.
Description	User can pay for the booking travel package using his bank details.
Exceptions	The user is not logged in to the system. The user is not booked a travel package.
Post-conditions	The user gets a message for payment.

Table 22: Use case description for user does the payment

Use case	Search
Summary	User search travel packages.
Actors	User
Preconditions	The user should go to the website it doesn't matter whether he logs in to the system or not.
Description	The user search travel packages by entering the number of days, places, or the price of the travel package.
Exceptions	User is not on the home page.
Post-conditions	User will see the filtered packages

Table 23: Use case description for user searches

Use case	Update the account
Summary	The user updates the profile.
Actors	User
Preconditions	The user should be logged in to the system by using the username and the password. The user should select the view profile.
Description	The user updates his profile by clicking the “update profile” button. After that user will see the details of the profile and he will be able to update details.
Exceptions	The user is not logged in to the system.
Post-conditions	User will get a message for updating.

Table 24: Use case description for user updates the account

Use case	View [not registered user]
Summary	The user views the website.
Actors	User
Preconditions	The user should not be logged in to the system by using the username and the password.
Description	The user will see the options to view the packages by selecting category and the subcategory, view comments, view destinations, and view about us.
Exceptions	The user is logged in to the system.
Post-conditions	User will see the viewed item.

Table 25: Use case description for user [not registered user] views

Use case	View [registered user]
Summary	The user views the website.
Actors	User
Preconditions	The user should be logged in to the system.
Description	The user will see the options to view the packages by selecting category and the subcategory, view comments, view destinations, view profile, and view about us.
Exceptions	The user is not logged in to the system.
Post-conditions	User will see the viewed item.

Table 26: Use case description for user [registered user] views

Use case	Send message
Summary	The user sends messages to the administrator regarding something.
Actors	User
Preconditions	The user should be on the home page. It doesn't matter whether the user is registered or not. The user should select the contact page.
Description	User will see a form and he will enter his contact details and the message. After that he has to click on the "send" button.
Exceptions	The user is not on the website.
Post-conditions	User will get a message for sending.

Table 27: Use case description for user sends a message

9.3.3 Use case descriptions for Guide's functionalities.

Use case	Log in to the system
Summary	Guide logs into the system
Actors	Guide
Preconditions	The guide should not be logged in to the system. The guide should select the login.
Description	The guide will have to enter the username and the password to login to the system.
Exceptions	Guide's username and the password not in the system. The guide already logs in to the system.
Post-conditions	Guide logged in to the system.

Table 28: Use case description for guide login into the system

Use case	Create an account
Summary	Guide registers to the system using the account type of “guide”.
Actors	Guide
Preconditions	The guide should not have an account in the system. The guide should be selected the account type as “guide”
Description	The guide will get a form to fill his information for registration to the website.
Exceptions	The guide has an account with the same details like email.
Post-conditions	Guide views the home page.

Table 29: Use case description for guide creates and account

Use case	Update the account
Summary	Guide update the profile.
Actors	Guide
Preconditions	The guide should be logged in to the system by using the username and the password. The guide should select the view profile.
Description	The guide updates his profile by clicking the “update profile” button. After that the guide will see the details of the profile and he will be able to update details.
Exceptions	The guide is not logged in to the system.
Post-conditions	The guide will get a message for updating.

Table 30: Use case description for guide updates the profile

Use case	Delete the account
Summary	The guide can delete his account which includes his details from the system.
Actors	Guide
Preconditions	The guide should be logged in to the system by using the username and the password. The guide should select the view profile.
Description	The guide can delete his account by clicking the “delete profile” button. After that he has to enter the password of his account. Then there will be a confirmation. After confirming that he will be able to delete the account from the system.
Exceptions	The guide is not logged in to the system.
Post-conditions	Guide gets a message for deleting his account.

Table 31: Use case description for guide deletes the account

Use case	Change availability
Summary	The guide can input whether he is available or not available.
Actors	Guide
Preconditions	The guide should be logged in to the system by using the username and the password. The guide should be selected the view profile.
Description	The guide will see an option as “change availability”. After clicking that the guide will be able to change the availability.
Exceptions	The guide is not logged in to the system.
Post-conditions	The guide will see a message of changing the availability.

Table 32: Use case description for guide changes availability

9.3.4 Use case descriptions for the hotel's functionalities.

Use case	Log in to the system
Summary	Hotel logins into the system
Actors	Hotel
Preconditions	The hotel should not be logged in to the system. The hotel should select the login.
Description	The hotel will have to enter the username and the password to login to the system.
Exceptions	Hotel's username and the password not in the system. The hotel already logs in to the system.
Post-conditions	Hotel logged in to the system.

Table 33: Use case description for hotel logins into the system

Use case	Create an account
Summary	Hotel registers to the system using the account type of “hotel”.
Actors	Hotel
Preconditions	The hotel should not have an account in the system. The hotel should be selected the account type as “hotel”
Description	The hotel will get a form to fill his information for register to the website.
Exceptions	The hotel has an account with the same details like email.
Post-conditions	The hotel views the home page.

Table 34: Use case description for hotel creates an account

Use case	Update the account
Summary	Hotel updates the profile.
Actors	Hotel
Preconditions	The hotel should be logged in to the system by using the username and the password. The hotel should select the view profile.
Description	The hotel updates his profile by clicking the “update profile” button. After that the hotel will see the details of the profile and he will be able to update details.
Exceptions	The hotel is not logged in to the system.
Post-conditions	The hotel will get a message for updating.

Table 35: Use case description for hotel updates the account

Use case	Delete the account
Summary	The hotel can delete his account which includes his details from the system.
Actors	Hotel
Preconditions	The hotel should be logged in to the system by using the username and the password. The hotel should select the view profile.
Description	The hotel can delete his account by clicking the “delete profile” button. After that he has to enter the password of his account. Then there will be a confirmation. After confirming that he will be able to delete the account from the system.
Exceptions	The hotel is not logged in to the system.
Post-conditions	The hotel gets a message for deleting his account.

Table 36: Use case description for hotel deletes the account

Use case	Change availability
Summary	The hotel can input whether he is available or not available.
Actors	Hotel
Preconditions	The hotel should be logged in to the system by using the username and the password. The hotel should select the view profile.
Description	The hotel will see an option as “change availability”. After clicking that the hotel will be able to change the availability. Hotel has to enter the available number of rooms according to its type.
Exceptions	The hotel is not logged in to the system.
Post-conditions	The hotel will see a message of changing the availability.

Table 37: Use case description for hotel changes availability

9.3.5 Use case descriptions for vehicle's functionalities.

Use case	Log in to the system
Summary	Vehicle logs into the system
Actors	Vehicle
Preconditions	The vehicle should not be logged in to the system. The vehicle should select the login.
Description	The vehicle will have to enter the username and the password to login to the system.
Exceptions	Vehicle's username and the password not in the system. The vehicle already logs in to the system.
Post-conditions	The vehicle logged in to the system.

Table 38: Use case description for vehicle logins into the system

Use case	Create an account
Summary	Vehicle registers to the system using the account type of “vehicle”.
Actors	Vehicle
Preconditions	The vehicle should not have an account in the system. The vehicle should select the account type as “vehicle ”
Description	The vehicle will get a form to fill his information for registration to the website.
Exceptions	The vehicle has an account with the same details like email.
Post-conditions	The vehicle views the home page.

Table 39: Use case description for vehicle creates an account

Use case	Update the account
Summary	The vehicle updates the profile.
Actors	Vehicle
Preconditions	The vehicle should be logged in to the system by using the username and the password. The vehicle should select the view profile.
Description	The vehicle updates his profile by clicking the “update profile” button. After that hotel will see the details of the profile and he will be able to update details.
Exceptions	The vehicle is not logged in to the system.
Post-conditions	The vehicle will get a message for updating.

Table 40: Use case description for vehicle updates the account

Use case	Delete the account
Summary	The vehicle can delete his account which includes his details from the system.
Actors	Vehicle
Preconditions	The vehicle should be logged in to the system by using the username and the password. The vehicle should select the view profile.
Description	The vehicle can delete his account by clicking the “delete profile” button. After that he has to enter the password of his account. Then there will be a confirmation. After confirming that he will be able to delete the account from the system.
Exceptions	The vehicle is not logged in to the system.
Post-conditions	The vehicle gets a message for deleting his account.

Table 41: Use case description for vehicle deletes the account

Use case	Change availability
Summary	The vehicle can input whether he is available or not available.
Actors	Vehicle
Preconditions	The vehicle should be logged in to the system by using the username and the password. The vehicle should select the view profile.
Description	The vehicle will see an option as “change availability”. After clicking that the vehicle will be able to change the availability.
Exceptions	The vehicle is not logged in to the system.
Post-conditions	The vehicle will see a message of changing the availability.

