UNIVERSITI MALAYSIA TERENGGANU

FACULTY OF SCIENCE COMPUTER AND MATHEMATIC

APPLICATION SYSTEM DEVELOPMENT PROJECT (CSE3953)

K1

PROJECT REPORT

BUS TICKET RESERVATION SYSTEM

NAME	MATRIC NUMBER
AESYA SURAYA BINTI ANUAR	S65410
NUR HAZIQAH BINTI WAGIMAN	S66833
MUHAMMAD IRFAN HAKIMI BIN YAHYA	S65610
MUHAMMAD HARITH FIKRI BIN MOHD HANAFIAH	S67623

Prepared by:
Group 4

Prepared for:
Assoc. Prof. Ts. Dr. Aida Ali

[Bachelor of Computer Science (Software Engineering) with Honour]

SEMESTER II 2023/2024

TABLE OF CONTENT

TABLE OF CONTENT **1**
1 . SOFTWARE PROJECT MANAGEMENT PLAN (BY AESYA SURAYA) **4**
1.1 INTRODUCTION 4
1.1.1 Problem Statements 4
1.1.2 Problem Solutions 5
1.1.3 Objective of the System 6
1.1.4 Scope of the System 7
1.2 LITERATURE REVIEWS 9
1.2.1 Introduction 9
1.2.2 Review of the Existing System 9
1.2.3 Discussion 10
1.3 METHODOLOGY 11
1.3.1 Introduction 11
1.3.2 Software Model 11
1.3.3 Software Architecture 14
1.3.4 Project Management Plan 16
1.4 EXPECTED OUTCOMES 17
1.5 ENTREPRENEURSHIP 18
1.5.1 Qualification 18
1.5.2 Pricing options 18
1.5.3 Clarify terms and conditions 19
1.6 CONCLUSION 20
2. SOFTWARE PROJECT SPECIFICATION (BY HAZIQAH) **21**
2.1 DEVELOPMENT REQUIREMENT 21
2.1.1 Hardware Specification 21
2.1.2 Software Specification 21
2.2 REQUIREMENTS 22
2.2.1 Requirements Gathering Technique 22
2.2.2 Functional Requirements 25
2.2.3 Non-Functional Requirements 25
2.3 SYSTEM MODELING 26
2.3.1 Business Process and Functional Modeling 26
2.3.1.1 Use Case Diagram 26
2.3.1.2 Use Case Description 27
2.3.1.3 Activity Diagram 29
2.3.2 Structural Modeling 39
2.3.2.1 CRUD Diagram 39

2.3.2.2 Class Diagram 40
2.3.3 Behavioral Modeling 41
2.3.3.1 Sequence Diagram 41
2.4 CONCLUSION 51
3. SOFTWARE DESIGN DOCUMENT (BY IRFAN HAKIMI) 52
3.1 INTRODUCTION 52
3.2 NORMALIZATION 52
3.3 ENTITY RELATIONSHIP DIAGRAM 55
3.4 DATA DICTIONARY 56
3.5 USER INTERFACE DESIGN 60
4. USER MANUAL (BY HARITH) 75
4.1 INTRODUCTION 75
4.1.1 System Overview 75
4.1.2 System Requirement 76
4.1.2.1 Hardware Requirements 76
4.1.2.2 Software Requirements 76
4.1.3 System Setting 77
4.2 SYSTEM ACCESSING SYSTEM MODULE 78
4.2.1 Welcome Page 78
4.2.2 Registration, Login and Authentication Module (By Harith) 80
4.2.2.1 Admin login 80
4.2.2.2 Wrong Password Or Username for Admin 81
4.2.2.3 Admin Dashboard 82
4.2.2.4 Register Staff 83
4.2.2.5 IC Number Already Exist 84
4.2.2.6 Staff Login 85
4.2.2.7 Wrong Password Or Username for Staff 86
4.2.2.8 Driver Dashboard 87
4.2.2.9 Staff Dashboard 88
4.2.2.10 User Sign Up 89
4.2.2.11 User Login 91
4.2.3 Feedback Module(By Harith) 92
4.2.3.1 View Feedback for Admin 92
4.2.3.2 Reply,Edit or Delete Feedback for Admin 93
4.2.3.3 Give Feedback for User 95
4.2.4 Manage Route Module(By Irfan) 97
4.2.4.1 Add, Update Or Delete Route 97
4.2.5 Manage Reservation Module Admin(By Aesya) 101

