



Daffodil
International
University

Transport Finder

Submitted By

Mourin Mondol

ID: 152-35-1298

Batch: 17th

Department of Software Engineering

Daffodil International University

Supervised By

Ftatama Binta Rafiq

Lecturer

Department of Software Engineering

Daffodil International University

Department of Software Engineering
DAFFODIL INTERNATIONAL UNIVERSITY

Spring – 2019

APPROVAL

This **project** titled “**Transport Finder**” submitted by **Mourin Mondol, 152-35-1298** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirement for the degree of B.Sc. in Software Engineering and approved as to its style and contents.

BOARD OF EXAMINERS

Dr. Touhid Bhuiyan
Professor and Head

Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Chairman

K. M. Imtiaz-Ud-Din
Assistant Professor

Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 1

Md. Shohel Arman
Lecturer

Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 2

Mr. Shafiul Hasan
Managing Director

Vivacom Solution, Dhaka

External Examiner

DECLARATION

I hereby declare that I have taken this project under the supervision Fatama Binta Rafiq, Lecturer, Department of Software Engineering, Daffodil International University. I also declare that neither this thesis/project nor any part of this has been submitted elsewhere for award of any degree.

.....

Mourin Mondol

ID: 152-35-1298

Batch: 17th

Department of Software Engineering

Faculty of Science and Information Technology

Daffodil International University

Certified by:

.....

Fatama Binta Rafiq

Lecturer

Department of Software Engineering

Faculty of Science and Information Technology

Daffodil International University

Acknowledgement

First of all, I am grateful to the Almighty Allah for making me eligible to complete this project. Then I would like to thank my supervisor **Fatama Binta Rafiq, Lecturer**, Department of Software Engineering. I am extremely grateful and indebted to her for his expert, sincere and valuable guidance and encouragement extended to me.

Beside my supervisor, I would like to express my sincere thanks to **Dr. Touhid Bhuiyan**, Professor and Head of Software Engineering Department for his constant encouragement.

I would like to thank them who helped me in my project by their very important suggestions without their passionate participation and input; the project could not be successfully conducted. I take this opportunity to record my sincere thanks to all the faculty members of the Department of Software Engineering for their help and encouragement.

Last but not least, I would like to thank our parents, for their unconditional support, love and without this we would not have come this far.

ABSTRACT

Transport Finder is a web application. It is a transport finder application that allow user to find stations and available transport in a station or transports that run between two stations. The main aim of the project is to provide the users an easy to use web applications that helps them to find available transport in a station, find transports by destination and starting station according to their need. User can see the travelling cost of the transport and others related information. They can also provide feedback through that web application.

The application will be user driven as user will be able to contribute by adding new station and transport and other info's.

The System will count the feedback and automatically delete any station, transport or information if it gets 50 dislikes.

The system will also calculate the distance between the two stations using latitude and longitude

It is a social services system by which users will be able to avoid hassle of finding stations and transports.

Table of Contents

Chapter I.....	1
Introduction.....	1
1.1. Project Overview	1
1.2. Project Purpose	1
1.2.1. Background.....	1
1.2.2. Benefits & Beneficiaries.....	2
Benefits:	2
Beneficiaries:	2
1.2.3. Goals	2
1.3. Stakeholders.....	2
1.4. Proposed System Model	3
1.4.1. Agile-Model.....	3
1.5. Project Schedule.....	4
1.5.1. Gantt Chart.....	4
1.5.2. HR Planning for Development Phase	5
1.5.3. Release Plan	5
Chapter II	6
Software Requirement Specification	6
2.1. Functional Requirements	6
2.2. Data Requirements.....	7
2.3. Performance Requirements.....	7
2.3.1. Speed and Latency Requirements.....	7
2.3.2. Precision or Accuracy Requirements.....	7
2.3.3. Capacity Requirements	8

2.4. Dependability Requirements.....	8
2.4.1. Reliability Requirements	8
2.4.2. Availability Requirements	8
2.4.3. Robustness or Fault-Tolerance Requirements	8
2.5. Maintainability and Supportability Requirements	9
2.5.1. Maintenance Requirements.....	9
2.5.2. Supportability Requirements	9
2.5.3. Adaptability Requirements	9
2.6. Security Requirements	9
2.6.1. Access Requirements	9
2.6.2. Integrity Requirements.....	10
2.6.3. Privacy Requirements	10
2.7. Usability and Human-Interaction Requirements	10
2.8. Look and Feel Requirements	10
2.8.1. Appearance Requirements	10
2.8.1. Style Requirements	10
Chapter III.....	11
System Analysis.....	11
3.1. Use Case Diagram.....	11
3.2. Use Case Description.....	12
3.2.1. Manage Transports.....	12
3.2.2. Manage Stations.....	13
3.2.3. Manage Routes.....	14
3.2.4. Manage Feedbacks.....	15
3.2.5. Manage Users.....	16

3.2.6. Make Contribution.....	17
3.2.7. Manage Profile.....	18
3.2.8. Search Information's.....	19
3.2.9. Give Feedback	20
3.3. Activity Diagram	21
3.3.1. Activity Diagram for admin to Manage Stations.....	21
3.3.2. Activity Diagram for Admin to Manage Transports	22
3.3.3. Activity Diagram for Admin to Manage Routes.....	23
3.3.4. Activity Diagram for Admin to Manage Users.....	24
3.3.5. Activity Diagram for Admin to Manage Feedbacks.....	25
3.3.6. Activity Diagram for User to Make Contribution.....	26
3.3.7. Activity Diagram for User to Give Feedback.....	27
3.3.8. Activity Diagram for User to Manage Profile	28
Chapter IV.....	29
System Design Specification	29
4.1 Sequence Diagram	29
4.1.1. Sequence Diagram of Manage Transports.....	29
4.1.2. Sequence Diagram of Manage Stations	30
4.1.3. Sequence Diagram of Manage Routes.....	31
4.1.4. Sequence Diagram Manage Feedbacks	32
4.1.5. Sequence Diagram of Manage Users.....	33
4.1.6. Sequence Diagram of Make Contribution	34
4.1.7. Sequence Diagram of Give Feedback.....	35
4.1.8. Sequence Diagram of Manage Profile	36
4.1.9. Sequence Diagram of Search Information.....	37

4.3. ERD Diagram.....	38
Chapter V	39
User Interface and Manuals	39
5.1. User Interface.....	39
5.1.1. Home Page	39
5.1.2. Destination Wise Search Result.....	40
5.1.3. Search by Category	41
5.1.4. Category Wise Search result.....	42
5.1.5. Transport Profile	43
5.1.6. Station Profile	44
5.1.7. User Login	45
5.1.8. User Registration	46
5.1.9. Give Feedback	47
5.1.10. User Panel	48
5.1.11. Manage Transport - User	49
5.1.12. Manage Station - User	50
5.1.13. Manage Route - User	51
5.1.14. Manage Feedback - User	52
5.1.15. Manage Profile - User.....	53
5.1.16. Admin Login.....	54
5.1.17. Admin Dashboard	55
5.1.18. Manage Transport - Admin.....	56
5.1.19. Manage Station - Admin.....	57
5.1.20. Manage Route - Admin.....	58

5.1.21. Manage Feedback - Admin.....	59
5.1.22. Manage User - Admin.....	60
5.1.23. Contact Page	61
Chapter VI.....	62
Development Tools & Technologies	62
6.1. User Interface Technologies	62
6.2. Implementation Technologies.....	62
6.2.1. Laravel (5.8).....	62
6.2.2. Composer (1.8.5)	62
6.2.3. Mysql (15.1).....	62
6.2.4. PHP (7.3).....	62
6.2.5. XAMPP (3.22)	62
6.3. Platform & Environment.....	63
6.3.1. Hardware.....	63
6.3.2. Tools	63
6.3.3. Version Control.....	63
Chapter VII	64
System Testing.....	64
7.1. Introduction.....	64
7.2. Test Plan Strategy	64
7.3. Test Case	64
7.3.1. Test Case of Login	65
7.3.2. Test Case of Register	66
7.3.3. Test Case of Search by Destination	67

7.3.4. Test Case of Search by Category	68
7.3.5. Test Case of Manage Stations.....	69
7.3.6. Test Case of Manage Transports.....	70
7.3.7. Test Case of Manage Routes.....	71
7.3.8. Test Case of Manage Feedback's	72
7.3.9. Test Case of Manage Users.....	73
7.3.10. Test Case of Manage Profile.....	74
7.3.11. Test Case of Give Feedback	75
7.3.12. Test Case of Make Contribution.....	76
7.4. Features yet not tested.....	76
Chapter VIII.....	77
Project Summary.....	77
8.1. GitHub Link.....	77
8.2. Limitations	77
8.3. Obstacle & Achievements.....	77
8.4. Conclusion	77
8.5. Future Work.....	78
Key of Terms	79
References.....	80

List of Figure

Figure 1: Agile-Model	3
Figure 2: Gantt chart	4
Figure 3: Use Case Diagram	11
Figure 4: Activity Diagram for Admin to Manage Stations	21
Figure 5: Activity Diagram for Admin to Manage Transports	22
Figure 6: Activity Diagram for Admin to Manage Routes	23
Figure 7: Activity Diagram for Admin to Manage Users	24
Figure 8: Activity Diagram for Admin to Manage Feedbacks	25
Figure 9: Activity Diagram for User to Make Contribution	26
Figure 10: Activity Diagram for User to Give Feedback	27
Figure 11: Activity Diagram for User to Manage Profile	28
Figure 12: Sequence Diagram for Manage Station	29
Figure 13: Sequence Diagram for Manage Transports	30
Figure 14: Sequence Diagram for Manage Routes	31
Figure 15: Sequence Diagram for Manage Feedbacks	32
Figure 16: Sequence Diagram for Manage Users	33
Figure 17: Sequence Diagram for Make Contribution	34
Figure 18: Sequence Diagram for Give Feedback	35
Figure 19: Sequence Diagram for Manage Profile	36
Figure 20: Sequence Diagram for Search Information	37
Figure 21: ERD	38
Figure 23: Home Page	39
Figure 23: Result of Search by Destination	40
Figure 24: Search by Category Module	41
Figure 25: Category Wise Search Result	42
Figure 26: Transport Profile	43
Figure 27: Station Profile	44
Figure 28: User Login	45

Figure 29: User Registration.....	46
Figure 30: Give Feedback.....	47
Figure 31: User Panel.....	48
Figure 32: Manage Transport - User.....	49
Figure 33: Manage Station - User.....	50
Figure 34: Manage Route - User.....	51
Figure 35: Manage Feedback - User.....	52
Figure 36: Manage Profile - User	53
Figure 37: Admin Login.....	54
Figure 38: Admin Dashboard.....	55
Figure 39: Manage Transport - Admin	56
Figure 40: Manage Station - Admin	57
Figure 41: Manage Route - Admin	58
Figure 42: Manage Feedback – Admin.....	59
Figure 43: Manage User - Admin	60
Figure 44: Contact Page.....	61

List of Table

Table 1: Functional Requirements	6
Table 2: Data requirements	7
Table 3 Speed and Latency Requirements	7
Table 4: Precision or Accuracy Requirements.....	7
Table 5: Capacity Requirements	8
Table 6: Reliability Requirements	8
Table 7: Availability Requirements	8
Table 8: Robustness or Fault-Tolerance Requirements	8
Table 9: Maintenance Requirements	9
Table 10: Supportability Requirements	9
Table 11: Adaptability Requirements	9
Table 12: Access Requirements.....	9
Table 13: Integrity Requirements	10
Table 14: Privacy Requirements.....	10
Table 15: Appearance Requirements	10
Table 16: Style Requirements	10
Table 17: Manage Transports	12
Table 18: Manage Stations.....	13
Table 19 Manage Routes.....	14
Table 20: Manage Feedbacks.....	15
Table 21: Manage Users.....	16
Table 22: Make Contribution.....	17
Table 23: Manage Profile.....	18
Table 24: Search Information's.....	19
Table 25: Give Feedback.....	20
Table 26: Test Case of Login.....	65
Table 27: Test Case of Register.....	66
Table 28: Test Case of Search by Destination	67

Table 29: Test Case of Search by Category.....	68
Table 30: Test Case of Manage Station.....	69
Table 31 Test Case of Manage Transport.....	70
Table 32: Test Case of Manage Route.....	71
Table 33: Test Case of Manage Feedback.....	72
Table 34: Test Case of Manage User.....	73
Table 35: Test Case of Manage Profile.....	74
Table 36: Test Case of Make Contribution.....	75
Table 37: Test Case of Give Feedback.....	76
Table 38: Features yet not tested.....	77

Chapter I

Introduction

1.1. Project Overview

Transport Finder is a web application. Where user will use it to find information's of station, transports and specific transports that run between two stations (provided by user). The special feature of this project is it's a user driven system. As user will be able to add transport, station and route. User also will be able to give feedback to a station, transport or route. If any information gets 50 dislikes it will be automatically deleted. The system is also able to calculate distance between two stations using latitude and longitude provided.

1.2. Project Purpose

1.2.1. Background

I could not find any source in web where anyone can find local bus and Launch route and fare. But there is, many different sources where people can find information of plane, train and long-distance buses but they are not user friendly also don't have sufficient information. So, there is no easy way to find information about local transports. Specially, local buses. And for this people fall in trouble when using public transport. They also get demotivated to use public transport. Which is a major reason behind traffic jam in cities. So, I decided to create a system where anyone can find information (Route, Fare and many more) about public transports (Bus, Plane, Train, Launch, Others) and stations. My system will solve this problem in an amazing way. As it's a user driven system it can be used by everyone around the world.

In this system user will be able to search by starting station and destination or by category. User will also be able to give feedback to information. If an information gets at least 20 dislikes, it will be deleted automatically. Besides this system can be a global hub where people can find information of public transport. As all the information will be contributed by user and also monitored by user by taking feedback.

This system can make our life a little bit easy. We can plan our day easily by knowing needed information of transports in advance.

1.2.2. Benefits & Beneficiaries

Benefits:

- Everything in one place.
- Provide transport and stations information in details. (e.g. Fare, Location, Phone etc.)
- User will be able to search by destination or category.
- User will be able to contribute.
- User will be able to give feedback to make sure provided information is trusted.

Beneficiaries:

- People who use public transports.
- People Who need public transports/stations info.

1.2.3. Goals

Goal of this system is to create a global platform where information of public transports and stations will be available around the world. And helping people with stations and transports data to make their travel or other plans in advance. No matter wherever you go around the world public transport and stations information will be there.

1.3. Stakeholders

A stakeholder is a party that has an interest in a company and can either affect or be affected by the business. The primary stakeholders in a typical corporation are its investors, employees, customers and suppliers.

- **Users**

1.4. Proposed System Model

A software process model is a simplified representation of a software process. Each model represents a process from a specific perspective.

1.4.1. Agile-Model

My proposed system model is agile model which is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas

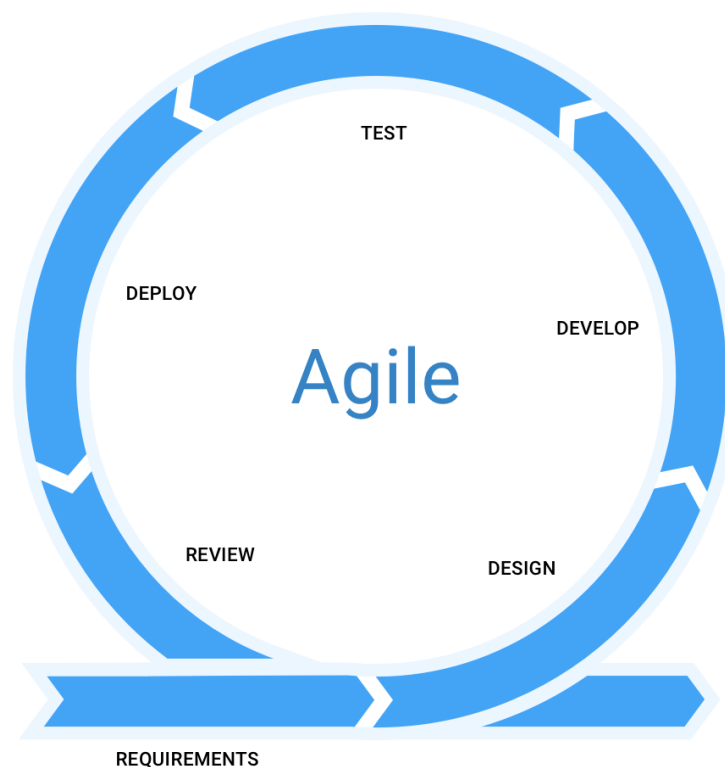


Figure 1: Agile-Model

1.5. Project Schedule

In project management, project scheduling is a mechanism to communicate what tasks need to get done and which organizational resources will be allocated to complete those tasks in what timeframe. A project schedule is a document collecting all the work needed to deliver the project on time. A schedule is commonly used in the project planning and project portfolio management parts of project management.

1.5.1. Gantt Chart

Weeks	1	2	3	4	5	6	7	8	9	10	11	12
Works												
Analysis Phase	Green	Green										
Feasibility Study		Red										
Project proposal			Light Green									
Project UI			Blue	Blue								
Mid-Term Defense					Blue							
Implementation of The Project				Blue	Blue	Blue	Blue	Blue	Blue	Blue		
Testing										Orange		
Document of The Project										Orange	Orange	
Final Defense												Purple

Figure 2: Gantt chart

1.5.2. HR Planning for Development Phase

1. Project plan
2. Analysis
3. Requirement gathering
 - Brainstorming
 - Interview
 - Observation
 - Analysis
4. Design
 - System design
 - Database design
 - System user interface
5. Development
 - User Module
 - Event Module
 - Others
6. Testing
 - Test plan
 - Test Case
 - Test Execution

1.5.3. Release Plan

Release 1: beta version 1.0.0 on 12/02/2018

Release 2: version 2.0.0 on 12/03/2018

Release 3: version 3.0.0 on 12/04/2018

Chapter II Software Requirement Specification

2.1. Functional Requirements

Table 1: Functional Requirements

SRS No.	Name	Description
#01	Manage Transports	Admin can manage transports. Admin can add, update, delete. User also can manage transport. But they will be only view, update, delete the transports only they have added. And Transport added by user will require approval for admin to be visible in system.
#02	Manage Stations	Admin can manage stations. Admin can add, update, delete. User also can manage stations. But they will be only view, update, delete the stations only they have added. And stations added by user will require approval for admin to be visible in system.
#03	Manage Routes	Admin can manage routes. Admin can add, update, delete. User also can manage routes. But they will be only view, update, delete the routes only they have added. And routes added by user will require approval for admin to be visible in system.
#04	Manage Feedbacks	Admin can manage feedbacks. Admin can delete and change the status of feedback (Active/ Inactive define whether to count this feedback). User can only see the list of feedback they give. And they can delete it.
#05	Manage Users	Admin can manage users. Admin can add, update, delete.
#06	Manage Profile	User can manage their profile. Ex, changing name, password or email.
#07	Search	User can search by starting station destination or only by station or transport.