Table 42: Use case description for vehicle changes availability

9.4 Activity Diagrams

9.4.1 Activity diagrams for User

9.4.1.1 Login to the system

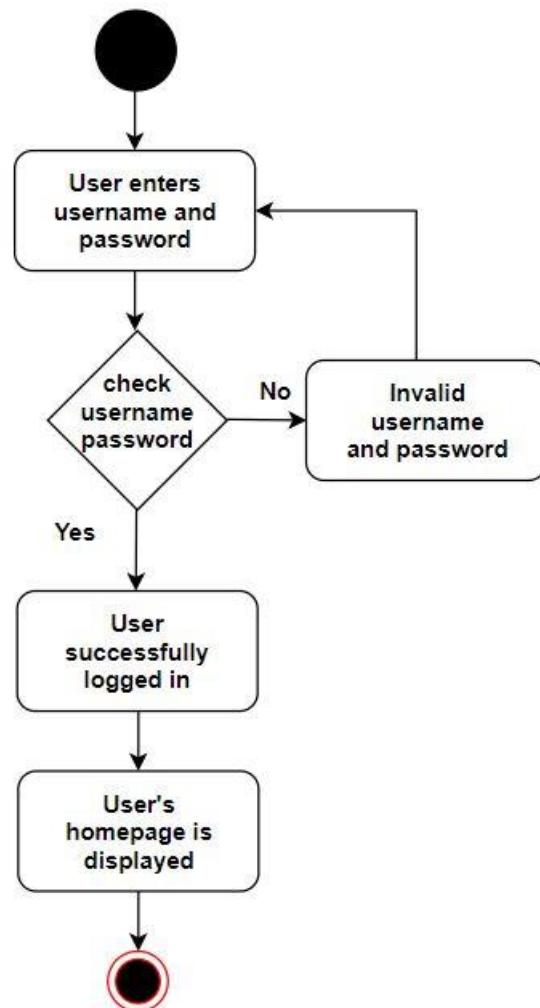


Figure 11: User login to the system

9.4.1.2 Create an account

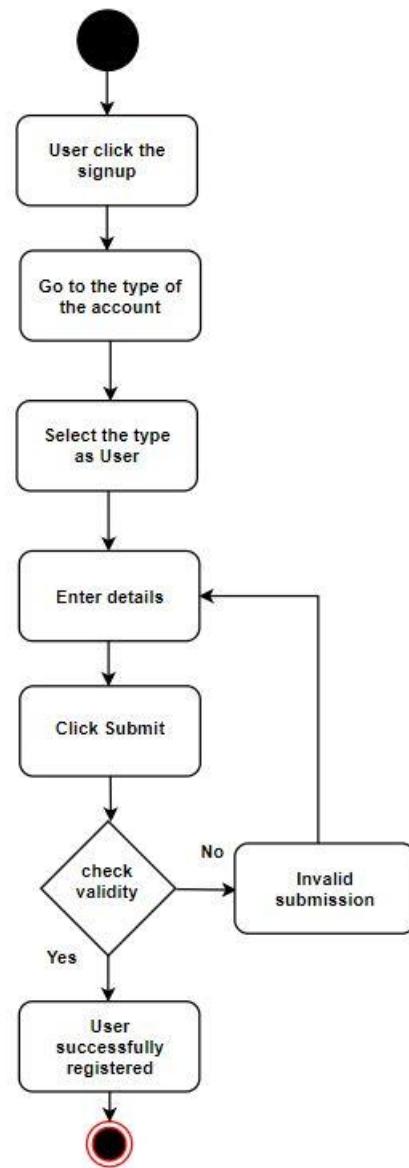


Figure 12: User creates an account

9.4.1.3 Add comments

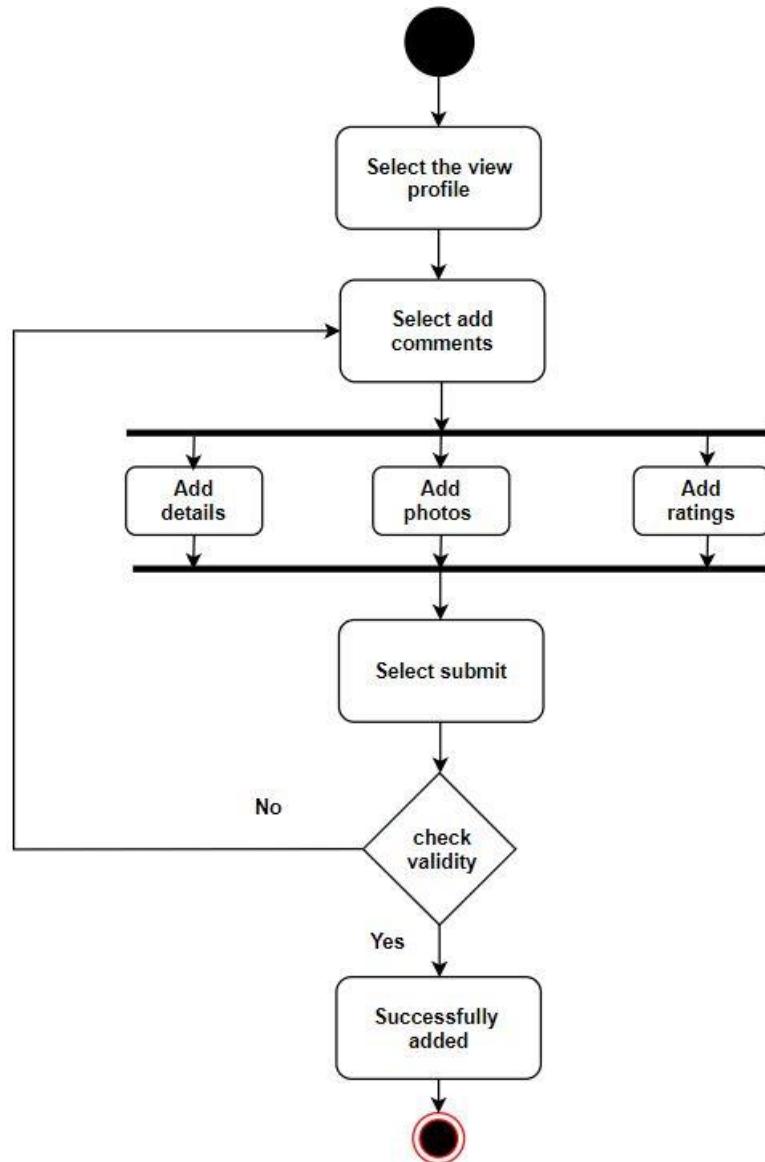


Figure 13: User adds comments

9.4.1.4 Book travel package

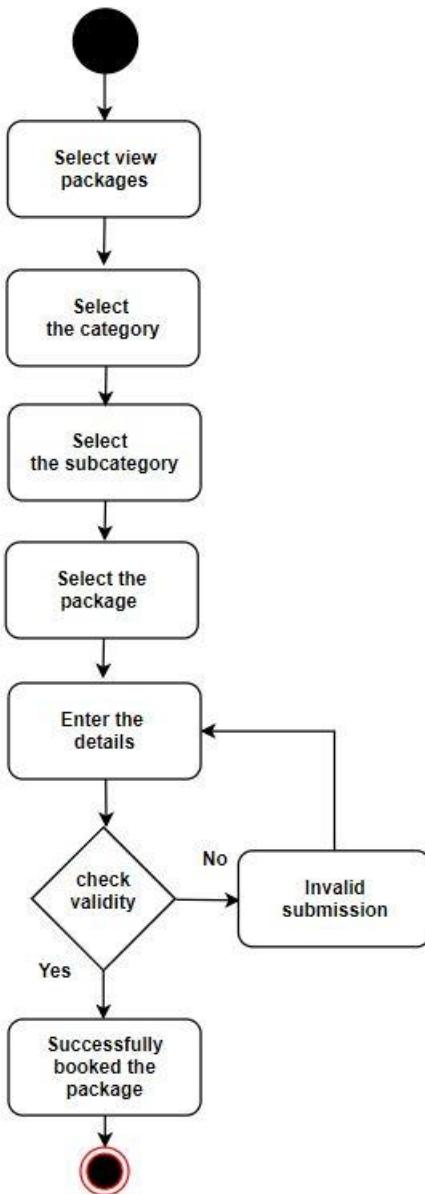


Figure 14: User books travel package

9.4.1.5 Cancel the booking

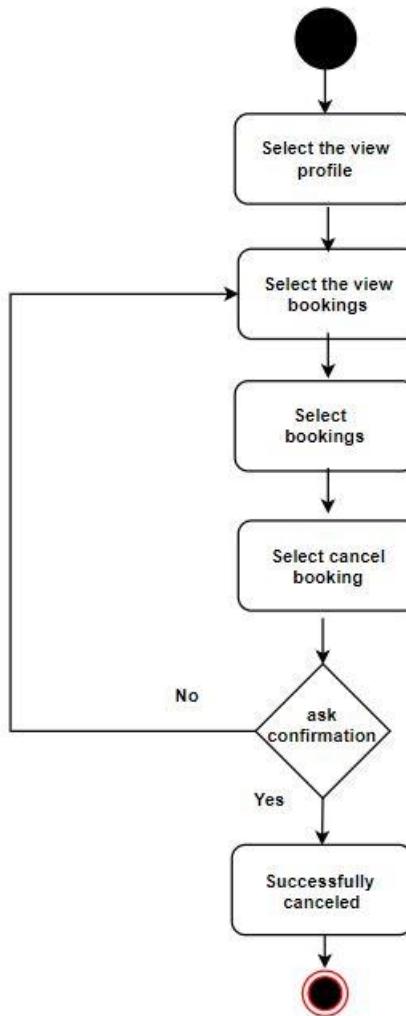


Figure 15: User cancels the booking

9.4.1.6 Delete the account

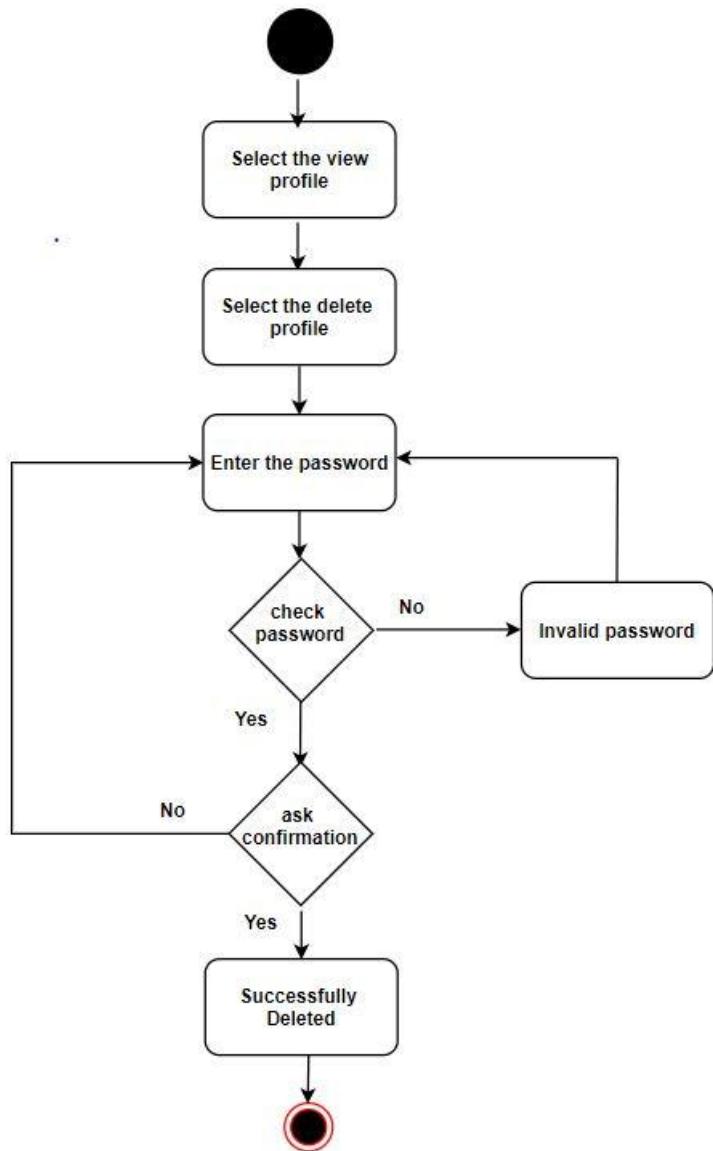


Figure 16: User deletes the account

9.4.1.7 Do payment

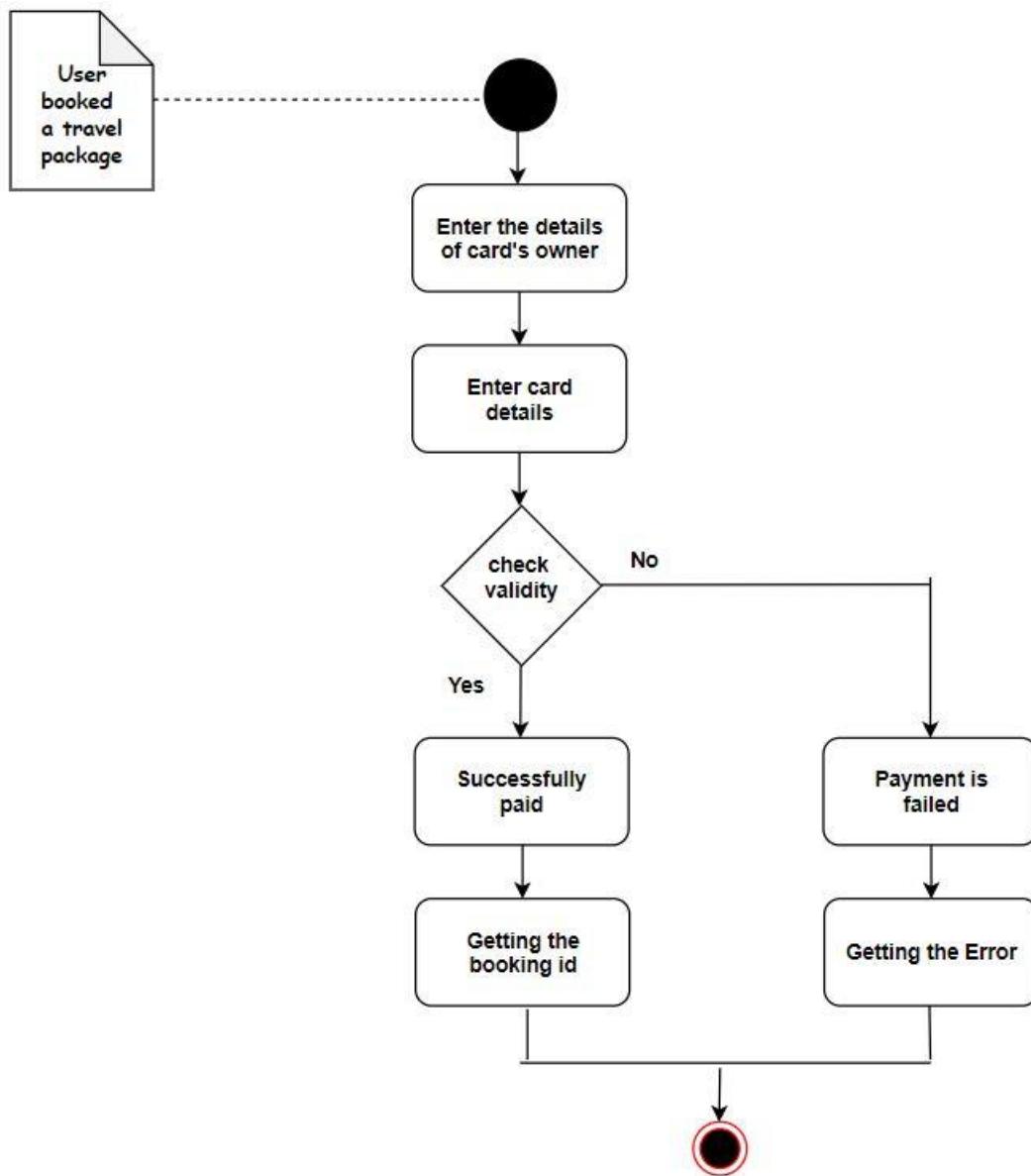


Figure 17: User does the payment

9.4.1.8 Search

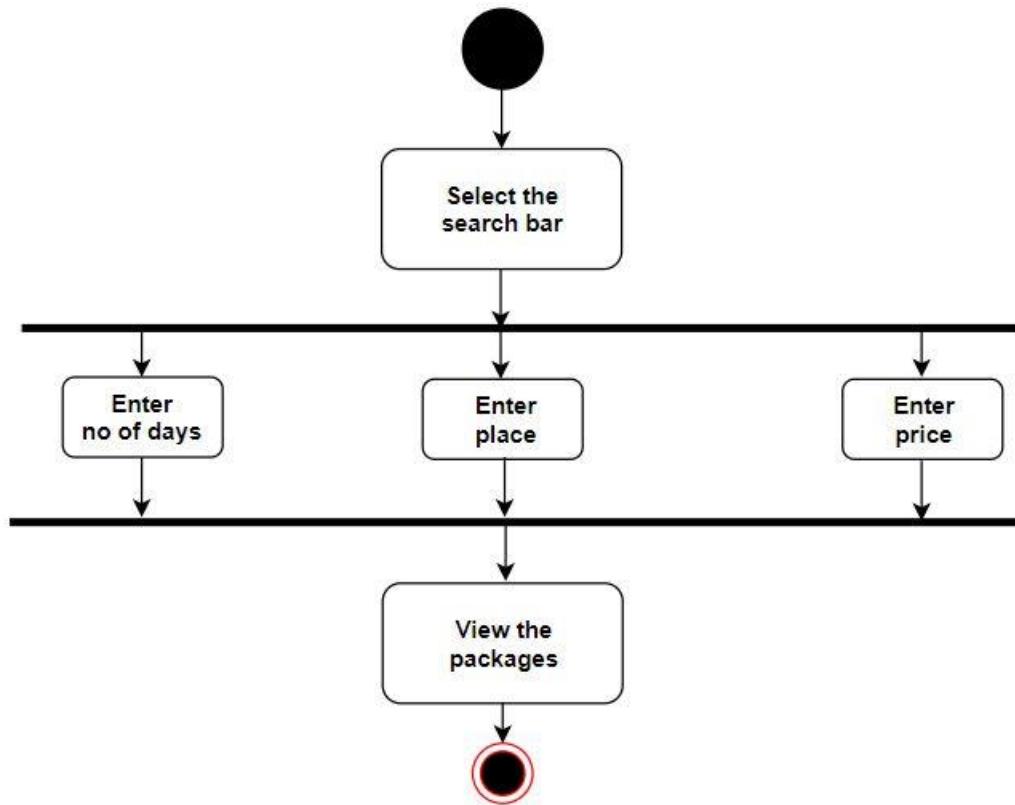


Figure 18: User search

9.4.1.9 Update the account

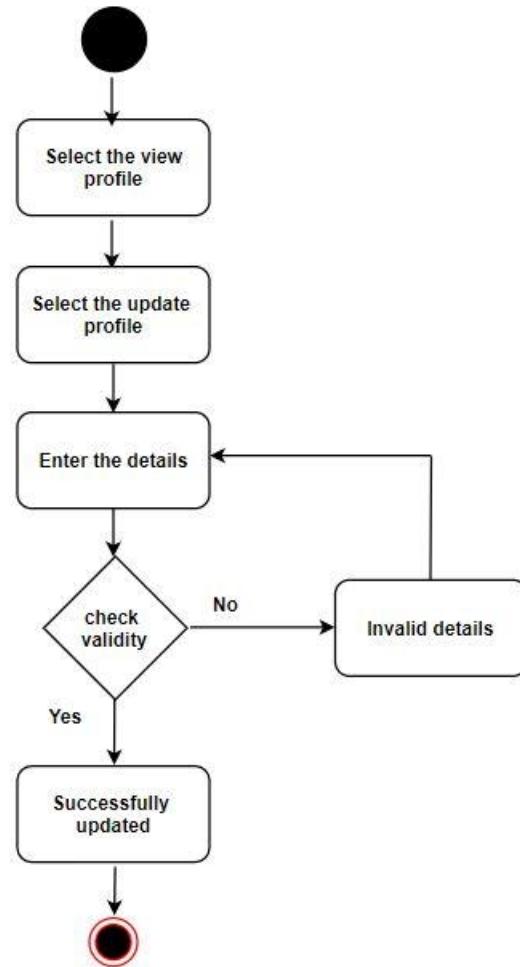


Figure 19: User updates the account

9.4.1.10 View [not registered user]

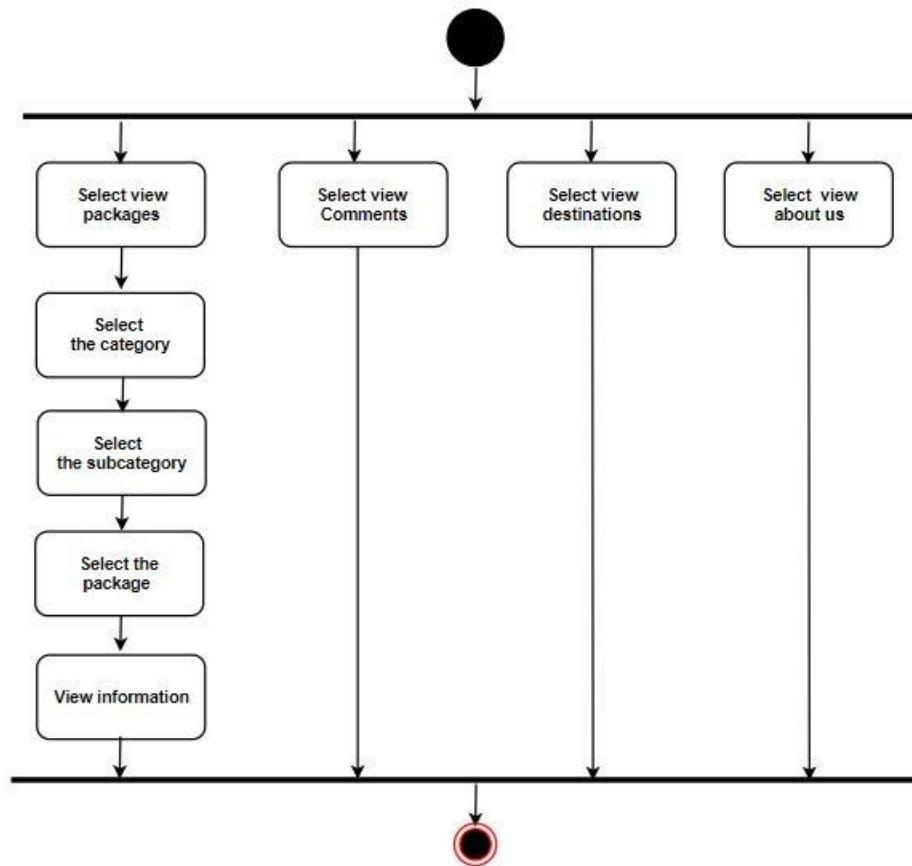


Figure 20: User view [not registered user]

