Software Requirements Specification

for

TransPortEase

Prepared by

Nafisa Binte Jashim BKH2125014F
Mohammad Asif Iqbal ASH2125027M
Md. Sajedul Karim Munna ASH2125024M

Institute of Information Technology
Noakhali Science and Technology University

Contents

List of Figure

1. Introduction

In today's fast-paced world, efficient transportation systems are crucial for the smooth functioning of any organization. Noakhali Science and Technology University (NSTU) recognizes the importance of an efficient bus transportation system to cater to the needs of its students, faculty, and staff. With this in mind, we introduce "TransPortEase" a comprehensive mobile application aimed at enhancing the NSTU bus system experience for its users.

TransPortEase offers a user-friendly interface that enables students to access vital information about bus schedules, reserve seats, track bus locations in real-time, receive notifications, provide feedback, and more. Additionally, the application provides bus drivers with tools to optimize routes, handle complaints, and offer emergency assistance promptly. By leveraging modern technology, TransPortEase aims to streamline the NSTU bus system, making commuting more convenient and efficient for everyone involved.

1.1 Problem Statement

The current bus transportation system at Noakhali Science and Technology University (NSTU) faces several pressing challenges that significantly impact the commuting experience for students, faculty, and staff. Chief among these challenges is the lack of real-time information accessibility, leading to uncertainty and delays in bus schedules and locations. Additionally, the existing seat reservation process is cumbersome and inefficient, causing frustration among users. Communication channels between bus drivers and passengers are limited, hindering the prompt addressing of issues such as route optimization, complaints, and emergency assistance. Furthermore, the feedback mechanism is inadequate, preventing users from providing valuable insights for system improvement. Ensuring security and authentication within the transportation system is also a concern, as maintaining user privacy and trust is essential. Addressing these challenges is paramount to enhancing the overall experience of the NSTU bus system and ensuring the smooth and efficient transportation of its stakeholders. TransPortEase aims to tackle these issues comprehensively, leveraging modern technology to streamline bus operations and improve the user experience.

1.2 Purpose

The primary purpose of our project, "TransPortEase," is to revolutionize the bus transportation system at Noakhali Science and Technology University (NSTU) by leveraging modern technology to enhance efficiency, convenience, and user experience. The project aims to address the following key objectives:

Improve Accessibility: By providing real-time bus schedules, seat reservation options, and live tracking features, TransPortEase aims to make bus transportation more accessible and convenient for students, faculty, and staff at NSTU.

Enhance Communication: The project seeks to facilitate seamless communication between bus drivers and passengers through features such as feedback mechanisms, complaint management systems, and emergency assistance functionalities, ensuring prompt resolution of issues and concerns.

Optimize Operations: TransPortEase aims to optimize bus route planning and scheduling, enabling bus drivers to navigate efficiently and effectively while minimizing delays and maximizing passenger satisfaction.

E**nhance User Experience:** By offering a user-friendly interface, personalized notifications, and convenient booking options, the project endeavors to enhance the overall user experience of the NSTU bus transportation system, making commuting more comfortable and enjoyable for all stakeholders.

Ensure Security and Reliability: TransPortEase prioritizes the security and reliability of the bus transportation system by implementing robust authentication measures, safeguarding user data, and providing emergency assistance features to ensure passenger safety and peace of mind.

Overall, the purpose of the TransPortEase project is to modernize and optimize the NSTU bus transportation system, making it more efficient, user-friendly, and responsive to the needs of its diverse user base, thereby enriching the university experience for all.

1.3 Project Scope

The scope of the "TransPortEase" project encompasses the development of a comprehensive mobile application tailored to meet the specific needs of the Noakhali Science and Technology University (NSTU) bus transportation system. The project will include the following key features and functionalities:

User Registration and Authentication: Allow students, faculty, and staff to register and authenticate their accounts securely within the application.

Bus Schedule Viewing: Provide users with access to real-time bus schedules, including departure times, routes, and destinations.

Seat Reservation System: Implement a user-friendly interface for users to reserve seats on university buses conveniently.

Real-time Bus Tracking: Enable users to track the location of buses in real-time on a map interface, providing them with accurate arrival times and locations.
.
Notification System: Implement a notification system to alert users about bus schedule updates, seat reservation confirmations, and other relevant information.

Feedback and Rating Mechanism: Allow users to provide feedback and ratings on their bus transportation experience, enabling continuous improvement of the system.

Driver Interface: Provide bus drivers with tools to view optimized routes, handle complaints, and offer emergency assistance efficiently.

Complaint Management System: Implement a mechanism for users to submit complaints and for administrators to track, manage, and resolve them promptly.

Emergency Assistance Feature: Include an emergency assistance button for users to request immediate help in case of emergencies during their bus commute.

Security and Authentication Measures: Ensure robust security measures to safeguard user data and authenticate user interactions within the application.

Optimized Route Planning: Provide bus drivers with tools to optimize routes based on traffic conditions, passenger demand, and other relevant factors.

Administrative Dashboard: Develop an administrative dashboard for system administrators to monitor and manage user accounts, bus schedules, feedback, complaints, and other system-related data.

The project scope will focus on the development and implementation of these features to enhance the overall efficiency, convenience, and user experience of the NSTU bus transportation system through the TransPortEase mobile application.

1.4 Glossary

Glossary for TransPortEase:

TransPortEase: The name of the mobile application developed for the Noakhali Science and Technology University (NSTU) bus transportation system, providing users with various features and functionalities for enhanced commuting experience.

NSTU: Abbreviation for Noakhali Science and Technology University, the academic institution for which the TransPortEase application is developed.

User: Any individual who interacts with the TransPortEase application, including students, faculty, staff, and bus drivers.

Registration: The process through which users create an account within the TransPortEase application, providing necessary information for authentication and access to features.

Authentication: The process of verifying the identity of users to ensure secure access to the TransPortEase application and its functionalities.

Bus Schedule: A timetable displaying the departure times, routes, and destinations of NSTU buses, accessible to users via the TransPortEase application.

Seat Reservation: The feature allowing users to book seats on NSTU buses in advance through the TransPortEase application, ensuring a guaranteed spot during their commute.

Real-time Bus Tracking: The functionality enabling users to monitor the live location and movement of NSTU buses on a map interface within the TransPortEase application.

Notification System: A feature of the TransPortEase application that sends alerts and updates to users regarding bus schedule changes, seat reservation confirmations, and other relevant information.

Feedback Mechanism: A system within the TransPortEase application that enables users to provide comments, suggestions, and ratings based on their bus transportation experience, contributing to system improvement.

Complaint Management System: The feature allowing users to submit complaints or report issues encountered during their bus commute through the TransPortEase application, managed and addressed by system administrators.

Emergency Assistance: A functionality within the TransPortEase application enabling users to request immediate help or assistance in case of emergencies during their bus commute.

Route Optimization: The process of planning and adjusting bus routes to improve efficiency, minimize delays, and enhance the overall effectiveness of the NSTU bus transportation system.

1.5 References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

1.6 Overview

TransPortEase is a revolutionary mobile application designed to streamline and enhance the bus transportation system at Noakhali Science and Technology University (NSTU). Developed to meet the diverse needs of students, faculty, staff, and bus drivers, TransPortEase offers a comprehensive set of features and functionalities aimed at optimizing the commuting experience for all stakeholders.

At its core, TransPortEase provides users with real-time access to essential information such as bus schedules, routes, and seat availability. Through the intuitive interface of the application, users can effortlessly reserve seats on NSTU buses, track their live locations, and receive timely notifications about schedule updates and seat confirmations.

One of the key highlights of TransPortEase is its robust communication platform, facilitating seamless interaction between bus drivers and passengers. Users can submit feedback, report issues, and even request emergency assistance directly through the application, ensuring prompt resolution and improved service quality.

TransPortEase also prioritizes security and reliability, implementing stringent authentication measures to safeguard user data and privacy. Additionally, the application features advanced route optimization tools, enabling bus drivers to navigate efficiently while minimizing delays and maximizing passenger satisfaction.

With its user-centric design and innovative features, TransPortEase aims to revolutionize the NSTU bus transportation system, making commuting more convenient, efficient, and enjoyable for all. Whether it's students planning their daily commute or bus drivers optimizing their routes, TransPortEase empowers every user to navigate the NSTU campus with ease and confidence.

2. Stakeholders and Characteristics

Here students are the main users of our system. They will be able to create their profile in our TransPortEase app . Administrator , Development team are the main author who have the main access of the site and have control power over the system. In our project Developers, Administrators , Testing Team , Passenger Support, Security Team are our primary stakeholders & Students , Bus Drivers are secondary stakeholder.

2.1 Developers

They are responsible for designing, developing, and maintaining the mobile application and its backend systems. The key characteristics of developers are:

- They will ensure that the app is being built according to the project plan and that it meets the needs of the users.
- Technical expertise in software development, knowledge of relevant programming languages and frameworks, and familiarity with mobile application development best practices.

2.2 Adminstrator

They would be responsible for communicating with stakeholders and making sure that the project stays on track and within budget. The key characteristics of administrator are:

- They will oversee the overall operation of the our TransPortEase bus system and are responsible for managing routes, schedules, and resources.

2.3 Testing Team

Testing team will be responsible for testing the app to ensure that it is functioning properly and meeting the desired quality standards . The key characteristics of testing team are:

- They would work with the development team to identify and fix any bugs or issues with the app.
- It includes attention to detail, ability to identify and document bugs and issues, and proficiency in various testing methodologies and tools.

2.4 Passenger Support

Helps passengers to figure out schedules, solve problems, and offering personalized assistance . The key characteristics of Passenger Support are:

- It will provide assistance and support to students using the mobile application
- It includes strong customer service skills, patience, ability to troubleshoot technical issues, and empathy towards student needs and concerns

2.5 Security Team

They will make sure our TransPortEase app is strong and safe from any threats . The key characteristics of Security Team are:

- They are responsible for ensuring the security and integrity of the mobile application and its data.
- It includes expertise in cybersecurity principles and practices, knowledge of relevant security protocols and standards, and ability to identify and mitigate security risks .

2.6 Students

Current bus travel lacks real-time coordination between the students of NSTU and drivers, leading to uncertainties in arrival times and locations . As a main users of our system the first demand of students will be to have a seamless travel experience . The key characteristics of Students are:

- They are the primary users of the mobile application and the university bus system .

- Students come from different background want their travel to be easy, safe, and reliable.

2.7 Bus Driver

They will play one of the most important parts in our system **.** The key characteristics of Bus Drivers are:

- They are responsible for operating the university buses and providing transportation services to students.
- It includes knowledge of bus routes and schedules, good driving skills, ability to communicate effectively with passengers, and adherence to safety protocols and regulations.

.

3. Design and Implementation Constrains

3.1 Network Connectivity

- The system heavily relies on network connectivity for real-time tracking, notifications, and communication between users and the server.
- Constraints may arise in areas with poor or unstable network coverage.

3.2 Hardware Limitations

- The performance and capabilities of users' devices (smartphones, tablets, etc.) may vary, influencing the design and functionality of the mobile application.
- The availability and compatibility of GPS on user devices may affect the accuracy of real-time tracking

3.3 Data Privacy and Security

- Adherence to data protection laws and university policies is crucial to ensure the security and privacy of user information.
- Secure authentication mechanisms and encryption should be implemented to protect sensitive data.

3.4 Integration with Existing Systems

- Integration with existing university databases and systems for user authentication and bus schedule data may pose constraints.
- Compatibility issues with legacy systems may need to be addressed during integration.

3.5 Regulatory Compliance

- The project must comply with university regulations and policies related to data handling, privacy, and technology usage.

3.6 Scalability Requirements

- The system should be designed to handle potential growth in the number of users, buses, and concurrent transactions as the university community expands.
- Scalability constraints may arise if the architecture is not prepared for increased loads.

3.7 User Interface Considerations

- The user interface should be designed to accommodate various devices and screen sizes, considering the diversity of users within the university community.
- Accessibility standards should be followed to ensure inclusivity.

3.8 Budgetary Constraints

- The project's budget may impact the choice of technologies, development resources, and the scope of features.
- Consideration of cost-effective solutions and efficient resource utilization is essential.

3.9 Technology Stack

- Constraints may exist based on the university's preferred or existing technology stack, influencing the selection of programming languages, frameworks, and databases.

3.10 Development Timeframe

- Time constraints may affect the development timeline, potentially limiting the inclusion of additional features or thorough testing.
- A phased approach to development may be necessary to meet deadlines.

3.11 Emergency Response Coordination

- The system's emergency assistance feature requires coordination with university authorities and emergency services, potentially introducing constraints related to response times and procedures.

3.12 Complaint Handling Procedures

- Constraints may arise in handling and resolving complaints, necessitating efficient procedures for communication and resolution.

3.13 Optimized Route Algorithms

- The bus optimization feature requires robust algorithms that consider real-time traffic conditions, and constraints in road infrastructure. Integration with external mapping services may be necessary.

4. Requirement Specification

All the requirements based on the elicitation process are described in this section.

4.1 Functional Requirement

Functional requirements are those requirements that are used to illustrate the internal working nature of the system, the description of the system, and explanation of each subsystem. It consists of what task the system should perform, the processes involved, which data the system should hold and the interfaces with the user.

4.1.1 User Authentication

FR-1	User Authentication		
Description	Users (students and bus drivers) must be able to create accounts, log in, and authenticate their identity.		
Stakeholders	Students, Bus Drivers, University Administrators	**Priority**	High

4.1.2 Registration

FR-2	Registration		
Description	Users should be able to register their details, including student ID for students and driver-specific information for bus drivers.		
Stakeholders	Students, Bus Drivers, University Administrators	**Priority**	High

4.1.3 View Bus Schedule

FR-3	View Bus Schedule

Description	Students should have access to the bus schedule, displaying routes, departure times, and other relevant information.		
Stakeholders	Students, University Administrators	**Priority**	Medium

4.1.4 Set Drop-Off Point

FR-4	Set Drop-Off Point		
Description	Users should be able to set their desired drop-off points to help optimize bus routes.		
Stakeholders	Students	**Priority**	Medium

4.1.5 Bus Tracking

FR-5	Bus Tracking		
Description	Students should be able to track the real-time location of the university bus.		
Stakeholders	Students, Bus Drivers, University Administrators	**Priority**	High

4.1.6 Seat Reservation

FR-6	Seat Reservation

Description	Users should be able to reserve seats on the bus for a specific date and time.		
Stakeholders	Students	**Priority**	High

4.1.7 Receive Notifications

FR-7	Receive Notifications		
Description	Users should receive notifications about bus schedules, seat reservations, and other relevant information.		
Stakeholders	Students, Bus Drivers	**Priority**	High

4.1.8 Feedback and Ratings

FR-8	Feedback and Ratings		
Description	Students should be able to provide feedback and ratings after each bus journey.		
Stakeholders	Students, Bus Drivers	**Priority**	Medium

4.1.9 Optimized Route for Bus Drivers

FR-9	Optimized Route for Bus Drivers
Description	Bus drivers should have access to an optimized route based on seat reservations and drop-off points..

Stakeholders	Bus Drivers, University Administrators	**Priority**	High

4.1.10 Complaints and Emergency Assistance

FR-10	Complaints and Emergency Assistance		
Description	Users (students and drivers) should be able to submit complaints and request emergency assistance when needed.		
Stakeholders	Students, Bus Drivers, University Administrators	**Priority**	High

4.1.11 Authentication for Complaints

FR-11	Authentication for Complaints		
Description	The system should authenticate and verify the identity of users submitting complaints.		
Stakeholders	Students, Bus drivers, University Administrator	**Priority**	High

4.2 Data Requirement

Data requirements for TransPortEase system:

4.2.1 User Data

- Full Name
- Student ID (for Students)
- Driver ID (for Bus Drivers)
- Contact Information (Phone number, Email)
- Password (for authentication)

Stakeholders: Students, Bus Drivers, University Administrators
Priority: High

4.2.2 Bus Schedule Data

- Route Information (start and end points)
- Departure Times
- Bus Capacity
- Available Seats
- Days of Operation

Stakeholders: Students, University Administrators
Priority: Medium

4.2.3 Drop-Off Point Data

- Geographical Coordinates (Latitude and Longitude) for each designated drop-off point

Stakeholders: Students, University Administrators
Priority: Medium

4.2.4 Real-Time Bus Location Data

- GPS Coordinates of the bus
- Timestamps for location updates

Stakeholders: Judges, organizers, competitive programmers
Priority: High

4.2.5 Seat Reservation Data

- Student ID
- Date and Time of Reservation
- Bus Route
- Seat Number

Stakeholders: Students, Bus Drivers
Priority: High

4.2.6 Notification Data

- Type of Notification (e.g., schedule update, seat confirmation)
- Recipient (Student ID or Driver ID)
- Timestamp

Stakeholders: Students, Bus Drivers
Priority: High

4.2.7 Feedback and Ratings Data

- Student ID (or Anonymous if preferred)
- Date and Time of Feedback
- Bus Route
- Ratings (if applicable)
- Comments or Feedback

Stakeholders: Students, Bus Drivers
Priority: Medium

4.2.8 Optimized Route Data for Bus Drivers

- Bus Route
- List of Drop-Off Points
- Estimated Time of Arrival at each point
- Real-Time Traffic Data (if available)

Stakeholders: Bus Drivers, University Administrators
Priority: High

4.2.9 Complaints and Emergency Assistance Data

- Student ID or Driver ID
- Date and Time of Submission
- Nature of Complaint or Assistance Request
- Status of Resolution (open, in progress, resolved)

Stakeholders: Students, Bus Drivers, University Administrators
Priority:

4.2.10 Authentication Logs

- User ID
- Date and Time of Login
- Type of Action (Login, Logout)
- IP Address (for security purposes)

Stakeholders: University Administrator
Priority: High

4.2.11 System Logs

- Timestamp
- Type of Activity (e.g., seat reservation, location update)
- User ID or Bus ID involved
- Outcome or Status (success, failure)

Stakeholders: University Administrator
Priority: High

4.3 Performance Requirement

4.3.1 Response time

The system should respond to user actions (e.g., login, seat reservation) within 2 seconds.

4.3.2 Real-Time Bus Tracking

The bus location should be updated on the user interface every 30 seconds.

4.3.3 Seat Reservation Processing

Seat reservation requests should be processed instantly, and confirmation should be sent to the user within 5 seconds.

4.3.4 Notification Delivery

Notifications (e.g., schedule updates, seat confirmations) should be delivered to users within 5 seconds of the event.

4.3.5 Optimized Route Calculation

The system should calculate and provide optimized routes for bus drivers within 10 seconds, considering current reservations and traffic data.

4.3.6 Data Synchronization

All data, including bus schedules, seat reservations, and real-time locations, should be synchronized across devices in near real-time (within 1 minute).

4.3.7 Complaint Resolution Time

The system should aim to resolve complaints and emergency assistance requests within 24 hours of submission.

4.3.8 System Availability

The system should have at least 99% availability during peak hours (e.g., when buses are in operation).

4.3.9 Scalability

The system should be able to handle a 20% increase in user activity (e.g., seat reservations, logins) without a significant degradation in performance.

4.3.10 Security Response Time

Security breaches or unauthorized access attempts should trigger an alert to administrators within 5 seconds.

4.4 Maintainability and Supportability

4.4.1 Maintenance Requirements

MR-1	Code should be maintainable		
Description	The code for TransPortEase should be well-organized, readable, and easy to modify and extend, so that developers can make changes and fix bugs as needed.		
Stakeholders	Developers	**Priority**	High

MR-2	Full Specification should be documented		
Description	TransPortEase should include comprehensive documentation that describes its architecture, design, and code, so that developers can understand and modify the system as needed.		
Stakeholders	Requirement Engineer	**Priority**	High

4.4.2 Supportability Requirements

4.4.2.1 Helpdesk and User Support

Establish a dedicated helpdesk or support team accessible through various channels (e.g., email, chat, or a support portal) to assist users with inquiries, issues, and feedback.