2.2. Data Requirements

Table 2: Data requirements

No	Description
#01	Admin has to insert the correct login credentials otherwise system will show an error with message.
#02	Admin has to monitor everything and provide approval.
#03	User has to insert login credential to login to the system
#04	Admin has to add transport, station, route, user
#05	User has to insert register information to get registered.
#06	User has to add transport, station, route, feedback

2.3. Performance Requirements

2.3.1. Speed and Latency Requirements

Table 3 Speed and Latency Requirements

No	Description
#01	The system must have a high speed of manipulation data and reply to the user requests.

2.3.2. Precision or Accuracy Requirements

Table 4: Precision or Accuracy Requirements

No	Description
#01	The input data should be validated when user or admin provide data to the system
#02	All data should be in place accurately where it is associated

2.3.3. Capacity Requirements

Table 5: Capacity Requirements

No	Description
#01	The web application size must able to load at hosting site.
#03	System should support 1 million users at the beginning version
#04	System should support 1k request per second.

2.4. Dependability Requirements

2.4.1. Reliability Requirements

Table 6: Reliability Requirements

No	Description
#01	All confidential data must have to be encrypted.
#03	No one can use customers data for any other purpose except system needs.

2.4.2. Availability Requirements

Table 7: Availability Requirements

No	Description
#01	The system should work 24 hours a day
#02	The system should provide the desired data to the user in time

2.4.3. Robustness or Fault-Tolerance Requirements

Table 8: Robustness or Fault-Tolerance Requirements

No	Description
#01	If the system has been crashed, it should not be more than an hour.
#02	System must be responsible and should be cross browser supported

2.5. Maintainability and Supportability Requirements

2.5.1. Maintenance Requirements

Table 9: Maintenance Requirements

No	Description
#01	The system maintenance should be quick.

2.5.2. Supportability Requirements

Table 10: Supportability Requirements

No	Description
#01	The system supports latest Laravel and mysql version.
#02	Should support all the browsers and screen size.

2.5.3. Adaptability Requirements

Table 11: Adaptability Requirements

No	Description
#01	The system should adapt all upgrading version and time.
#02	New version of system should support latest Php, Laravel and mysql.

2.6. Security Requirements

2.6.1. Access Requirements

Table 12: Access Requirements

No	Description
#01	All users access should be limited with their use case boundaries
#02	Users need to be authorized first to access user panel.
#04	Users boundaries should be within the browser.

2.6.2. Integrity Requirements

Table 13: Integrity Requirements

No	Description
#01	Only authorized user can add or delete data with their respective accessibility and authorization.
#02	Only admin can add users and provide approval for all information.

2.6.3. Privacy Requirements

Table 14: Privacy Requirements

No	Description
#01	The user data must not be visible for public
#02	The user data should not contain any private issues
#03	All the confidential data should be encrypted.

2.7. Usability and Human-Interaction Requirements

No visible usability and Human-Interaction requirements.

2.8. Look and Feel Requirements

2.8.1. Appearance Requirements

Table 15: Appearance Requirements

No	Description
#01	The user interface must be attractive.
#02	The user interface must be user friendly.
#03	The user interface must be user interactive.

2.8.1. Style Requirements

Table 16: Style Requirements

No	Description
#01	The interface color should be flat or material.

Chapter III System Analysis

3.1. Use Case Diagram

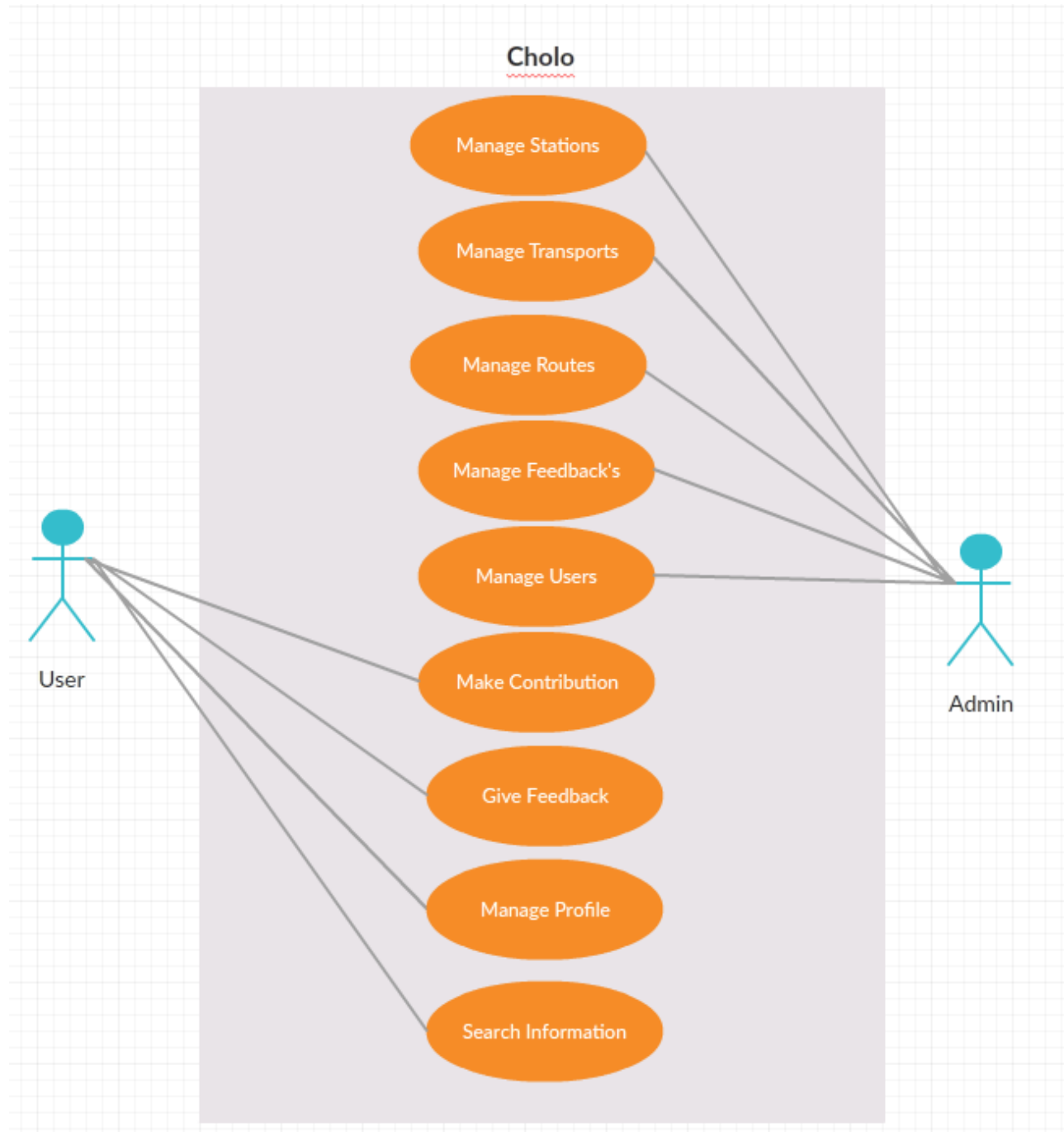


Figure 3: Use Case Diagram

3.2. Use Case Description

3.2.1. Manage Transports

Authorized admin can add, update, delete or provide approval of transport added by users.

Table 17: Manage Transport

Use Case ID	1A
Use Case	Manage Transports
Goal	Providing a system where admin can add, edit, delete, search, view transport.
Preconditions	Admin must be authorized.
Success End Condition	Admin successfully create/edit/delete/search/view transport.
Failed End Condition	Admin failed to create/edit/delete/search/view transport.
Primary Actors: Secondary Actors:	Admin None
Trigger	Admin wants to manage transports.
Description / Main Success Scenario	<ol style="list-style-type: none">1. Admin indicates desire to manage transport.2. System check if admin is authorized3. If admin is authorized system provide functionalities to admin to manage transport. The Use Case Ends.
Alternative Flows	Start after Main Flow Step-3 <ol style="list-style-type: none">1. If admin is not authorized system will redirect admin to login.2. The admin enters invalid log in info's.3. The system displays error message. The use Case ends.

3.2.2. Manage Stations

Authorized admin can add, update, delete or provide approval of stations added by users.

Table 18: Manage Stations

Use Case ID	2A
Use Case	Manage Stations
Goal	Providing a system where admin can add, edit, delete, search, view station.
Preconditions	Admin must be authorized.
Success End Condition	Admin successfully create/edit/delete/search/view transport.
Failed End Condition	Admin failed to create/edit/delete/search/view transport.
Primary Actors: Secondary Actors:	Admin None
Trigger	Admin wants to manage stations.
Description / Main Success Scenario	<ol style="list-style-type: none">1. Admin indicates desire to manage station.2. System check if admin is authorized3. If admin is authorized system provide functionalities to admin to manage station. The use case ends.
Alternative Flows	Start after Main Flow Step-3 <ol style="list-style-type: none">1. If admin is not authorized system will redirect admin to login.2. The admin enters invalid log in info's.3. The system displays error message. The use Case ends.

3.2.3. Manage Routes

Authorized admin can add, update, delete or provide approval of routes added by users.

Table 19: Manage Route

Use Case ID	3A
Use Case	Manage Route
Goal	Providing a system where admin can add, edit, delete, search, view route information's.
Preconditions	Admin must be authorized.
Success End Condition	Admin successfully create/edit/delete/search/view route information.
Failed End Condition	Admin failed to create/edit/delete/search/view route information.
Primary Actors: Secondary Actors:	Admin None
Trigger	Admin wants to manage travel information's.
Description / Main Success Scenario	<ol style="list-style-type: none">1. Admin indicates desire to manage route info's.2. System check if admin is authorized3. If admin is authorized system provide functionalities to admin to manage route info's.4. After admin provides information system use latitude and longitude to calculate distance between two station. The Use case ends.
Alternative Flows	Start after Main Flow Step-4 <ol style="list-style-type: none">1. If admin is not authorized system will redirect admin to login.2. The admin enters invalid log in info's.3. The system displays error message. The use Case ends.

3.2.4. Manage Feedbacks

Authorized admin can delete or provide approval of feedbacks added by users.

Table 20: Manage Feedbacks

Use Case ID	4A
Use Case	Manage Feedback's
Goal	Providing a system where admin can view, delete, search feedback.
Preconditions	Admin must be authorized.
Success End Condition	Admin successfully /delete/search/view feedback.
Failed End Condition	Admin failed to delete/search/view feedback.
Primary Actors: Secondary Actors:	Admin None
Trigger	Admin wants to manage feedbacks.
Description / Main Success Scenario	<ol style="list-style-type: none">1. Admin indicates desire to manage feedbacks.2. System check if admin is authorized3. If admin is authorized system provide functionalities to admin to manage feedback. The Use Case Ends.
Alternative Flows	Start after Main Flow Step-3 <ol style="list-style-type: none">1. If admin is not authorized system will redirect admin to login.2. The admin enters invalid log in info's.3. The system displays error message. The use Case ends.

3.2.5. Manage Users

Authorized admin can add, update, delete users.

Table 21: Manage Users

Use Case ID	5A
Use Case	Manage Users
Goal	Providing a system where admin can edit, delete, search, view user.
Preconditions	Admin must be authorized.
Success End Condition	Admin successfully edit/delete/search/view user.
Failed End Condition	Admin failed to edit/delete/search/view user.
Primary Actors: Secondary Actors:	Admin None
Trigger	Admin wants to manage users.
Description / Main Success Scenario	<ol style="list-style-type: none">1. Admin indicates desire to manage user.2. System check if admin is authorized3. If admin is authorized system provide functionalities to admin to manage users. The use case ends.
Alternative Flows	Start after Main Flow Step-3 <ol style="list-style-type: none">1. If admin is not authorized system will redirect admin to login.2. The admin enters invalid log in info's.3. The system displays error message. The use Case ends.

3.2.6. Make Contribution

Registered user can add transport, station, route. They can delete and update info's only they have added. They also can update and delete feedback they have given.

Table 22: Make Contribution

Use Case ID	IU
Use Case	Make Contribution
Goal	Providing a system where user can add transport, station and route information.
Preconditions	User must be logged in.
Success End Condition	User successfully contributes.
Failed End Condition	User failed to contributes.
Primary Actors: Secondary Actors:	User None
Trigger	User wants to contribute.
Description / Main Success Scenario	<ol style="list-style-type: none">1. User indicates desire contribute.2. System check if user is logged in.3. If user is authorized system provide functionalities to user to make contribution.4. After user provide information. System use latitude and longitude to calculate distance between two stations. The use case ends.
Alternative Flows	Start after Main Flow Step-4 <ol style="list-style-type: none">1. If user is not authorized system will redirect user to login.2. The user enters invalid log in info's.3. The system displays error message. The use Case ends.

3.2.7. Manage Profile

Registered user can manage their profile. Ex. Changing password, email or name.

Table 23: Manage Profile

Use Case ID	2U
Use Case	Manage Profile
Goal	Providing a system where user can manage their profile.
Preconditions	User must be logged in.
Success End Condition	User successfully update his/her profile.
Failed End Condition	User failed to update.
Primary Actors: Secondary Actors:	User None
Trigger	User wants to update profile.
Description / Main Success Scenario	<ol style="list-style-type: none">5. User indicates desire to update profile.6. System check if user is logged in.7. If user is authorized system provide functionalities to user to update profile.8. After user provide information. System update the info in database. The use case ends.
Alternative Flows	Start after Main Flow Step-4 <ol style="list-style-type: none">4. If user is not authorized system will redirect user to login.5. The user enters invalid log in info's.6. The system displays error message. The use Case ends.

3.2.8. Search Information's

User can search transport by providing starting station and destination or they can search for only station, transport. User do not require to be login.

Table 24: Search Information's

Use Case ID	3U
Use Case	Search Information
Goal	Providing a system where user can search by destination, station and information transport.
Preconditions	None
Success End Condition	User successfully get the search result.
Failed End Condition	User does not get the search result.
Primary Actors: Secondary Actors:	User None
Trigger	Admin wants to manage stations.
Description / Main Success Scenario	<ol style="list-style-type: none">1. User indicates desire to search information.2. User provide search keyword.3. System return result.
Alternative Flows	None

3.2.9. Give Feedback

Only registered user can give feedback. Feedback can be likes or dislikes. And this feedback instantly updated the table after admin approve it. This feed goes through a function every time which check if a information got 50 dislikes, If it is its delete the info.

Table 25: Give Feedback

Use Case ID	4U
Use Case	Give Feedback
Goal	Providing a system where user can give feedback to transport, station and travel information.
Preconditions	User must be logged in.
Success End Condition	User successfully Give feedback.
Failed End Condition	User failed to give feedback.
Primary Actors: Secondary Actors:	User None
Trigger	User wants to give feedback.
Description / Main Success Scenario	<ol style="list-style-type: none">1. User indicates desire to give feedback.2. System ask user to log in.3. If user successfully logged in system will provides functionalities to give feedback.4. After giving feedback system calculate the number of dislike and like in the backend. And delete information if its get at least 50 dislikes The Use Case Ends.
Alternative Flows	Start after Main Flow Step-4 <ol style="list-style-type: none">1. If user is not authorized system will redirect user to login.2. The user enters invalid log in info's.3. The system displays error message. The use Case ends.