9.4.1.11 View [registered user]

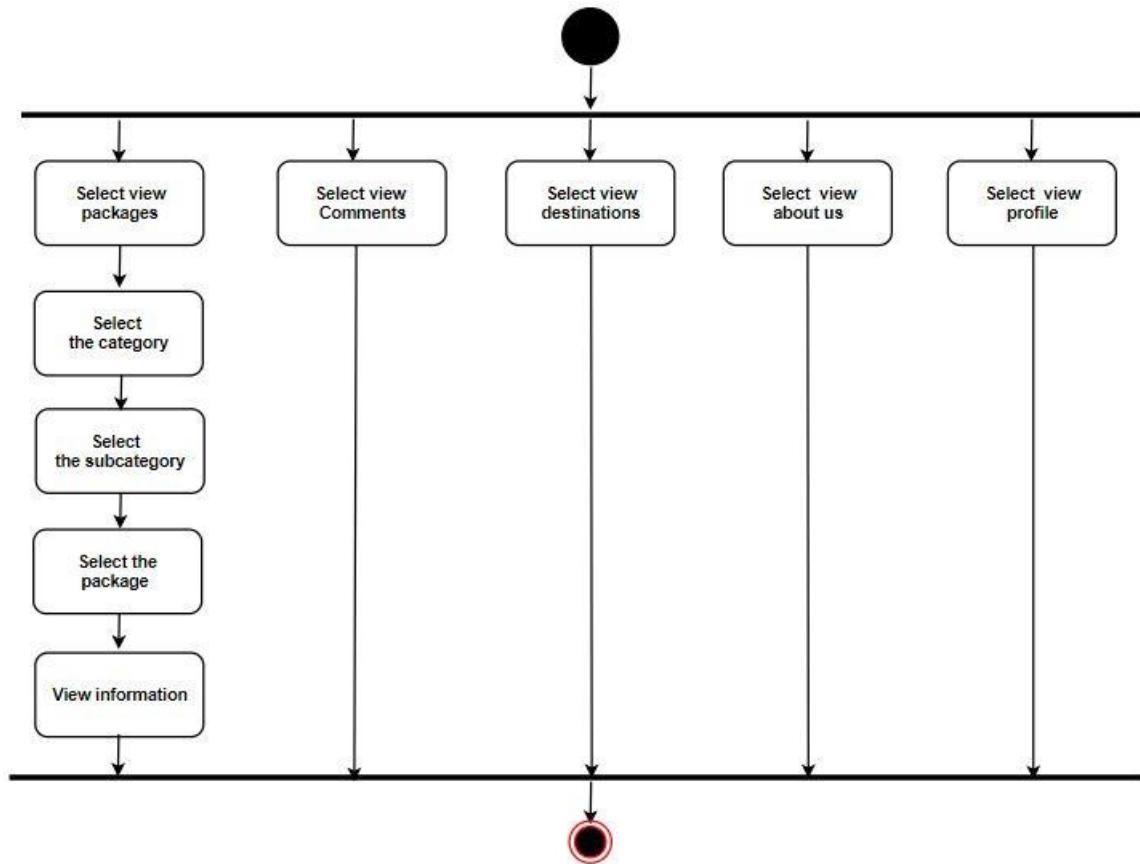


Figure 21: User view [registered user]