4.4.2.2 Training Resources

Provide comprehensive training resources, including user manuals, video tutorials, and FAQs, to empower both administrators and end-users in utilizing the system effectively.

4.4.2.3 Maintenance and Update Notifications

Implement a notification system to inform users and administrators about upcoming maintenance schedules, updates, and new features.

4.4.2.4 Remote Assistance

Enable remote assistance tools to facilitate troubleshooting and support, allowing support personnel to assist users directly when needed.

4.4.2.5 Knowledge Base

Develop and maintain a knowledge base with solutions to common issues, troubleshooting guides, and best practices for both administrators and end-users.

4.4.2.6 Bug Reporting System

Implement a user-friendly bug reporting system, allowing users to submit detailed bug reports with supporting information to expedite the resolution process.

4.4.2.7 Continuous Monitoring

Establish continuous monitoring systems to proactively identify and address potential issues, minimizing downtime and disruption.

4.4.2.8 Vendor Support Agreement

If third-party components or services are utilized, maintain active support agreements with vendors to ensure timely assistance in case of issues or updates.

4.4.2.9 Data Recovery Plan

Develop and regularly test a data recovery plan to ensure the swift recovery of system data in the event of unexpected data loss or corruption.

4.4.2.10 User Feedback Integration

Integrate mechanisms for collecting and analyzing user feedback, allowing for continuous improvement of the system based on user experiences and suggestions.

4.4.2.11 Regular Maintenance Procedures

Establish and document regular maintenance procedures, ensuring that routine tasks such as database optimizations, server updates, and security checks are performed systematically.

These supportability requirements aim to create a robust support infrastructure, enhance user experience, and ensure efficient resolution of issues throughout the lifecycle of your university bus system. Regularly reviewing and updating these support processes will contribute to the system's long-term success.

4.5 Security Requirements

4.5.1 Authentication and Authorization

TransPortEase should implement secure authentication and authorization mechanisms to ensure that only authorized users can access the system and its features.

4.5.2 Data Encryption

All sensitive data, such as passwords and personal information, should be encrypted in transit and at rest to prevent unauthorized access or theft.

4.5.3 Access Control

TransPortEase should have fine-grained access control mechanisms to ensure that users can only access the data and features they are authorized to access.

4.5.4 Input Validation

TransPortEase should validate all input data, especially user input, to prevent malicious attacks, such as cross-site scripting and SQL injection.

4.5.5 Security Logs

TransPortEase should maintain logs of all security-related events and activities, such as login attempts and failed access attempts, to help detect and prevent security threats.

4.5.6 Vulnerability Scanning

TransPortEase should be regularly scanned for vulnerabilities to identify and remediate security risks.

4.5.7 Firewall

TransPortEase should be protected by a firewall to prevent unauthorized access from the internet.

4.5.8 Regular Security Updates

TransPortEase should be regularly updated to address known security vulnerabilities and protect against new threats.

4.5.9 Disaster Recovery

TransPortEase should have a disaster recovery plan in place to ensure that the system and data can be recovered in the event of a disaster.

5. Requirement Engineering Process

Requirements Engineering (RE) determines software requirements according to customer requirements or needs. Requirements engineering process includes requirements elicitation, needs modeling, requirements analysis, requirements assurance & validation, and requirements management.

5.1 Requirement Elicitation Techniques

Requirements elicitation is the practice of researching and finding system requirements for users, customers, and other stakeholders, also referred to as "requirement gathering". Requirement elicitation can be done by contacting participants directly or by doing some research, analysis and testing.

5.1.1 Hold Interviews

We hold discussions that can be held individually or with a small group of participants. They are an effective way to access services without spending a lot of time with participants because we meet with people to discuss only certain important requirements of this program. We mainly perform our interview based on some specific criteria.

- Short description about goals and objectives
- Process of using the TransPortEase app
- Available Features
- How it will reduce the seat reservation problem
- Review System

5.1.2 Designing Surveys or Questionnaire

The questionnaire is an effective tool for gathering information on styles, changes in attitudes and preferences, and user satisfaction. To minimize fatigue or frustration for the respondent, our questions were kept concise and grouped together based on topics. This allowed the respondent to focus on specific areas and provided a clear rationale for each question. The main advantage of using this survey approach was the ability to collect responses in a standard manner, allowing for the consolidation of information from a large number of people.

5.1.3 Perform Document Analysis

Existing documentation can assist in demonstrating how systems are being used or what I should do with them. Documents contain textual details regarding existing programs, operational procedures, required specifications, and market research on competitors. Once again, textual analysis can be useful. Determine which features should be removed and which performances should stay by consulting the Software Requirements Specification. The previous document in our investigation, we discovered a number of issues with the current system.

- When lot of students will request for seat reservations, system can fail
- The limitation of bus numbers

5.1.4 Prototyping

We have developed a prototype of the system to demonstrate its functionality and collect feedback from stakeholders.

5.2 Sample of Requirement Collection

The process of requirement collection serves as a foundational step in the development of any project, providing crucial insights into the needs, preferences, and expectations of key stakeholders. This document presents an introduction to a sample requirement collection process conducted for the TransPortEase project, focusing on gathering insights from both students and bus drivers of Noakhali Science and Technology University (NSTU).

Requirement collection involves systematically gathering and analyzing information from stakeholders to identify essential features and functionalities for the proposed TransPortEase application. Through a combination of structured interviews, surveys, and observations, the goal is to gain a comprehensive understanding of the user requirements and system constraints.

5.2.1 Requirement Collection -1

The requirement collection phase of the TransPortEase project aimed to gather insights and preferences from a diverse pool of stakeholders, primarily students of Noakhali Science and Technology University (NSTU).

Interview

Methodology:

The requirement collection process utilized a structured interview approach, wherein participants were asked a series of predefined questions related to their preferences, challenges, and expectations concerning the NSTU bus transportation system and the proposed TransPortEase application. Interviews were conducted in-person and digitally, allowing for flexibility and inclusivity in participation. Responses were recorded, transcribed, and analyzed for common themes and patterns.

Participants:

A total of 50 NSTU students from various disciplines and academic years participated in the requirement collection process. Participants were selected randomly to ensure representation across different demographics, including gender, age, and academic background. Their diverse perspectives and experiences provided valuable insights into the needs and preferences of the broader student community.

Findings:

Key findings from the requirement collection interviews include:

- High demand for real-time bus tracking and schedule information.
- Preference for a user-friendly interface with easy seat reservation options.
- Desire for timely notifications regarding bus schedule changes and seat availability.
- Interest in providing feedback and ratings to improve the overall transportation experience.
- Concerns about security and privacy in accessing and using the application.

Key Requirements:

Based on the findings, the following key requirements were identified for the TransPortEase application:

- Real-time bus tracking and schedule display.
- Seamless seat reservation system with user-friendly interface.
- Notification system for updates and alerts.
- Feedback and rating mechanism for user engagement.
- Robust security measures to protect user data.

Assumptions:

The requirement collection process operated under the following assumptions:

- Participants provided honest and representative feedback.
- Preferences and needs expressed by participants are reflective of broader student sentiments.
- Technical feasibility and resource availability to implement identified requirements.

Limitations:

Limitations encountered during the requirement collection phase include:

- Limited sample size due to resource constraints.
- Potential biases in participant selection and responses.
- Difficulty in capturing nuances and complexities of user preferences in a structured interview format.

Conclusion:

The requirement collection process served as a foundational step in understanding the needs and expectations of NSTU students regarding the TransPortEase application. Key insights gleaned from participant interviews have informed the development of essential features and functionalities to enhance the NSTU bus transportation system. Moving forward, continued collaboration with stakeholders and iterative refinement of requirements will be essential to ensure the success and relevance of the TransPortEase project.

5.2.2 Requirement Collection -2

This document outlines the requirements gathered from interviews conducted with 15 drivers of the Noakhali Science and Technology University (NSTU) bus transportation system. The objective was to understand the perspectives, challenges, and preferences of bus drivers regarding the proposed TransPortEase application.

Methodology:

Interviews with NSTU bus drivers were conducted in-person and digitally, utilizing a structured questionnaire to gather insights into their experiences, needs, and expectations. Responses were recorded, transcribed, and analyzed to identify common themes and requirements.

Participants:

A total of 15 NSTU bus drivers participated in the requirement collection process. Drivers were selected based on their experience and involvement in the university's transportation system, ensuring representation across different routes and shifts.

Findings:

Key findings from the driver interviews include:

- Desire for an optimized route planning feature to improve efficiency and reduce travel time.
- Need for a streamlined complaint management system to address passenger issues promptly.
- Interest in access to real-time traffic updates and navigation assistance.
- Concerns about user authentication and security measures within the application

.

Key Requirements:

Based on the findings, the following key requirements were identified for the TransPortEase application:

- Route optimization tool for efficient navigation and reduced fuel consumption.
- User-friendly interface for managing and resolving passenger complaints.
- Integration of real-time traffic data and navigation features for route planning.
- Secure authentication mechanisms to ensure driver access and data integrity.

Assumptions:

The requirement collection process operated under the assumption that drivers provided honest and representative feedback based on their experiences and needs within the NSTU bus transportation system.

Limitations:

Limitations encountered during the requirement collection phase include:

- Potential biases in participant selection and responses.
- Difficulty in capturing the full spectrum of driver perspectives and requirements within the scope of the interviews.

Conclusion:

The requirements gathered from NSTU bus drivers serve as valuable insights into the features and functionalities needed to enhance the TransPortEase application's effectiveness and usability. Moving forward, collaboration with drivers and ongoing refinement of requirements will be essential to ensure the successful implementation and adoption of the application within the NSTU transportation system.

5.3 Requirement Validation

Requirement validation ensures that the requirements are correct and reflect the quality you want from this program. In the beginning, our requirements looked good but when we read them and tried to work with them they came out having ambiguities and gaps.

5.3.1 Walkthrough

A walkthrough is a review of the requirement documents conducted by a team of stakeholders, developers, and experts. They review the requirements and provide feedback on their completeness, correctness, and consistency.

5.3.2 Prototype Testing

Creating a working prototype of the application and testing it with users to validate the requirements and identify any issues or areas for improvement. It is a working model of the system. It can be used to validate the requirements by allowing stakeholders and users to interact with the system and provide feedback on its functionality and usability.

5.3.3 User Acceptance Testing (UAT)

UAT is a testing process where end-users test the system to verify if it meets their requirements. This process can help validate the requirements and ensure that the system meets the needs of the users.

5.3.4 Requirements Traceability

Using the traceability information to verify that all requirements have been implemented in the application

6 . Use case Diagram

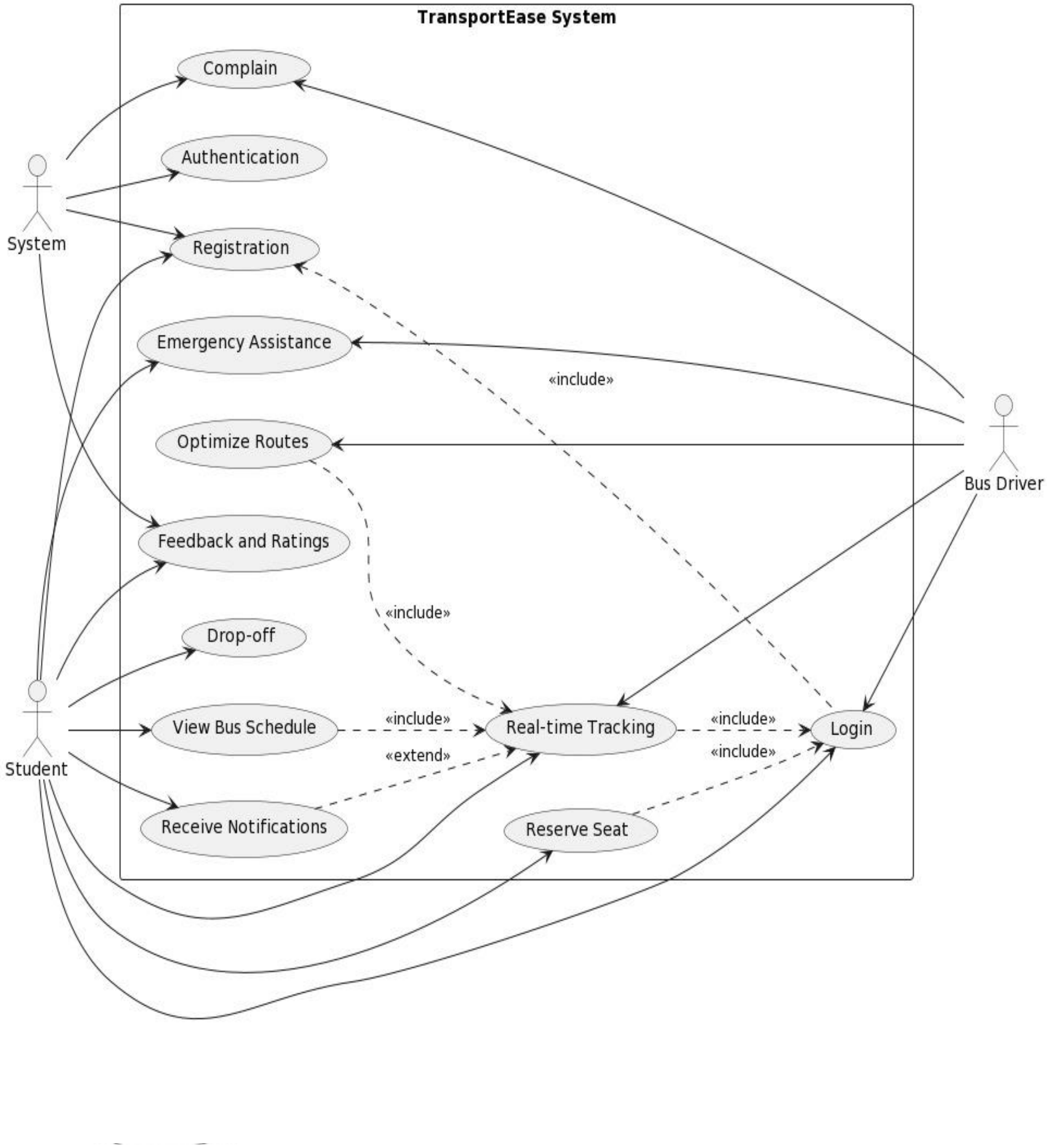

Fig 1 : Use Case Diagram

7 . Use Case Description

Table 1 : Registration

<table>
<tr><td><b>Use Case No.</b></td><td colspan="2">01</td></tr>
<tr><td><b>Use Case Name</b></td><td colspan="2">Registration</td></tr>
<tr><td><b>Goal</b>
<a longer statement of the goal in context if needed&gt;</td><td colspan="2">Register a user into the system.</td></tr>
<tr><td><b>Preconditions</b>
<what we expect is already the state of the world&gt;</td><td colspan="2">Must be a student.</td></tr>
<tr><td><b>Success End Condition</b>
<the state of the world upon successful completion&gt;</td><td colspan="2">Registered into the system successfully.</td></tr>
<tr><td><b>Failed End Condition</b>
<the state of the world if goal abandoned&gt;</td><td colspan="2">Registration failed.</td></tr>
<tr><td><b>Primary Actors:</b>
<b>Secondary Actors:</b></td><td colspan="2">Student
System</td></tr>
<tr><td><b>Trigger</b>
<the action upon the system that starts the case&gt;</td><td colspan="2">The user clicks the “Registration” button.</td></tr>
<tr><td rowspan="8"><b>Main Success Flows</b>
<the steps of the scenario from trigger to goal delivery and any clean up after&gt;</td><td><b>Steps</b></td><td><b>Action</b></td></tr>
<tr><td>1</td><td>The user accesses the registration page.</td></tr>
<tr><td>2</td><td>The system presents the registration form to the user.</td></tr>
<tr><td>3</td><td>The user fills in the required details such as name, email, password, etc.</td></tr>
<tr><td>4</td><td>The user submits the registration form.</td></tr>
<tr><td>5</td><td>The system verifies the submitted information.</td></tr>
<tr><td>6</td><td>If the information is valid, the system creates a new account for the user.</td></tr>
<tr><td>7</td><td>The system notifies the user of successful registration.</td></tr>
</table>

Alternative Flows <a: condition causing branching> <a1: action or name of sub use case>	**Steps**	**Action**
	6a	If the user submits incomplete or invalid information, the system prompts the passenger to correct the errors.
	6a1	The user corrects the information and resubmits the form.
	6a2	The system re-verifies the submitted information.
	6a3	If the information is now valid, the system proceeds with creating the account as described in the main flow.
Quality Requirement	1	The system must employ encryption techniques to securely store registration information, including passwords, to prevent unauthorized access.
	2	The registration process should be responsive, with minimal latency, to provide a smooth user experience even during peak usage periods.
	3	The system should be able to handle concurrent registration requests efficiently without degradation in performance.
	4	The system should be able to accommodate a growing number of registrations over time without significant impact on performance or reliability.

Table 1: Authentication

Use Case No.	02
Use Case Name	Authentication
Goal <a longer statement of the goal in context if needed>	Authenticating users attempting to access the system.
Preconditions **<**what we expect is already the state of the world>	Must be registered to the system.
Success End Condition <the state of the world upon successful completion>	Successfully log in into the system.

Failed End Condition <the state of the world if goal abandoned>	Log in failed.	
Primary Actors: **Secondary Actors:**	Student System	
Trigger <the action upon the system that starts the case>	The user clicks the “Log in” button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Steps**	**Action**
	1	The user navigates to the login page.
	2	The system presents the login form.
	3	The user enters their registered email and password.
	4	The user submits the registration form.
	5	The system verifies the entered credentials.
	6	The system grants access to the user-specific features of the system.
	7	The user is successfully logged into the system and directed to their respective dashboard or landing page.
Alternative Flows <a: condition causing branching> <a1: action or name of sub use case>	**Steps**	**Action**
	5a	If the user enters incorrect credentials, the system prompts them to re-enter the correct email and password combination.
	5a1	The user corrects the credentials and resubmits the form.
	5a2	The system re-verifies the entered credentials.
	5a3	If the credentials are now valid, the system proceeds with granting access as described in the main flow.
Quality Requirement	1	The system must encrypt sensitive information (e.g., passwords) during transmission and storage to prevent unauthorized access.
	2	The registration form should have clear labels and instructions to guide users through the process.
	3	The registration process should respond promptly to user input, with minimal latency.

Table 2: Complain

Use Case No.	03	
Use Case Name	Complain	
Goal <a longer statement of the goal in context if needed>	Submit a complain to the system.	
Preconditions <what we expect is already the state of the world>	The user must be logged into the system as either an administrator or a bus driver.	
Success End Condition <the state of the world upon successful completion>	Successfully submit complain to the system.	
Failed End Condition <the state of the world if goal abandoned>	Submission complain failed.	
Primary Actors: **Secondary Actors:**	Student, Bus Driver System	
Trigger <the action upon the system that starts the case>	The user encounters an issue during their interaction with the transportation service that warrants a complaint.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Steps**	**Action**
	1	The user or a bus driver, accesses the complaint submission functionality within the system.
	2	The system presents a form or interface for the user to input details regarding the complaint.
	3	The user fills out the necessary information, which may include: Description of the incident or issue, date or time of the issue.
	4	Once the user completes the form, they submit the complaint to the system.
	5	The system records the complaint and stores it in the database for further processing.
Alternative Flows	**Steps**	**Action**

<a: condition causing branching> <a1: action or name of sub use case>	3a	If the user encounters technical difficulties while submitting the complaint, they may seek assistance from system support or try submitting the complaint again.
	5a	If the complaint form is missing essential information, the system prompts the user to provide the required details before allowing submission.
Quality Requirement		N/A

Table 3: Emergency Assistance

Use Case No.	04
Use Case Name	Emergency Assistance
Goal <a longer statement of the goal in context if needed>	Calls for assistance if emergency situation arrives.
Preconditions <what we expect is already the state of the world>	The user must be logged into the system as a passenger, bus driver, or administrator. There must be an emergency situation that requires immediate assistance.
Success End Condition <the state of the world upon successful completion>	Emergency assistance sent to the administrator.
Failed End Condition <the state of the world if goal abandoned>	Failed to sent emergency assistance.
Primary Actors: **Secondary Actors:**	Student, Bus Driver System
Trigger <the action upon the system that starts the case>	The user encounters an emergency situation while using the transportation service.

Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Steps**	**Action**
	1	The user identifies an emergency situation and decides to request assistance.
	2	The user accesses the emergency assistance functionality within the system.
	3	The system provides options for the user to specify the type of emergency assistance needed (e.g., medical, security, accident).
	4	The user selects the appropriate type of assistance and provides any additional details or context about the emergency.
	5	Once the information is entered, the user submits the request for emergency assistance.
	6	The system immediately notifies relevant parties, such as emergency response teams, designated personnel, or nearby authorities, about the situation.
	7	The system tracks the status of the emergency assistance request and provides updates to the user as necessary.
Alternative Flows <a: condition causing branching> <a1: action or name of sub use case>	**Steps**	**Action**
	1a	If the user is unable to access the emergency assistance functionality due to technical issues or system errors, they may need to seek help through alternative means, such as contacting emergency services directly.
	2a	If the user is unable to provide detailed information about the emergency, the system may prompt them to provide as much relevant information as possible before submitting the request.
Quality Requirement		N/A

Table 4: Optimize Routes

Use Case No.	05
Use Case Name	Optimize Routes
Goal <a longer statement of the goal in context if needed>	It will enable bus drivers to optimize their routes for efficiency and timely arrivals.
Preconditions <what we expect is already the state of the world>	The system must have access to current traffic and road conditions

Success End Condition <the state of the world upon successful completion>	"Optimize Routes" sent successfully.
Failed End Condition <the state of the world if goal abandoned>	"Optimize Routes" sending process failed.
Primary Actors: **Secondary Actors:**	User (Bus Driver) System

Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The bus driver selects the "Optimize Routes" option from the driver interface
	2	The system retrieves current traffic and road condition data
	3	The system analyzes the data and suggests optimized routes based on traffic congestion and other factors
	4	The bus driver reviews the suggested optimized routes
	5	The bus driver selects the preferred optimized route
	6	The system updates the navigation system with the selected optimized route.
	7	The bus driver follows the optimized route to reach destinations efficiently

Alternative Flows <a: condition causing branching> <a1: action or name of sub use case>	**Step**	**Action**
	1	The bus driver selects the "Optimize Routes" option from the driver interface.
	2	The system retrieves current traffic and road condition data
	3a	The system encounters technical issues during the route optimization process
	3a1	The system notifies the bus driver about the failure to provide optimized routes due to technical difficulties.

	3a2	The bus driver acknowledges the notification and proceeds with the standard route or relies on their experience to navigate efficiently.
Quality Requirement	1	The system should optimize routes in a timely manner, considering factors such as traffic conditions, road closures, and passenger demand.
	2	The optimized routes should accurately reflect the most efficient and practical paths for buses to follow based on current conditions and constraints
		It should consider various factors such as distance, travel time, and passenger preferences to generate optimal routes.

Table 5: Feedback & Ratings

Use Case No.	06	
Use Case Name	Feedback & Ratings	
Goal <a longer statement of the goal in context if needed>	It will allow allows passengers to provide feedback and ratings for their journey experience	
Preconditions <what we expect is already the state of the world>	Passengers must have completed a journey.	
Success End Condition <the state of the world upon successful completion>	“Feedback & Ratings “ submitted successfully	
Failed End Condition <the state of the world if goal abandoned>	“Feedback & Ratings “ submission process failed	
Primary Actors: **Secondary Actors:**	(Student) System	
Trigger <the action upon the system that starts the case>	The user clicks the “Feedback & Ratings” button.	
Main Success Flows	**Step**	**Action**

<the steps of the scenario from trigger to goal delivery and any clean up after>	1	The passenger selects the "Feedback & Ratings" option from the user interface after completing the journey.
	2	The system presents a form or interface where the passenger can provide feedback and ratings for various aspects of the journey, such as cleanliness, punctuality, and staff behavior.
	3	Once the feedback is submitted, the system records the feedback and ratings associated with the specific journey.
	4	Optionally, the system may provide a confirmation message to the passenger indicating that their feedback has been successfully submitted.
Alternative Flows <a: condition causing branching> <a1: action or name of sub use case>	**Step**	**Action**
	2a	If the passenger does not have enough time to provide detailed feedback, they may choose to provide only ratings without comments
	2a1	The passenger may decide to cancel the feedback submission process at any point before submitting it. In this case, the system discards the partially filled feedback form and returns the passenger to the main interface.
	2a2	If there is an error during the submission process (e.g., network error, database error), the system notifies the passenger about the issue and may offer the option to retry the submission later
Quality Requirement		N/A

Table 6: Drop Off

Use Case No.	07	
Use Case Name	Drop Off	
Goal <a longer statement of the goal in context if needed>	It will enable passengers in selecting their preferred drop-off points	
Preconditions <what we expect is already the state of the world>	The passenger is logged into the system	
Success End Condition <the state of the world upon successful completion>	The system updates the passenger's journey itinerary to reflect the selected drop-off point	
Failed End Condition <the state of the world if goal abandoned>	The system failed to updates the passenger's journey itinerary to reflect the selected drop-off point	
Primary Actors: **Secondary Actors:**	Student System	
Trigger <the action upon the system that starts the case>	The user clicks the "Drop Off" button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The passenger indicates their intention to disembark by interacting with the user interface.
	2	The system presents the available drop-off points along the bus route.
	3	The passenger selects their desired drop-off point from the provided options.
	4	The system confirms the selected drop-off point and updates the journey detail
	5	The passenger completes the drop-off selection process and continues their journey.
Alternative Flows <a: condition causing branching> <a1: action or name of sub use case>	**Step**	**Action**
	3a	If the passenger cancels the drop-off selection, the system returns to the previous state, maintaining the current journey details

	3a1	If there are any errors in the drop-off selection process), appropriate error messages are displayed, and the passenger is prompted to correct their selection
Quality Requirement	1	The system should process drop-off requests promptly, minimizing waiting times for passengers.
	2	It should efficiently handle multiple drop-off requests simultaneously without significant performance degradation.
	3	The interface should be responsive, allowing passengers to select their desired drop-off points quickly and easily.

Table 8: View Bus Schedule

Use Case No.	08	
Use Case Name	View Bus Schedule	
Goal <a longer statement of the goal in context if needed>	It enables passengers to view the schedule of buses within the system.	
Preconditions <what we expect is already the state of the world>	The passenger must be logged into the system.	
Success End Condition <the state of the world upon successful completion>	The passenger has successfully viewed the bus schedule and can plan their travel accordingly.	
Failed End Condition <the state of the world if goal abandoned>	System will show error message .	
Primary Actors: **Secondary Actors:**	Student System	
Trigger <the action upon the system that starts the case>	The user clicks the “View Bus Schedule” button.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	**Step**	**Action**
	1	The passenger navigates to the "View Bus Schedule" option in the system interface.
	2	The system presents a list of available bus routes and their respective schedules.
	3	The passenger selects a specific bus route or searches for it .

	4	The system displays the schedule for the selected bus route, showing departure times and stops along the route.
	5	The passenger can view additional details such as bus capacity, estimated travel time, and frequency of service.
	6	The passenger can navigate back to the main menu or perform other actions within the system.
Alternative Flows <a: condition causing branching> <a1: action or name of sub use case>	**Step**	**Action**
	2a	The system informs the passenger that the schedule for the selected route is currently unavailable.
	2a1	The system notifies the passenger about the invalid selection.
Quality Requirement	1	The system must provide up-to-date and accurate bus schedule information to prevent misinformation for passengers.

Table 9: Receive Notifications

Use Case No.	09	
Use Case Name	Receive Notifications	
Goal <a longer statement of the goal in context if needed>	The goal of this use case is to provide timely and relevant notifications to registered students, enhancing their travel experience by keeping them informed about important events related to the bus service.	
Preconditions	Step	Action

<what we expect is already the state of the world>	1	The passenger is registered and logged into the TransPortEase System.
	2	Notification preferences for the passenger are enabled.
Success End Condition <the state of the world upon successful completion>	The registered student successfully receives timely notifications about important events affecting their bus journey.	
Failed End Condition <the state of the world if goal abandoned>	Technical issues prevent the delivery of notifications and the student has disabled notification preferences.	
Primary Actors:	Student	
Secondary Actors:	System	
Trigger <the action upon the system that starts the case>	The system (TransPortEase) detects relevant events, such as bus delays, route changes, or other updates affecting the student's journey.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	Step	Action
	1	The system monitors and detects relevant events.
	2	Notifications are generated based on the detected events.
	3	Notifications are sent to the registered passenger's preferred communication channels (e.g., mobile app notifications, SMS, or email).
Alternative Flows <a: condition causing branching> <a1: action or name of sub use case>	If the passenger has disabled notification preferences, no notifications will be sent.	

Quality Requirements	Step	Action
	1	Notifications must be delivered promptly to ensure passengers receive real-time updates.
	2	The system should have mechanisms to handle technical issues and ensure the reliable delivery of notifications.
	3	The passenger should be able to customize their notification preferences, specifying the type of information and communication channels.

Table 10: Real-time tracking

Use Case No.	10	
Use Case Name	Real-time tracking	
Goal <a longer statement of the goal in context if needed>	The goal of the "Real-time Tracking" use case is to provide students and drivers with up-to-date information about the current location and status of buses within the TransPortEase System.	
Preconditions <what we expect is already the state of the world>	Step	Action
	1	The student is registered and logged into the TransPortEase System.
	2	The bus driver is logged into the TransPortEase System.
	3	The bus is equipped with GPS tracking devices.
Success End Condition <the state of the world upon successful completion>	Passengers and drivers have accurate and real-time information about the current location and status of buses.	

Failed End Condition <the state of the world if goal abandoned>	Technical issues prevent the real-time tracking system from providing accurate information.	
Primary Actors: **Secondary Actors:**	Students Bus driver	
Trigger <the action upon the system that starts the case>	The system continuously receives and updates bus location data from GPS tracking devices.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	Step	Action
	1	The system retrieves real-time location data from GPS tracking devices installed on buses.
	2	Passengers access the TransPortEase System to view the real-time location of buses on a map.
	3	Bus drivers access the TransPortEase System to view real-time route information and track their progress.
	4	The system updates the real-time tracking information at regular intervals.
Alternative Flows <a: condition causing branching> <a1: action or name of sub use case>	If there are technical issues with GPS devices, the system may display the last known location of the bus.	
Quality Requirements	**Step**	Action
	1	The real-time tracking system must provide accurate and precise bus location information.
	2	The system should update the bus location information at regular intervals to ensure real-time tracking.

	3	The system should be robust and capable of handling intermittent connectivity issues to maintain continuous tracking.

Table 11: Reserve Seat

Use Case No.	11	
Use Case Name	Reserve Seat	
Goal <a longer statement of the goal in context if needed>	The goal of the "Reserve Seat" use case is to allow students to secure a seat on a bus for a specific journey within the TransPortEase System. This feature enhances the overall student experience by providing a convenient and efficient way for students to guarantee their seat and plan their journey in advance.	
Preconditions <what we expect is already the state of the world>	Step	Action
	1	The passenger is registered and logged into the TransPortEase System.
	2	The passenger has selected a specific bus journey and route.
Success End Condition <the state of the world upon successful completion>	The student successfully reserves a seat on the selected bus journey.	
Failed End Condition <the state of the world if goal abandoned>	The reservation process fails due to technical issues or insufficient available seats.	
Primary Actors: **Secondary Actors:**	Students None	
Trigger <the action upon the system that starts the case>	The students initiates the seat reservation process by selecting a specific bus journey.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	step	Action
	1	The system displays the available seats for the selected bus journey.

	2	The passenger chooses a preferred seat from the available options.
	3	The system validates the seat selection and reserves the chosen seat for the passenger.
	4	The system confirms the successful seat reservation to the passenger.
Alternative Flows <a: condition causing branching> <a1: action or name of sub use case>	If there are technical issues during the reservation process, the system informs the passenger about the failure and prompts them to try again.	
Quality Requirements	Step	Action
	1	The seat reservation system should be reliable, ensuring successful seat reservations and minimizing technical failures.
	2	The user interface for seat reservation should be intuitive and easy to use, enhancing the overall user experience.
	3	The system should update seat availability in real-time to provide accurate information to passengers.

Table 12: Log In

Use Case No.	12
Use Case Name	Log In
Goal <a longer statement of the goal in context if needed>	The goal of the "Login" use case is to authenticate and authorize users, providing them access to their individual accounts within the TransPortEase System. This feature ensures the security of user accounts and allows students, bus drivers, and administrators to access personalized information and functionalities.

Preconditions <what we expect is already the state of the world>	The user is registered in the TransPortEase System and The user has a valid username and password.	
Success End Condition <the state of the world upon successful completion>	The user successfully logs into the TransPortEase System and gains access to their account.	
Failed End Condition <the state of the world if goal abandoned>	The login attempt fails due to incorrect credentials, account suspension, or technical issues.	
Primary Actors: **Secondary Actors:**	Students, Bus Driver None	
Trigger <the action upon the system that starts the case>	The user initiates the login process by entering their username and password.	
Main Success Flows <the steps of the scenario from trigger to goal delivery and any clean up after>	Step	Action
	1	The system prompts the user to enter their username and password.
	2	The user provides their valid credentials.
	3	The system validates the credentials against the stored user data.
	4	If the credentials are valid, the system grants access to the user's account.
Alternative Flows <a: condition causing branching> <a1: action or name of sub use case>	Step	Action
	1	If the user enters incorrect credentials, the system notifies the user of the authentication failure and prompts them to re-enter the information.
	2	If the user's account is suspended, the system notifies the user and provides instructions on the suspension status.
Quality Requirements	Step	Action
	1	The login process should ensure the security of user accounts by securely handling and storing credentials.

	2	The login interface should be user-friendly, making it easy for users to enter their credentials and navigate the system.

8 . Activity Diagram

Activity Diagram (View Bus Schedule)

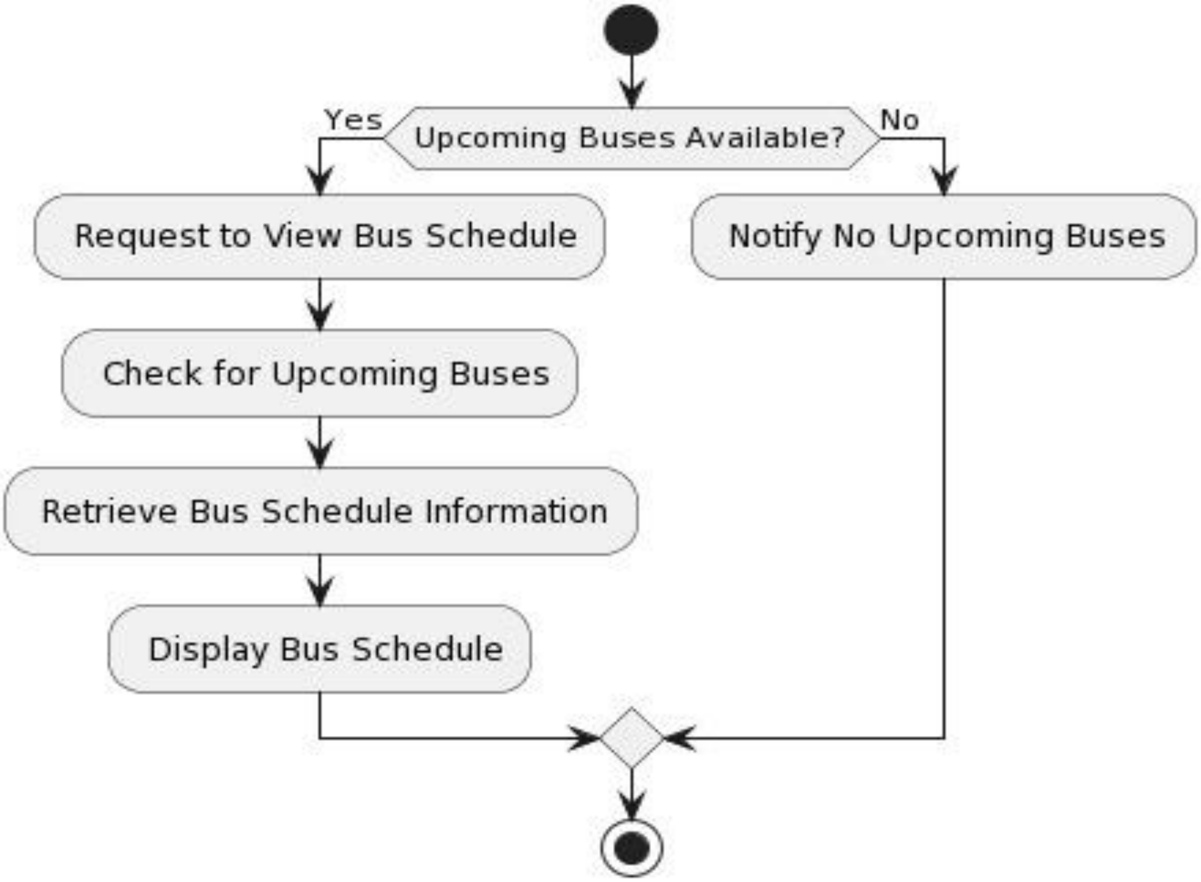
Yes
Upcoming Buses Available?
No
Request to View Bus Schedule
Notify No Upcoming Buses
Check for Upcoming Buses
Retrieve Bus Schedule Information
Display Bus Schedule

Fig 2 :View Bus_Schedule

Activity Diagram (Log in)

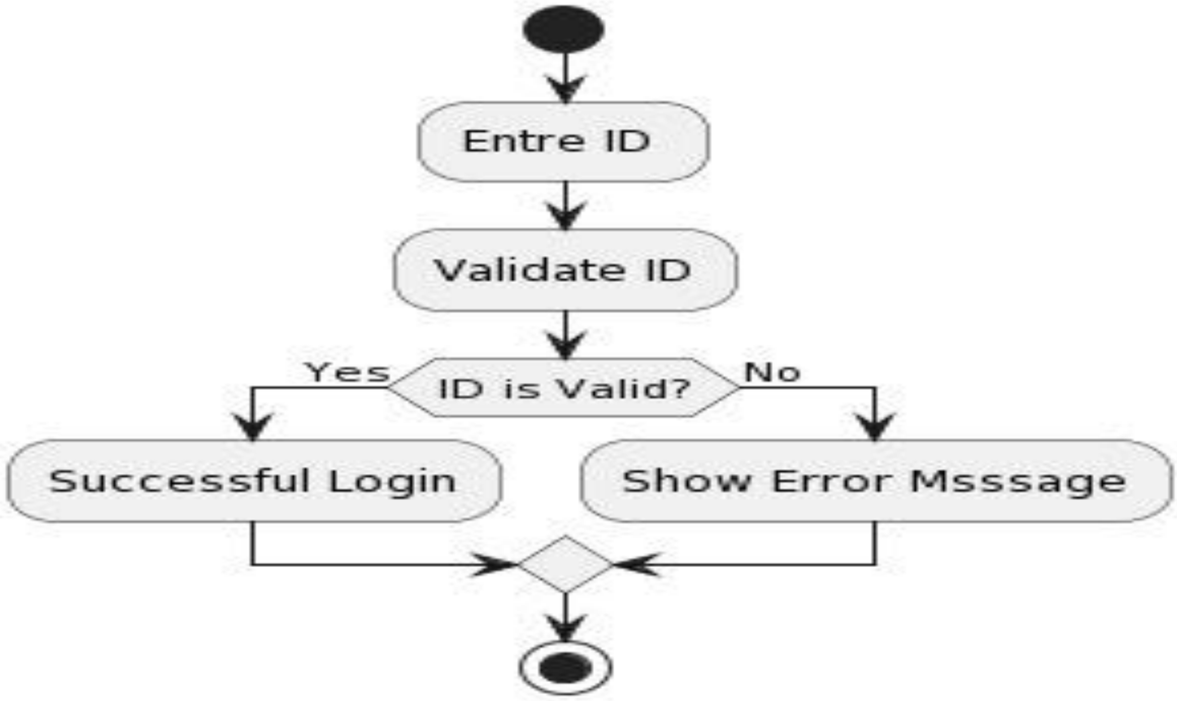

Fig 3: Log in

Activity Diagram (Optimizing Routes)