3.3. Activity Diagram

3.3.1. Activity Diagram for Admin to Manage Stations

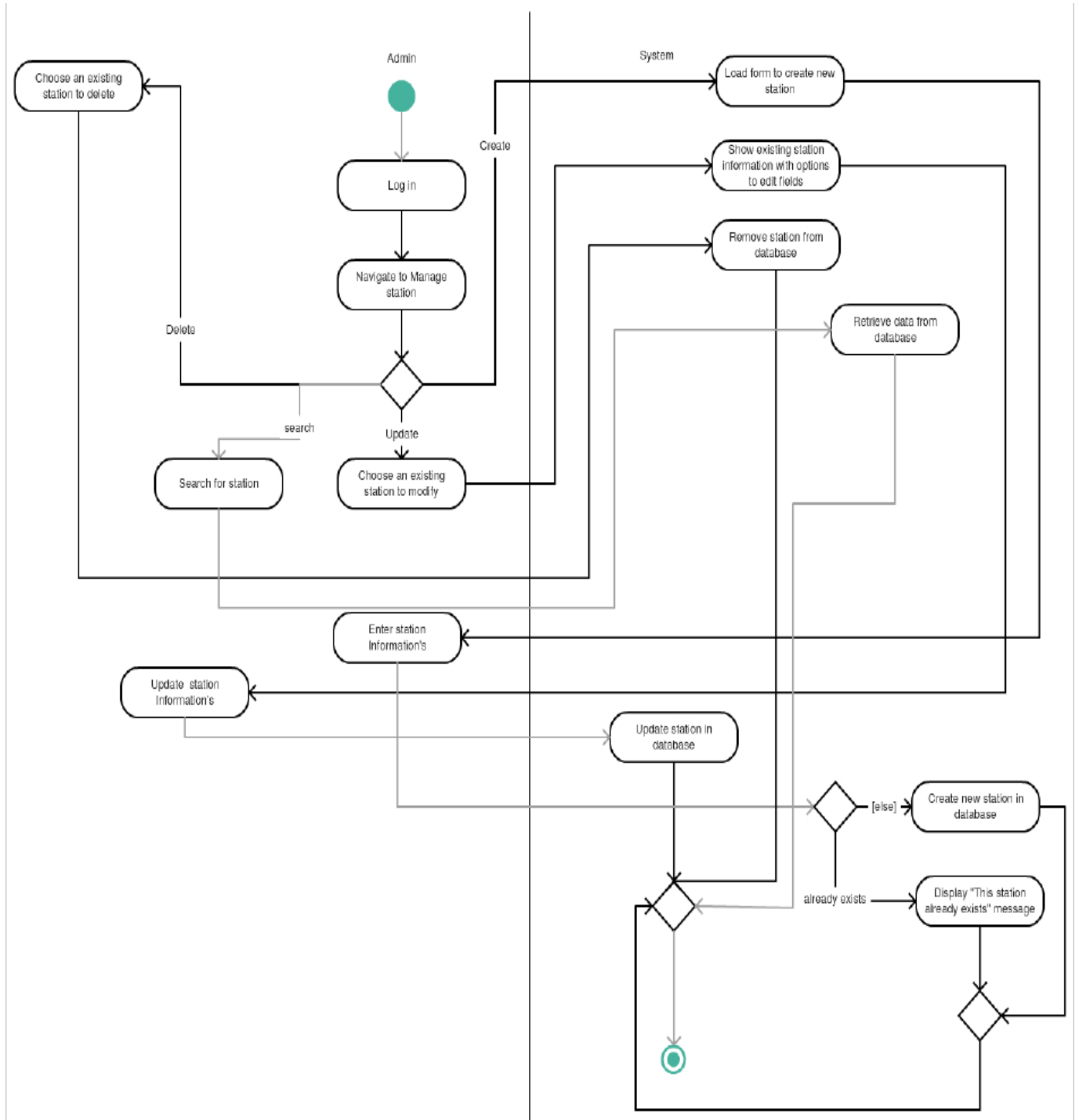


Figure 4: Activity Diagram for Admin to Manage Stations

3.3.2. Activity Diagram for Admin to Manage Transports

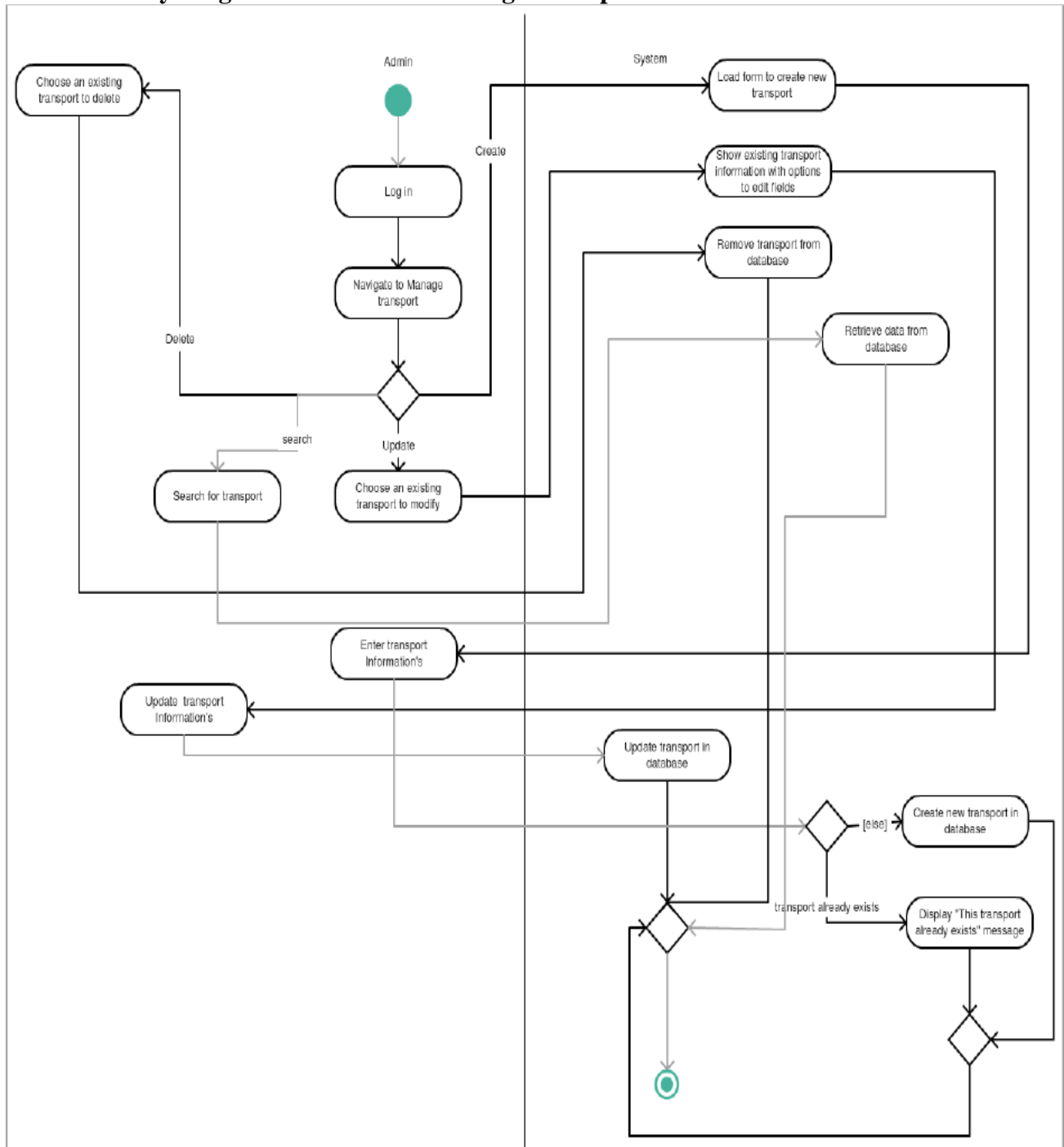


Figure 5: Activity diagram for admin to manage transports

3.3.3. Activity Diagram for Admin to Manage Routes

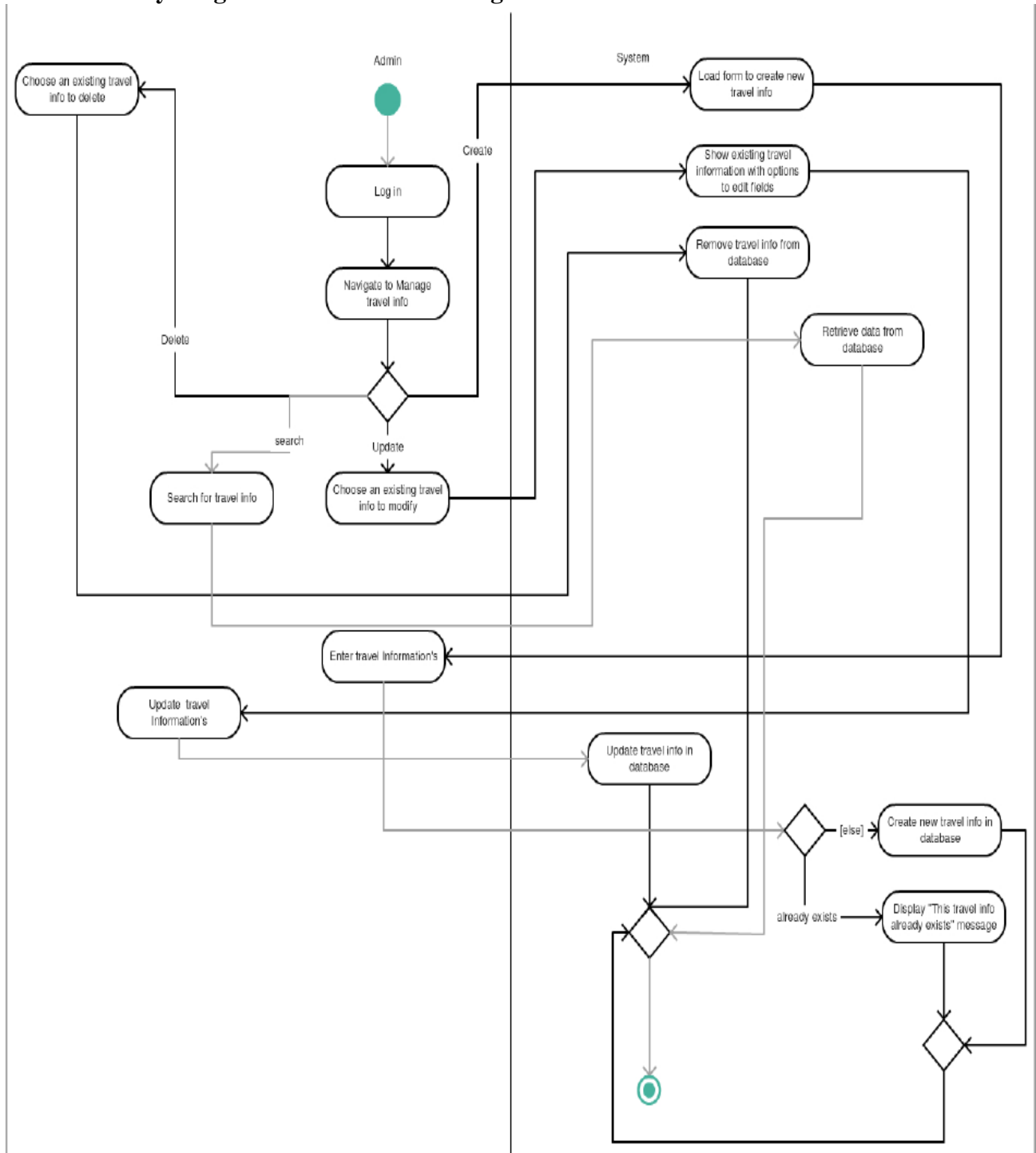


Figure 6: Activity Diagram for admin to manage routes

3.3.4. Activity Diagram for Admin to Manage Users

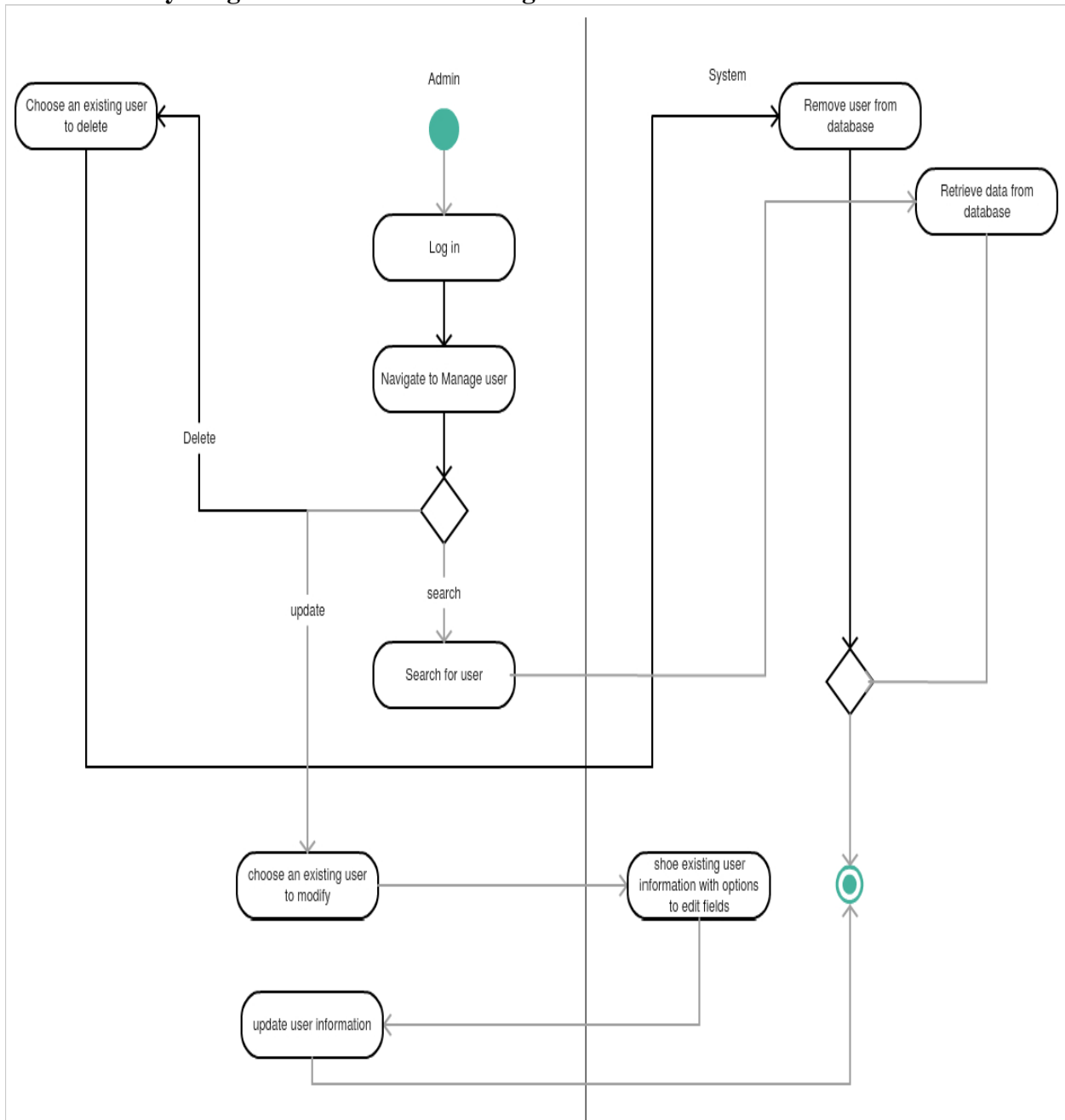


Figure 7: Activity Diagram for admin to manage users

3.3.5. Activity Diagram for Admin to Manage Feedbacks

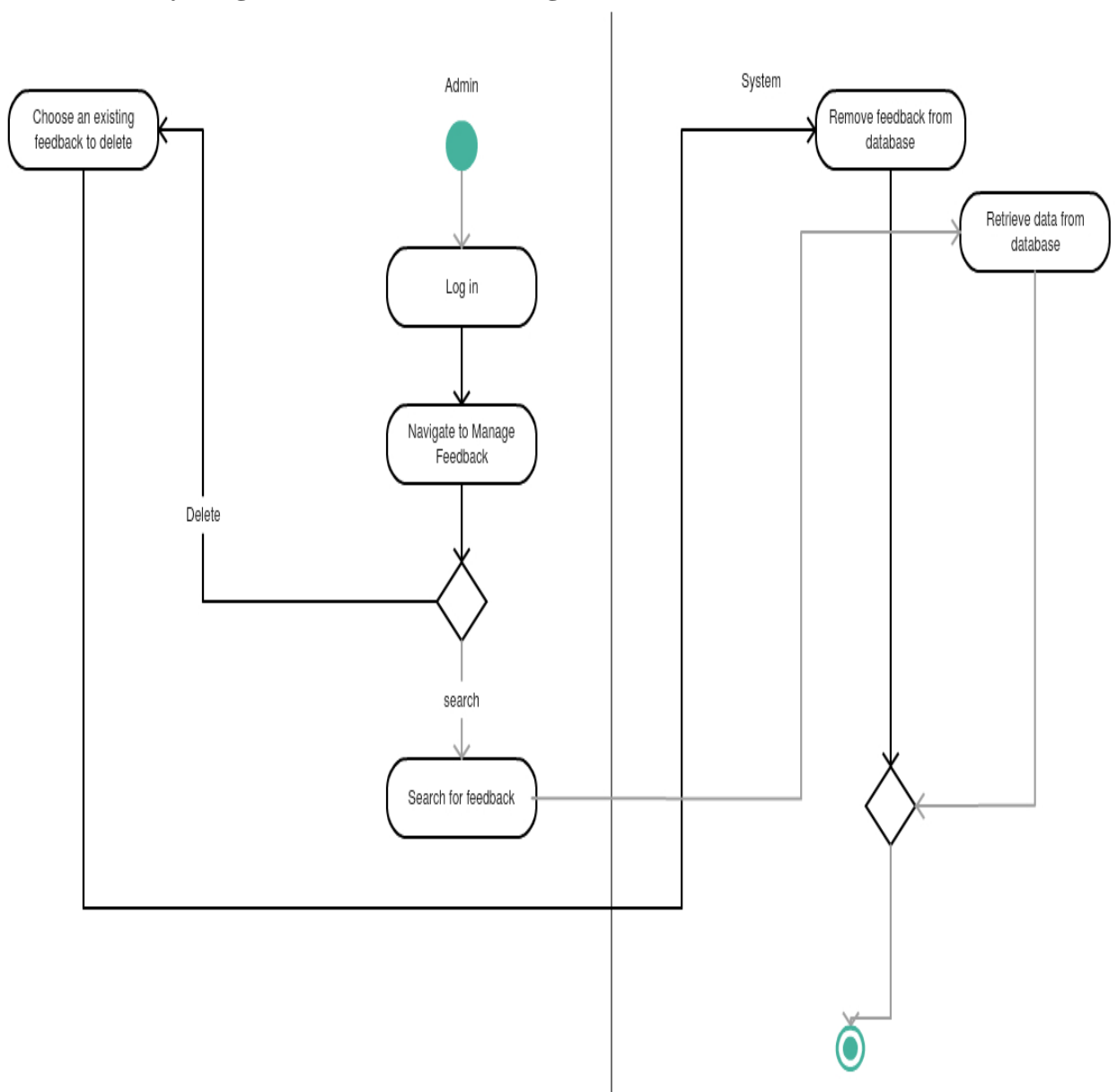


Figure 8: Activity Diagram for admin to manage feedbacks

3.3.6. Activity Diagram for User to Make Contribution

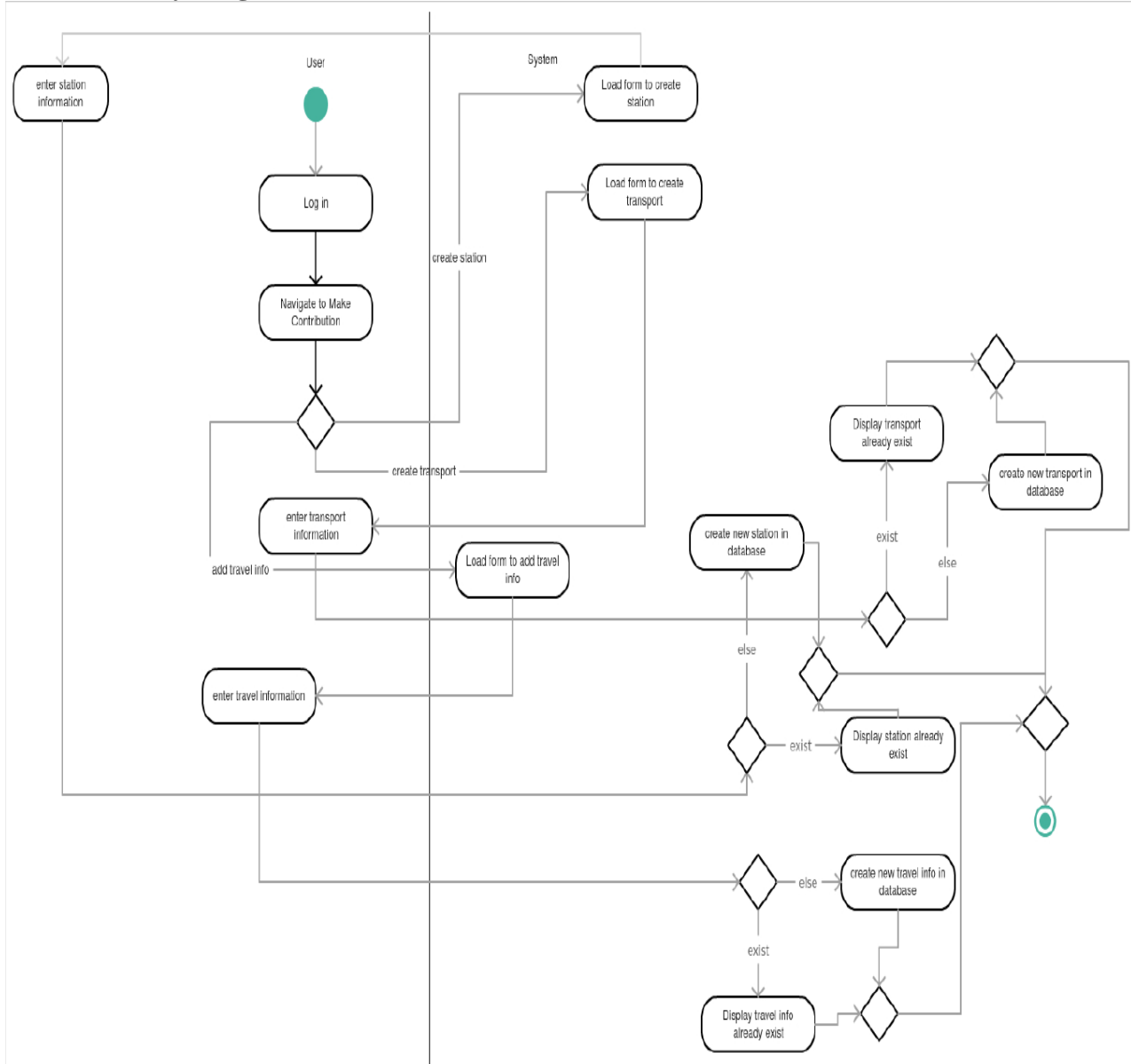


Figure 9: Activity Diagram for User to make contribution

3.3.7. Activity Diagram for User to Give Feedback

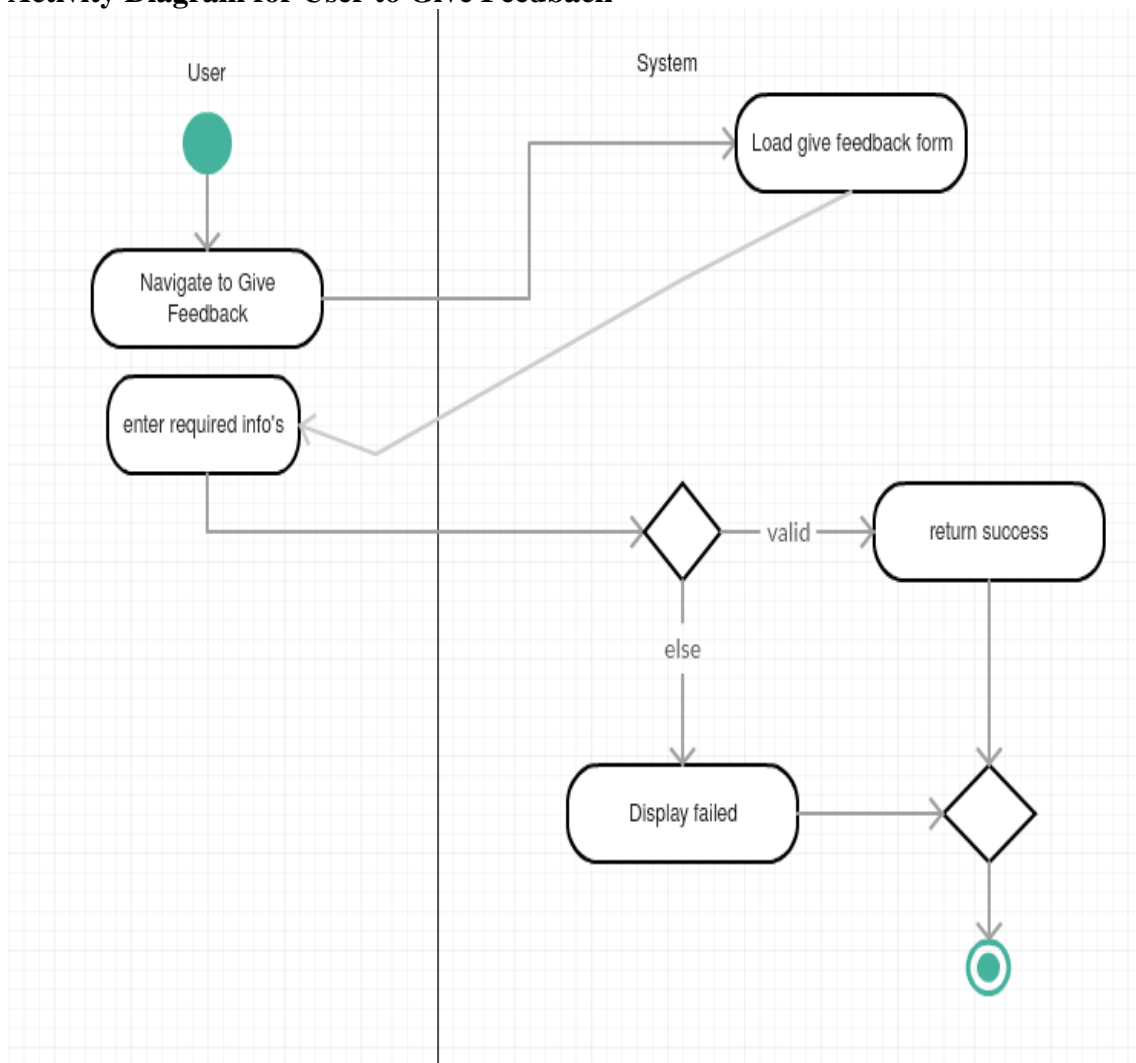


Figure 10: Activity Diagram for User to Give Feedback

3.3.8. Activity Diagram for User to Manage Profile