9.4.1.12 User sends message

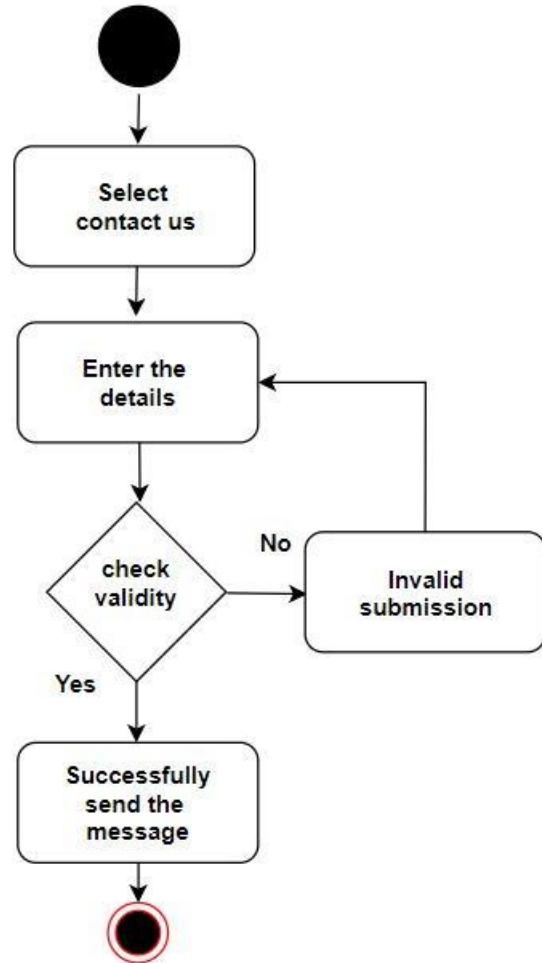


Figure 22: User sends message

9.4.2 Activity diagrams for Admin

9.4.2.1 Login to the system

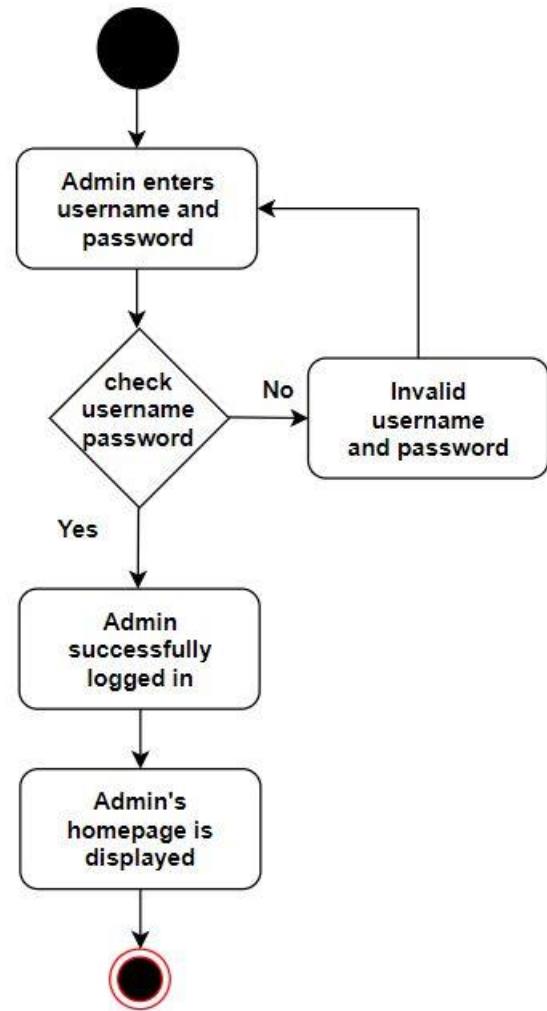


Figure 23: Admin login to the system

9.4.2.2 View accounts

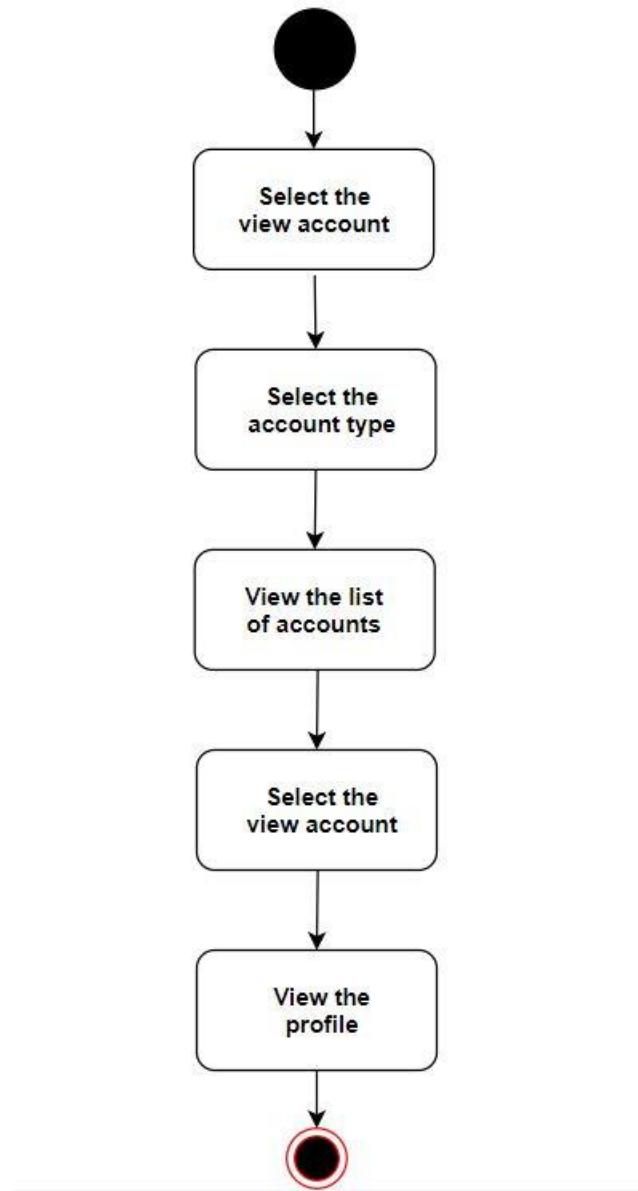


Figure 24: Admin views accounts

9.4.2.3 View bookings

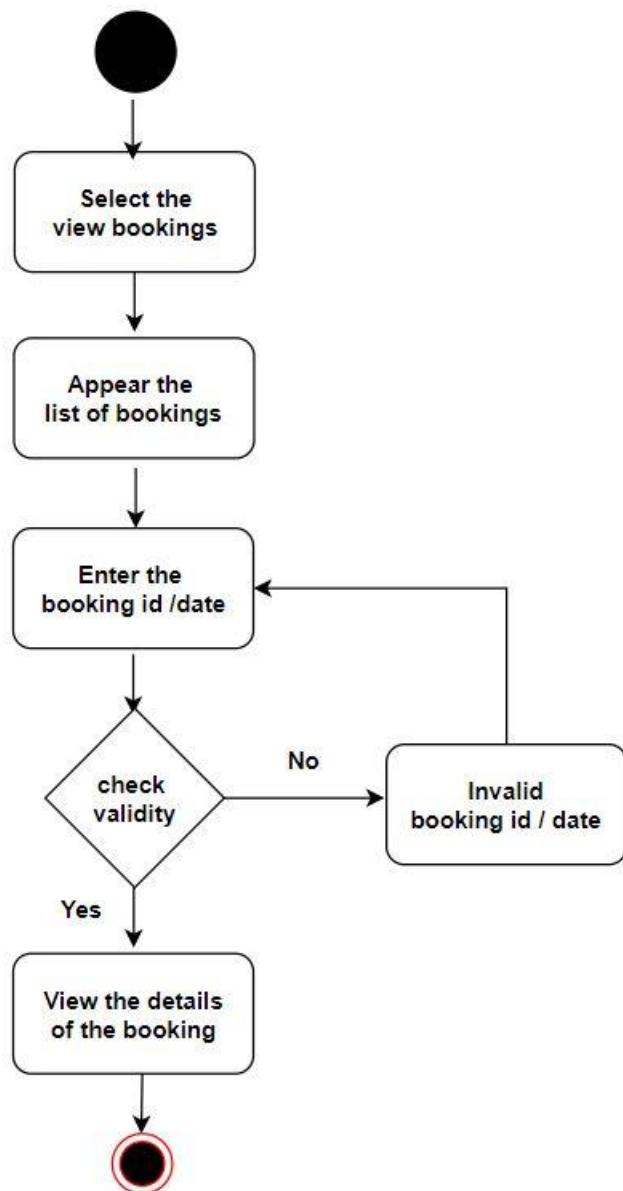


Figure 25: Admin views bookings

9.4.2.4 View reports

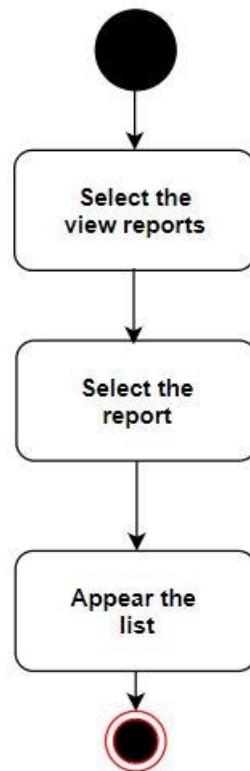


Figure 26: Admin views reports

9.4.2.5 Add travel package

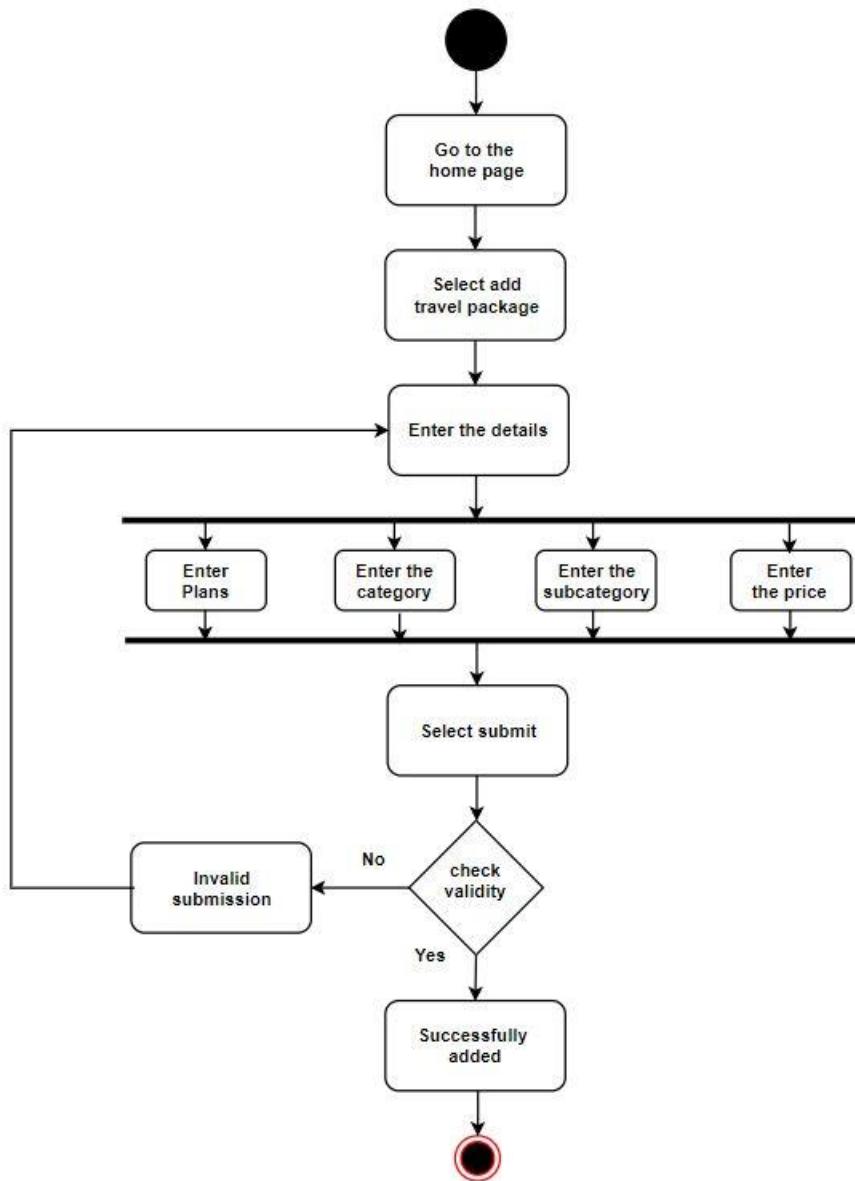


Figure 27: Admin adds travel package

9.4.2.6 Add category

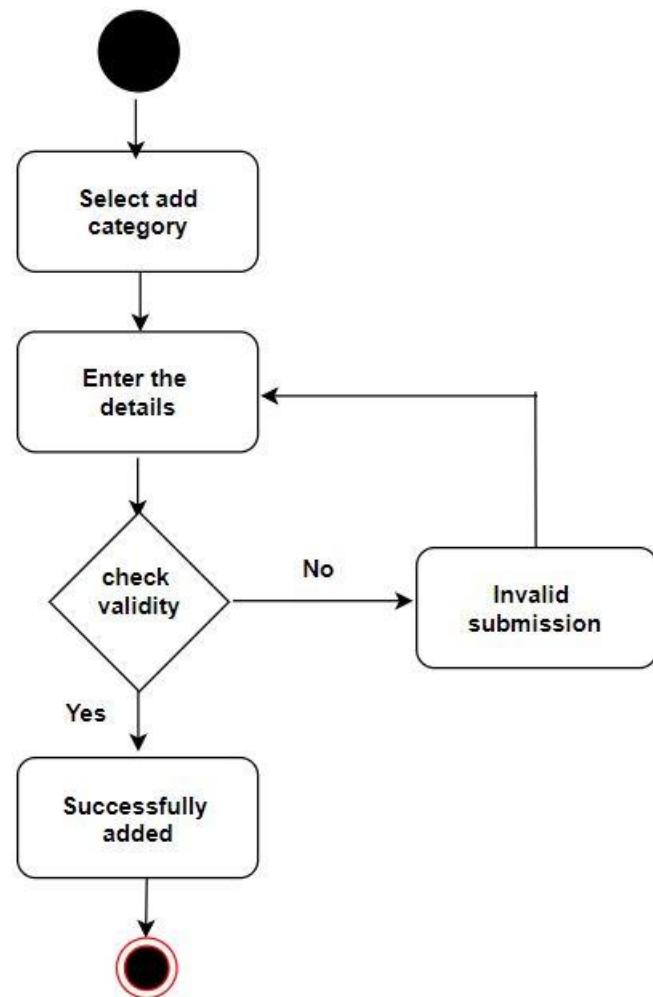


Figure 28: Admin adds category

9.4.2.7 Add subcategory

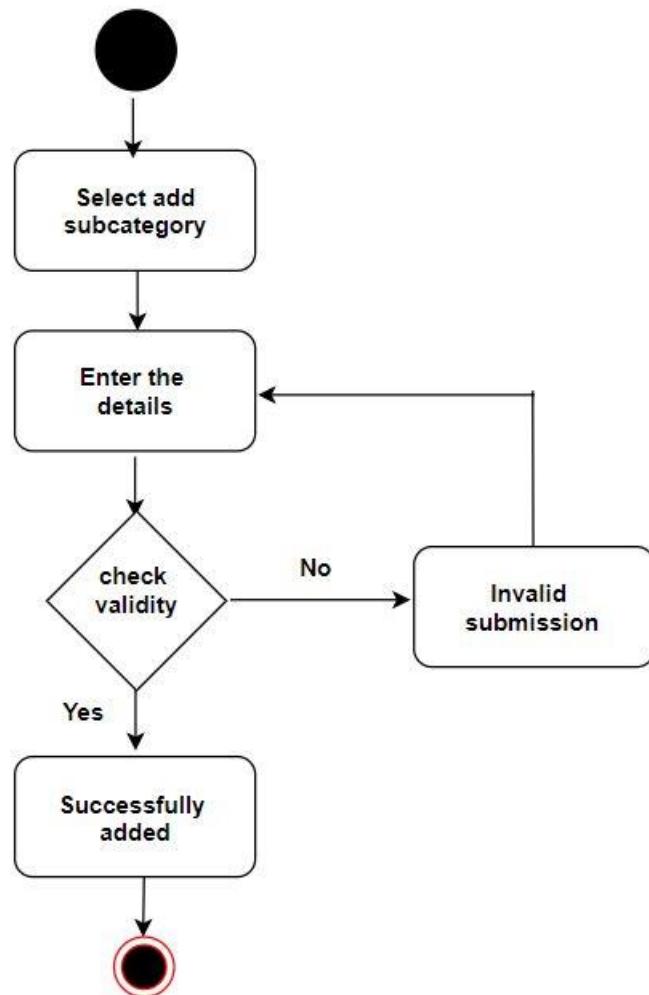


Figure 29: Admin adds subcategory

9.4.2.8 Add destinations

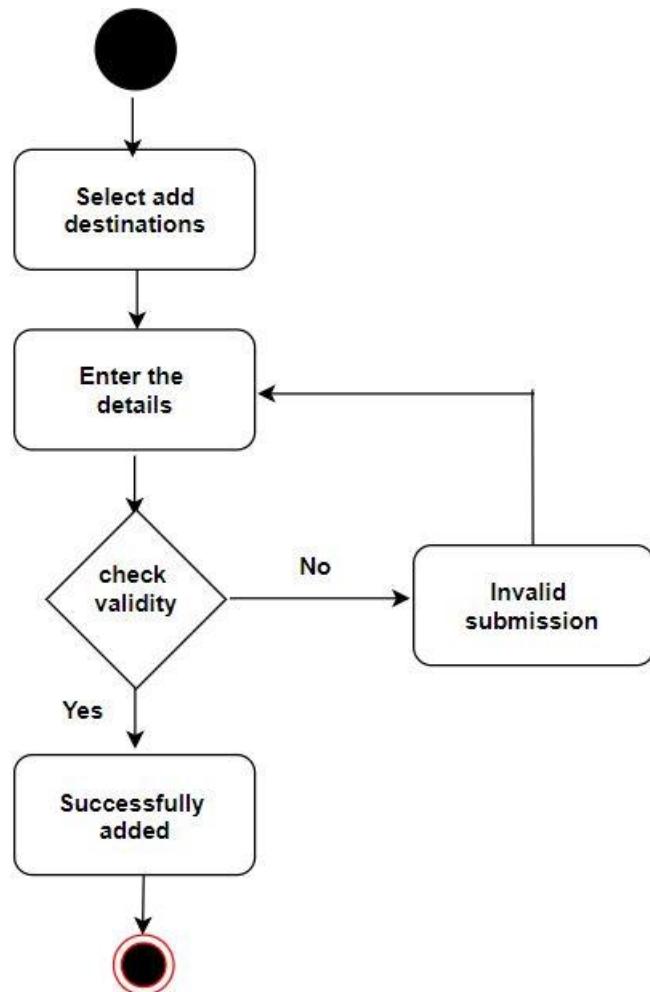


Figure 30: Admin adds destinations

9.4.2.9 Check availability

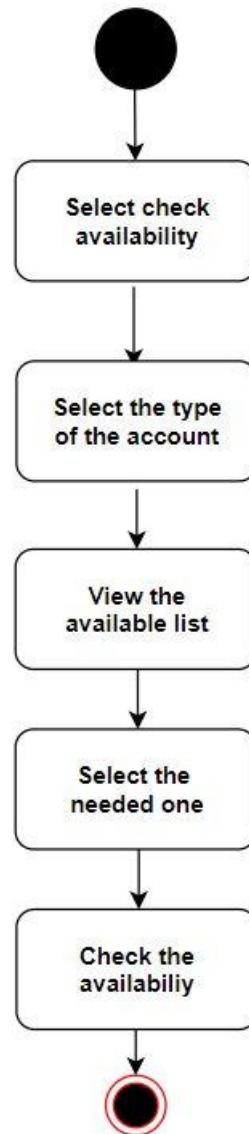


Figure 31: Admin checks availability

9.4.2.10 Delete a category

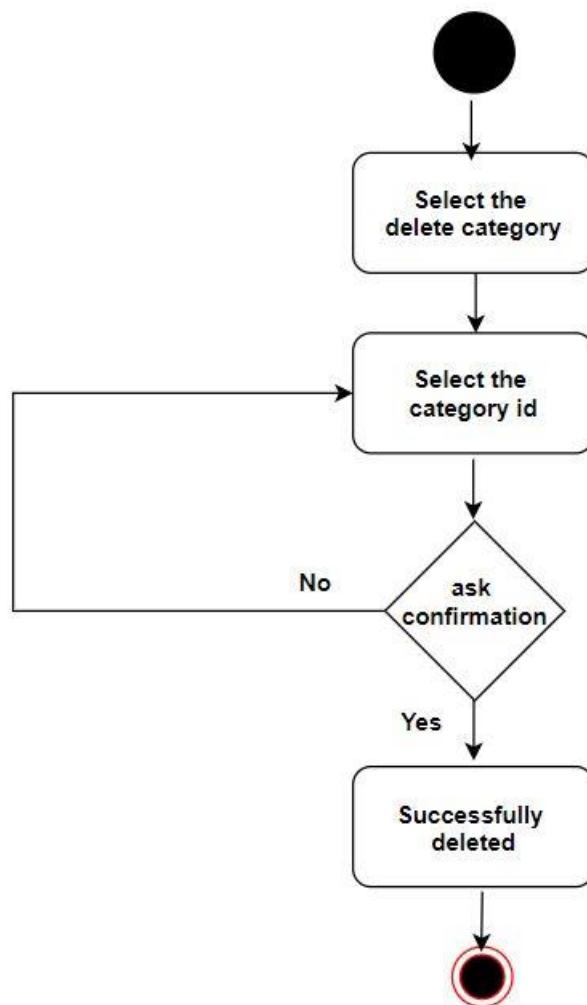


Figure 32: Admin deletes category

9.4.2.11 Delete a subcategory

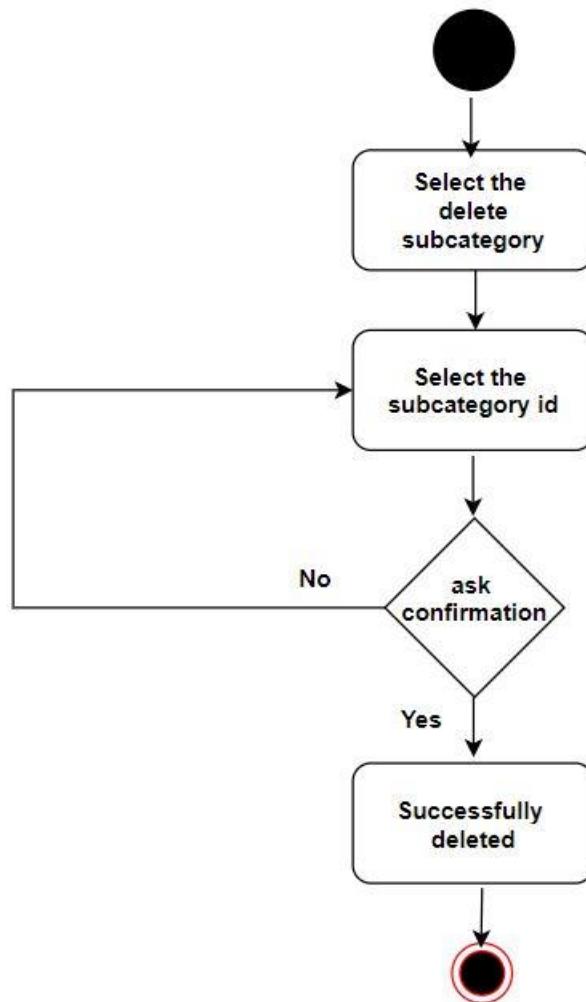


Figure 33: Admin deletes subcategory

9.4.2.12 Delete a travel package

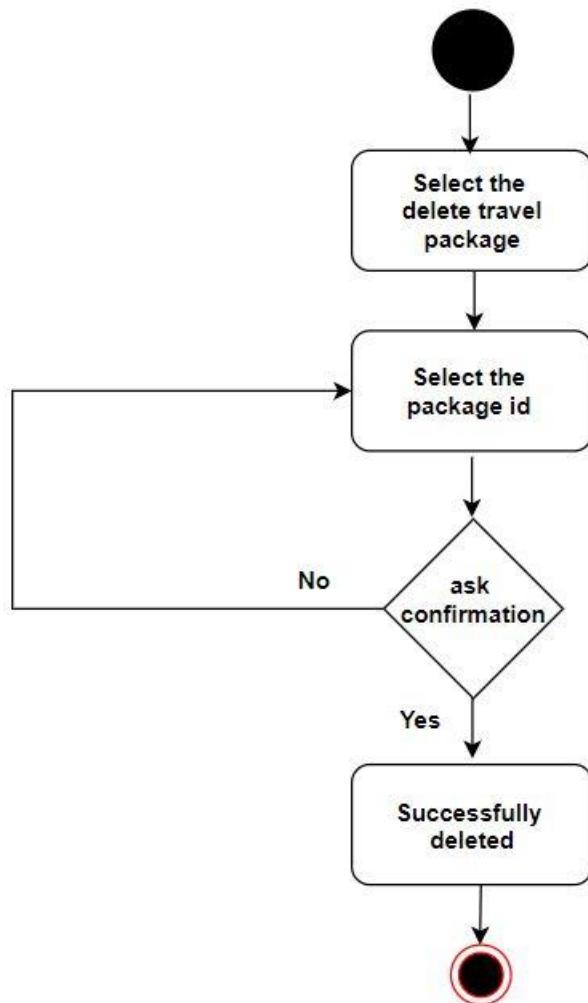


Figure 34: Admin deletes travel package

9.4.2.13 Update category

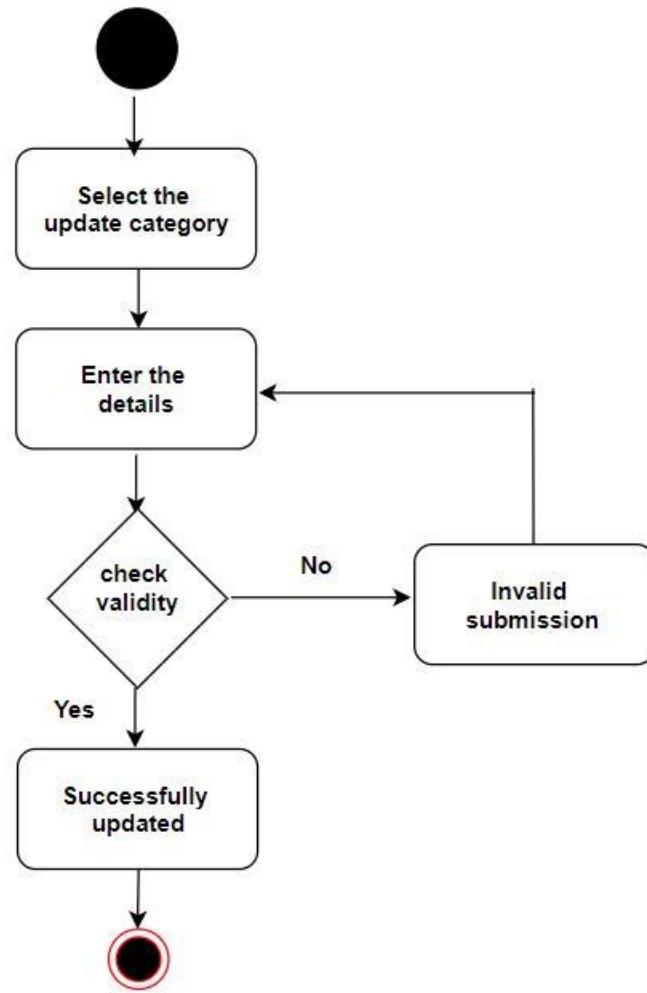


Figure 35: Admin updates category

9.4.2.14 Update subcategory

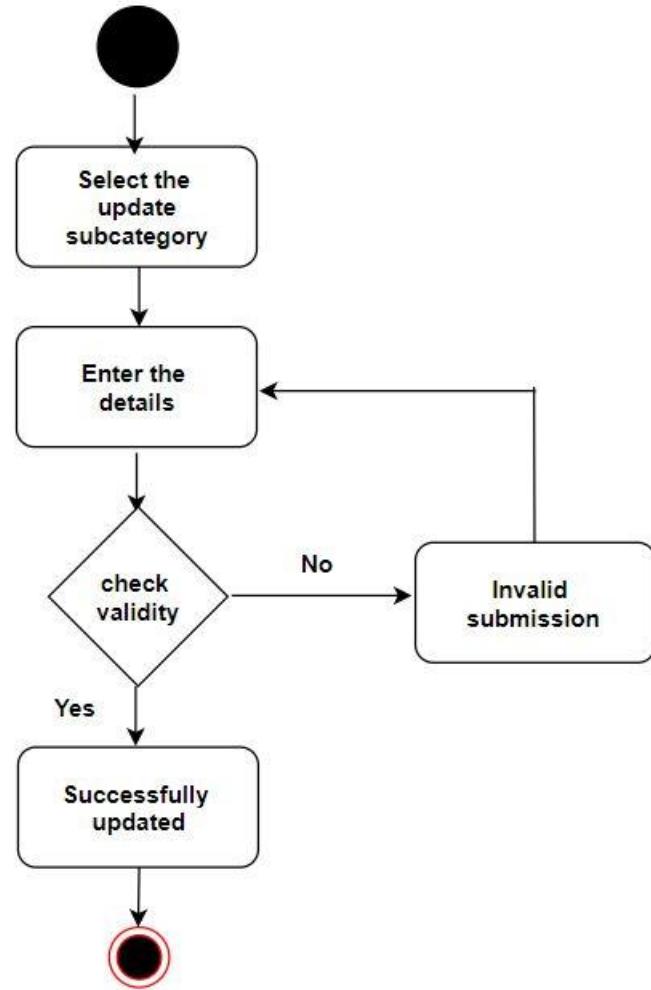


Figure 36: Admin updates subcategory

9.4.2.15 Update travel package