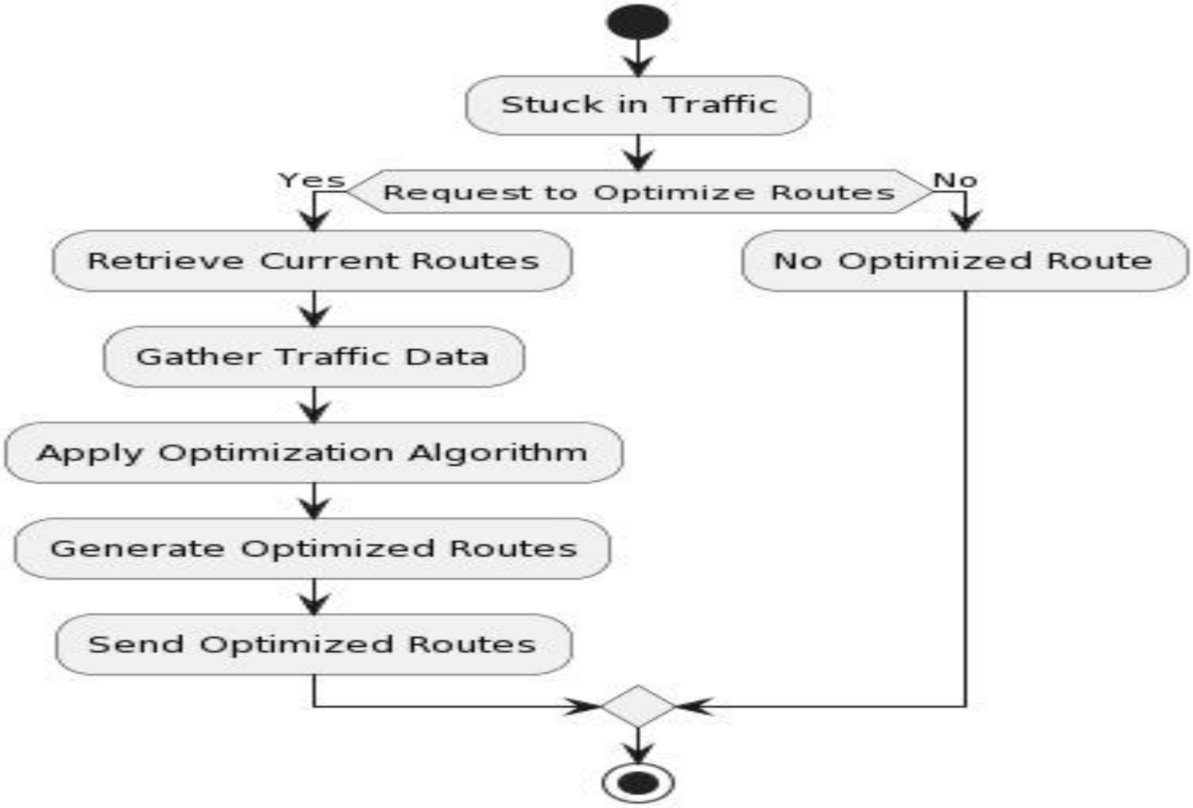

Fig 4: Optimizing Routes

Activity Diagram (Emergency Assistance)

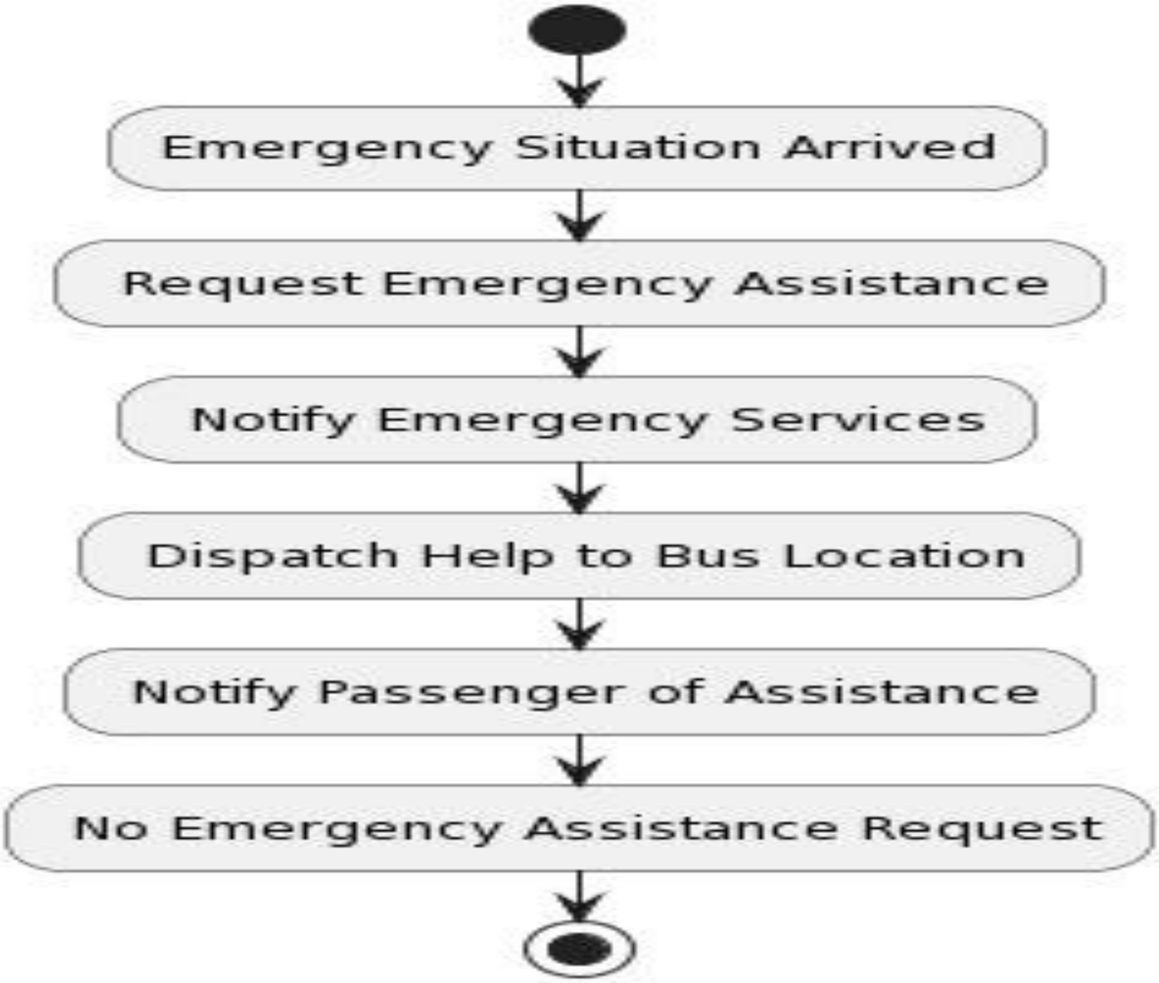
Emergency Situation Arrived
Request Emergency Assistance
Notify Emergency Services
Dispatch Help to Bus Location
Notify Passenger of Assistance
No Emergency Assistance Request

Fig 5 : Emergency Assistance

Activity Diagram (Received Notification)

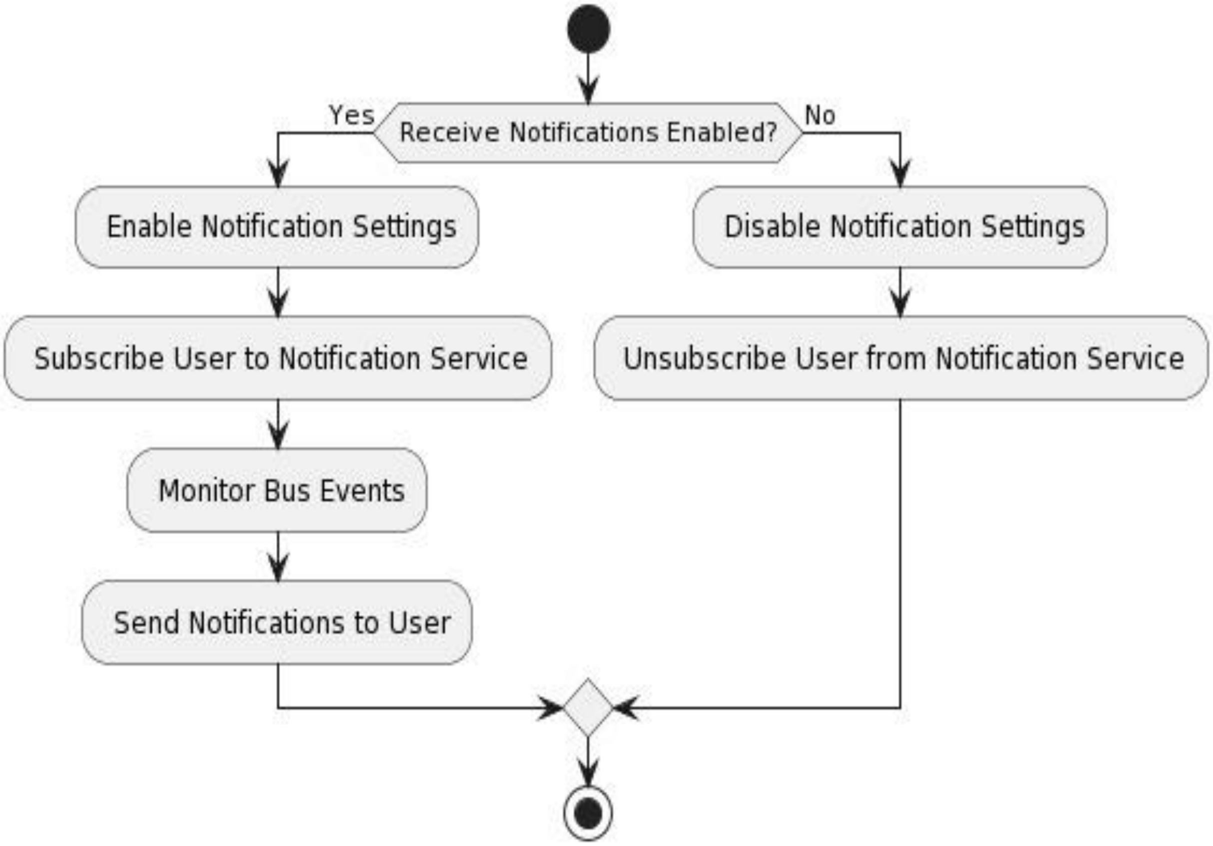
Receive Notifications Enabled?
Yes
No
Enable Notification Settings
Disable Notification Settings
Subscribe User to Notification Service
Unsubscribe User from Notification Service
Monitor Bus Events
Send Notifications to User

Fig 6 : Received Notification

Activity Diagram (Feedback & Ratings)

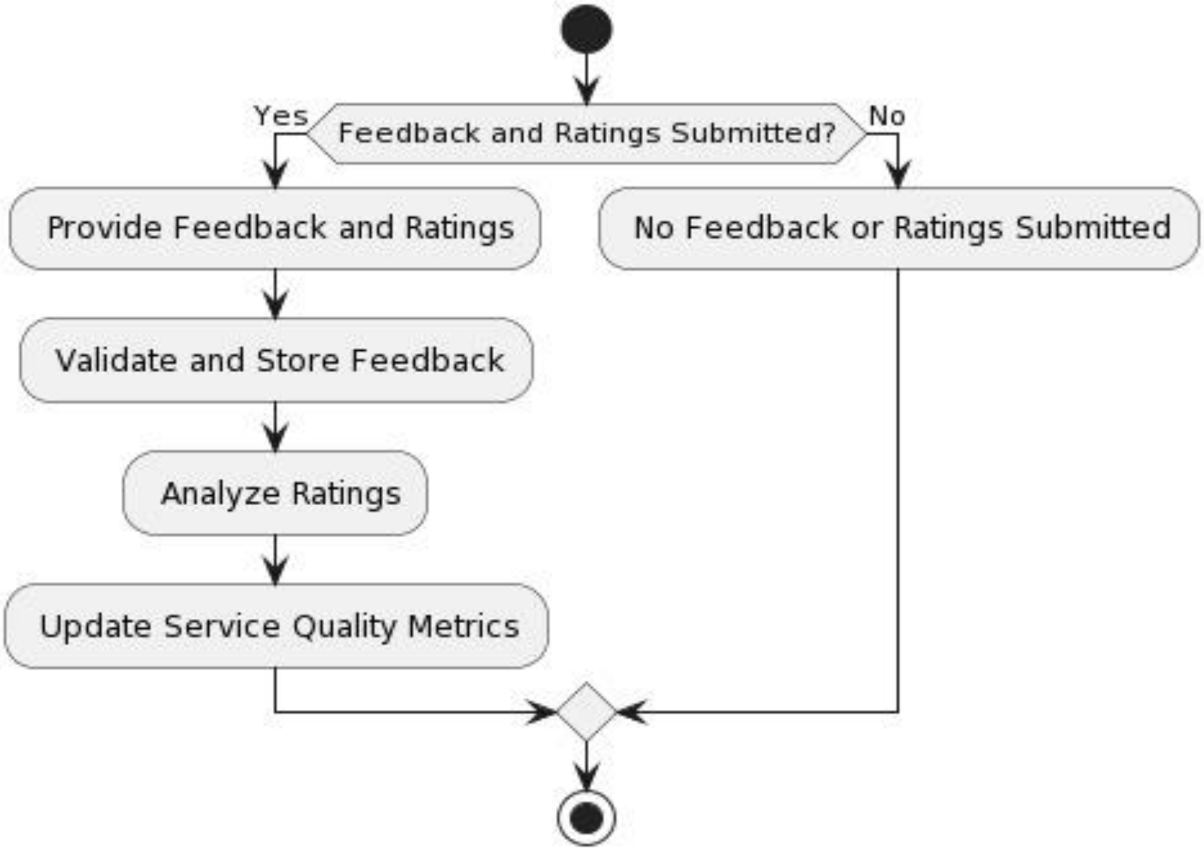
Feedback and Ratings Submitted?
Yes
No
Provide Feedback and Ratings
No Feedback or Ratings Submitted
Validate and Store Feedback
Analyze Ratings
Update Service Quality Metrics

Fig 7 : Feedback & Ratings

Activity Diagram (Seat Reservation)

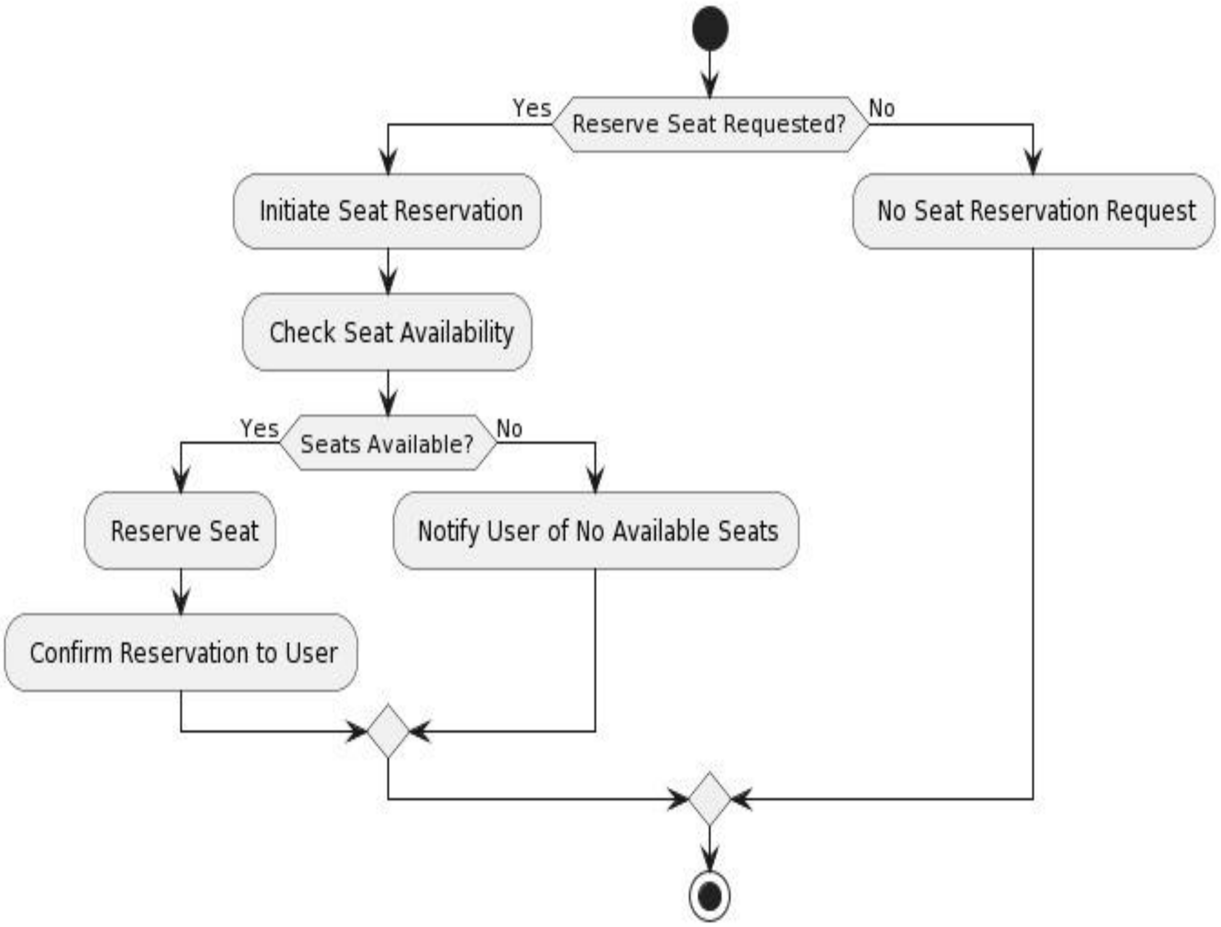
Yes
Reserve Seat Requested?
No
Initiate Seat Reservation
No Seat Reservation Request
Check Seat Availability
Yes
Seats Available?
No
Reserve Seat
Notify User of No Available Seats
Confirm Reservation to User

Fig 8 : Seat Reservation

Activity Diagram (Drop Off)