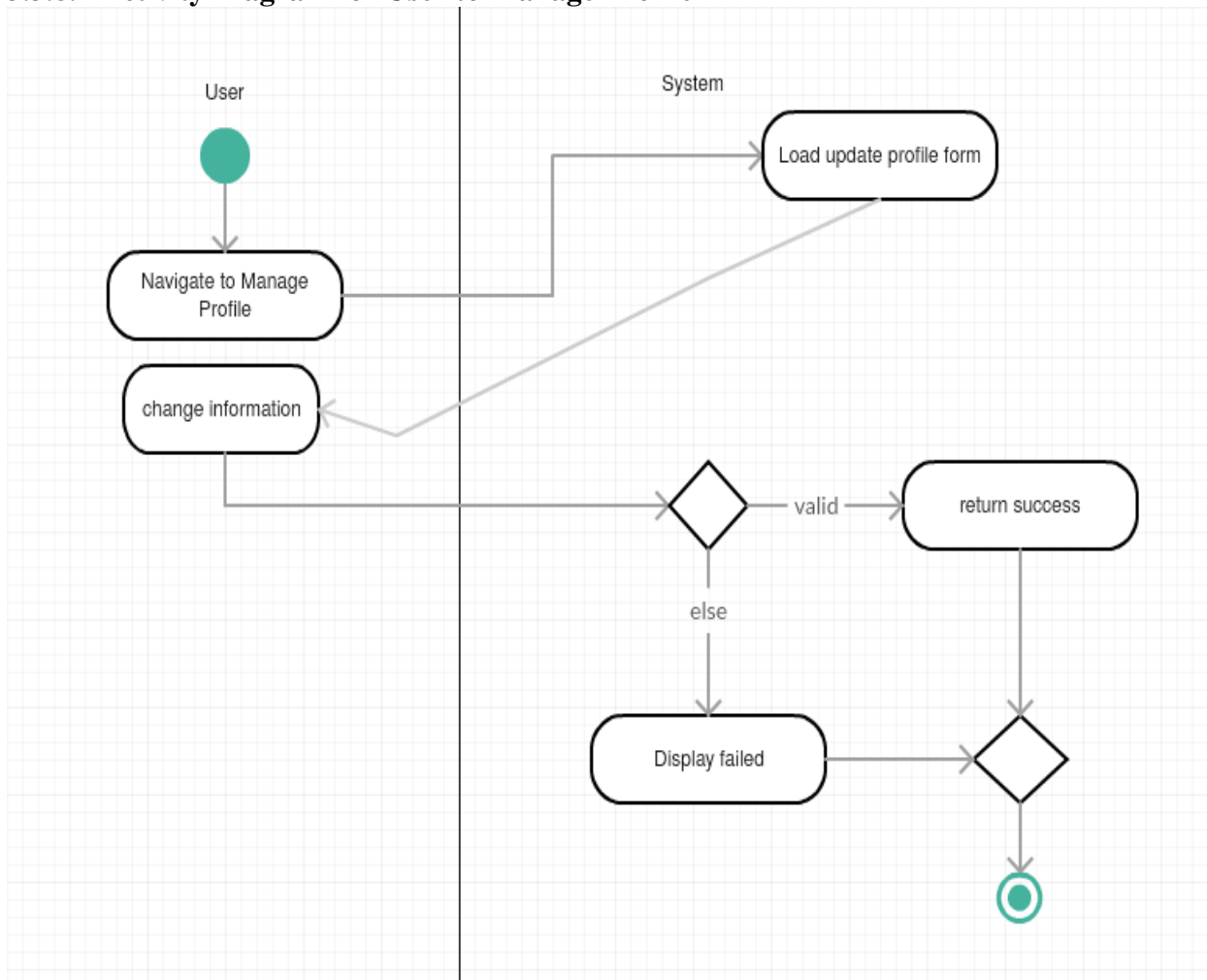


Figure 11: Activity Diagram for User to Manage Feedback

Chapter IV System Design Specification

4.1 Sequence Diagram

4.1.1. Sequence Diagram for Manage Transports

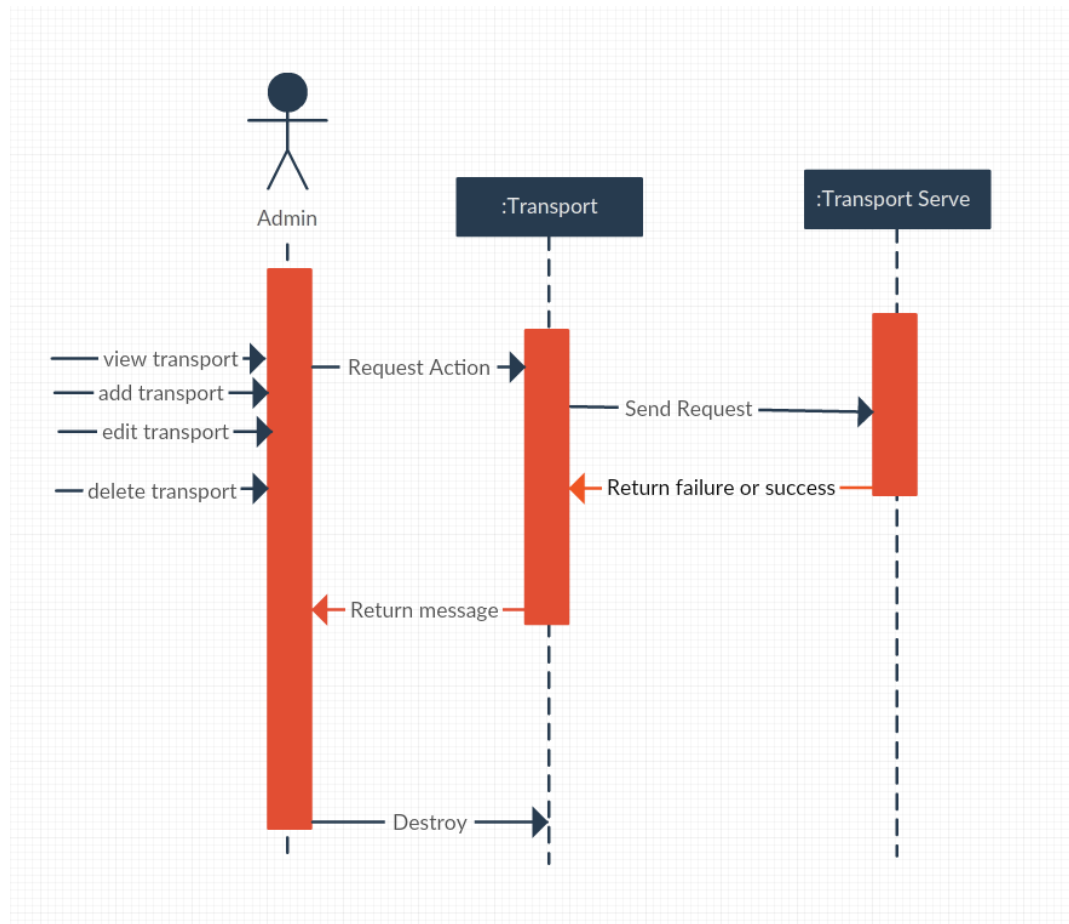


Figure 12: Sequence Diagram for Manage Transports

4.1.2. Sequence Diagram for Manage Stations

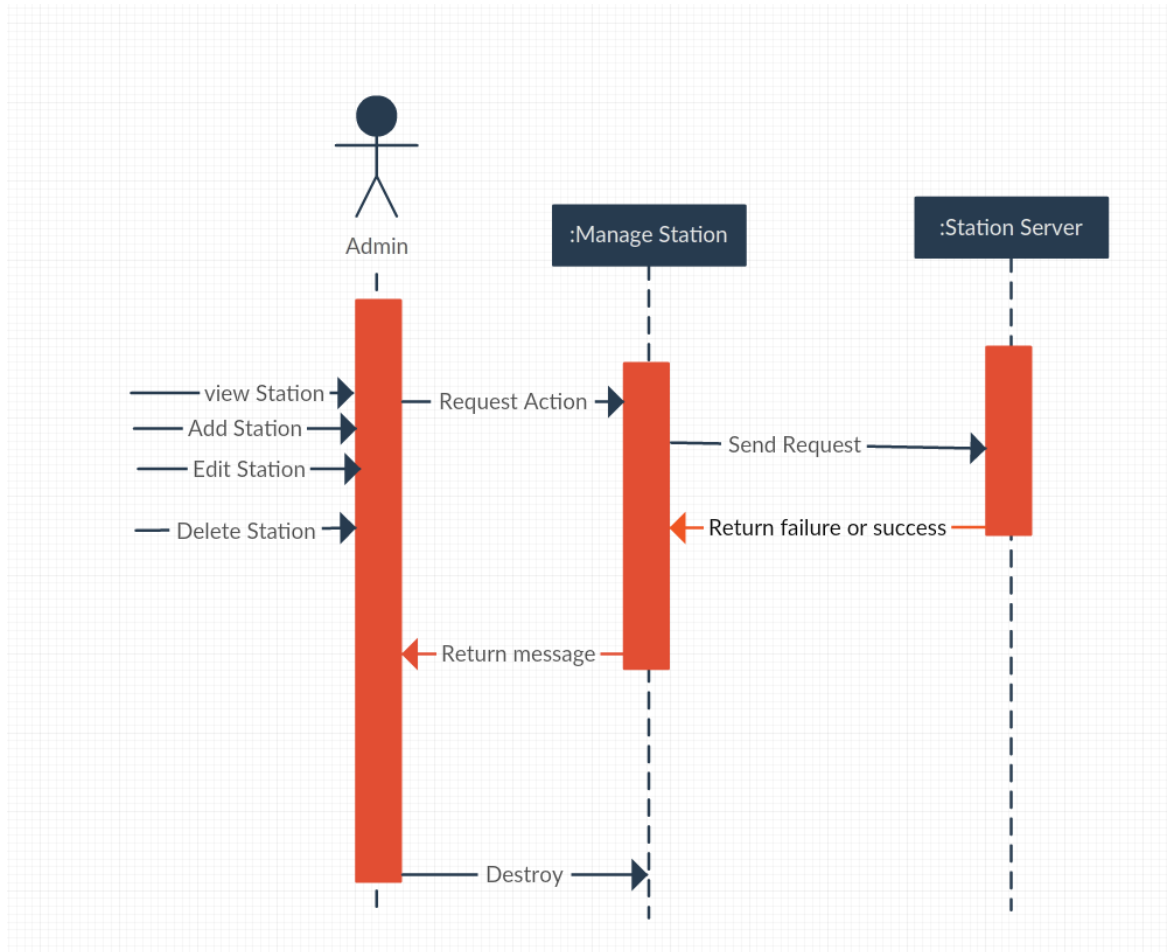


Figure 13: Sequence Diagram for Manage Stations

4.1.3. Sequence Diagram for Manage Routes

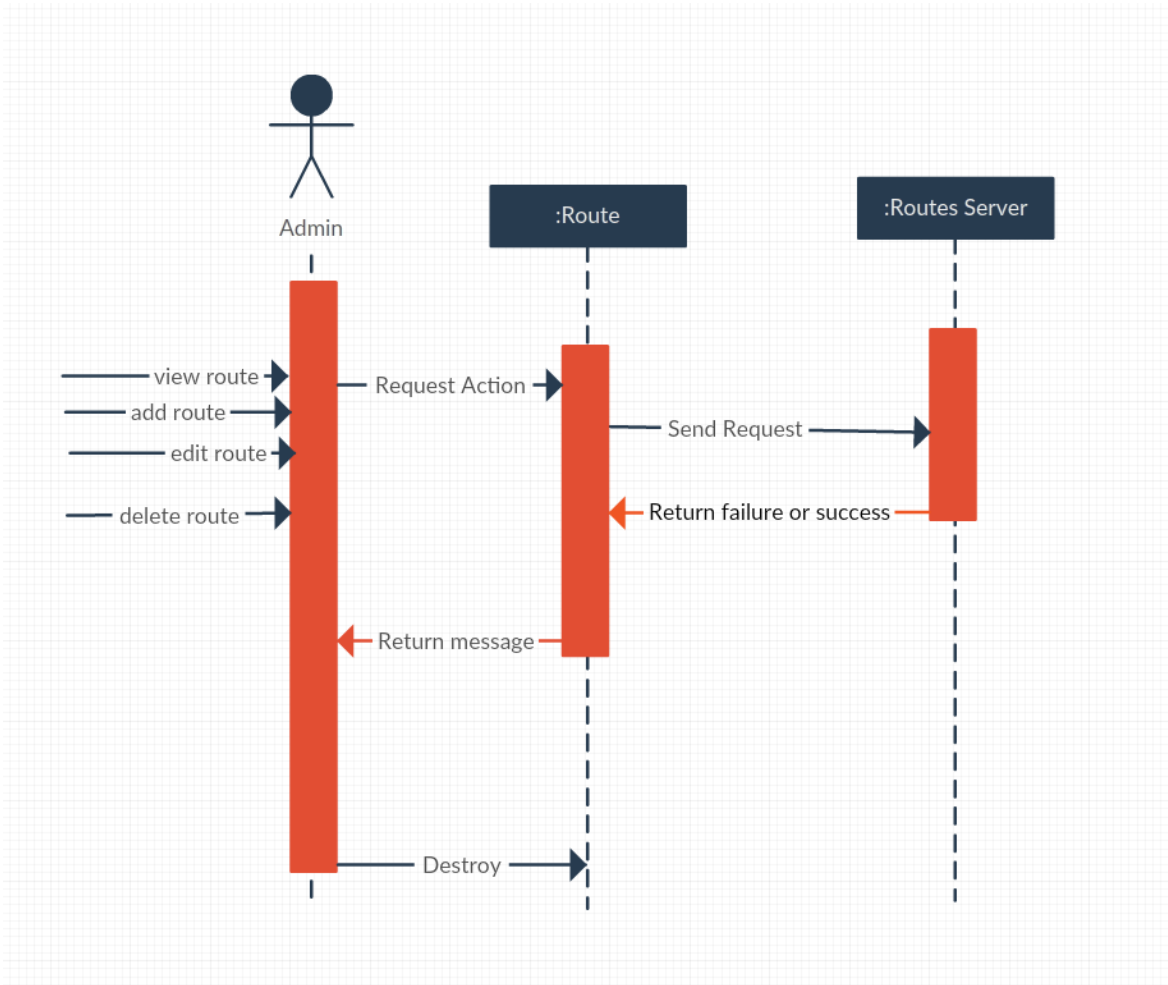


Figure 14: Sequence Diagram for Manage Routes

4.1.4. Sequence Diagram for Manage Feedbacks

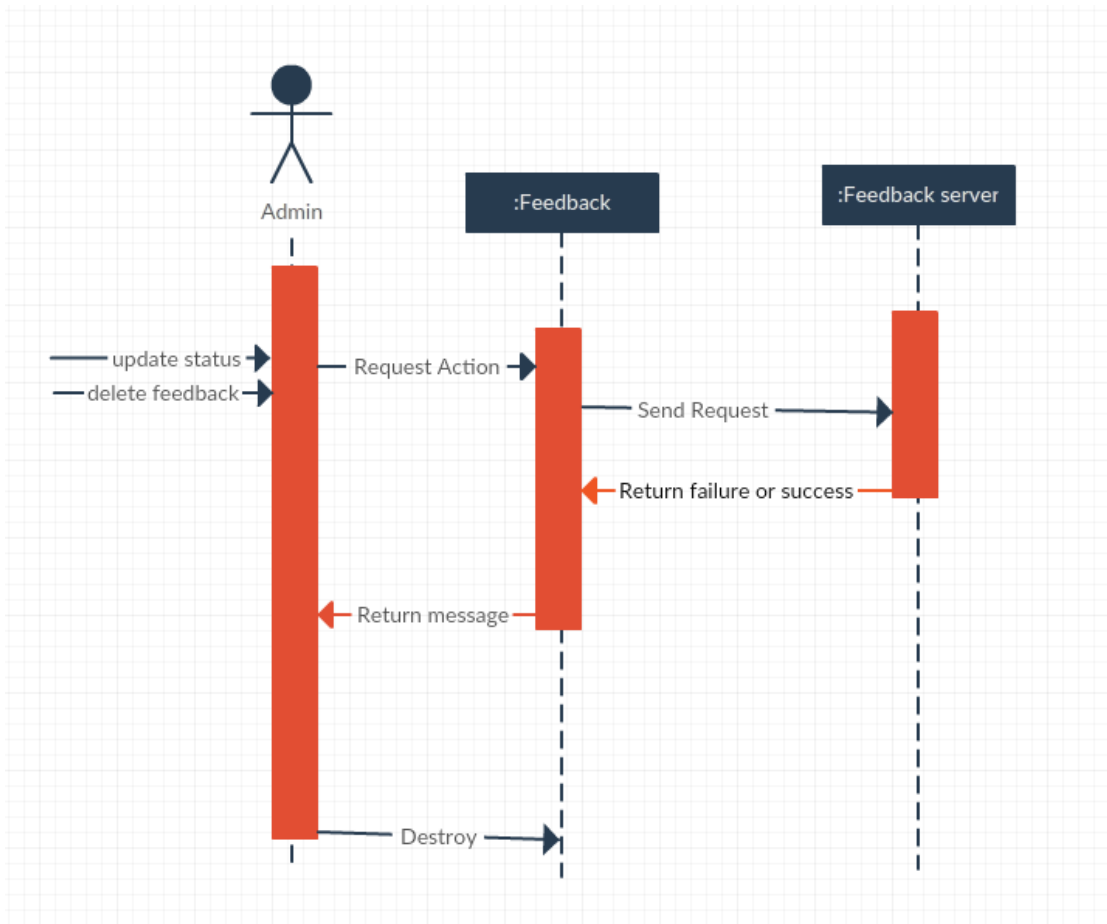


Figure 15: Sequence Diagram for Manage Feedbacks

4.1.5. Sequence Diagram for Manage Users

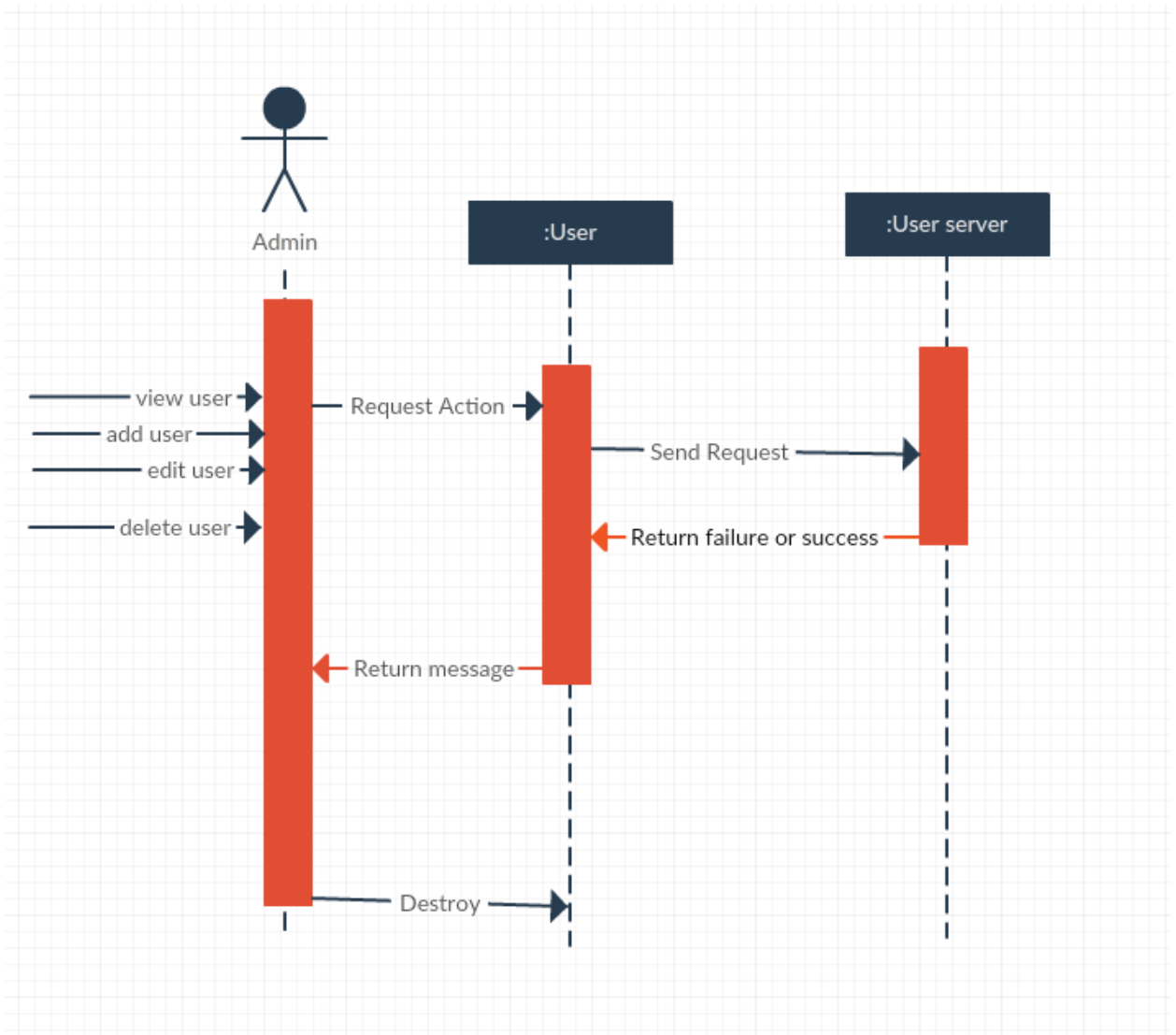


Figure 16: Sequence Diagram for Manage Stations

4.1.6. Sequence Diagram for Make Contribution

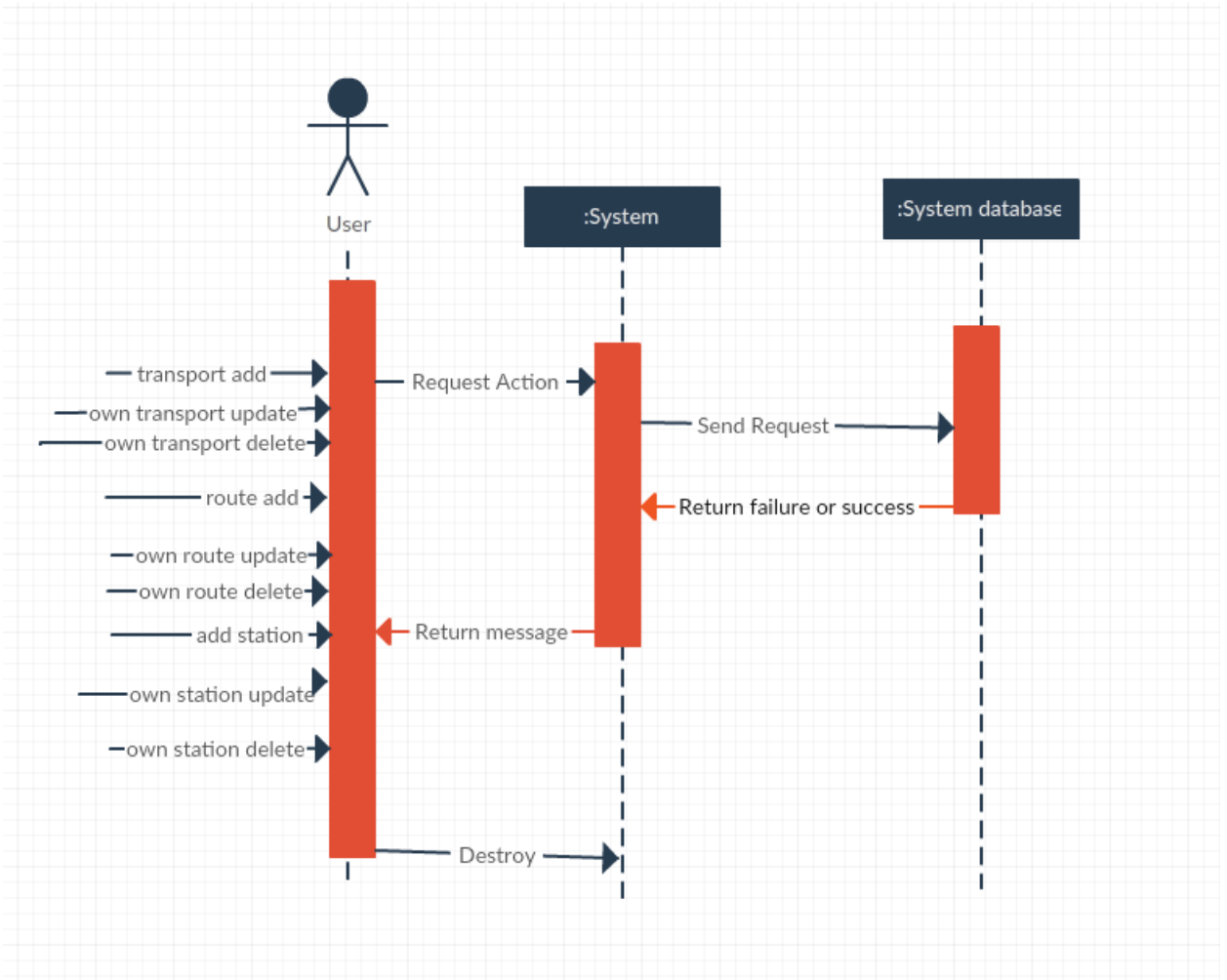


Figure 17: Sequence Diagram for Make Contribution

4.1.7. Sequence Diagram for Give Feedback

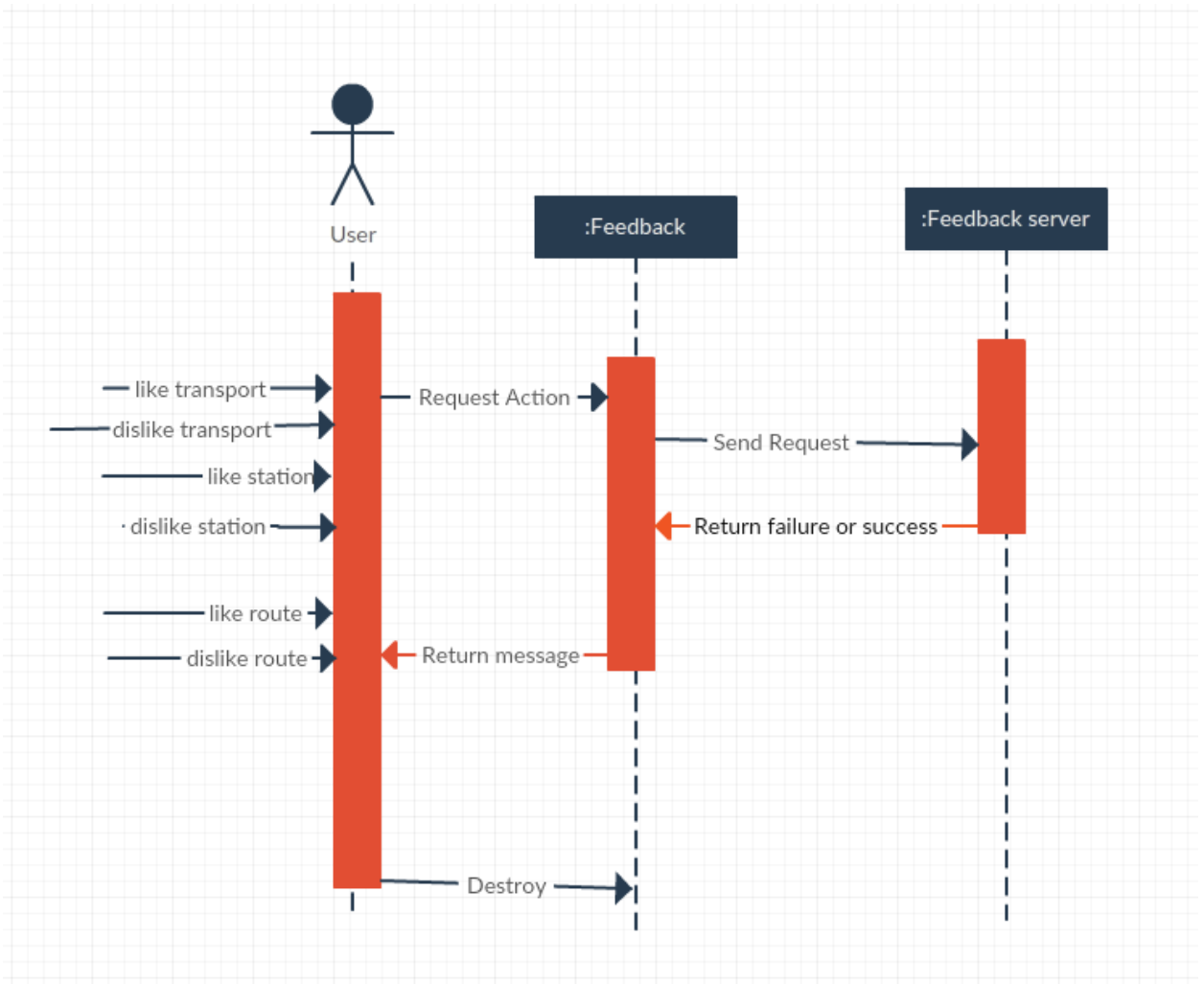


Figure 18: Sequence Diagram for Give Feedback