4.2.5.1 View Reservation 101
4.2.5.2 Update Reservation Status 104
4.2.6 Report Module(By Haziqah) 106
4.2.6.1 Assign Driver to Bus 106
4.2.7 Manage Profile Module(By Haziqah) 108
4.2.7.1 View Profile 108
4.2.7.2 Edit Profile 111
4.2.8 Manage Bus Module(By Irfan) 114
4.2.8.1 Add, Update Or Delete Bus 114
4.2.9. Manage Reservation Module User(By Aesya) 119
4.2.9.1 Make Reservation 119
4.2.10 Payment Module (By Aesya) 123
4.2.10.1 Make Payment 123
4.2.10.2 View Reservation 125
5. REFERENCES 128
6. MEETING 129
7. APPENDIX 133

1 . SOFTWARE PROJECT MANAGEMENT PLAN (BY AESYA SURAYA)

1.1 INTRODUCTION

1.1.1 Problem Statements

Currently, the traditional manual booking system is still being used at the counter. This method is very time-consuming and error-prone, leading to inefficiency and dissatisfaction among passengers. Limited accessibility results in a lack of options to book tickets online. This also limits passengers to physically go to the ticket counter as well as limiting the accessibility of bus services. Bus service providers will face challenges in optimizing resources, leading to underutilized capacity or overbooking without a centralized system. Insecure payment methods and lack of encryption can pose a risk to users' financial transactions and affect trust in the booking system. This will leave customers with no other choice but to queue up for a long time to secure a bus ticket and they will also need to pay cash when they buy the bus ticket.

1.1.2 Problem Solutions

To address the challenges posed by the traditional manual booking system for bus services, several solutions can be implemented. Firstly, the implementation of an online booking platform can significantly improve efficiency and accessibility. By allowing passengers to book tickets online, it reduces the need for physical presence at ticket counters, saving time and providing more options for customers.

Introducing a centralized booking system can help bus service providers optimize their resources effectively. This system can monitor and manage seat availability, preventing underutilized capacity or overbooking issues. This enhances overall operational efficiency and ensures a better distribution of resources.

To enhance security and trust in the booking system, implementing secure and encrypted payment methods is crucial. This not only safeguards users' financial transactions but also establishes a secure environment for online bookings. Incorporating modern cybersecurity measures will go a long way in building confidence among customers, encouraging them to use online services without concerns about their personal and financial information.

Furthermore, promoting cashless transactions for online bookings can streamline the process and reduce the need for customers to queue up for extended periods. By offering various payment options, including credit/debit cards and digital wallets, the booking system becomes more convenient and aligned with contemporary financial practices.

In conclusion, transitioning from a manual to a digital booking system with enhanced security measures and payment options can address the inefficiencies and challenges associated with the traditional bus ticketing process. This not only improves the overall customer experience but also ensures the optimal utilization of resources for bus service providers.

1.1.3 Objective of the System

1. To start allowing customers to book bus tickets online without having to line up and wait in long lines again.
2. To start providing real-time updates on bus schedules, bus routes, and seat availability to customers to ensure they are well informed throughout their journey.
3. To start implementing a secure and convenient payment gateway to facilitate online transactions.
4. To start enabling customers to cancel their reservations.
5. To start granting admin user privileges to update and edit bus routes.

1.1.4 Scope of the System

This system involves three actors which are customers, staff, admin and bus driver.