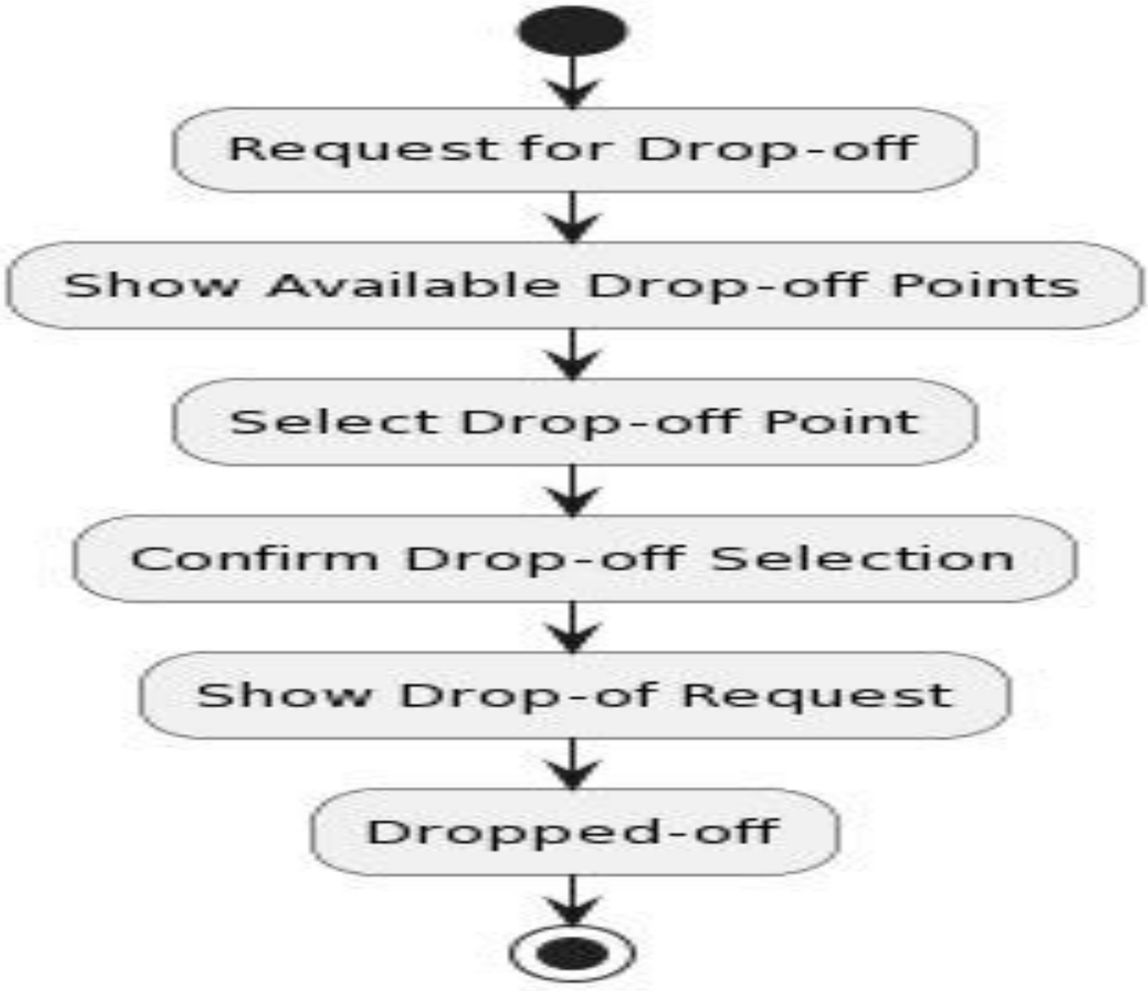

Fig 9 : Drop Off

Activity Diagram (Complain)

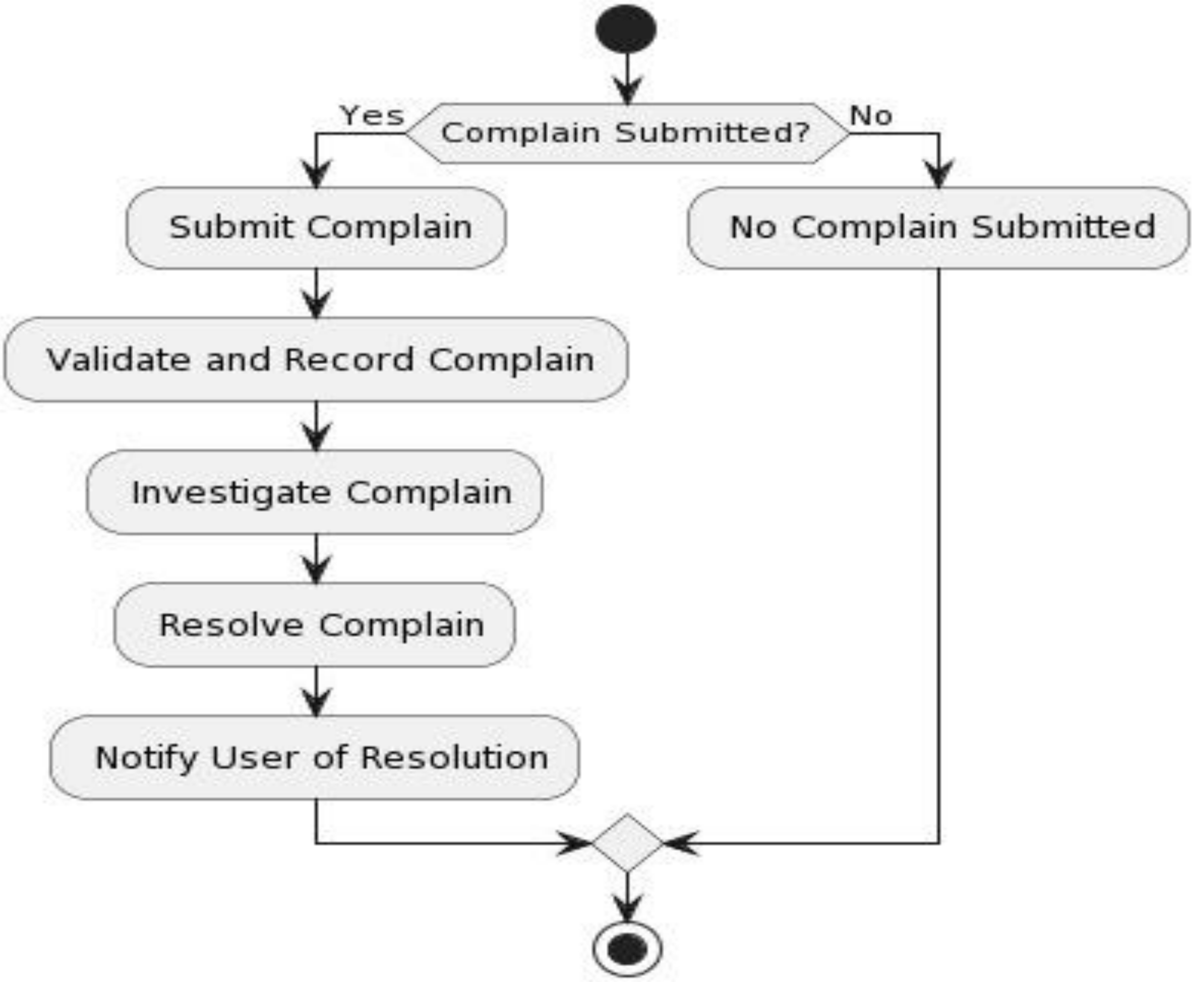

Fig 10 : Complain

Activity Diagram (Authentication)

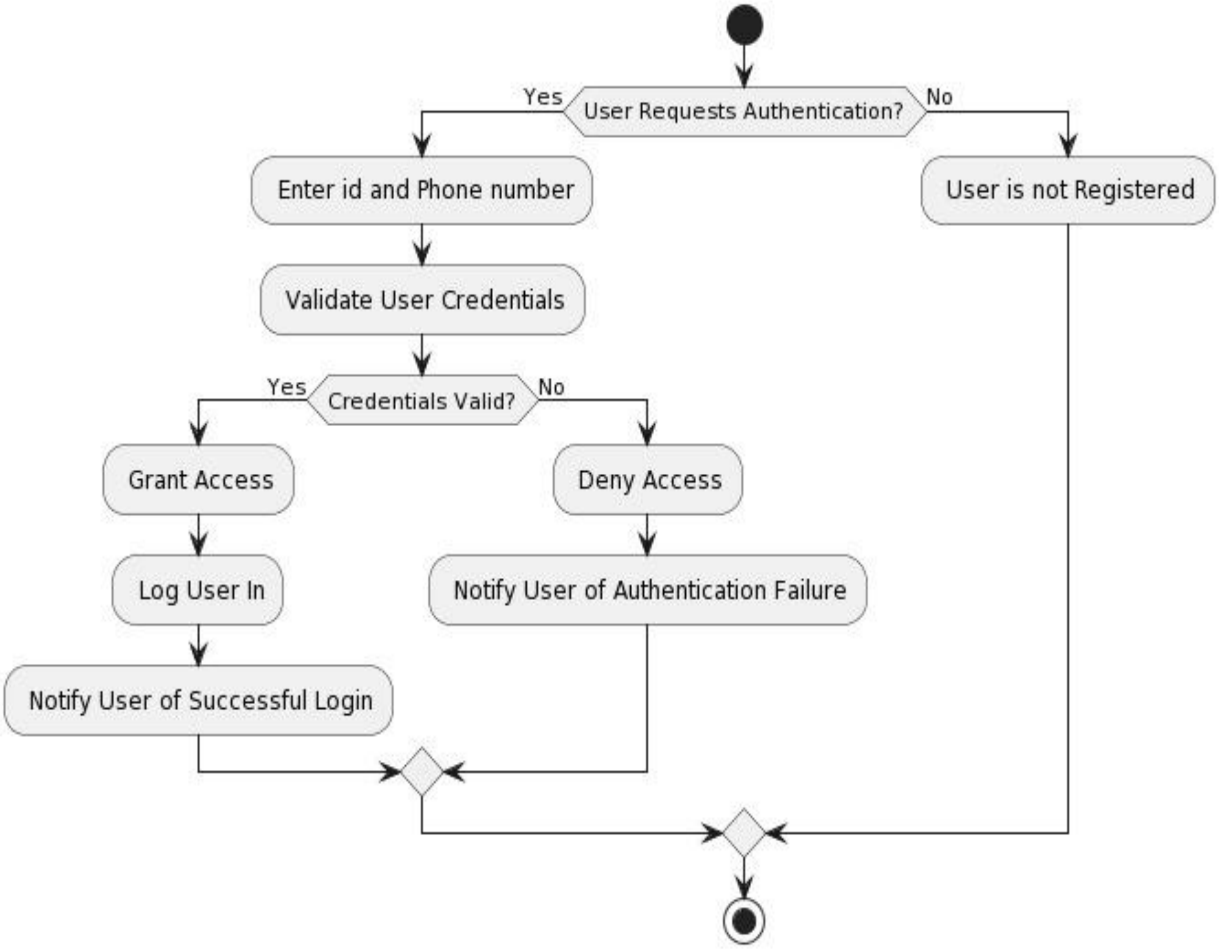
User Requests Authentication?
Yes
No
Enter id and Phone number
User is not Registered
Validate User Credentials
Credentials Valid?
Yes
No
Grant Access
Deny Access
Log User In
Notify User of Authentication Failure
Notify User of Successful Login

Fig 11 : Authentication

9 . Swim Lane Diagram

Swim Lane Diagram (View Bus Schedule)

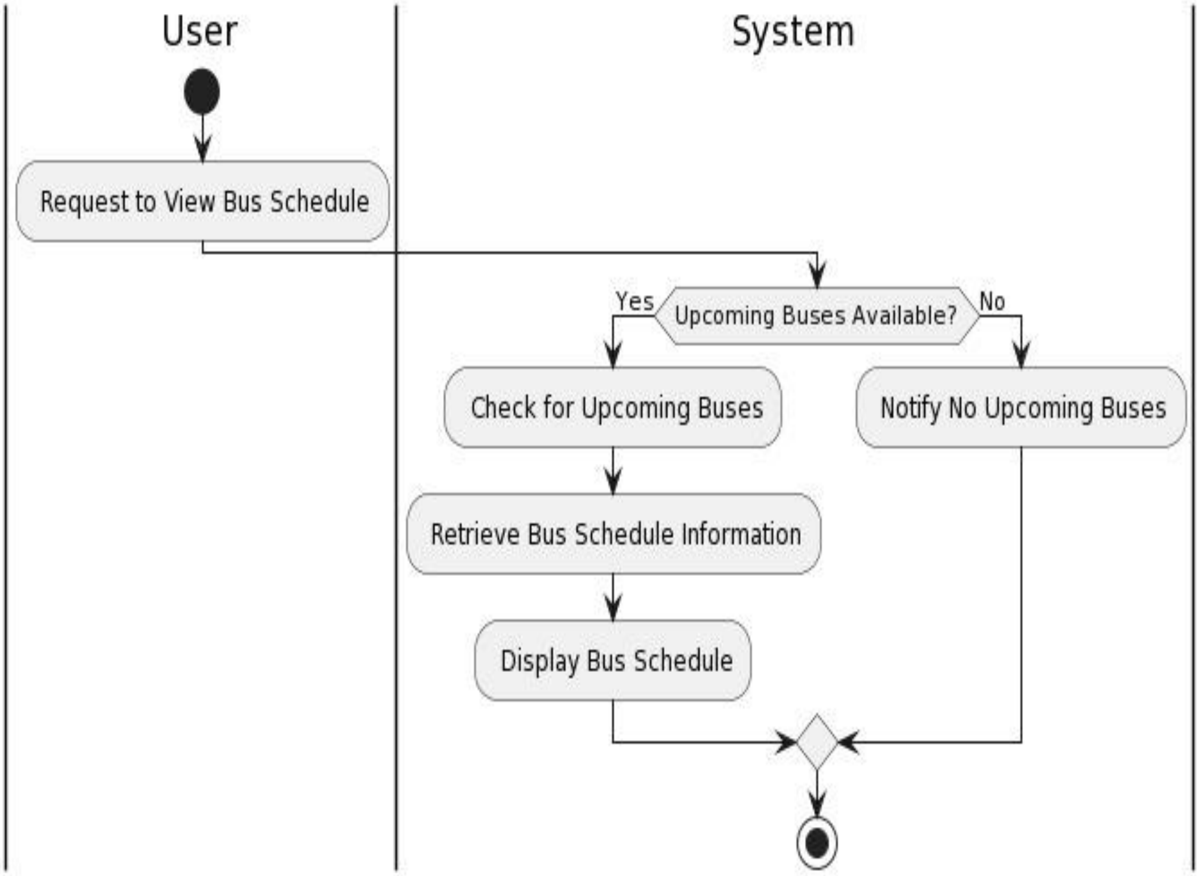
User
System
Request to View Bus Schedule
Yes
Upcoming Buses Available?
No
Check for Upcoming Buses
Notify No Upcoming Buses
Retrieve Bus Schedule Information
Display Bus Schedule

Fig 12 : View Bus Schedule

Swim Lane Diagram (Log in)

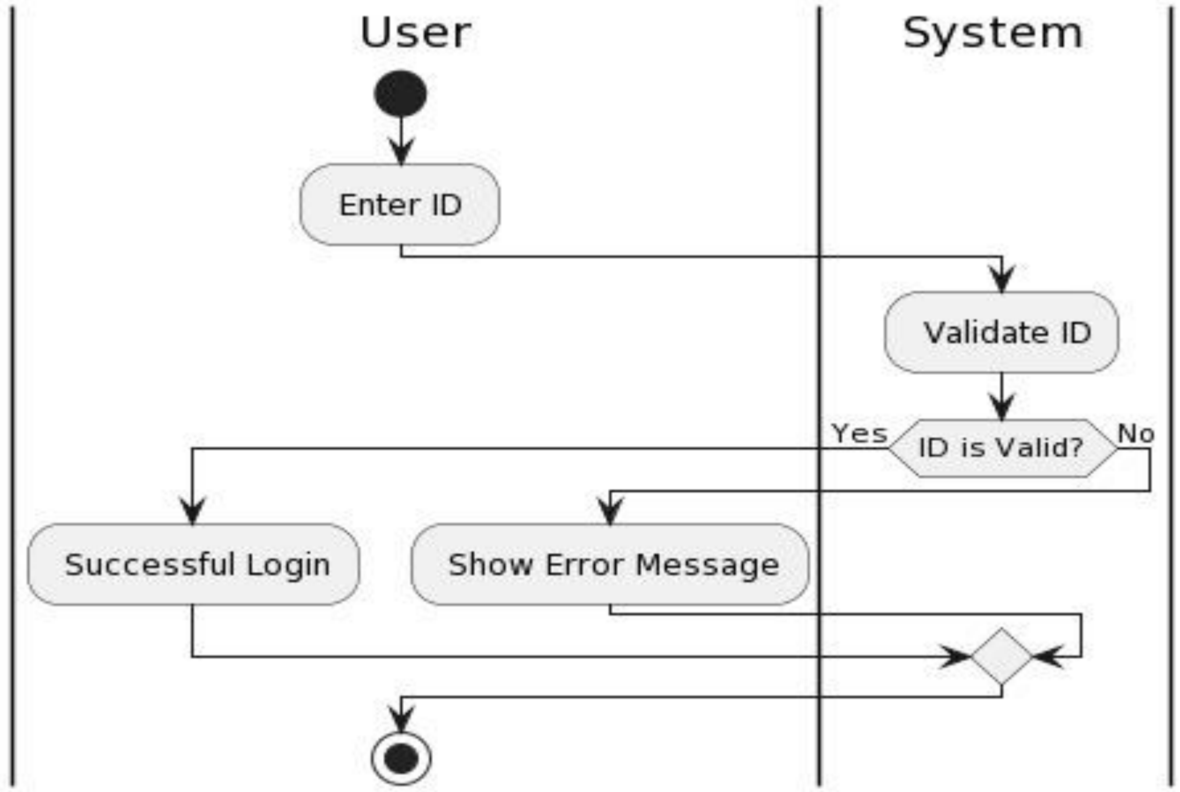

Fig 13 : Log in

Swim Lane Diagram (Optimize Routes)

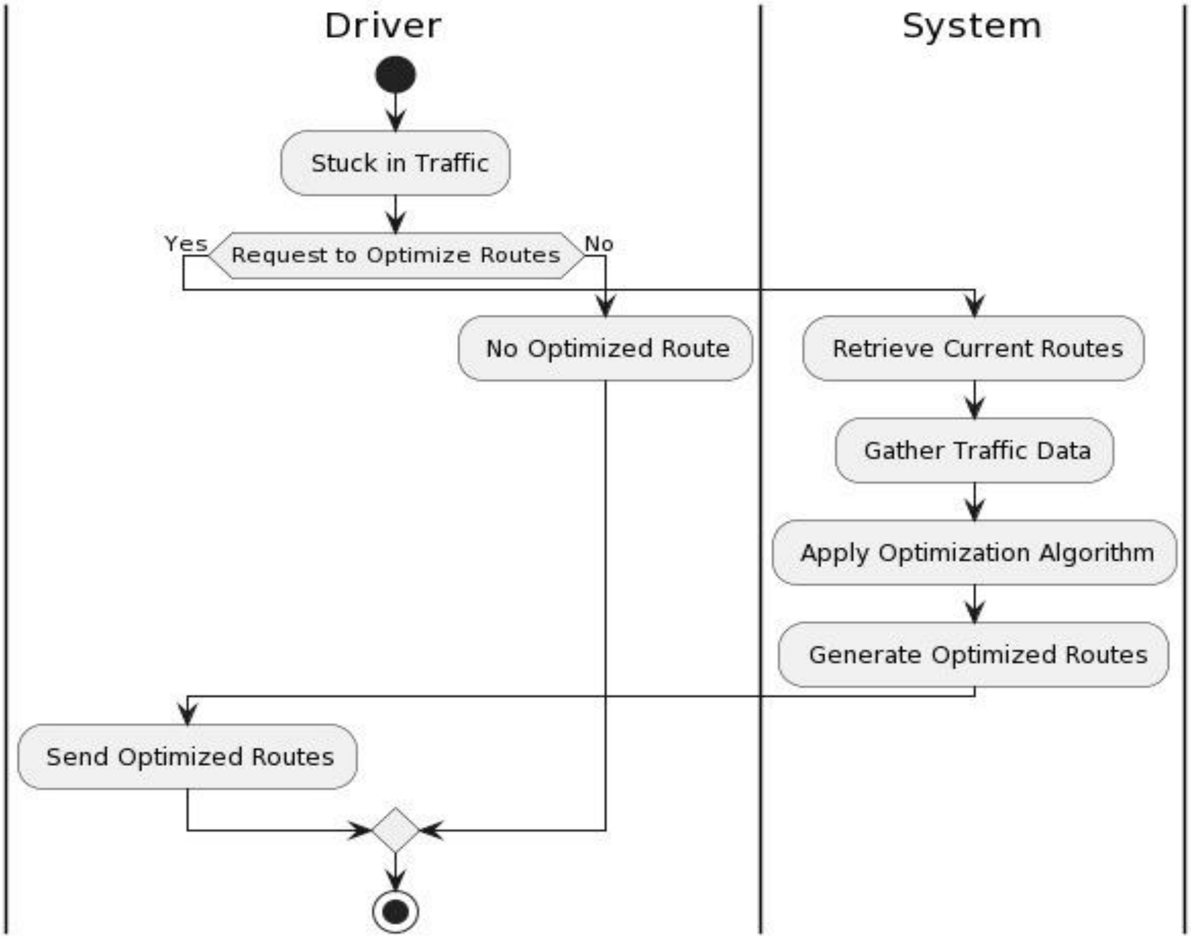

Fig 14 : Optimize Routes

Swim Lane Diagram (Real Time Tracking)