4.1.8. Sequence Diagram for Manage Profile

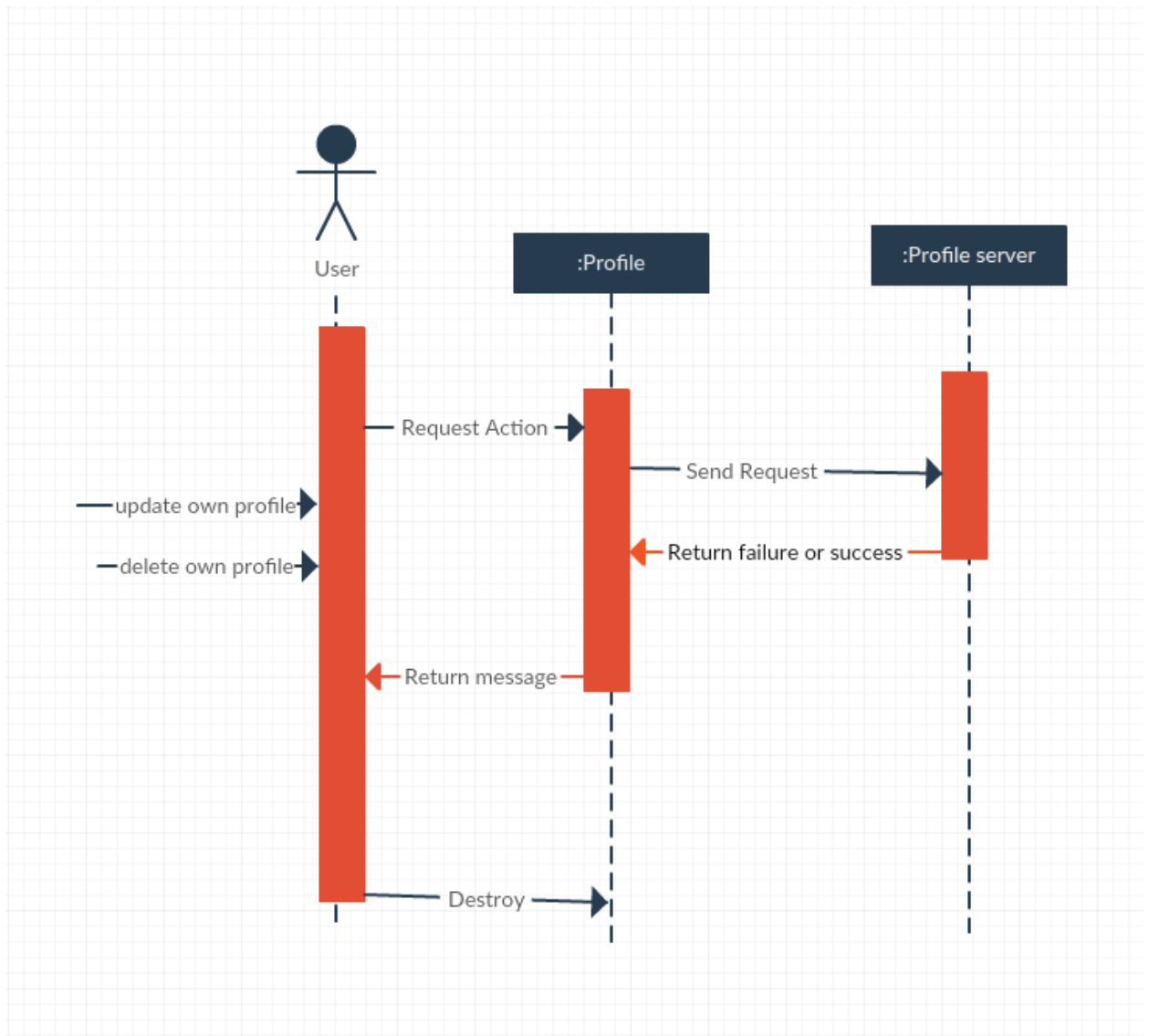


Figure 19: Sequence Diagram for Manage Profile

4.1.9. Sequence Diagram for Search Information's

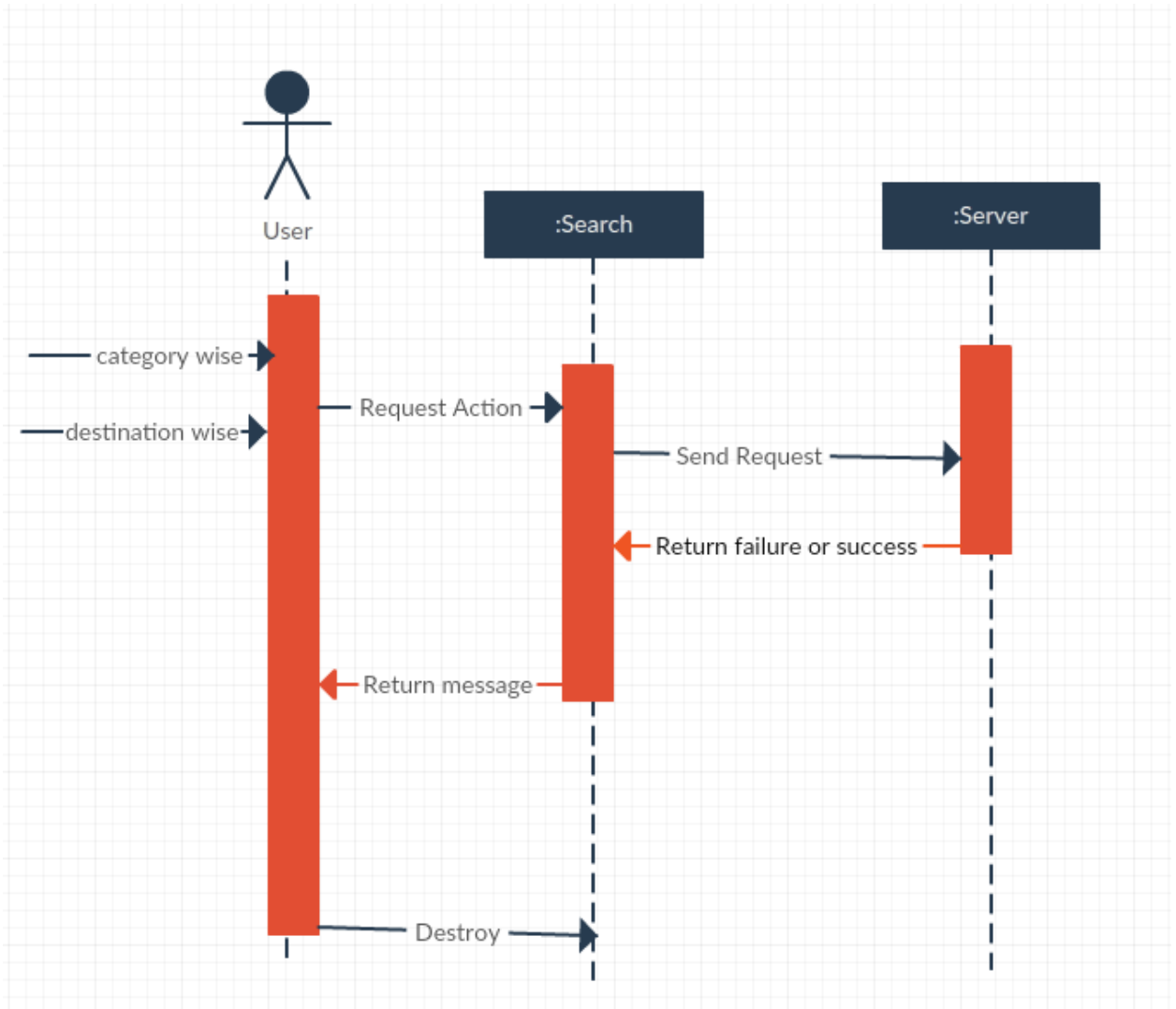


Figure 20: Sequence Diagram for Search Information's

4.2. ER Diagram

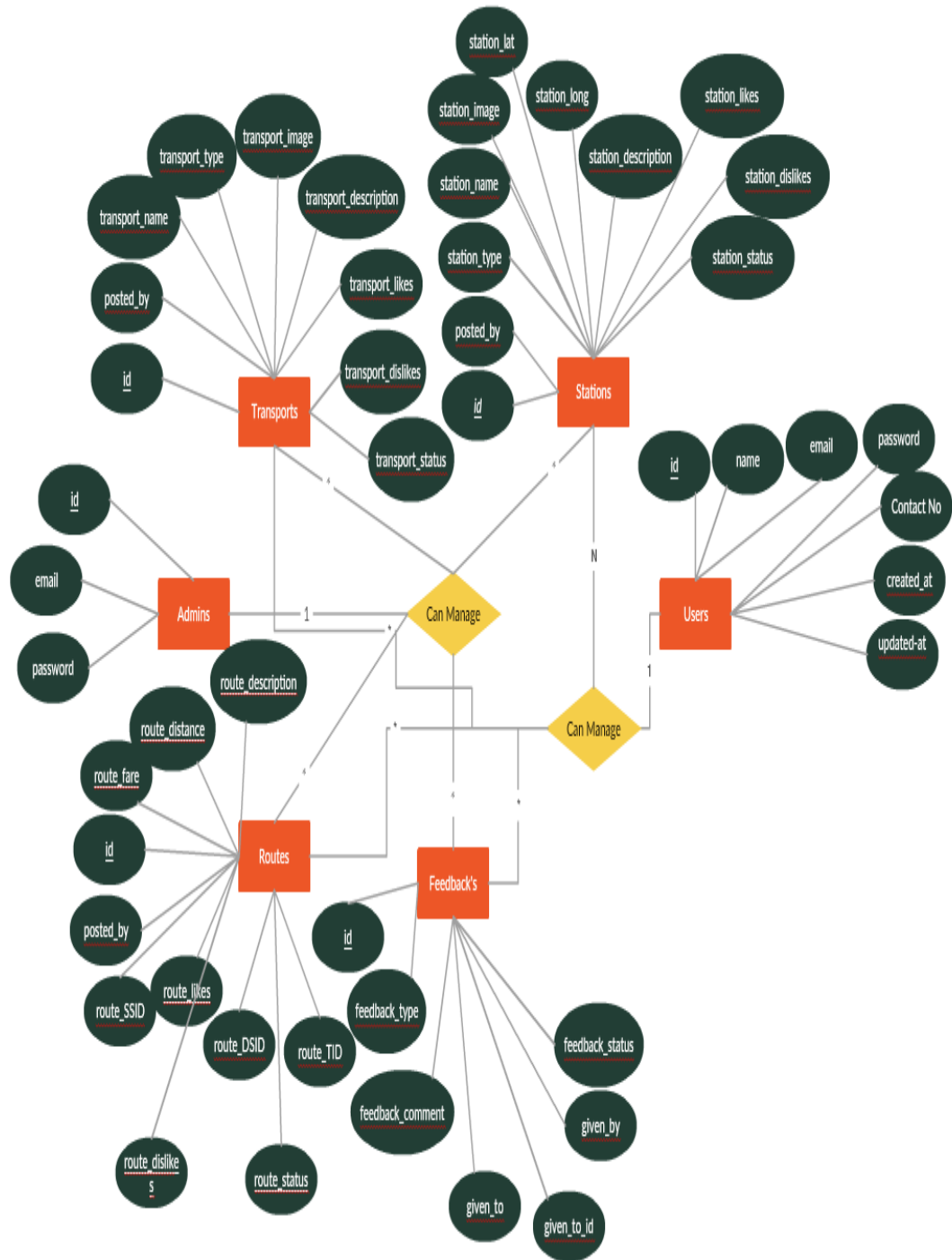


Figure 21: ERD

Chapter V User Interface and Manuals

5.1. User Interface

5.1.1. Home Page

This is the home page where you can search by providing to and from. And you can also navigate to other menus.

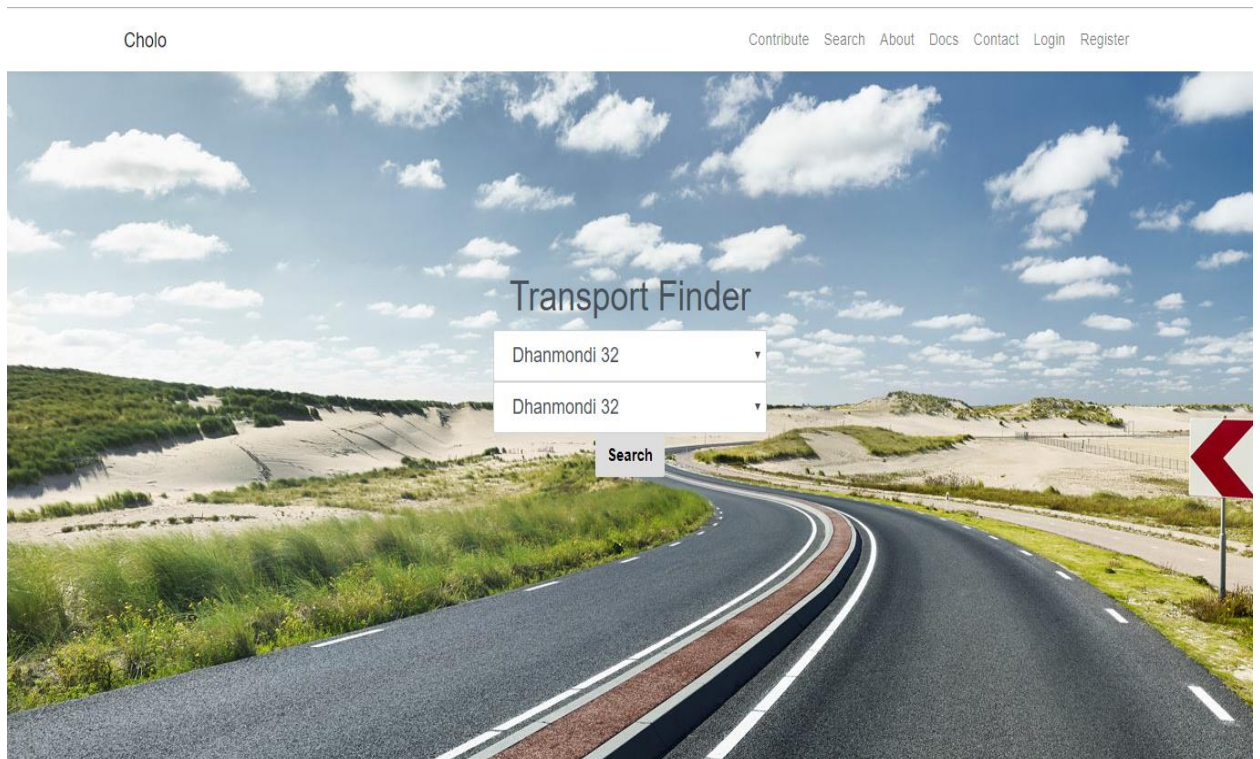


Figure 22: Home Page

5.1.2. Destination Wise Search Result

After clicking on search by providing starting station and destination. This page would appear.

You can go to transport profile by clicking on transport name from here. And You can give feedback by clicking on feedback. But you must be logged in to do so.

The screenshot shows a web interface for searching transport options. At the top, there is a header with the word 'Cholo' on the left and navigation links: 'Contribute', 'Search', 'About', 