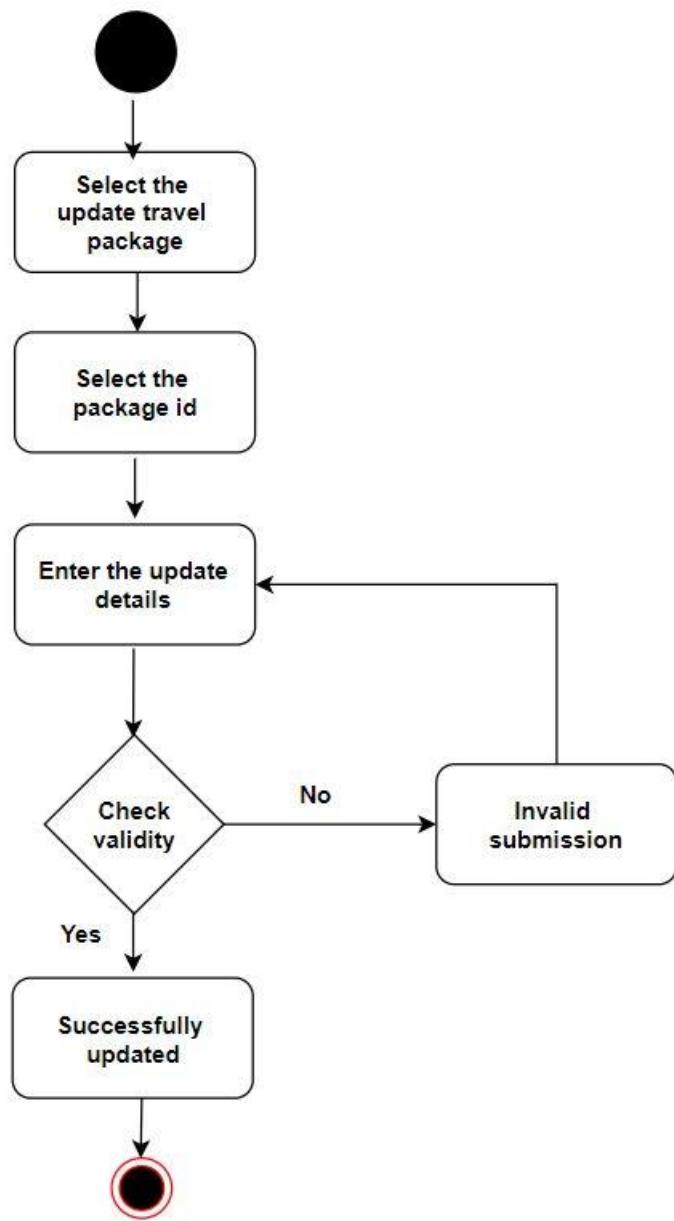


Figure 37: Admin updates travel package

9.4.3 Activity diagrams for Hotel

9.4.3.1 Login to the system

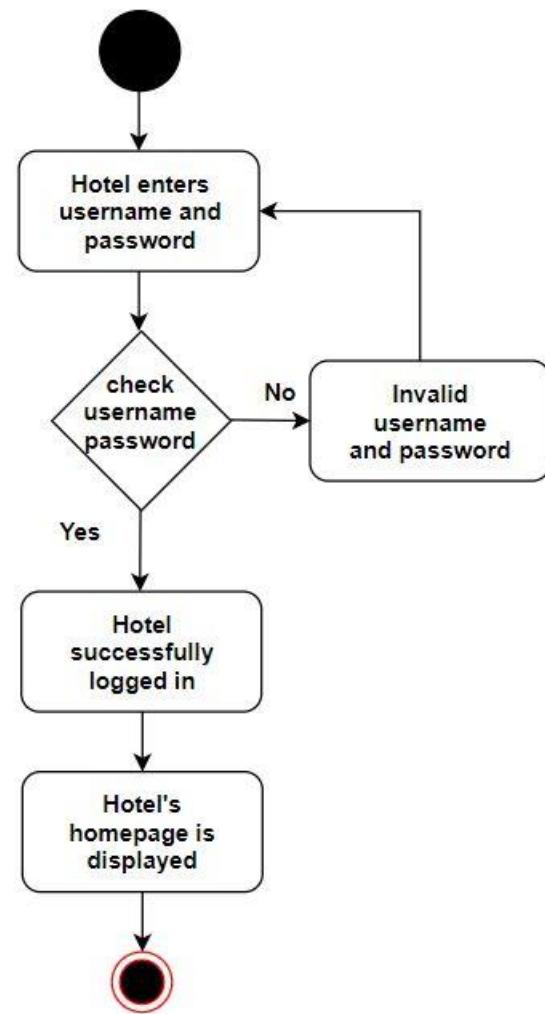


Figure 38: Hotel login to the system

9.4.3.2 Create an account

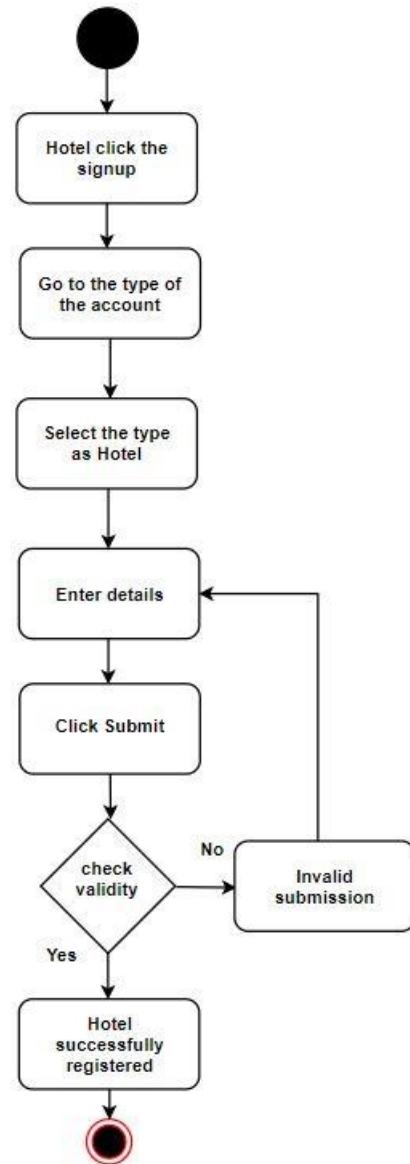


Figure 39: Hotel creates an account

9.4.3.3 Update the account

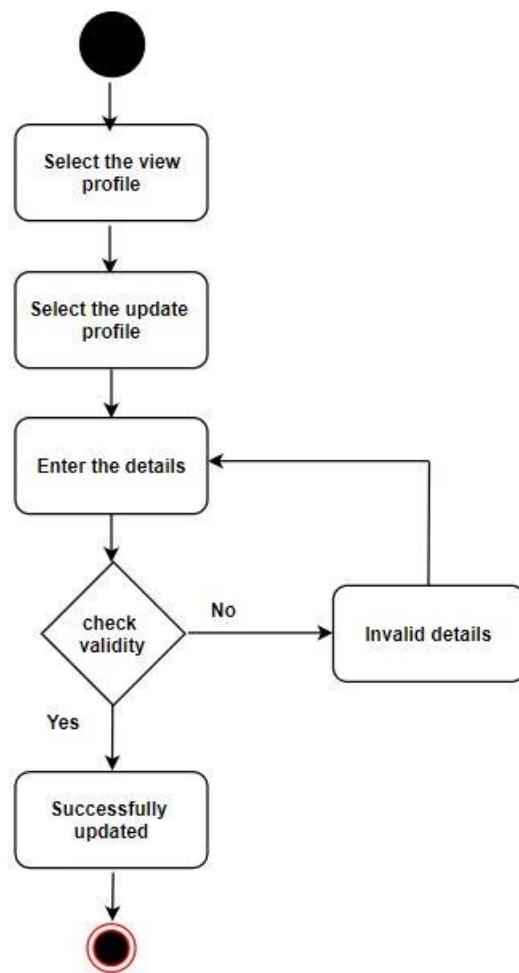


Figure 40: Hotel updates the account

9.4.3.4 Delete the account

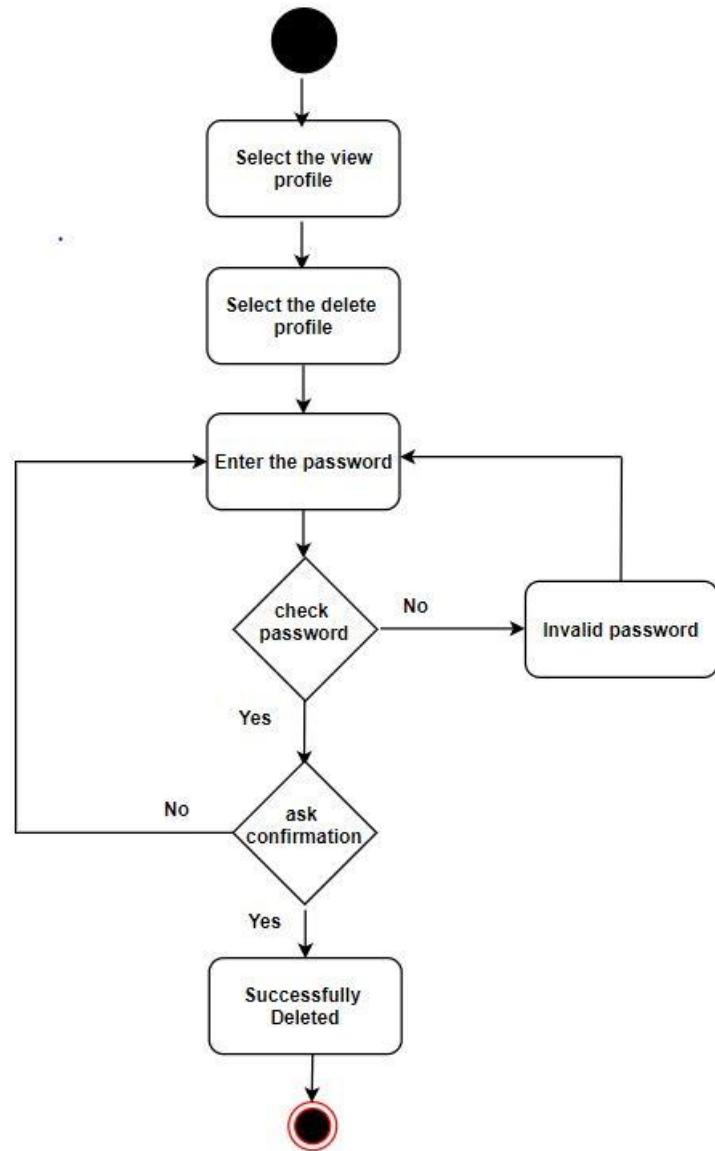


Figure 41: Hotel deletes the account

9.4.3.5 Change availability

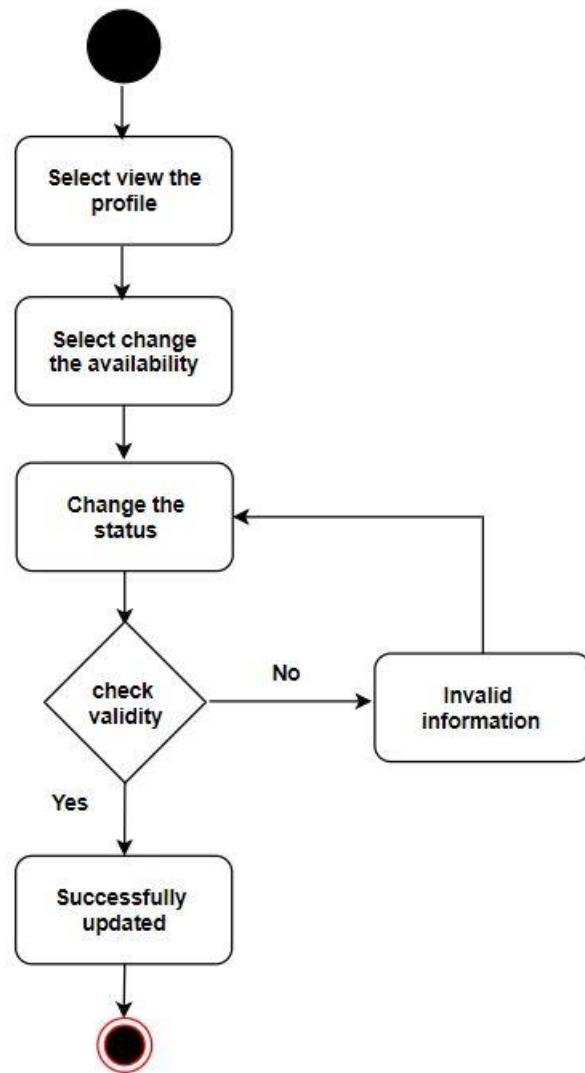


Figure 42: Hotel changes the availability

9.4.4 Activity diagrams for Guide

9.4.4.1 Login to the system

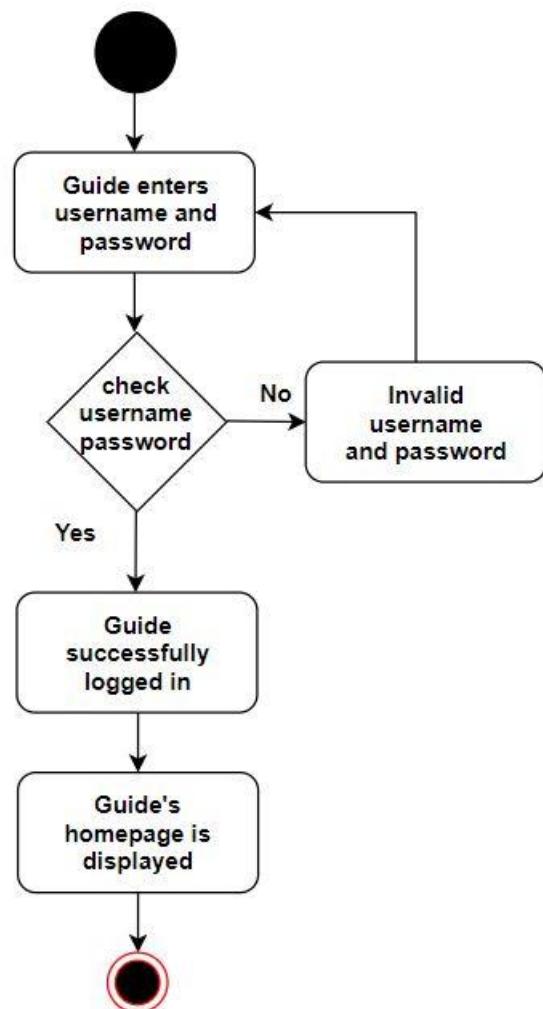


Figure 43: Guide login to the system

9.4.4.2 Create an account

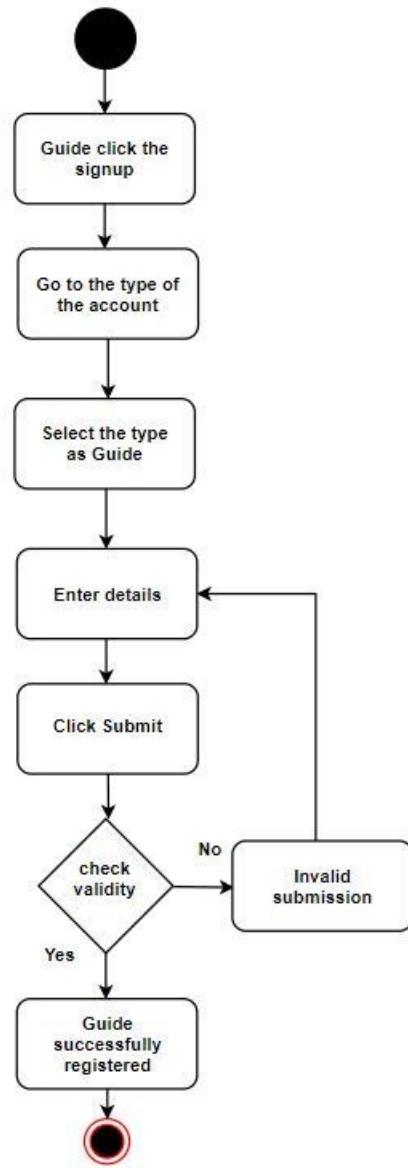


Figure 44: Guide creates an account

9.4.4.3 Update the account

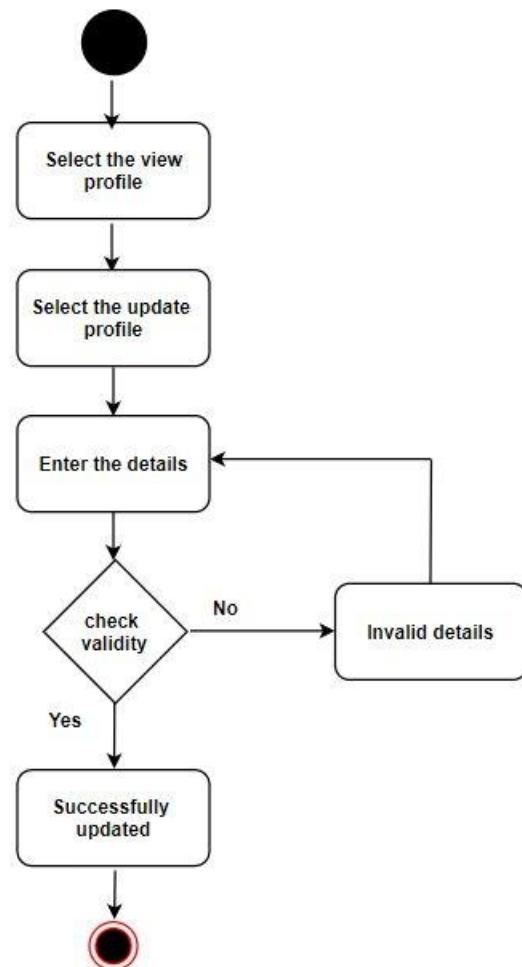


Figure 45: Guide updates the account

9.4.4.4 Delete the account

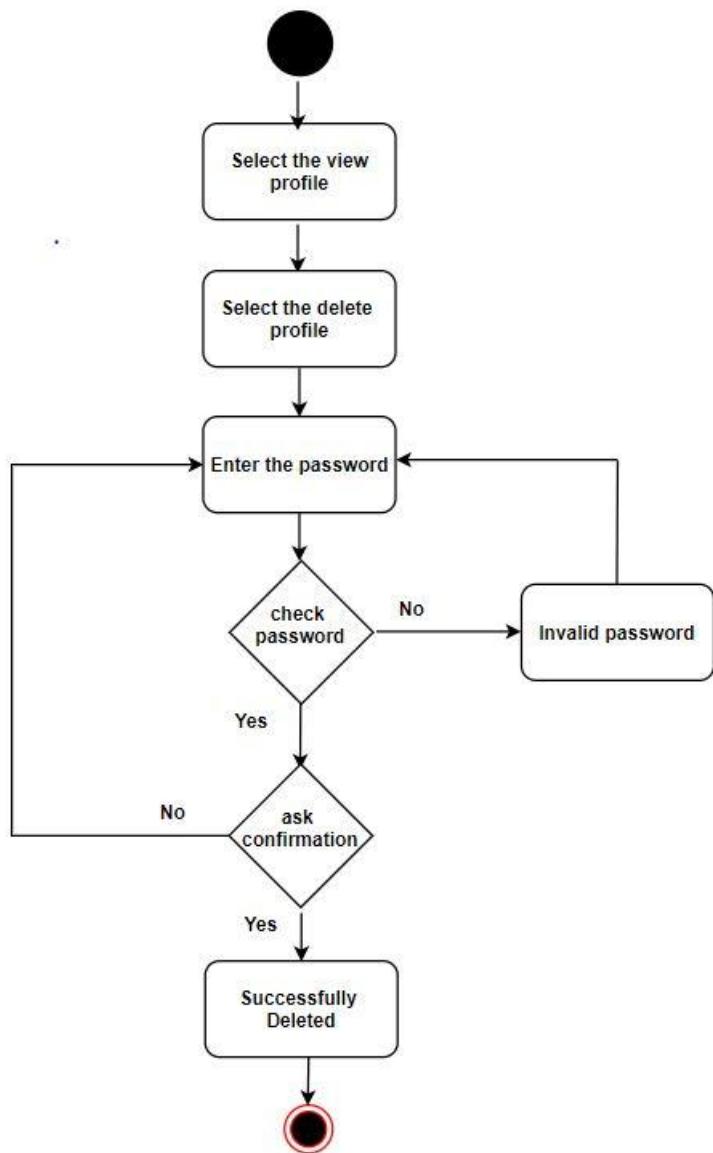


Figure 46: Guide deletes the account

9.4.4.5 Change availability

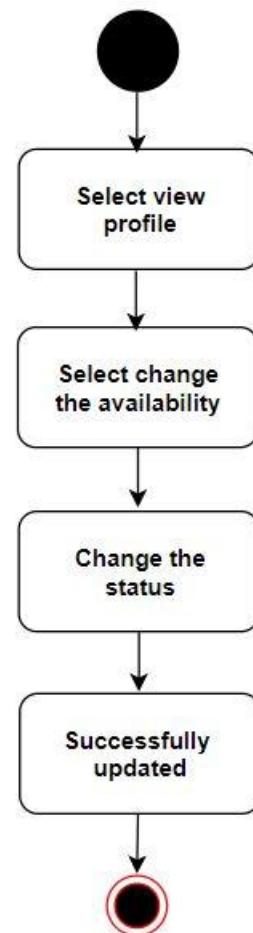


Figure 47: Guide changes the availability

9.4.5 Activity diagrams for Vehicle

9.4.5.1 Login to the system

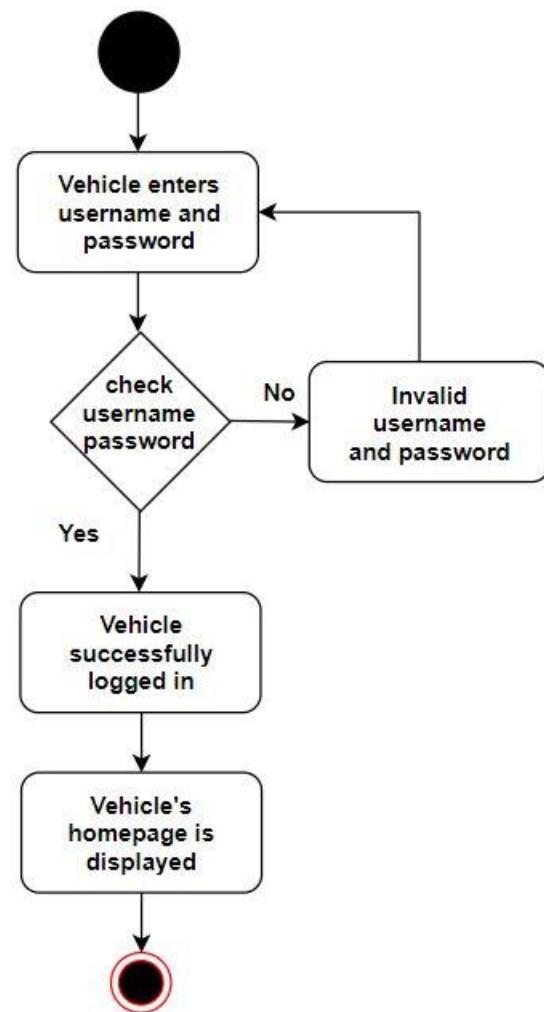


Figure 48: Vehicle login to the system

9.4.5.2 Create an account

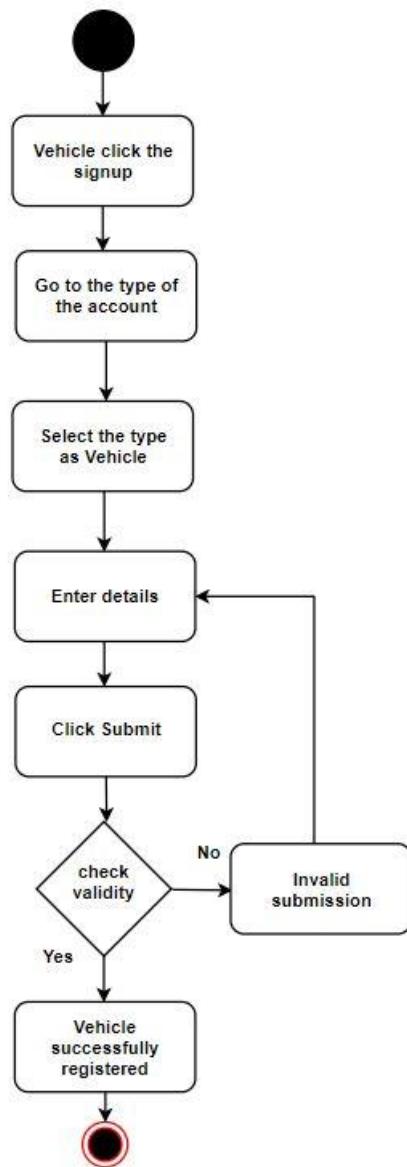


Figure 49: Vehicle creates an account

9.4.5.3 Update the account

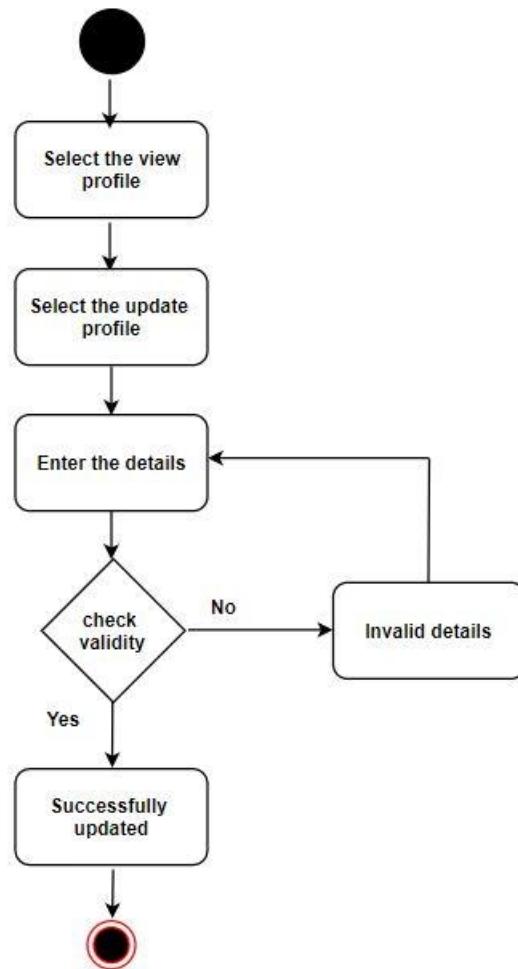


Figure 50: Vehicle updates the account

9.4.5.4 Delete the account