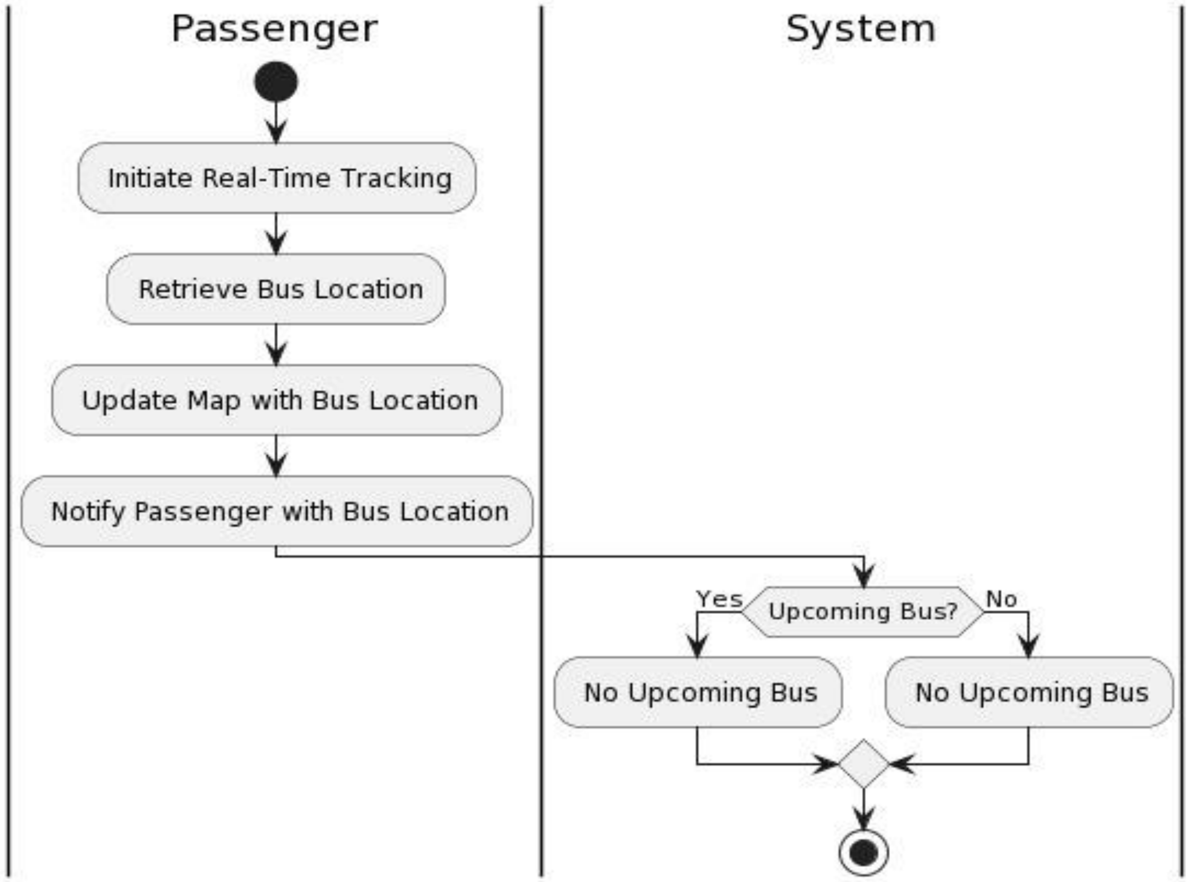
Passenger
System
Initiate Real-Time Tracking
Retrieve Bus Location
Update Map with Bus Location
Notify Passenger with Bus Location
Upcoming Bus?
Yes
No
No Upcoming Bus
No Upcoming Bus

Fig 15 : Real Time Tracking

Swim Lane Diagram (Received Notification)

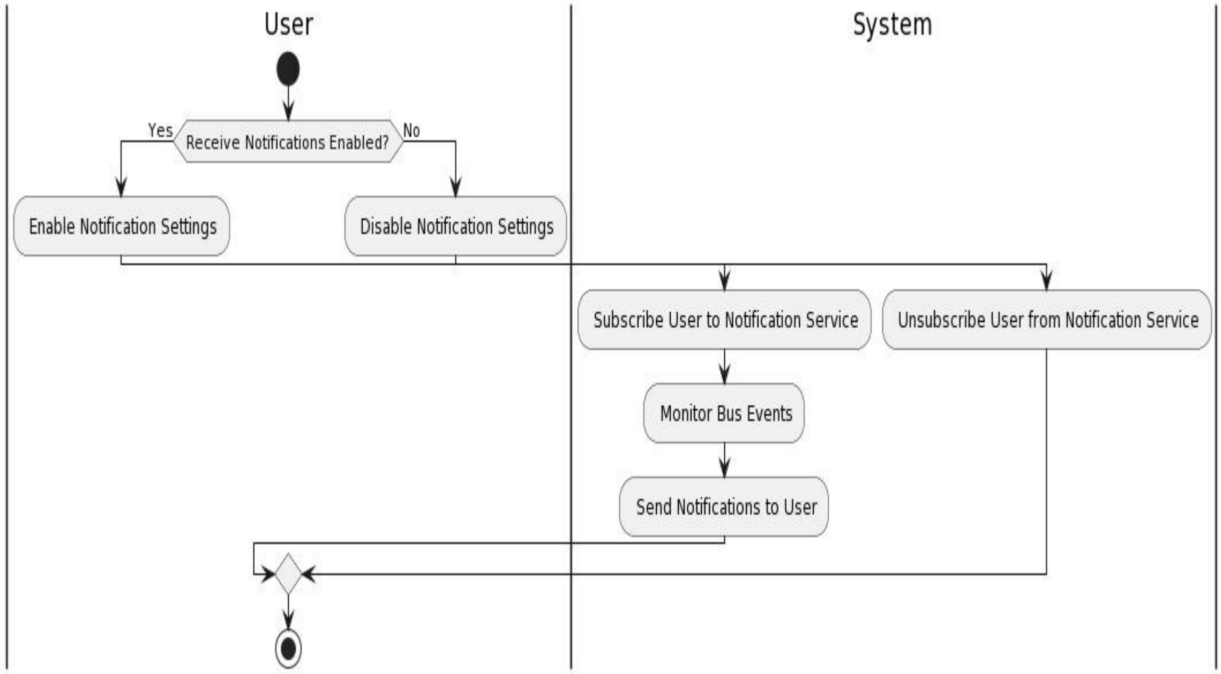
User
System
Receive Notifications Enabled?
Yes
No
Enable Notification Settings
Disable Notification Settings
Subscribe User to Notification Service
Unsubscribe User from Notification Service
Monitor Bus Events
Send Notifications to User

Fig 16 : Received Notification

Swim Lane Diagram (Feedback & Ratings)

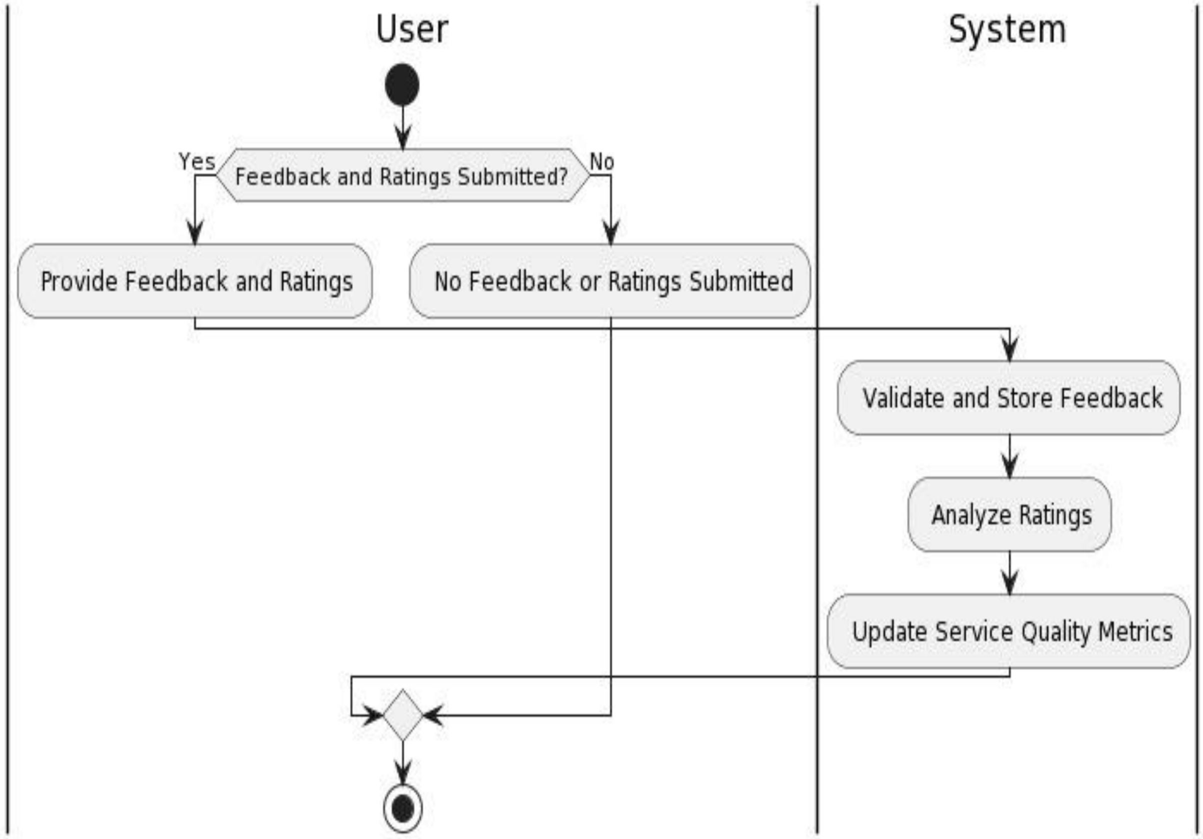
User
System
Yes
Feedback and Ratings Submitted?
No
Provide Feedback and Ratings
No Feedback or Ratings Submitted
Validate and Store Feedback
Analyze Ratings
Update Service Quality Metrics

Fig 17 : Feedback & Ratings

Swim Lane Diagram (Seat Reservation)

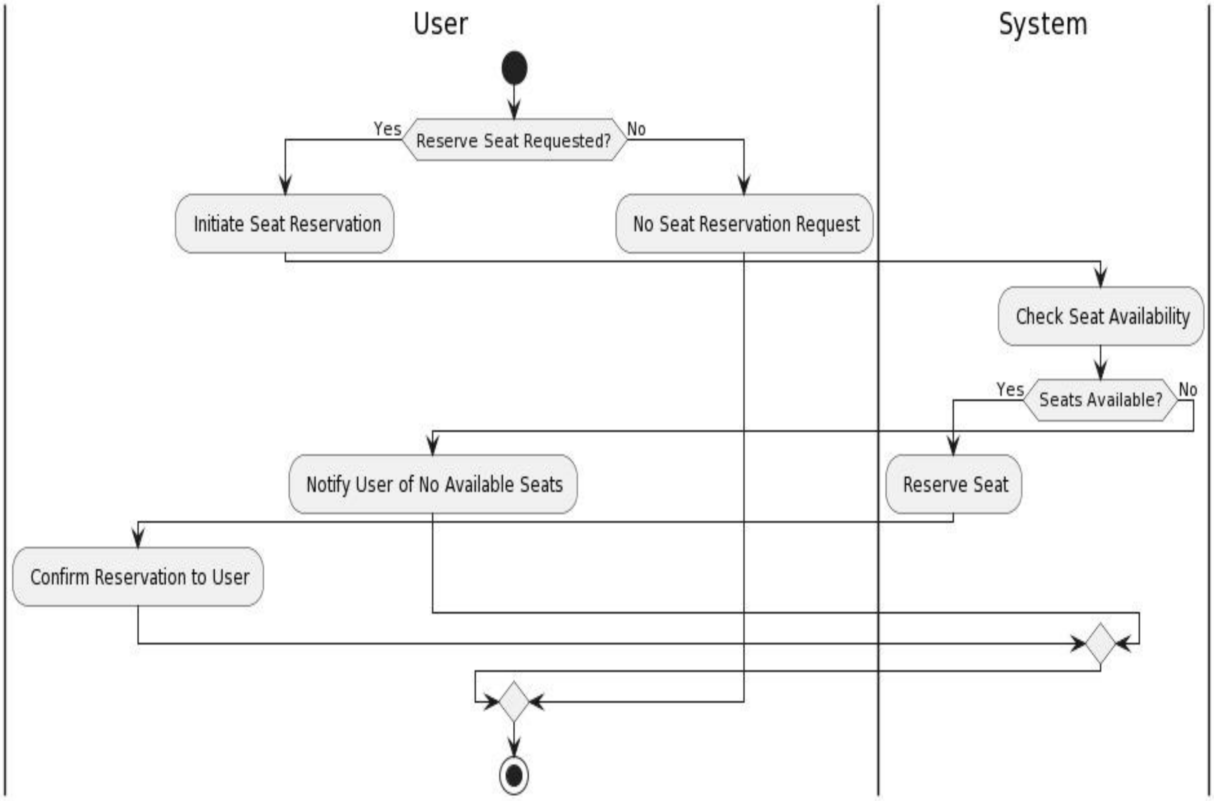
User
System
Reserve Seat Requested?
Yes
No
Initiate Seat Reservation
No Seat Reservation Request
Check Seat Availability
Seats Available?
Yes
No
Notify User of No Available Seats
Reserve Seat
Confirm Reservation to User

Fig 18 : Seat Reservation

Swim Lane Diagram (Drop Off)

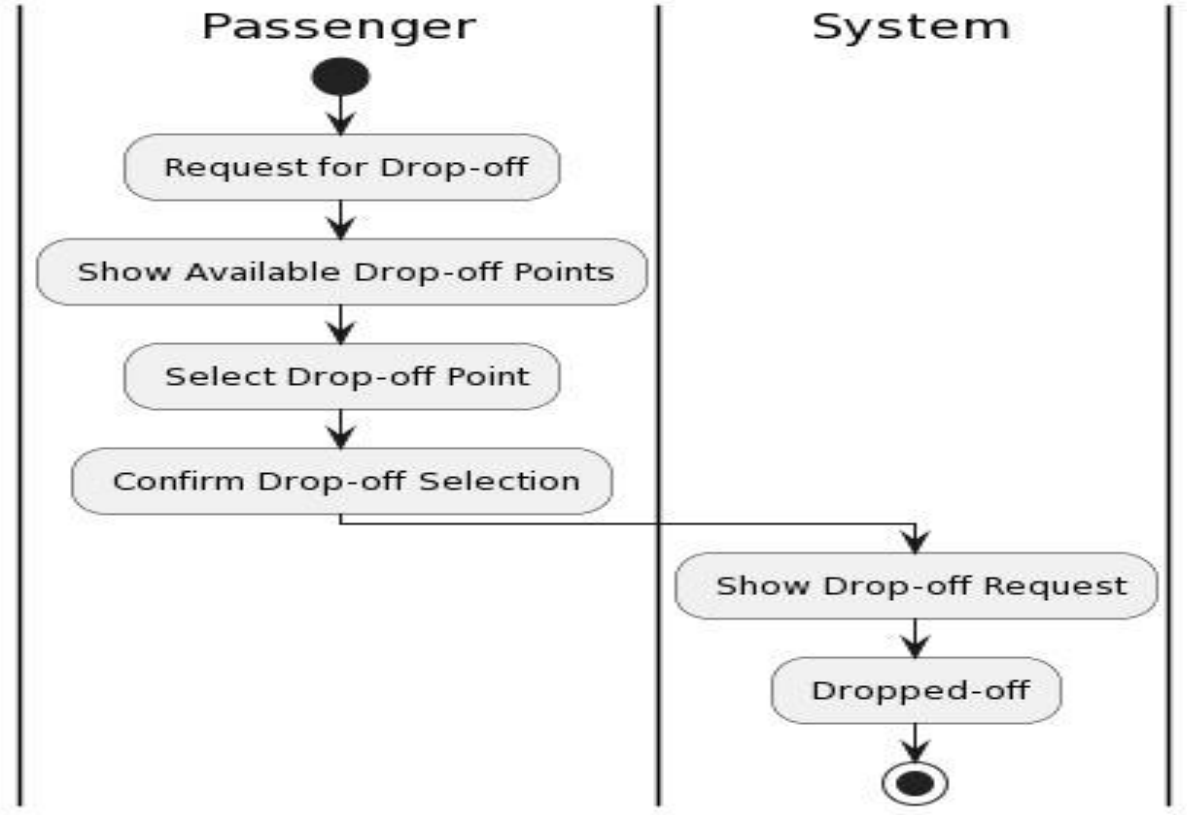

Fig 19 : Drop Off

Swim Lane Diagram (Complain)

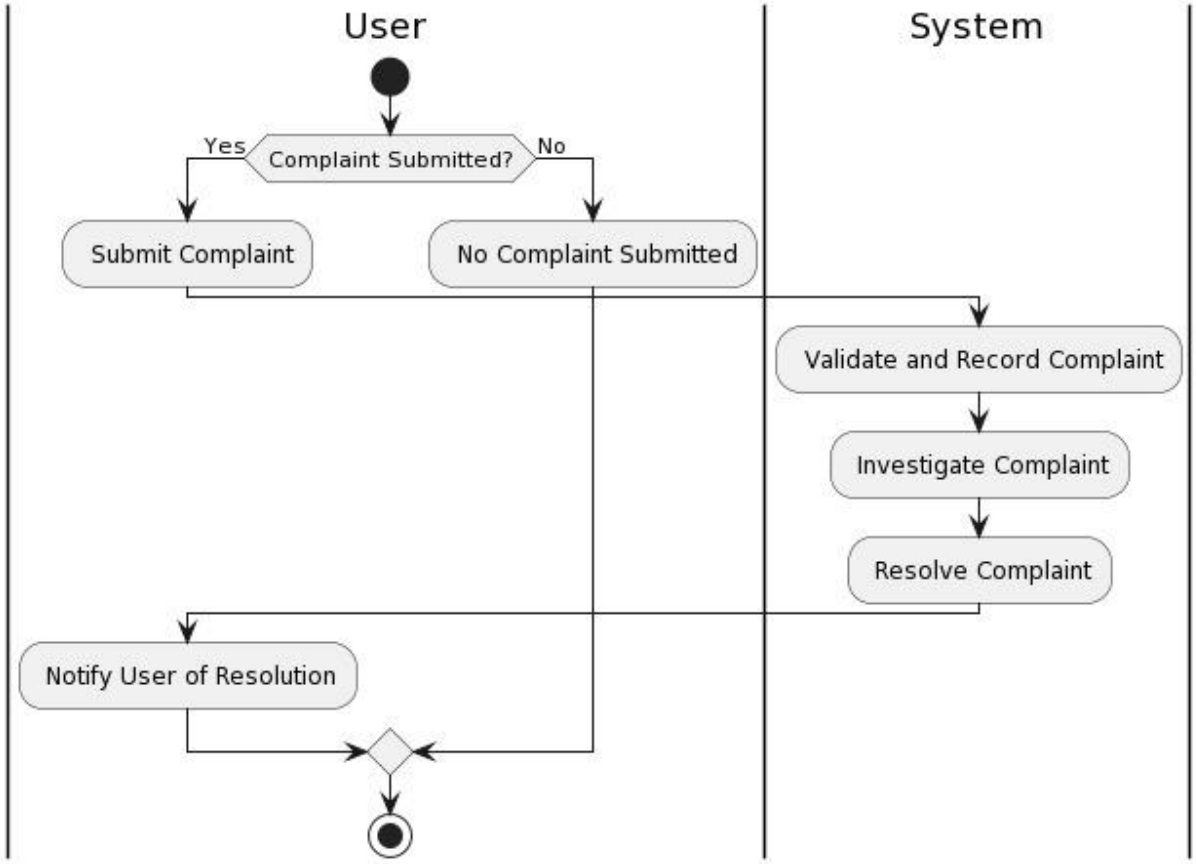

Fig 20 : Complain

Swim Lane Diagram (Authentication)