'Docs', 'Contact', 'Login', and 'Register'. Below the header, the main content area is titled 'All Available Transports from Dhanmondi 32 To Shyamoli'. This title is above a table that lists transport options. The table has columns for 'Transport Name', 'Type', 'Fare', 'Distance', 'Description', 'Likes', 'Dislikes', and 'Action'. A single row is visible with the following data: 'Moumita' (Transport Name), 'Bus' (Type), '10' (Fare), '422.74' (Distance), 'This is a Intercity Bus which runs through Savar To Saidabad.' (Description), and a 'Give Feedback' button (Action).

Transport Name	Type	Fare	Distance	Description	Likes	Dislikes	Action
Moumita	Bus	10	422.74	This is a Intercity Bus which runs through Savar To Saidabad.			Give Feedback

Figure 23: Result of Search by Destination Module

5.1.3. Search by Category

Here user can search by category such as station, transport, bus, train etc.

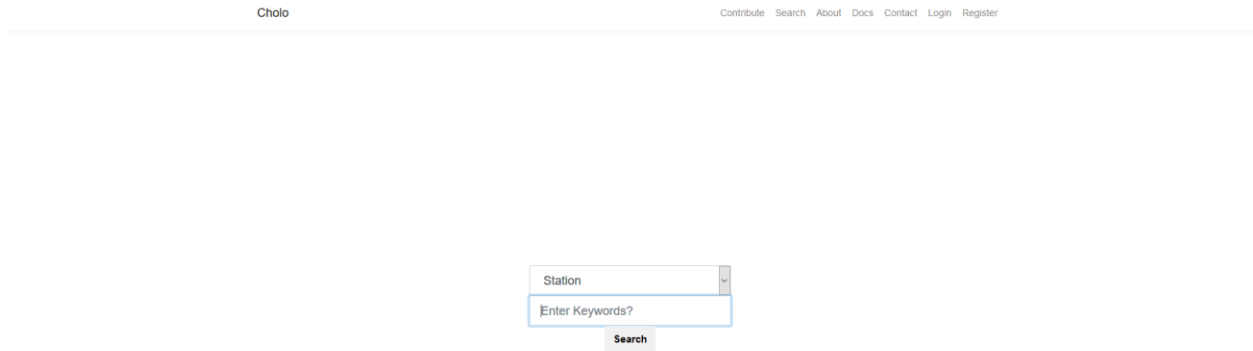
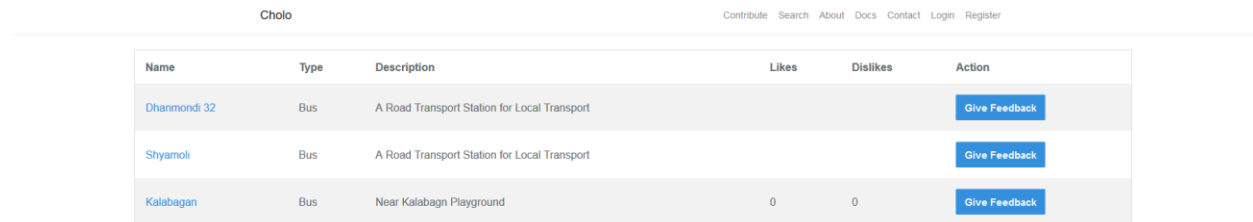


Figure 24: Search by Category Module

5.1.4. Category Wise Search Result

If you search for station category you will see a list like this.