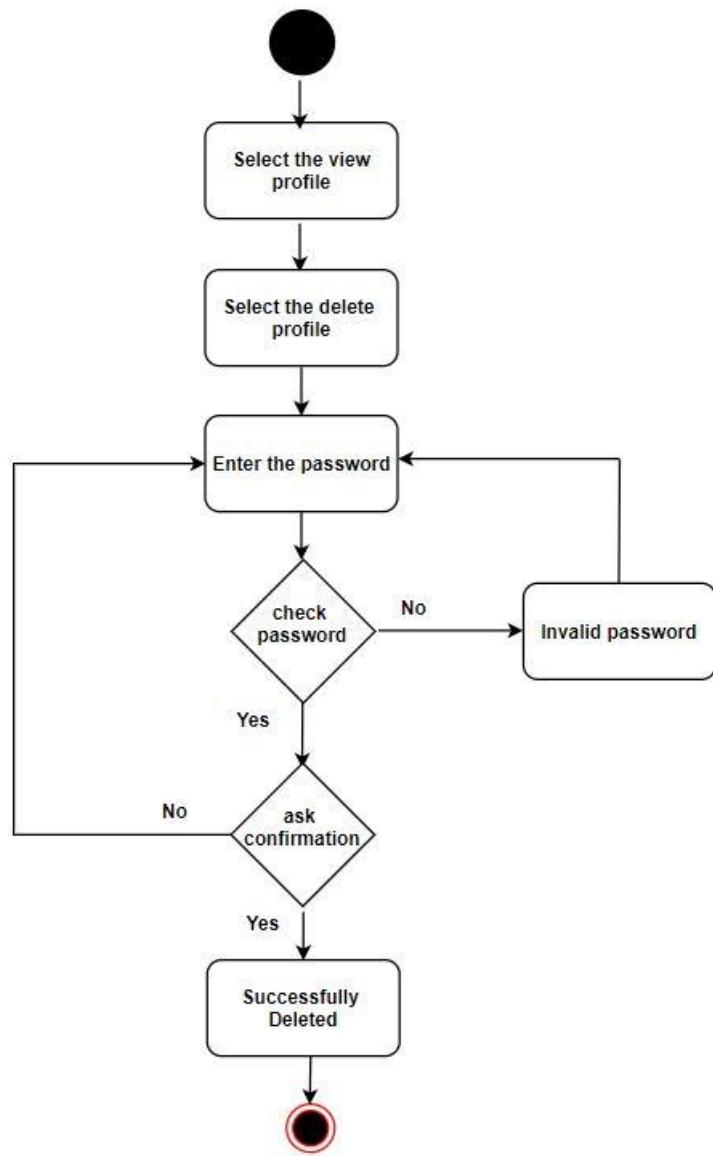


Figure 51: Vehicle deletes the account

9.4.5.5 Change availability

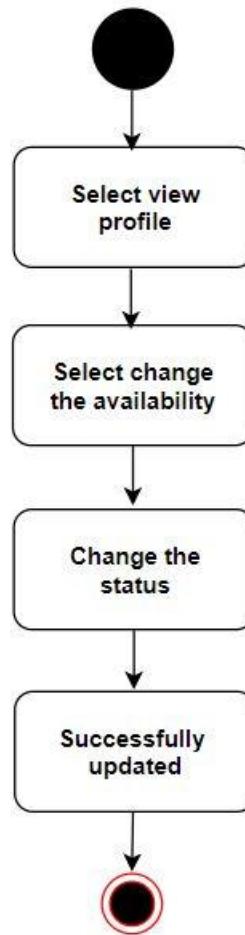


Figure 52: Vehicle changes availability

10. Non-Functional Requirements

10.1 Identification of Quality Attributes

1. Security
2. Performance
3. Usability
4. User-friendliness
5. Portability
6. Maintainability

10.2 Suggestion to achieve Quality Attributes

1. Security
 - Required passwords should be longer than 12 characters.
 - Password recovery option
 - Password will save on the database in an encrypted manner.
 - The forms are validated with the required fields.
 - Data backup option (In phpMyAdmin database has an option to export data from the database.)
2. Performance
 - We can improve the performance of the website by speeding the process, decrease the response time, etc.
 - We use PHP programming language for back-end development. So because of that applications can be easily loaded over the slow internet and data speed.
 - We can use minify and combine files to reduce the size of each file as well as the total number of files. From that system's performance will be improved.
 - We can use session variables to pass data.
3. Usability
 - We are planning to give our website to chosen users to check whether it is usable or not.
 - From this testing, we can identify what the user thinks about our website.
4. User-friendliness
 - The proposed website can be accessed easily.
 - Users will find travel packages very easily. Because all the packages are divided into categories and subcategories.
 - Considering the clean and clear user interfaces, it will help the user to access and affect the website.