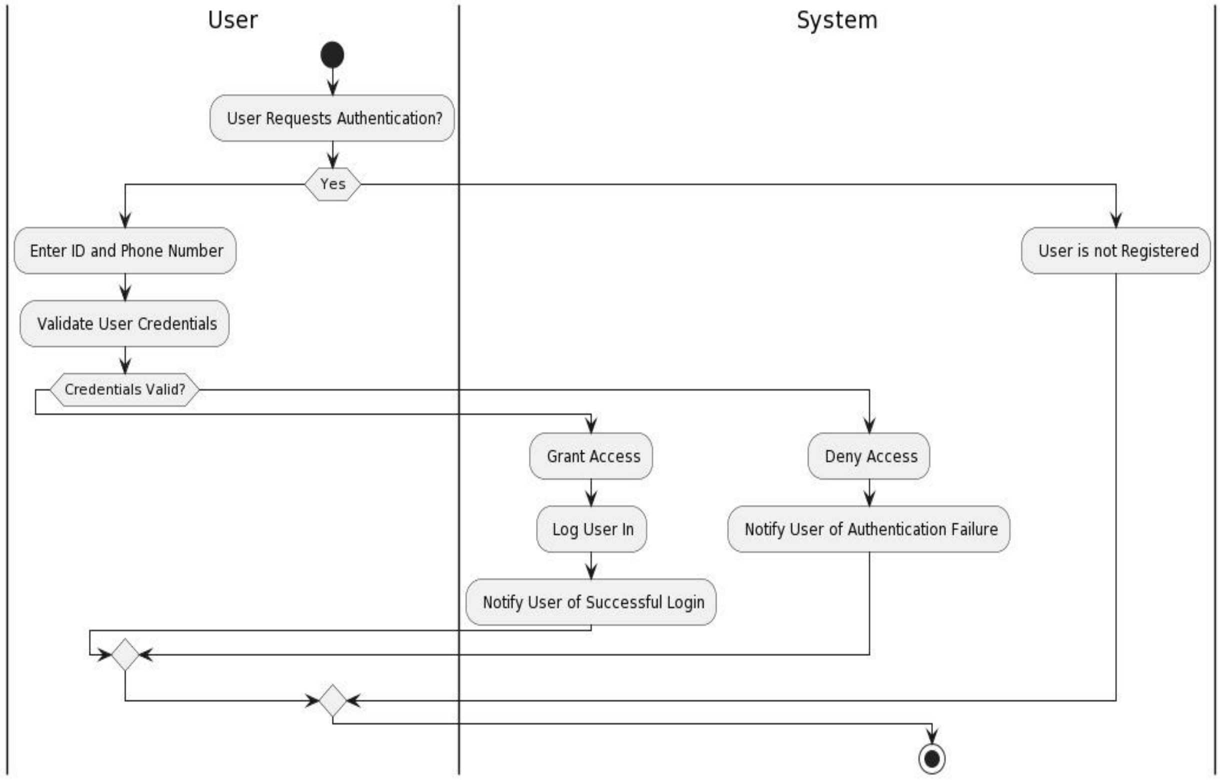
User
System
User Requests Authentication?
Yes
Enter ID and Phone Number
User is not Registered
Validate User Credentials
Credentials Valid?
Grant Access
Deny Access
Log User In
Notify User of Authentication Failure
Notify User of Successful Login

Fig 21 : Authentication

Swim Lane Diagram (Emergency Assistance)

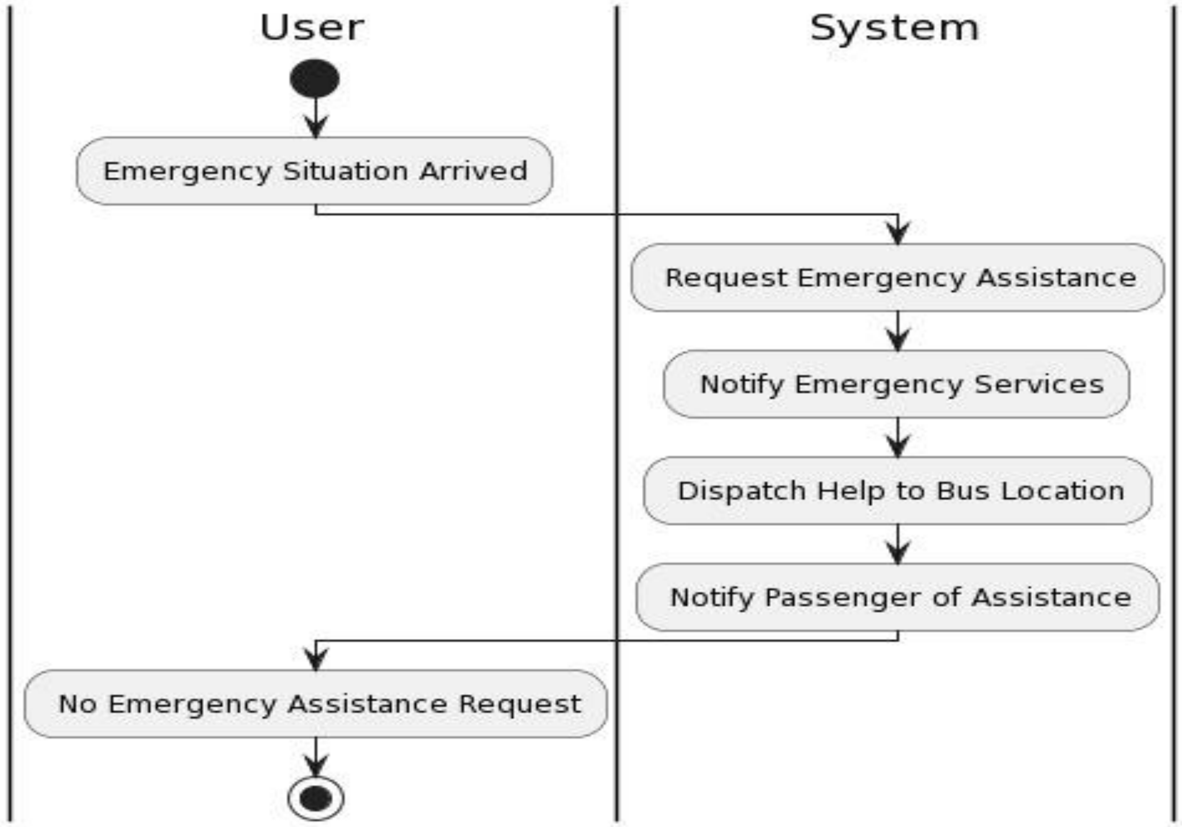
User
System
Emergency Situation Arrived
Request Emergency Assistance
Notify Emergency Services
Dispatch Help to Bus Location
Notify Passenger of Assistance
No Emergency Assistance Request

Fig 22 : Emergency Assistance

10 .Sequence Diagram

Sequence Diagram (Log in)

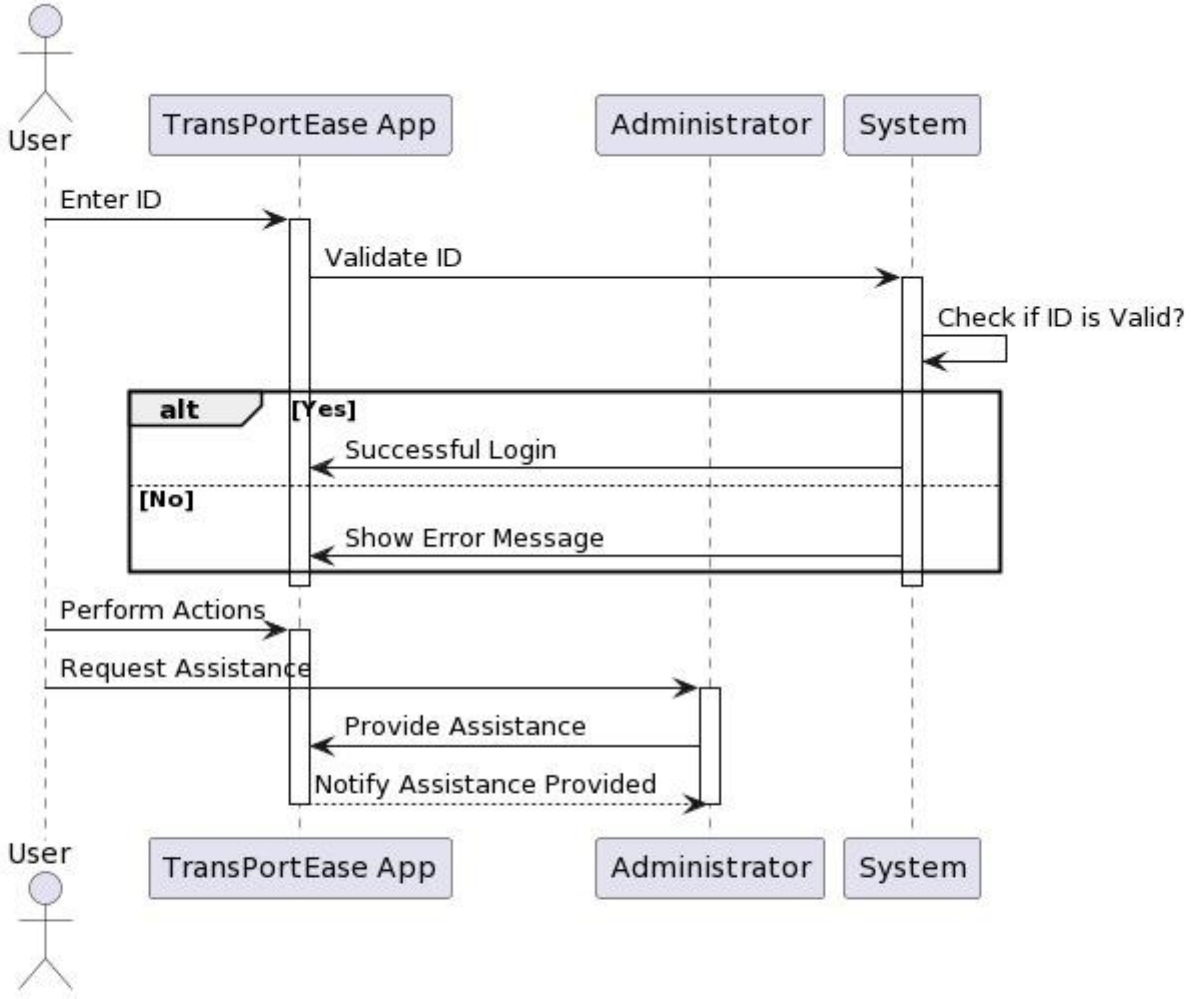

Fig 23 : Log in

Sequence Diagram (View Bus Schedule)

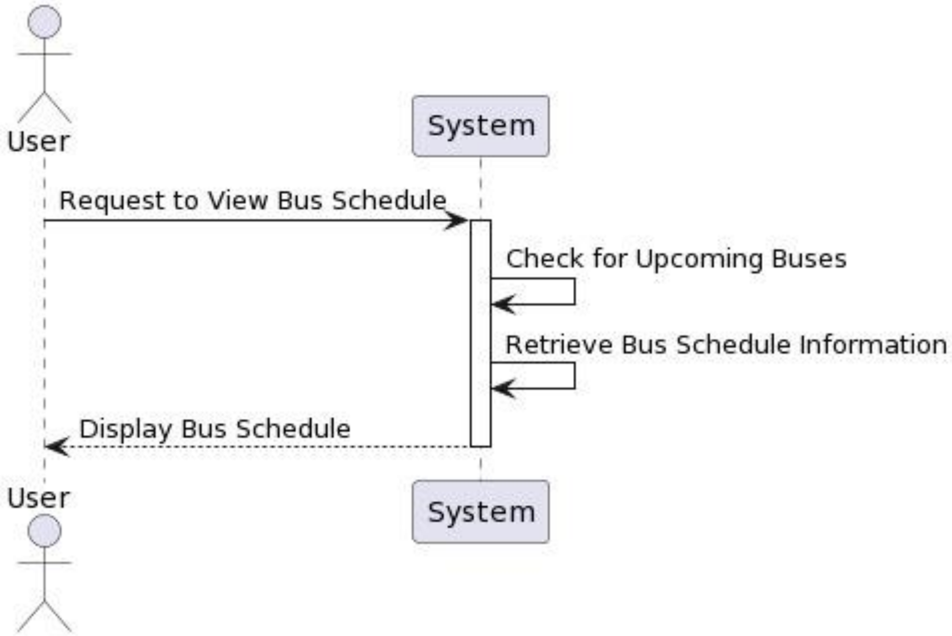

Fig 24: View Bus Schedule

Sequence Diagram (complain)

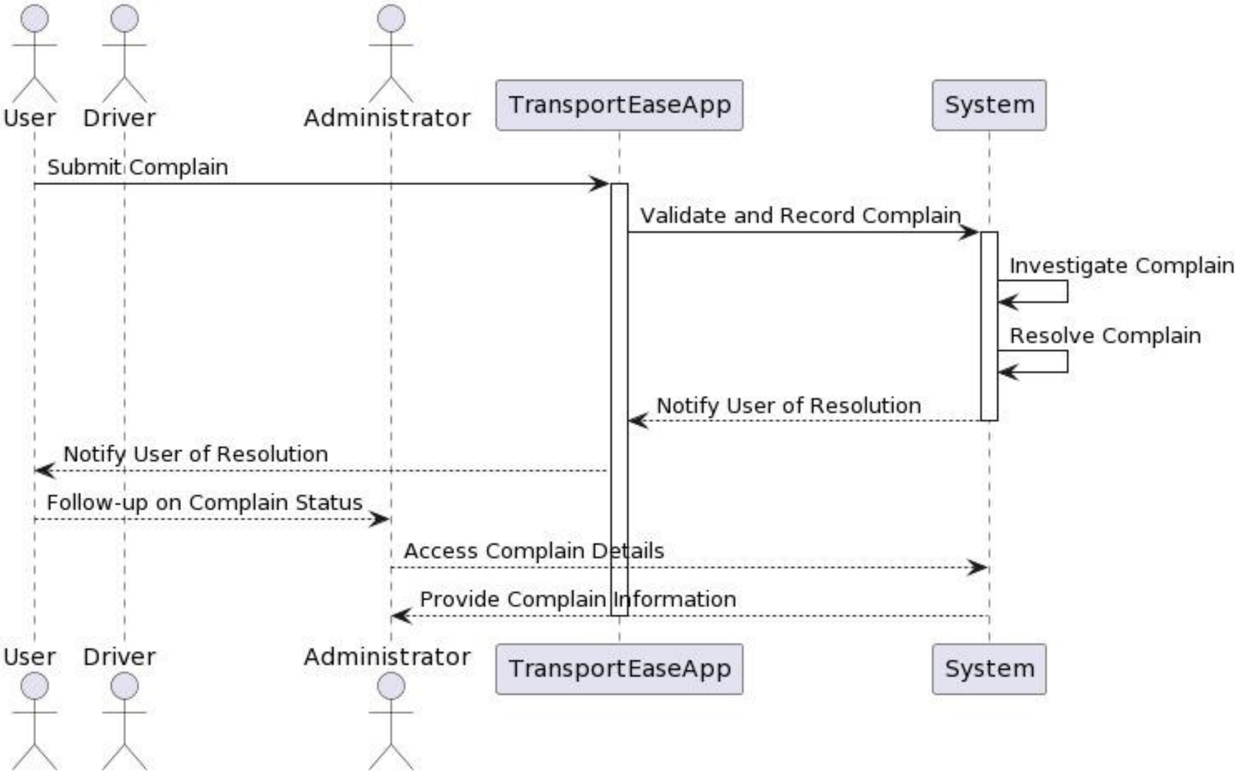

Fig 25 : complain

Sequence Diagram (Feedback & Ratings)

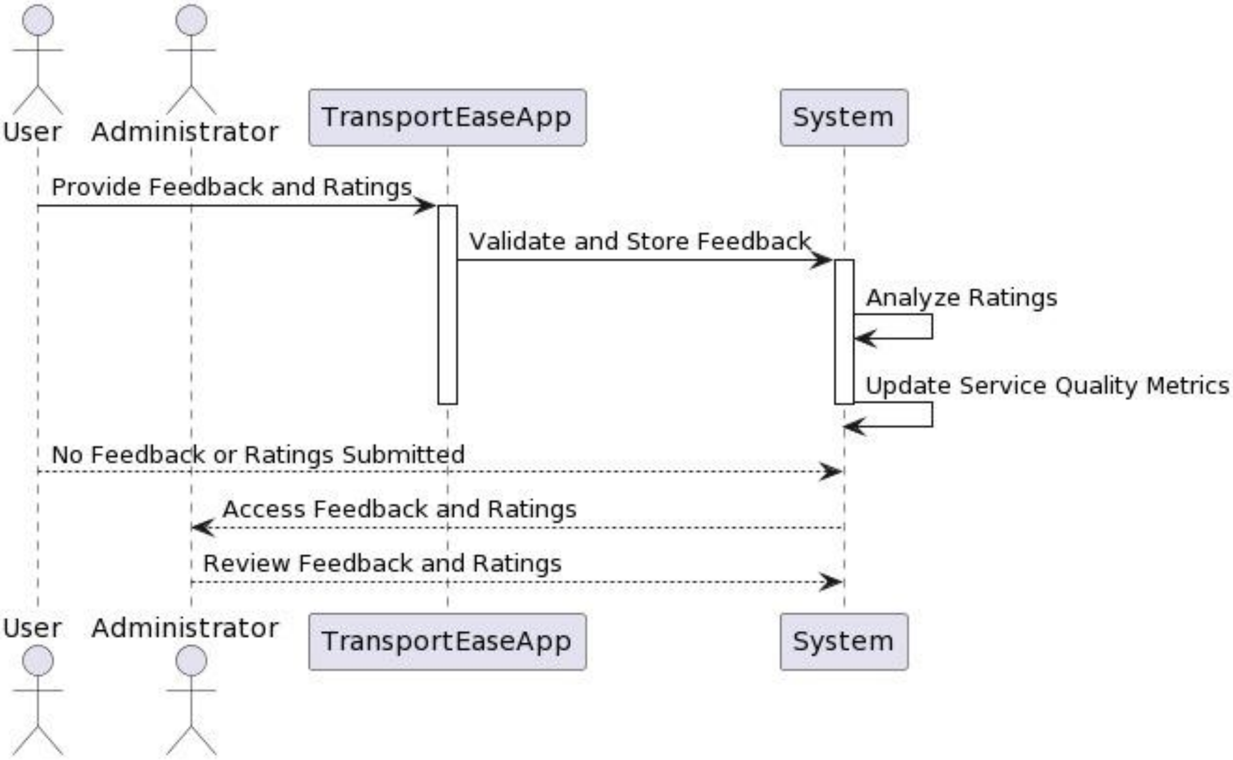
User
Administrator
TransportEaseApp
System
Provide Feedback and Ratings
Validate and Store Feedback
Analyze Ratings
Update Service Quality Metrics
No Feedback or Ratings Submitted
Access Feedback and Ratings
Review Feedback and Ratings
User
Administrator
TransportEaseApp
System

Fig 26 : Feedback & Ratings

Sequence Diagram (Emergency Assistance)