You can go to station profile by clicking on station name from here. And You can give feedback by clicking on feedback. But you must be logged in to do so.




Name	Type	Description	Likes	Dislikes	Action
Dhanmondi 32	Bus	A Road Transport Station for Local Transport			Give Feedback
Shyamoli	Bus	A Road Transport Station for Local Transport			Give Feedback
Kalabagan	Bus	Near Kalabagn Playground	0	0	Give Feedback

Figure 25: Category Wise Search Result

5.1.5. Transport Profile

Transport profile page. This page includes all the station his transport has stop. And others information of transport. You can give feedback from here too.

Cholo Contribute Search About Docs Contact Login Register



Moumita Bus
Intercity Bus run in dhaka


Give Feedback					
Station Name	Type	Description	Likes	Dislikes	Action
Dhanmondi 32	Bus	A Road Transport Station for Local Transport			Give Feedback
Shyamoli	Bus	A Road Transport Station for Local Transport			Give Feedback

Figure 26: Transport Profile

5.1.6. Station Profile

Station profile page. This page includes all the station his transport has stop. And others information of transport. You can give feedback from here too.

Cholo Contribute Search About Docs Contact Login Register



Dhanmondi 32 Bus
A Road Transport Station for Local Transport

Give Feedback					
Transport Name	Type	Description	Likes	Dislikes	Action
Moumita	Bus	Intercity Bus run in dhaka	0	0	Give Feedback

Figure 27: Station Profile

5.1.7. User Login

Login page for user.

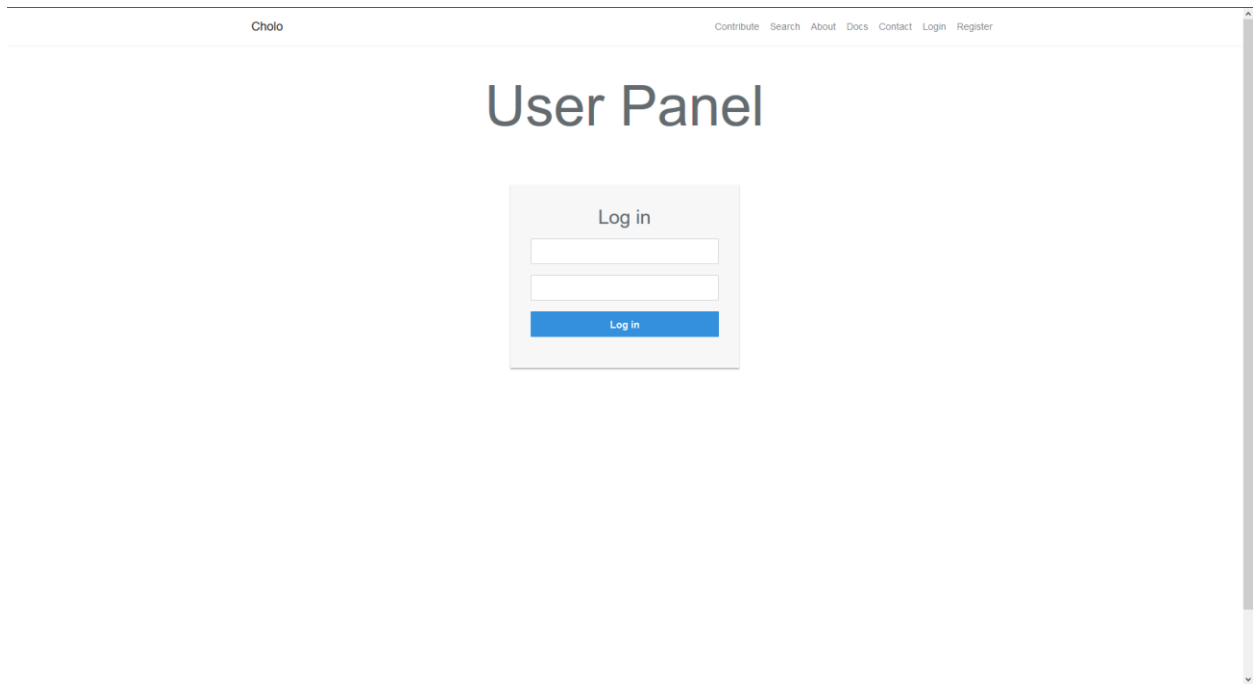


Figure 28: User Login

5.1.8. User Registration

Registration page for user.

Cholo Contribute Search About Docs Contact Login Register

Register

Name

E-Mail Address

Password

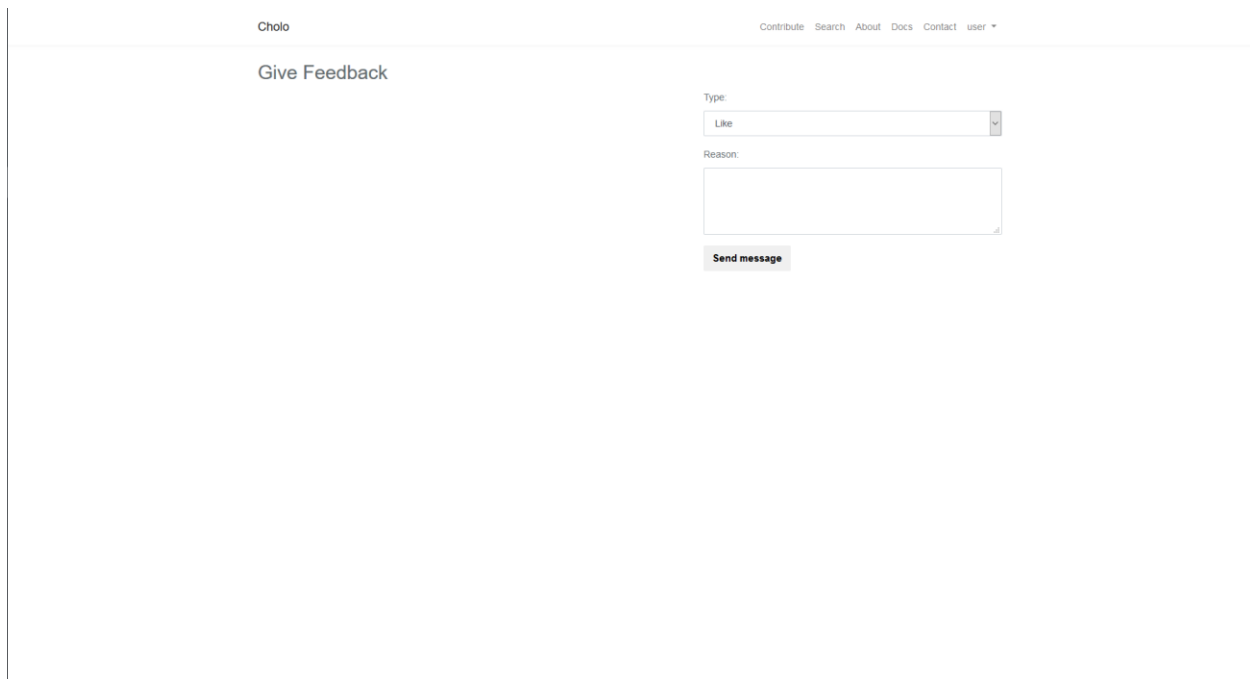
Confirm Password

[Register](#)

Figure 29: User Registration

5.1.9. Give Feedback

User can give feedback to any information such as transport, station, route. User must be logged in to give feedback.



The screenshot shows a web interface for giving feedback. At the top left, the word "Cholo" is displayed. At the top right, there are navigation links: "Contribute", "Search", "About", "Docs", "Contact", and "user" with a dropdown arrow. Below the navigation, the page title "Give Feedback" is centered. On the right side, there is a form with the following elements: a "Type:" label above a dropdown menu currently showing "Like"; a "Reason:" label above a large text input area; and a "Send message" button located below the text input area.

Figure 30: Give Feedback

5.1.10. User Panel

User panel. After user logged in user will be redirected to this page. Where user will be able to see total number of transport, station, route and feedback added by him/her.

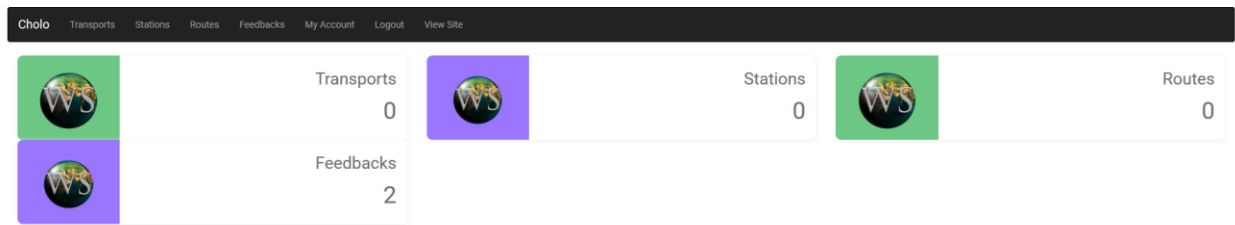


Figure 31: User Panel

5.1.11. Manage Transport - User

Here user can add, update, delete transport.

Cholo Transports Stations Routes Feedbacks My Account Logout View Site

Transport List

Transport Name	Transport Type	Transport Description	Transport Image	Transport Status	Actions
----------------	----------------	-----------------------	-----------------	------------------	---------

Add Transport

Transport Name:

Transport Type:

Description:

Image:

Add Transport

Figure 32: Manage Transport - User

5.1.12. Manage Station - User

Here user can add, update, delete station.

Cholo Transports Stations Routes Feedbacks My Account Logout View Site

Station List

Name	Type	Latitude	Longitude	Description	Image	Status	Actions
------	------	----------	-----------	-------------	-------	--------	---------

Add Station

Station Name:

Type:

Latitude:

Longitude:

Description:

Image:

Add Station

127.0.0.1:8000/user/home

Figure 33: Manage Station - User

5.1.13. Manage Route - User

Here user can add, update, delete route.

Cholo Transports Stations Routes Feedbacks My Account Logout View Site

Route List

From	To	Transport	Fare	Distance	Description	Status	Action
------	----	-----------	------	----------	-------------	--------	--------

Add Route

From: Dhanmondi 32

To: Dhanmondi 32

Transport: Moumita

Fare:

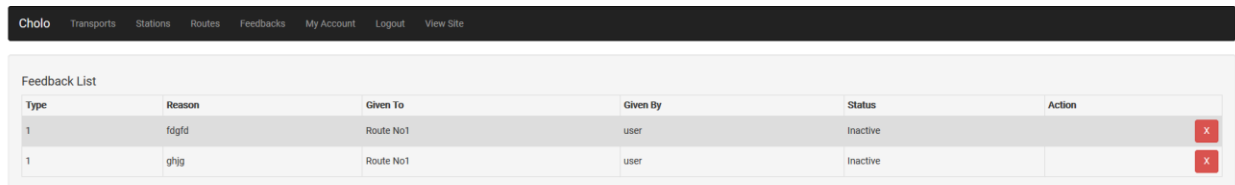
Description:

Add Route

Figure 34: Manage Route - User

5.1.14. Manage Feedback - User

Here user can delete feed he /she has given.





Type	Reason	Given To	Given By	Status	Action
1	fdgfd	Route No1	user	Inactive	
1	ghjg	Route No1	user	Inactive	

Figure 35: Manage Feedback - User

5.1.15. Manage Profile - User

Here user can update his profile. Such as changing name, email, password.

Id	Name	Email	Actions
2	user	user@cholo.com	E

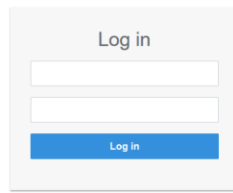
127.0.0.1:8000/user/home

Figure 36: Manage Profile -User

5.1.16. Admin Login

Login page for admin.

Admin Panel



A screenshot of an admin login form. The form is titled "Log in" and contains two input fields for username and password, followed by a blue "Log in" button.

Figure 37: Admin Login

5.1.17. Admin Dashboard

Admin Dashboard where admin can see total no of transport, station, feedback, route, user the system has.

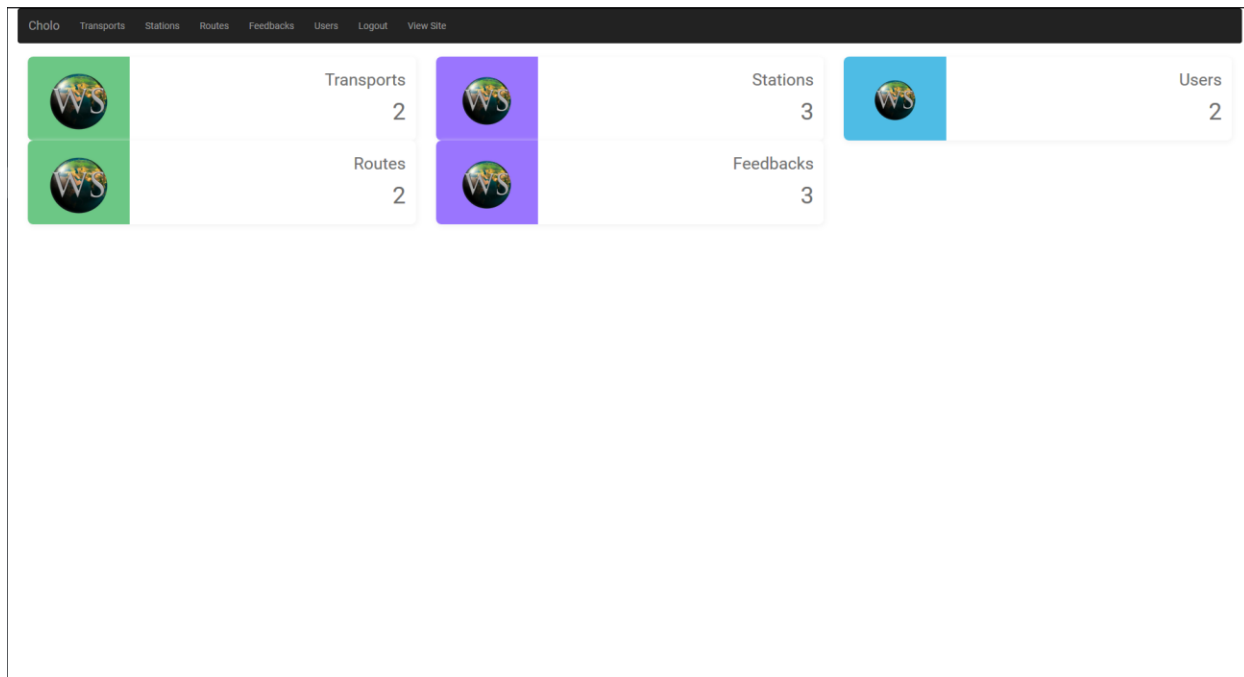




Figure 38: Admin Dashboard

5.1.18. Manage Transport - Admin

Admin can add, update, delete transport. And can approve transport added by user.

The screenshot displays the 'Manage Transport - Admin' interface. At the top, a navigation bar includes links for Cholo, Transports, Stations, Routes, Feedbacks, Users, Logout, and View Site. The main content is divided into two sections: a 'Transport List' table and an 'Add Transport' form.

Transport Name	Transport Type	Transport Description	Transport Image	Transport Status	Actions
Moumita	Bus	Intercity Bus run in dhaka		Active	E X
Transsilva	Bus	Intercity Bus Service		Inactive	E X

The 'Add Transport' form includes the following fields:

- Transport Name:** A text input field.
- Transport Type:** A dropdown menu with 'Bus' selected.
- Description:** A large text area.
- Image:** A file upload field with a 'Browse...' button and the text 'No file selected.'
- Status:** A dropdown menu with 'Active' selected.
- Add Transport:** A submit button.




Figure 39: Manage Transport - Admin

5.1.19. Manage Station - Admin

Admin can add, update, delete station. And can approve station added by user.

The screenshot shows the 'Manage Station - Admin' interface. At the top, there is a navigation bar with the following items: Cholo, Transports, Stations, Routes, Feedbacks, Users, Logout, and View Site. Below the navigation bar, the main content area is divided into two sections.

Station List

Name	Type	Latitude	Longitude	Description	Image	Status	Actions
Dhanmondi 32	Bus	32.9697	-96.8032	A Road Transport Station for Local Transport		Active	E X
Shyamoli	Bus	29.4679	-98.5351	A Road Transport Station for Local Transport		Active	E X
Kalabagan	Bus	24.0984	90.3287	Near Kalabagn Playground		Inactive	E X

Add Station

Station Name:

Type:

Latitude:

Longitude:

Description:

Image:

Status:

© 2019 Daffodil International University

Figure 40: Manage Station - Admin

5.1.20. Manage Route - Admin

Admin can add, update delete route. And can approve route added by user.

The screenshot displays the 'Manage Route - Admin' interface. At the top, a navigation bar includes 'Cholo' and links for 'Transports', 'Stations', 'Routes', 'Feedbacks', 'Users', 'Logout', and 'View Site'. The main content is divided into two sections:

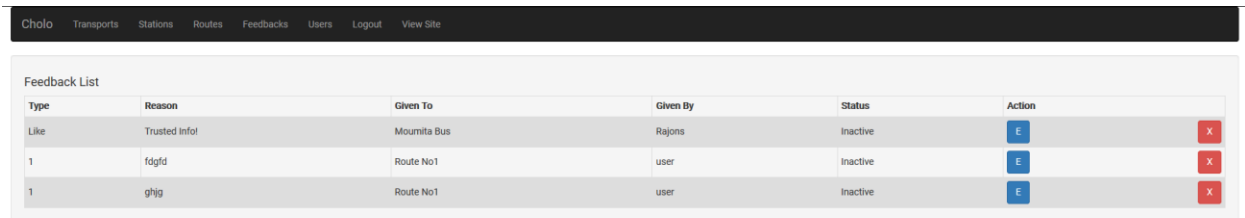
- Route List:** A table with the following data:

From	To	Transport	Fare	Distance	Description	Status	Action
Dhanmondi 32	Dhanmondi 32	Moumita	10	422.74	This is a Intercity Bus which runs through Savar To Saidabad.	Active	E X
Shyamoli	Shyamoli	Moumita	10	0	This is a Intercity Bus which runs through Savar To Saidabad.	Active	E X
- Add Route:** A form with the following fields:
 - From:** Dropdown menu with 'Dhanmondi 32' selected.
 - To:** Dropdown menu with 'Dhanmondi 32' selected.
 - Transport:** Dropdown menu with 'Moumita' selected.
 - Fare:** Text input field.
 - Description:** Text area.
 - Status:** Dropdown menu with 'Active' selected.
 - Add Route:** Button.