- As mentioned before, we are planning to have user testing. So we can get an idea about how they feel about the user-friendliness of our website.

5. Portability

- We are planning to implement our website responsively. So it will compatible with operating systems and also the mobiles.
- So the can use any device to see a good output.

6. Maintainability

- We will use readable codes which can easily understand.
- We will use comments to identify the source codes. So code can easily change.
- We will use automated codes. That will make the code easy to compile.
- We will use automated validation parts.

11. Technologies to be used.

1. Programming languages
 - ❖ Front-end programming languages
 - HTML
 - CSS
 - JavaScript
 - ❖ Back-end programming languages
 1. PHP
 - ❖ Database
 2. MySQL
2. PHP software requirements
 - ❖ WAMP server [Microsoft Windows o/s, Apache MySQL PHP]
3. MySQL administrator tool
 - ❖ phpMyAdmin
4. Diagram drawing tool
 - ❖ draw.io
5. Documentation tools
 - ❖ Microsoft Word
 - ❖ Microsoft Excel
6. Project management software
 - ❖ Microsoft Project (To create Gantt chart and timeline)
7. Text editing tools
 - ❖ Visual Studio Code
 - ❖ Notepad++
8. Contribution platform
 - ❖ GitHub

12. Work Breakdown Structure

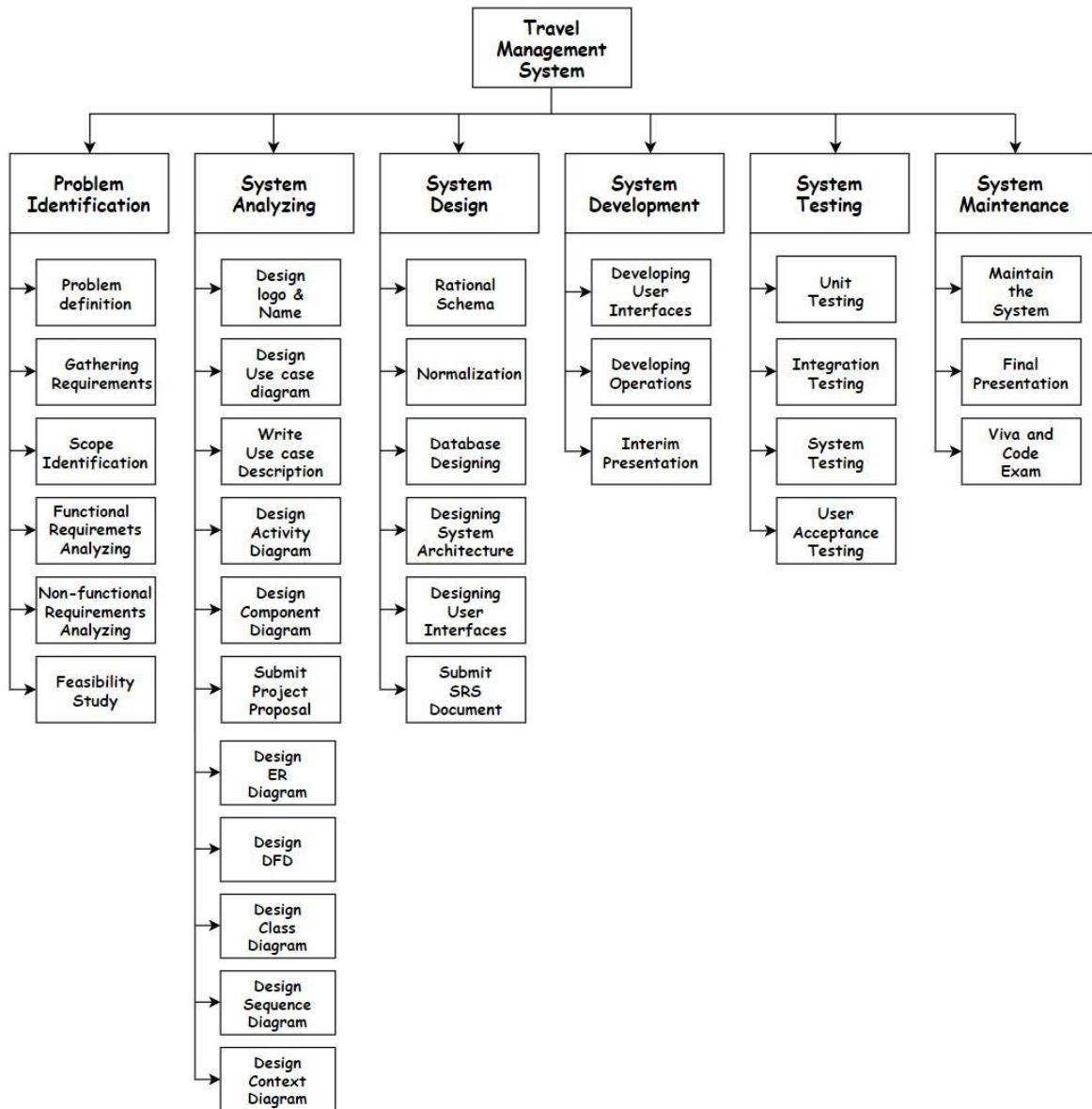


Figure 53: Work Breakdown Structure

13. Project Timeline

13.1 Block Timeline

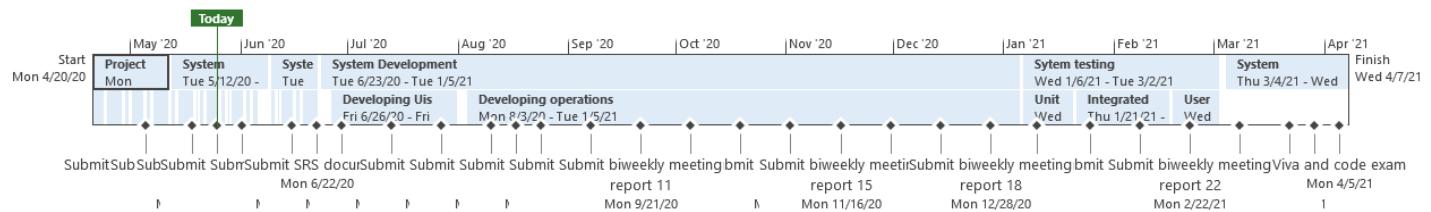


Figure 54: Block Timeline-1

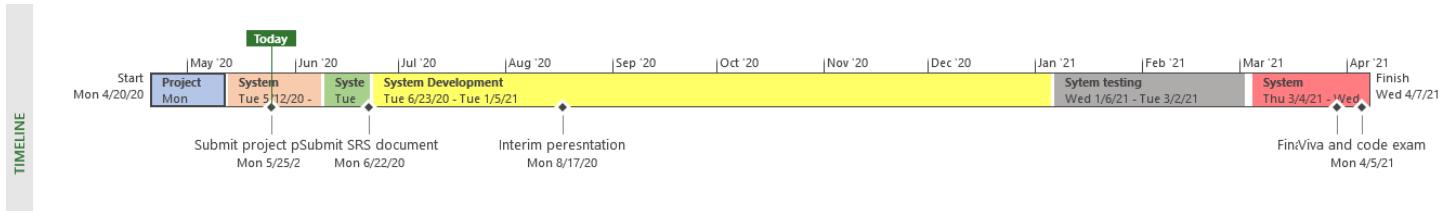


Figure 55: Block Timeline-2

13.2 Project Milestone Timeline

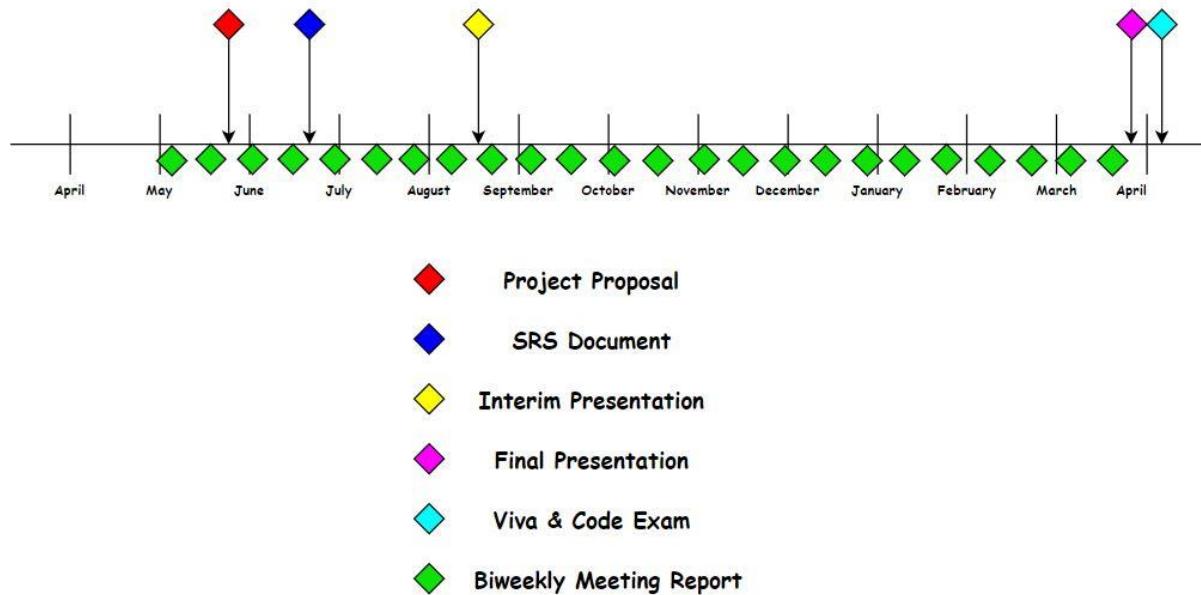
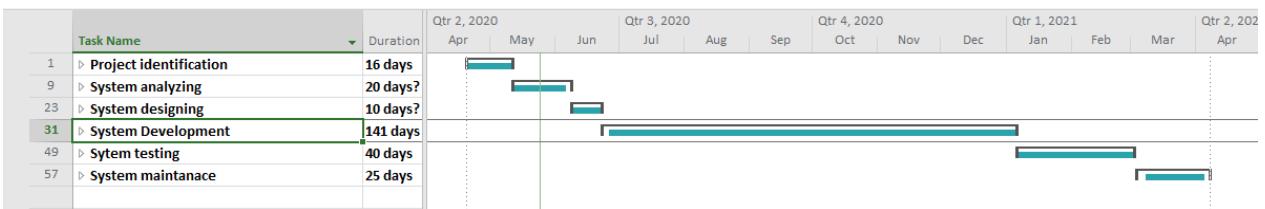