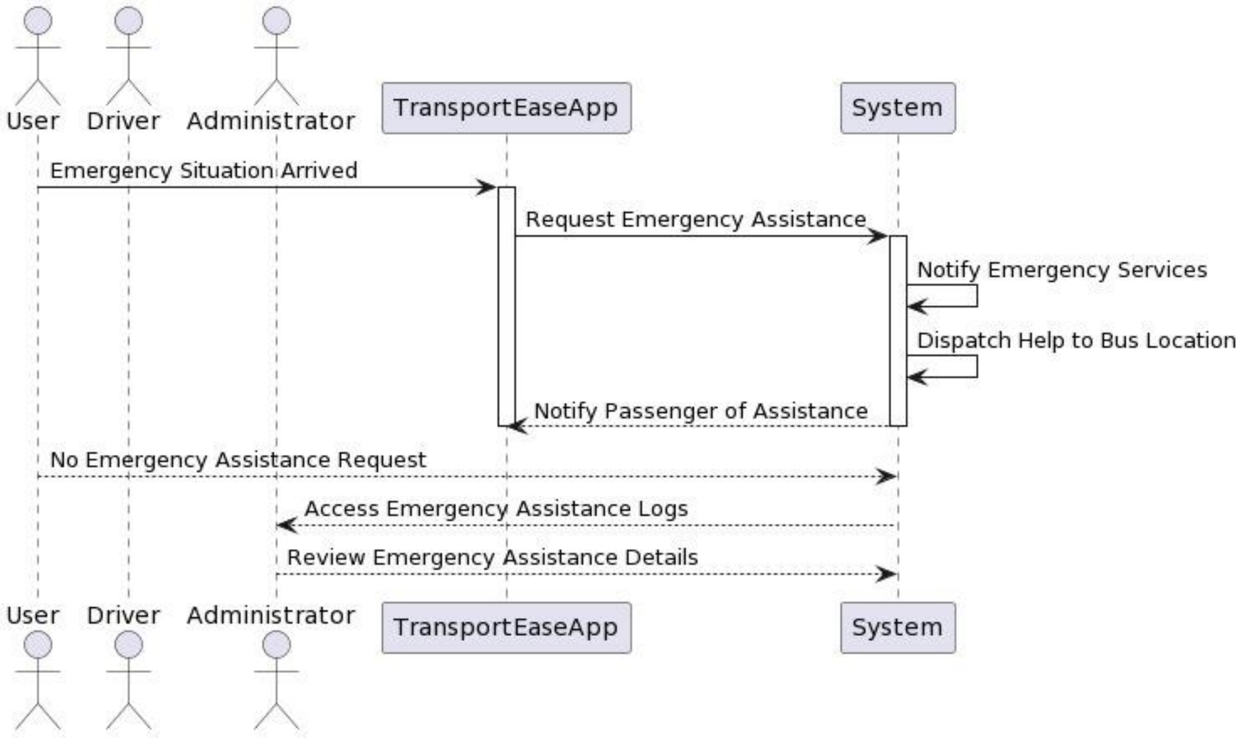
User
Driver
Administrator
TransportEaseApp
System
Emergency Situation Arrived
Request Emergency Assistance
Notify Emergency Services
Dispatch Help to Bus Location
Notify Passenger of Assistance
No Emergency Assistance Request
Access Emergency Assistance Logs
Review Emergency Assistance Details
User
Driver
Administrator
TransportEaseApp
System

Fig 27: Emergency Assistance

Sequence Diagram (Reserve Seat)

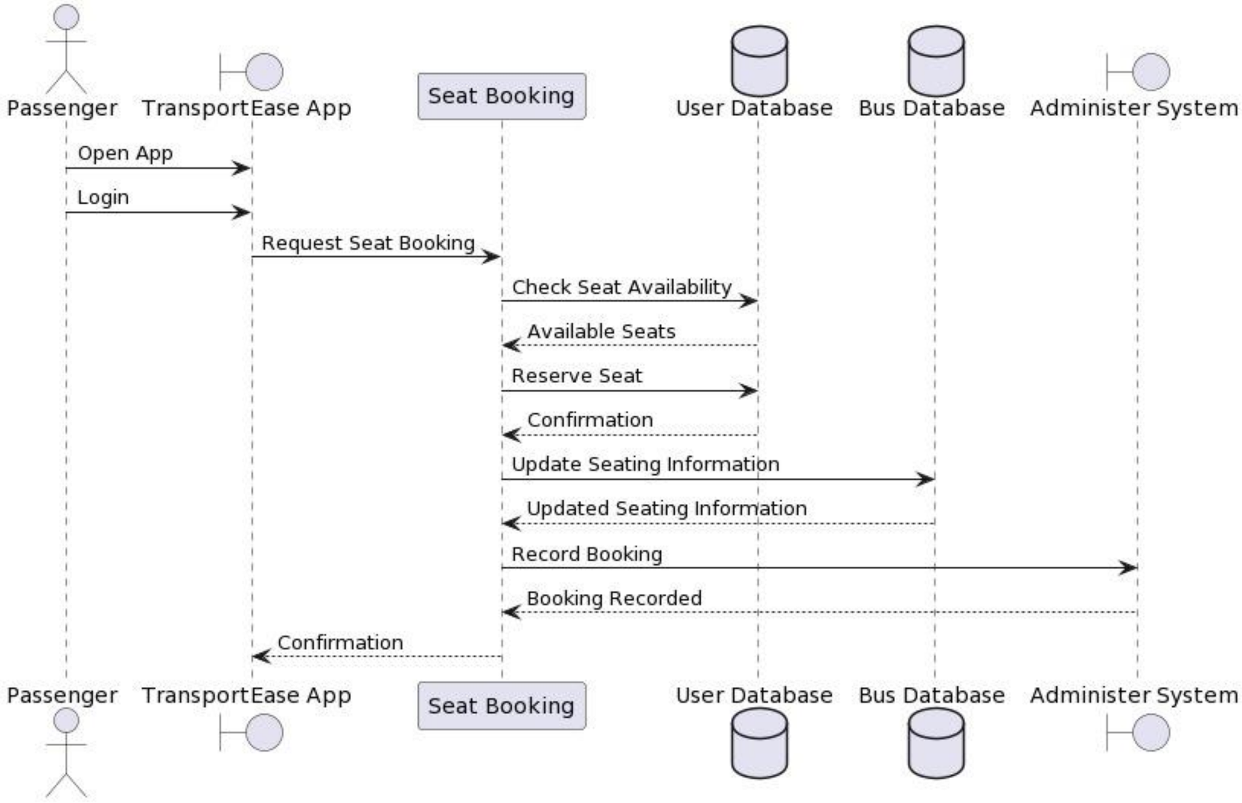

Fig 28 : Reserve Seat

Sequence Diagram (Drop Off)

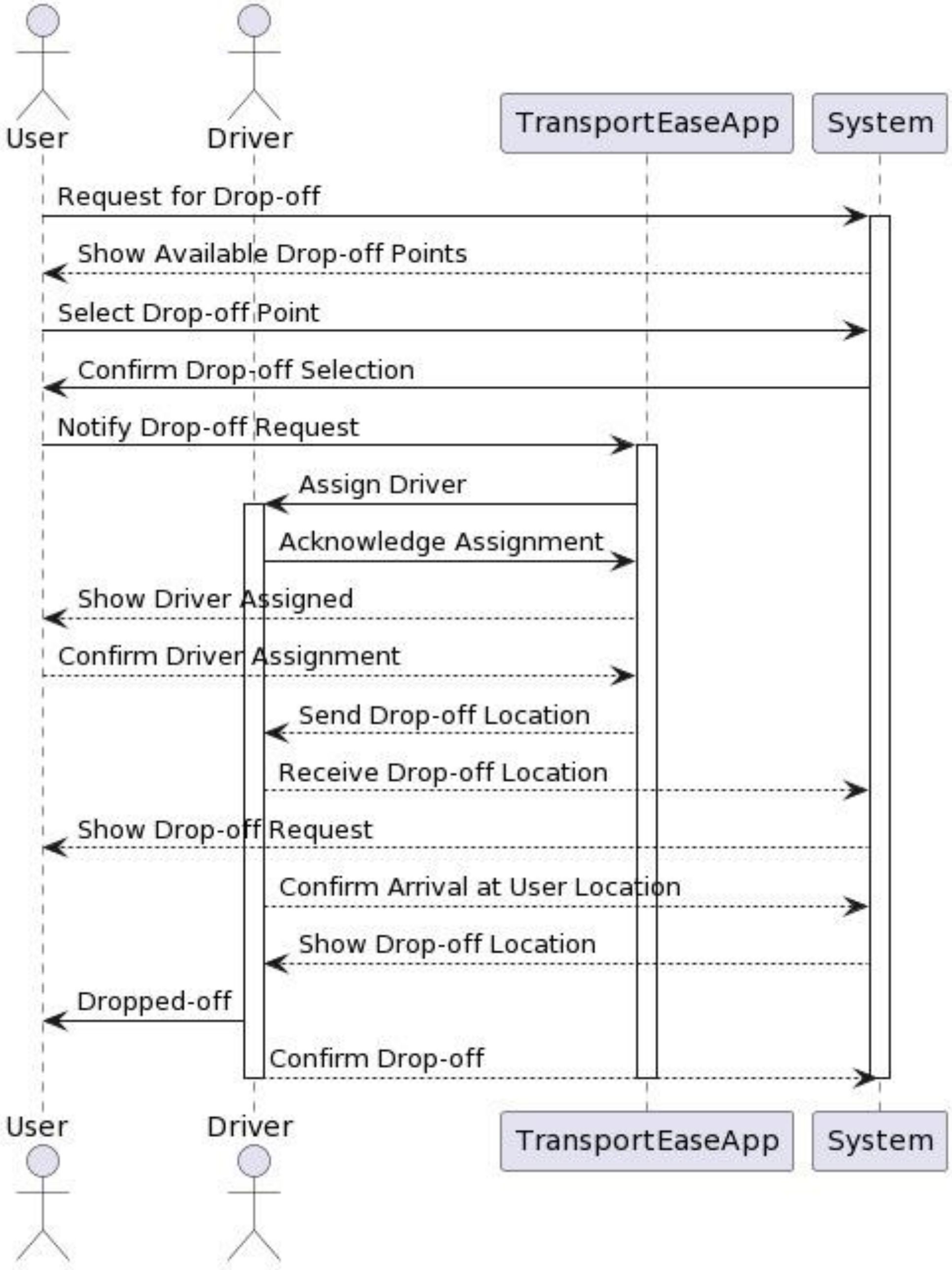

Fig 29 : Drop Off

Sequenc e Diagram (Real Time Tracking)

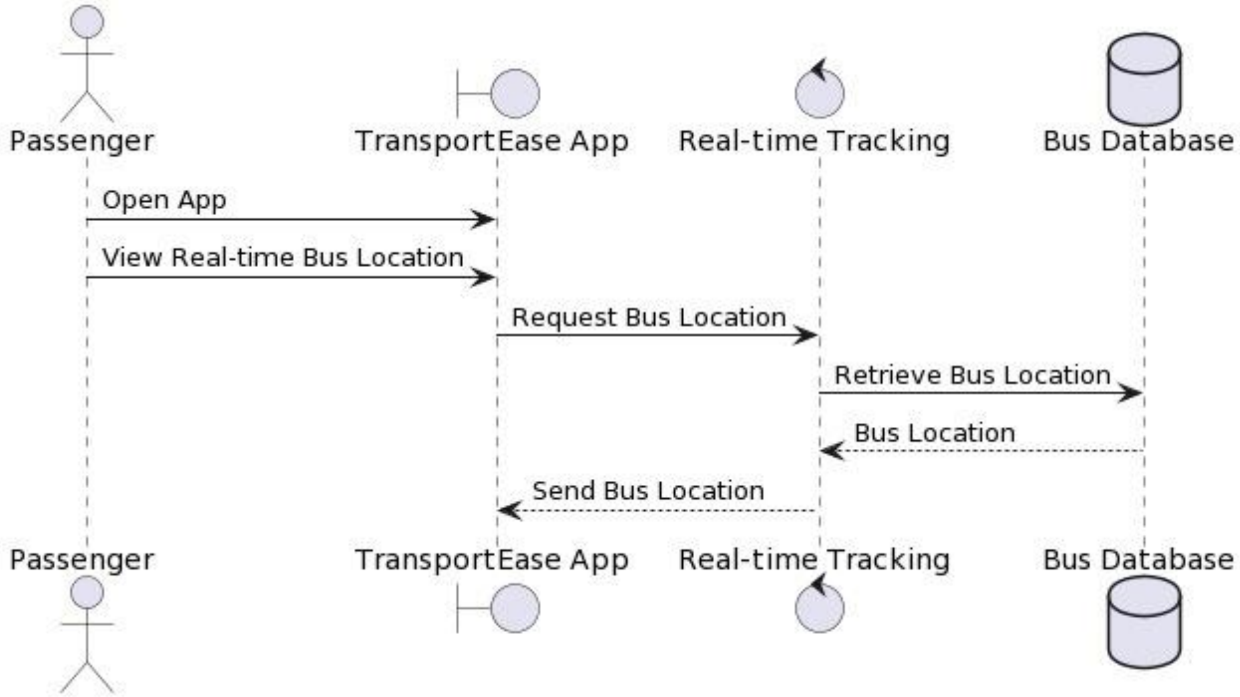

Fig 30: Real Time Tracking

Sequenc e Diagram (Optimze Routes)

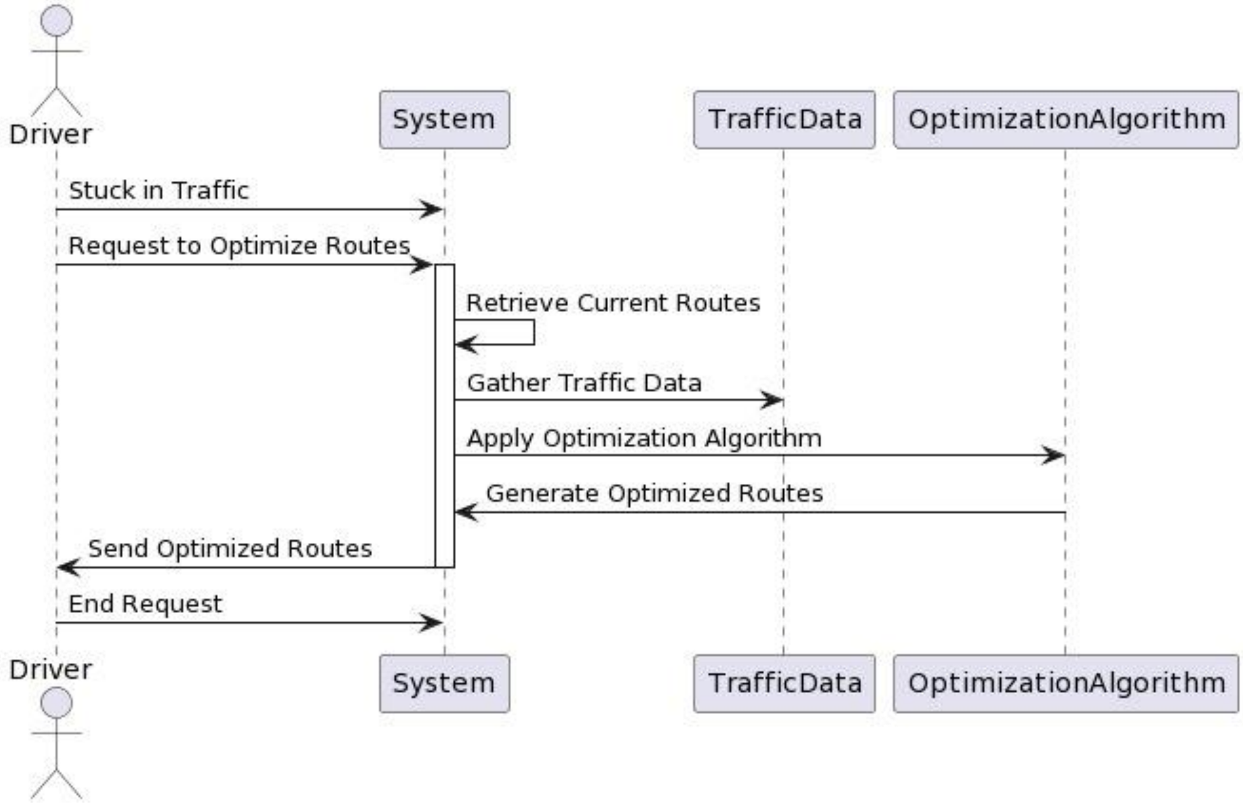

Fig 31 : Optimze Routes

11 . State Diagram

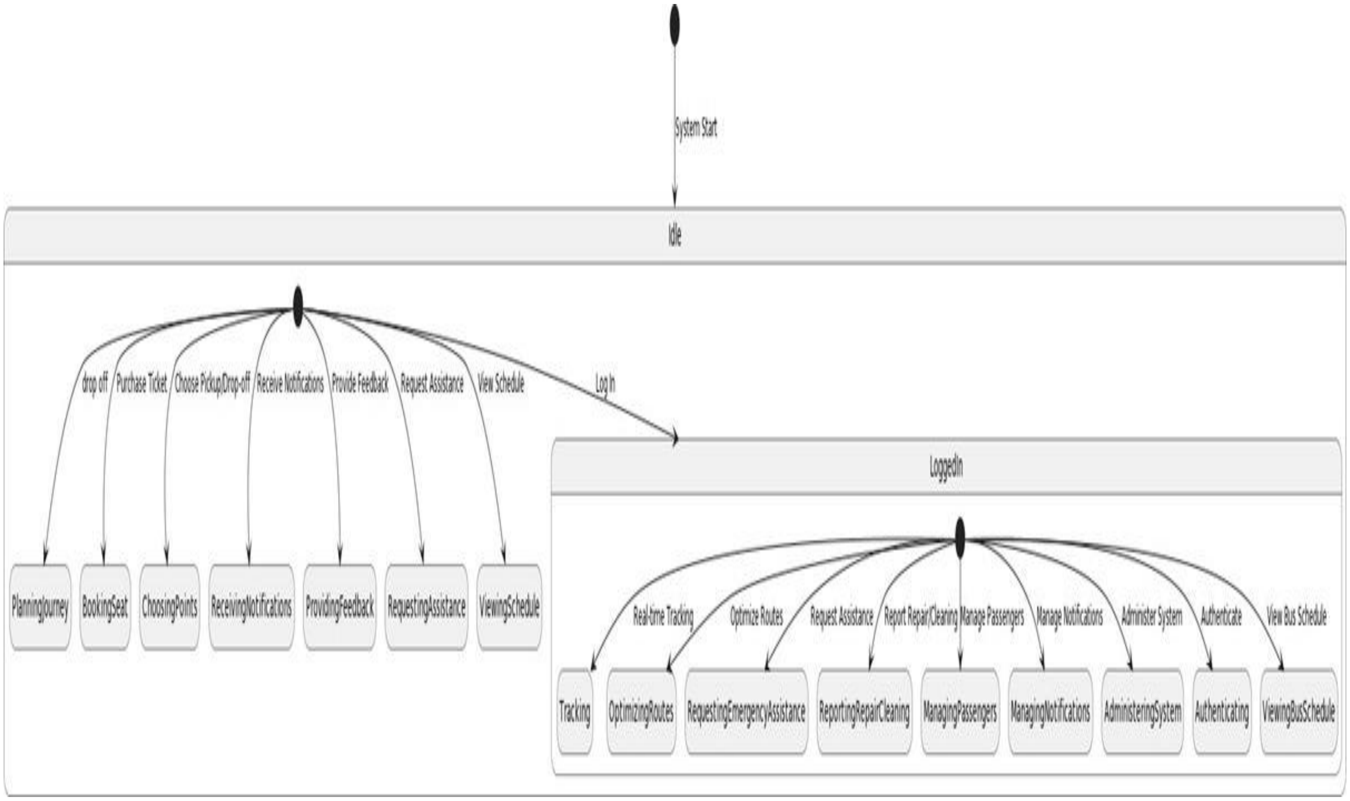

Fig 32 : State Diagram