Figure 41: Manage Route - Admin

5.1.21. Manage Feedback - Admin

Admin can update the status and delete the feedback.



The screenshot displays a web application interface with a navigation menu at the top containing: Cholo, Transports, Stations, Routes, Feedbacks, Users, Logout, and View Site. Below the menu is a 'Feedback List' table with the following data:

Type	Reason	Given To	Given By	Status	Action
Like	Trusted Infol	Mournita Bus	Rajons	Inactive	E X
1	fdgfd	Route No1	user	Inactive	E X
1	ghjg	Route No1	user	Inactive	E X

Figure 42: Manage Feedback - Admin

5.1.22. Manage User - Admin

Admin can add update, delete user.

The screenshot displays the 'Manage User - Admin' interface. At the top, a navigation bar includes the 'Cholo' logo and links for 'Transports', 'Stations', 'Routes', 'Feedbacks', 'Users', 'Logout', and 'View Site'. The main content area is divided into two sections:

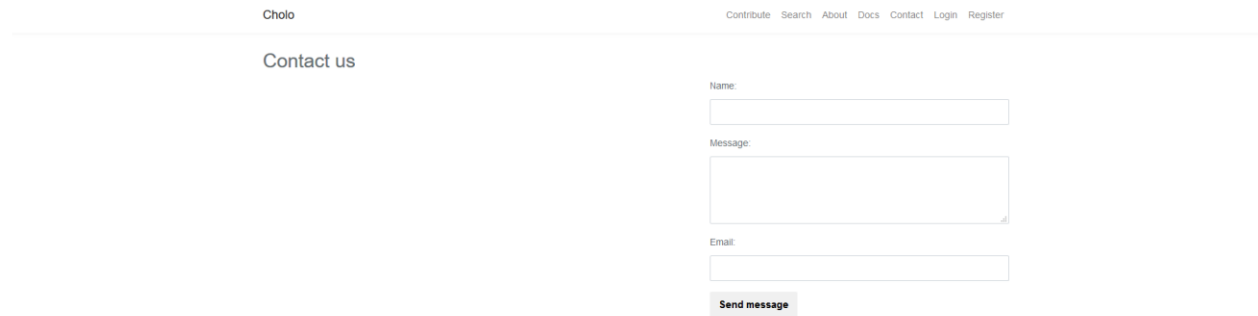
- User list:** A table with the following data:

id	Name	Email	Actions
1	Rajons	admin@app.com	[E] [X]
2	user	user@cholo.com	[E] [X]
- Create User:** A form with three input fields labeled 'Name:', 'Email:', and 'Password:', and a 'Create User' button.

Figure 43: Manage User - Admin

5.1.23. Contact Page

User will be able to contact with admin using this page.



The image shows a contact page layout. At the top left, the word "Cholo" is displayed. At the top right, there is a navigation menu with links: "Contribute", "Search", "About", "Docs", "Contact", "Login", and "Register". Below the navigation, the page title "Contact us" is centered. On the right side, there is a form with three input fields: "Name:" (a single-line text box), "Message:" (a multi-line text area), and "Email:" (a single-line text box). Below the email field is a "Send message" button.

Figure 44: Contact Page

Chapter VI

Development Tools & Technologies

6.1. User Interface Technologies

- HTML5, CSS3, JavaScript
- JQuery 3.2.1
- Twitter Bootstrap
- Font Awesome

6.2. Implementation Technologies

6.2.1. Laravel (5.8)

Laravel is a free, open-source PHP web framework.

6.2.2. Composer (1.8.5)

Composer is an application-level package manager for the PHP programming language that provides a standard format for managing dependencies of PHP software and required libraries

6.2.3. Mysql (15.1)

MySQL is an open-source relational database management system. its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language.

6.2.4. PHP (7.3)

PHP: Hypertext Preprocessor is a server-side scripting language designed for Web development, but also used as a general-purpose programming language. It was originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Group.

6.2.5. XAMPP (3.22)

XAMPP is a free and open source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages.

6.3. Platform & Environment

6.3.1. Hardware

- Processor: Intel Core i5.
- RAM: 4GB.
- Hard drive: 500GB.
- Windows 10

6.3.2. Tools

- IDE/Editor: Visual Studio Code
- Git Bash Terminal
- Server: Localhost:3000

6.3.3. Version Control

- Git
- Github

Chapter VII System Testing

7.1. Introduction

This is aimed at identifying and correcting error. The major objectives of this activity are to ensure that the process done by the application is correct and meets the objectives of the organization. Test plan aids in effective and systematic testing of the system and it aims at checking the errors of omission and commission that hinders the realization of the objectives.

7.2. Test Plan Strategy

The importance of the test plan is to show how the system is to be tested and also gives precise procedure to be followed during test plan. The test data is identified, what is being tested and the expected outcome as well as actual input. Test plan is one of the standard documents that should be produced in most software engineering projects. If the project does not have any test plan this means that the system produced is low quality. This may not be acceptable to the user it will not satisfy their needs. The test plan should be written as soon as requirements have been identified. The system will be tested with sample data to see how it would handle input and output functions as well as extreme data or conditions to determine the system behavior in overloaded situation which will directly slow the system that behaves in failure or extreme situation.

7.3. Test Case

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works properly. The process of developing test case can help find problems in the requirements or design of an application.

- Ensure that logical decisions on their true and false side.
- Practice all the logical decisions on their true and false side.
- Check equivalent partitions and boundary value within their operations bounds.
- Exercise internal data structure to assure their validity.

7.3.1. Test Case of Login

Test Case #01	Test Case Name: Testing the login panel.
System: TF	Subsystem: Login
Designed By: Mourin Mondol	Design date: 05.04.19
Executed By: Mourin Mondol	Execute Date:05.04.19
Short Description: This field handle's the login functionality of the website.	
Precondition: Go to http://localhost:8000/login . http://localhost:8000/admin/login	

Table 26: Test Case of Login

Steps	Action	Action Result	Expected System Response	Pass/Fail
01	Enter valid email and valid password	Get logged in.	Logged in into the system.	Pass
02	Valid email and invalid password	Not logged in and error message.	Not logged in and error message.	Pass
03	Click login without any data	Required message	Required message	Pass

7.3.2. Test Case of Register

Test Case #01	Test Case Name: Testing the Register panel.
System: TF	Subsystem: Registration
Designed By: Mourin Mondol	Design date: 6.04.19
Executed By: Mourin Mondol	Execute Date:6.04.19
Short Description: This field handle's the login functionality of the website.	
Precondition: Go to http://127.0.0.1:8000/register	

Table 27: Test Case of Register

Steps	Action	Action Result	Expected System Response	Pass/Fail
01	Click register without any data	Required message	Required message	Pass
02	Click register partially filling with data	Not registered and required messages.	Not registered and required messages.	Pass
03	Click register valid data and password less than 6 characters.	registered in and error message.	Required message	Pass
04	Click register with valid data and password not matching with confirm password.	Not registered in and error message.	Not registered in and error message.	Pass
05	Click register with valid data and password.	Registered and redirected to main landing page.	registered and redirected to main landing page.	Pass

7.3.3. Test Case of Search by Destination

Test Case #01	Test Case Name: Testing the search by destination module
System: TF	Subsystem: Search by Destination
Designed By: Mourin Mondol	Design date: 6.04.19
Executed By: Mourin Mondol	Execute Date:6.04.19
Short Description: This field handle's the search by destination functionality of the system.	
Precondition: Go to http://127.0.0.1:8000/	

Table 28: Test Case of Search by Destination

Steps	Action	Action Result	Expected System Response	Pass/Fail
01	Click search without any data	Required message	Required message	Pass
02	Click search with data.	Show search result	Show Search result	Pass
03	Click on give feedback button in result page	Redirected to give feedback page	Redirect give feedback page	Pass
04	Click on transport name In search result page	Redirected to Transport profile	Redirect to transport profile	Pass

7.3.4. Test Case of Search by Category

Test Case #01	Test Case Name: Testing the search by category module
System: TF	Subsystem: Search by Category
Designed By: Mourin Mondol	Design date: 6.04.19
Executed By: Mourin Mondol	Execute Date:6.04.19

Short Description: This field handle's the search by category functionality of the system.

Precondition: Go to <http://127.0.0.1:8000/search>

Table 29: Test Case of Search by Category

Steps	Action	Action Result	Expected System Response	Pass/Fail
01	Click search without any data	Required message	Required message	Pass
02	Click on station name	Redirected to station profile	Redirect to station profile	Pass
03	Click search with correct data	Returned search result	Return search result	Pass
04	Click on feedback button In search by category page	Redirected to feedback page	Redirect to feedback page	Pass
05	Search by selecting transport/bus/stations/train/launch/airplane	Return desired Type pf info's	Should return selected type of info	Pass

7.3.6. Test Case of Manage Stations

Test Case #01	Test Case Name: Testing manage stations
System: TF	Subsystem: Manage Stations
Designed By: Mourin Mondol	Design date: 6.04.19
Executed By: Mourin Mondol	Execute Date:6.04.19
Short Description: This field handle's manage stations functionality of the system for admin.	
Precondition: Go to http://127.0.0.1:8000/ admin/stations	

Table 30: Test Case of Manage Stations

Steps	Action	Action Result	Expected System Response	Pass/Fail
01	Click save without any data	Required message	Required message	Pass
02	Click save partially filling with data	Not saved and required messages.	Not saved and required messages.	Pass
03	Click save with filling all required data	Saved and redirect To station list	Saved and redirect to station list	Pass
04	Click on update and Change value then save	Station updated And redirect to station list	Station updated successfully And redirect to station list	Pass
05	Click on delete station	Station deletion Successful.	Station deleted successfully	Pass

7.3.6. Test Case of Manage Transports

Test Case #01	Test Case Name: Testing manage transports
System: TF	Subsystem: Manage Transports
Designed By: Mourin Mondol	Design date: 6.04.19
Executed By: Mourin Mondol	Execute Date:6.04.19
Short Description: This field handle's manage transports functionality of the system for admin.	
Precondition: Go to http://127.0.0.1:8000/ admin/transport	

Table 31: Test Case of Manage Transport

Steps	Action	Action Result	Expected System Response	Pass/Fail
01	Click save without any data	Required message	Required message	Pass
02	Click save partially filling with data	Not saved and required messages.	Not saved and required messages.	Pass
03	Click save with filling all required data	Saved and redirect To list	Saved and redirect to list	Pass
04	Click on update and Change value then save	transport updated And redirect to list	transport updated successfully And redirect to list	Pass
05	Click on delete transport	transport deletion Successful.	transport deleted successfully	Pass

7.3.7. Test Case of Manage Routes

Test Case #01	Test Case Name: Testing manage routes
System: TF	Subsystem: Manage Routes
Designed By: Mourin Mondol	Design date: 6.04.19
Executed By: Mourin Mondol	Execute Date:6.04.19
Short Description: This field handle's the manage routes functionality of the system.	
Precondition: Go to http://127.0.0.1:8000/ admin/routes	

Table 32: Test Case of Manage Routes

Steps	Action	Action Result	Expected System Response	Pass/Fail
01	Click save without any data	Required message	Required message	Pass
02	Click save partially filling with data	Not saved and required messages.	Not saved and required messages.	Pass
03	Click save with filling all required data	Saved and redirect To list	Saved and redirect to list	Pass
04	Click on update and Change value then save	Route updated And redirect to list	route updated successfully And redirect to list	Pass
05	Click on delete route	route deletion Successful.	route deleted successfully	Pass

7.3.8. Test Case of Manage Feedbacks

Test Case #01	Test Case Name: Testing manage feedback
System: TF	Subsystem: Manage Feedback
Designed By: Mourin Mondol	Design date: 6.04.19
Executed By: Mourin Mondol	Execute Date:6.04.19
Short Description: This field handle's the Manage Feedbacks functionality of the system.	
Precondition: Go to http://127.0.0.1:8000/ admin/feedbacks	

Table 33: Test Case of Manage Feedbacks

Steps	Action	Action Result	Expected System Response	Pass/Fail
01	Click on update and Change status	Status changed successfully	Status changed successfully	Pass
05	Click on delete	Successfully deleted	Successfully deleted	Pass

7.3.9. Test Case of Manage Users

Test Case #01	Test Case Name: Testing manage users
System: TF	Subsystem: Manage Users
Designed By: Mourin Mondol	Design date: 6.04.19
Executed By: Mourin Mondol	Execute Date:6.04.19
Short Description: This field handle's the manage profile functionality of the system.	
Precondition: Go to http://127.0.0.1:8000/ admin/users	

Table 34: Test Case of Manage Users

Steps	Action	Action Result	Expected System Response	Pass/Fail
01	Click save without any data	Required message	Required message	Pass
02	Click save partially filling with data	Not saved and required messages.	Not saved and required messages.	Pass
03	Click save with filling all required data	Saved and redirect To list	Saved and redirect to list	Pass
04	Click on update and Change value then save	Station updated And redirect to station list	updated successfully And redirect to station list	Pass
05	Click on delete	deletion Successful.	deleted successfully	Pass

7.3.10. Test Case of Manage Profile

Test Case #01	Test Case Name: Testing manage profile
System: TF	Subsystem: Manage Profile
Designed By: Mourin Mondol	Design date: 6.04.19
Executed By: Mourin Mondol	Execute Date:6.04.19
Short Description: This field handle's the manage profile functionality of the system.	
Precondition: Go to http://127.0.0.1:8000/ user/profile	

Table 35: Test Case of Manage Profile

Steps	Action	Action Result	Expected System Response	Pass/Fail
01	Click update without any data	Required message	Required message	Pass
02	Click update filling with data	successful	Successfully updated	Pass
03	Click on delete	Account deleted	Account should be deleted.	Pass

7.3.11. Test Case of Make Contribution

Test Case #01	Test Case Name: Testing make contribution
System: TF	Subsystem: Make Contribution
Designed By: Mourin Mondol	Design date: 6.04.19
Executed By: Mourin Mondol	Execute Date:6.04.19
Short Description: This field handle's the make contribution functionality of the system.	
Precondition: Go to http://127.0.0.1:8000/ user/transport , http://127.0.0.1:8000/ user/station , http://127.0.0.1:8000/ user/route	

Table 36: Test Case of Make Contribution

Steps	Action	Action Result	Expected System Response	Pass/Fail
01	Click save without any data	Required message	Required message	Pass
02	Click save partially filling with data	Not saved and required messages.	Not saved and required messages.	Pass
03	Click save with filling all required data	Saved and redirect To list	Saved and redirect to list	Pass
04	Click on update and Change value then save	Station updated And redirect to station list	Station updated successfully And redirect to station list	Pass
05	Click on delete	deletion Successful.	deleted successfully	Pass

7.3.11. Test Case of Give Feedback

Test Case #01	Test Case Name: Testing the search by destination
System: TF	Subsystem: Give Feedback
Designed By: Mourin Mondol	Design date: 6.04.19
Executed By: Mourin Mondol	Execute Date:6.04.19
Short Description: This field handle's the give feedback functionality of the system.	
Precondition: Go to http://127.0.0.1:8000/give_feedback?id=1&given_to=Route	

Table 37: Test Case of Give Feedback

Steps	Action	Action Result	Expected System Response	Pass/Fail
01	Select like and click send	Positive feedback given	Positive feedback given	Pass
02	Select dislike and click send	Negative feedback given	Negative feedback given	Pass
03	Send feedback without comment	Successfully given	Successfully given	Pass
04	50 dislikes	Untrusted Info deleted	Untrusted Info deleted	Pass

7.4. Features yet not tested

Table 38: Features yet not tested

No	Name	Users
01	Contact with admin	Admin/User

Chapter VIII

Project Summary

8.1. GitHub Link

https://github.com/mourinmondol/transport_finder_final

8.2. Limitations

- The system does not able to track transport.
- System does not offer nearby station finder.

8.3. Obstacle & Achievements

Obstacle:

- Learning new technology and environment
- Limited time and budget

Achievements

- Learnt new technology
- Successfully build a project for production level

8.4. Conclusion

Despite the hardships encountered in the entire development process the system has been developed for transport finder system thus enabling it improves its efficiency and effectiveness. Maintenance and usage of the system will be easy as the document and user manual of the system will be available to all users. Also, there will be room for enhancement as this was considered during development. The system will make it easy for user to get transport/station information and plan the day.

8.5. Future Work

Though the system was developed as fine but the future work will include some more major changes. GPS tracking will be integrated and route planner with more feature will be introduced.

Key of Terms

A

Abstract

Acknowledgement

Approval

Associate Review Analysis

H

Hardware and Software

Specification Html5

C

Conclusion

S

System description

Software Requirement Specification

Software Development Plan

System Design

System Overview

System Scope

Software Specification

F

Future work

Functional Requirements

I

Implementation

Introduction

D

Definition and abbreviations

Design map

Database Name

J

JavaScript

JQuery

T

Testing

Test case

Tools

Technical Description

Transport Finder

Transport Finder Relationship Diagram

U

Use case diagrams

N

Non-Functional requirements

References

Wikipedia, “Project Management,” [Online]. Available: [https://en.wikipedia.org/wiki/Schedule_\(project_management\)](https://en.wikipedia.org/wiki/Schedule_(project_management)). [Accessed: 25-November-2018].

K. Waters, “All About Agile,” All About Agile Agile Development Cycle Comments, 04-December-2011. [Online]. Available: <http://www.allaboutagile.com/agile-development-cycle/>. [Accessed: 4-December-2018].

I. Somerville, Software engineering. Boston: Pearson Education Limited, 2018. [Accessed: 05-December-2018].

I. Somerville, Software engineering. Boston: Pearson Education Limited, 2016. [Accessed: 05-December-2018].

Agile Model Figure. Available: https://www.tutorialspoint.com/sdlc/sdlc_agile_model.htm

Project Management. Available: <https://www.projectmanager.com/blog/what-is-project-scheduling>

Agile Methodology. Available: <https://www.topsimages.com/images/agile-sdlc-methodology-e5.html>