Figure 56: Project Milestone Timeline

14. Gantt Chart

	Task Mode	Task Name	Duration	Start	Finish	Predecessors
1	➡	Project identification	16 days	Mon 4/20/20	Mon 5/11/20	
2	➡	Problem definition	4 days	Mon 4/20/20	Thu 4/23/20	
3	➡	Requirement gathering	3 days	Fri 4/24/20	Tue 4/28/20	2
4	➡	Scope identification	2 days	Wed 4/29/20	Thu 4/30/20	3
5	➡	Functional requirements gathering	2 days	Fri 5/1/20	Mon 5/4/20	4
6	➡	Submit biweekly meeting report 1	0 days	Tue 5/5/20	Tue 5/5/20	
7	➡	Non-functional requirements gathering	2 days	Tue 5/5/20	Wed 5/6/20	5
8	➡	Feasibility study	3 days	Thu 5/7/20	Mon 5/11/20	7
9	➡	System analyzing	20 days?	Tue 5/12/20	Mon 6/8/20	
10	➡	Design logo and name	2 days	Tue 5/12/20	Wed 5/13/20	8
11	➡	Design use case diagram	3 days	Thu 5/14/20	Mon 5/18/20	10
12	➡	Submit biweekly meeting report 2	0 days	Mon 5/18/20	Mon 5/18/20	
13	➡	Writing use case description	1 day	Tue 5/19/20	Tue 5/19/20	11
14	➡	Designing activity diagrams	2 days	Wed 5/20/20	Thu 5/21/20	13
15	➡	Designing component diagram	2 days	Fri 5/22/20	Mon 5/25/20	14
16	➡	Submit project proposal	0 days	Mon 5/25/20	Mon 5/25/20	
17	➡	Designing ER diagram	2 days	Tue 5/26/20	Wed 5/27/20	15
18	➡	Designing DFDs	1 day?	Thu 5/28/20	Thu 5/28/20	17
19	➡	Designing class diagram	2 days	Fri 5/29/20	Mon 6/1/20	18
20	➡	Submit biweekly meeting report 3	0 days	Mon 6/1/20	Mon 6/1/20	
21	➡	Designing sequence diagram	2 days	Tue 6/2/20	Wed 6/3/20	19
22	➡	Designing context diagram	2 days	Thu 6/4/20	Fri 6/5/20	21
23	➡	System designing	10 days?	Tue 6/9/20	Mon 6/22/20	
24	➡	Relational schema	2 days	Tue 6/9/20	Wed 6/10/20	22
25	➡	Normalization	1 day?	Thu 6/11/20	Thu 6/11/20	24
26	➡	Database design	2 days	Fri 6/12/20	Mon 6/15/20	25
27	➡	Submit biweekly meeting report 4	0 days	Mon 6/15/20	Mon 6/15/20	
28	➡	Designing system architecture	2 days	Tue 6/16/20	Wed 6/17/20	26
29	➡	Designing user interfaces	3 days	Thu 6/18/20	Mon 6/22/20	28
30	➡	Submit SRS document	0 days	Mon 6/22/20	Mon 6/22/20	
31	➡	System Development	141 days	Tue 6/23/20	Tue 1/5/21	
32	➡	Submit biweekly meeting report 5	0 days	Mon 6/29/20	Mon 6/29/20	
33	➡	Developing Uis	26 days	Fri 6/26/20	Fri 7/31/20	29
34	➡	Submit biweekly meeting report 6	0 days	Mon 7/13/20	Mon 7/13/20	
35	➡	Submit biweekly meeting report 7	0 days	Mon 7/27/20	Mon 7/27/20	
36	➡	Developing operations	112 days	Mon 8/3/20	Tue 1/5/21	33
37	➡	Interim peresntation	0 days	Mon 8/17/20	Mon 8/17/20	
38	➡	Submit biweekly meeting report 8	0 days	Mon 8/10/20	Mon 8/10/20	
39	➡	Submit biweekly meeting report 9	0 days	Mon 8/24/20	Mon 8/24/20	
40	➡	Submit biweekly meeting report 10	0 days	Mon 9/7/20	Mon 9/7/20	
41	➡	Submit biweekly meeting report 11	0 days	Mon 9/21/20	Mon 9/21/20	
42	➡	Submit biweekly meeting report 12	0 days	Mon 10/5/20	Mon 10/5/20	
43	➡	Submit biweekly meeting report 13	0 days	Mon 10/19/20	Mon 10/19/20	
44	➡	Submit biweekly meeting report 14	0 days	Mon 11/2/20	Mon 11/2/20	

45	Submit biweekly meeting report 15	0 days	Mon 11/16/20	Mon 11/16/20		
46	Submit biweekly meeting report 16	0 days	Mon 11/30/20	Mon 11/30/20		
47	Submit biweekly meeting report 17	0 days	Mon 12/14/20	Mon 12/14/20		
48	Submit biweekly meeting report 18	0 days	Mon 12/28/20	Mon 12/28/20		
49	↳ System testing	40 days	Wed 1/6/21	Tue 3/2/21		
50	Unit testing	11 days	Wed 1/6/21	Wed 1/20/21	36	
51	Integrated testing	19 days	Thu 1/21/21	Tue 2/16/21	50	
52	Submit biweekly meeting report 19	0 days	Sun 1/10/21	Sun 1/10/21		
53	User testing	10 days	Wed 2/17/21	Tue 3/2/21	51	
54	Submit biweekly meeting report 20	0 days	Mon 1/25/21	Mon 1/25/21		
55	Submit biweekly meeting report 21	0 days	Mon 2/8/21	Mon 2/8/21		
56	Submit biweekly meeting report 22	0 days	Mon 2/22/21	Mon 2/22/21		
57	↳ System maintenance	25 days	Thu 3/4/21	Wed 4/7/21		
58	Submit biweekly meeting report 23	0 days	Mon 3/8/21	Mon 3/8/21		
59	Submit biweekly meeting report 24	0 days	Mon 3/22/21	Mon 3/22/21		
60	Final presentation	0 days	Mon 3/29/21	Mon 3/29/21		
61	Viva and code exam	0 days	Mon 4/5/21	Mon 4/5/21		



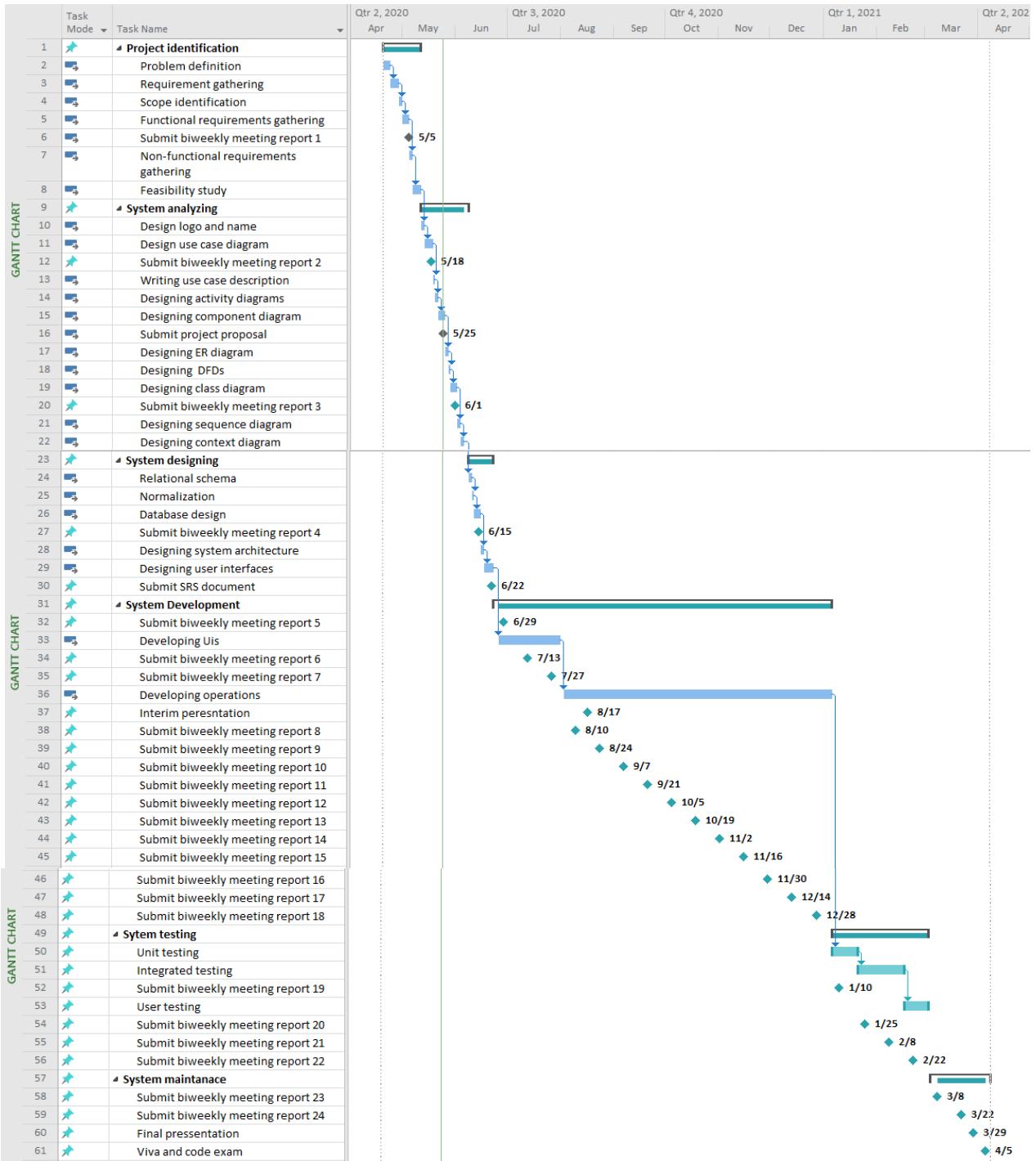
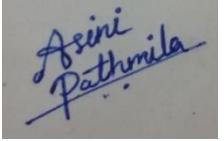
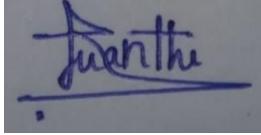
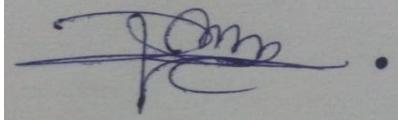
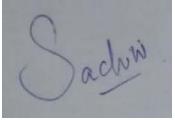


Figure 57: Gantt chart

15. Declaration

We as members of the project titled Travel Management System, Certify that we will carry out this project according to the guidelines provided by the coordinators and supervisors of the course as well as we will not incorporate, without acknowledgement, any material previously submitted for a degree or diploma in any university. To the best of our knowledge and brief, the project work will not contain any material previously published or written by another person or ourselves except where due reference is made in the text of appropriate places.

Index Number	Name of the Student	Signature
Asini Pathmila Silva	18020798	
Ruwanthi Hemachandra	18020291	
Hansaka Sandaruwan	18020704	
Sachini Maleesha	18020356	

16. References

- 1) iamglyka. 2020. *Schedule Feasibility*. [online] Available at: <<https://iamglyka.wordpress.com/2013/02/06/schedule-feasibility/>> [Accessed 4 May 2020].
- 2) Web.cse.ohio-state.edu. 2020. *Requirements Document Example*. [online] Available at: <http://web.cse.ohio-state.edu/~bair.41/616/Project/Example_Document/Req_Doc_Example.html> [Accessed 4 May 2020].
- 3) Programsformca.com. 2020. *UML Diagrams For Tour Management*. [online] Available at: <<http://www.programsformca.com/2012/03/uml-diagrams-for-tour-management.html>> [Accessed 4 May 2020].
- 4) Apps, B., 2020. *Technology Stack For Web Application Development*. [online] Rubygarage.org. Available at: <<https://rubygarage.org/blog/technology-stack-for-web-development>> [Accessed 4 May 2020].
- 5) Michael Bierman's Randomness. 2020. *Project Feasibility: Should We Include Ethical Feasibility?*. [online] Available at: <<https://michaelbierman.wordpress.com/2010/09/15/project-feasibility-should-we-include-ethical-feasibility/>> [Accessed 4 May 2020].
- 6) Simplilearn.com. 2020. *What Is Feasibility Study And Its Importance In Project Management*. [online] Available at: <<https://www.simplilearn.com/feasibility-study-article>> [Accessed 4 May 2020].
- 7) Usmani, F., 2020. *Assumptions And Constraints In Project Management / PM Study Circle*. [online] PM Study Circle. Available at: <<https://pmstudycircle.com/2012/10/assumptions-and-constraints-in-project-management/>> [Accessed 8 May 2020].
- 8) Masterofproject.com. 2020. *Master / Project Constraints Definition / 7 Common Project Constraints*. [online] Available at: <<https://masterofproject.com/blog/7390/project-constraints-definition-7-common-project-constraints>> [Accessed 8 May 2020].