The scope of this system is:

- Customer Module
    - Customer can register
    - Customer can book their tickets
    - Customer can select their preferred trips and seats
    - Customer can make payments
    - Customer can view their booking status and history
    - Customer can view booking receipts
    - Customer can view their feedback
    - Customer can add feedback
    - Customer can view replied feedback from admin
    - Customer can view their profiles
    - Customer can update their profiles
- Admin Module
    - Admin can register Staff and Bus Driver
    - Admin can delete Staff and Bus Driver
    - Admin can add new routes
    - Admin can view routes
    - Admin can update routes
    - Admin can delete routes
    - Admin can update booking's status
    - Admin can view booking history
    - Admin can delete booking
    - Admin can view feedback from customer
    - Admin can reply customer's feedbacks
    - Admin can delete customer's feedbacks
    - Admin can view their profiles

    - Admin can update their profiles
- Staff Module
    - Staff can add new bus
    - Staff can update bus information
    - Staff can delete bus
    - Staff can view their profiles
    - Staff can update their profiles
- Bus Driver Module
    - Bus Driver can view their schedule
    - Bus Driver can view their profiles
    - Bus Driver can update their profiles

1.2 LITERATURE REVIEWS

1.2.1 Introduction

The literature review presents the foundational theories and existing documentation relevant to the Bus Ticket Reservation System being developed by Tiny Travel Company. It evaluates and contrasts the conceptual frameworks of the current systems, emphasizing their advantages and disadvantages, and offers suggestions for how the new system may resolve these problems and enhance the overall bus ticket buying process.

1.2.2 Review of the Existing System

RedBus is now a popular online bus ticket buying service that provides a convenient interface for route searches, seat availability checks, and reservation bookings. Even if it incorporates a variety of payment methods and is convenient, there are a few areas that still require work to satisfy changing customer demands.

One significant drawback is the occasional disruption of the booking process caused by technological problems and downtime. Passengers are left in a state of confusion by the platform's lack of real-time data on bus timetables and delays. Furthermore, one feature that might improve the user experience is that the system does not provide customised recommendations based on user preferences**.**

Even though it is provided, customer service is frequently sluggish, which irritates users. For inexperienced users, navigating the UI might be challenging due to its clutter. Additionally, there are times when integration with the systems of bus operators is not flawless, leading to disparities in reservation details and seat availability.

In conclusion, even if RedBus works well, additional dependability, real-time updates, personalisation, customer service, UI design, and system integration are necessary for a more successful and approachable platform. Resolving these problems will improve RedBus's overall user experience and operational effectiveness.

1.2.3 Discussion

It is clear from analyzing current systems that a completely automated and integrated bus ticket reservation system is essential. The Bus Ticket Reservation System from Tiny Travel Company is designed to solve the weaknesses of the current systems by providing a full online booking platform with real-time updates, secure payment options, and a user-friendly interface.

This system will ensure that users have correct information when making reservations and will prevent overbooking by providing real-time updates on seat availability. The solution will improve the security of payments. Users' experience will be improved overall by a user-friendly interface that makes it simple for them to browse for and reserve tickets. Future growth in the system is supported by its architecture, which makes it possible to add more buses, routes without having to completely redesign everything. Enhancing resource management using a single booking system will reduce capacity that has not been being used and increase operational effectiveness. These features guarantee users an excellent booking experience.

1.3 METHODOLOGY

1.3.1 Introduction

The methodology section describes the approach and processes used by Tiny Travel Company to develop the Bus Ticket Reservation System. This involves deciding on an appropriate model for software development as well as the technologies that will ensure the system's effective implementation. By offering an organized framework that encourages iterative creation, ongoing feedback, and continual improvement, the selected methodology seeks to ensure that the finished product meets the requirements and expectations of all stakeholders.

1.3.2 Software Model

The Bus Ticket Reservation System project uses the Agile methodology for its software development. Agile, which emphasizes incremental and iterative development, is a well-known methodology in the software development life cycle. This approach is perfect for projects whose needs may change over time since it places a strong emphasis on adaptability, teamwork, and customer input.

The Agile technique divides the project into smaller, more manageable iterations known as sprints, which guarantees an orderly and disciplined development process. Usually lasting two to four weeks, each sprint is dedicated to delivering a functional increment of the product. Continuous supply and quick reaction to shifting requirements are made possible by this method.

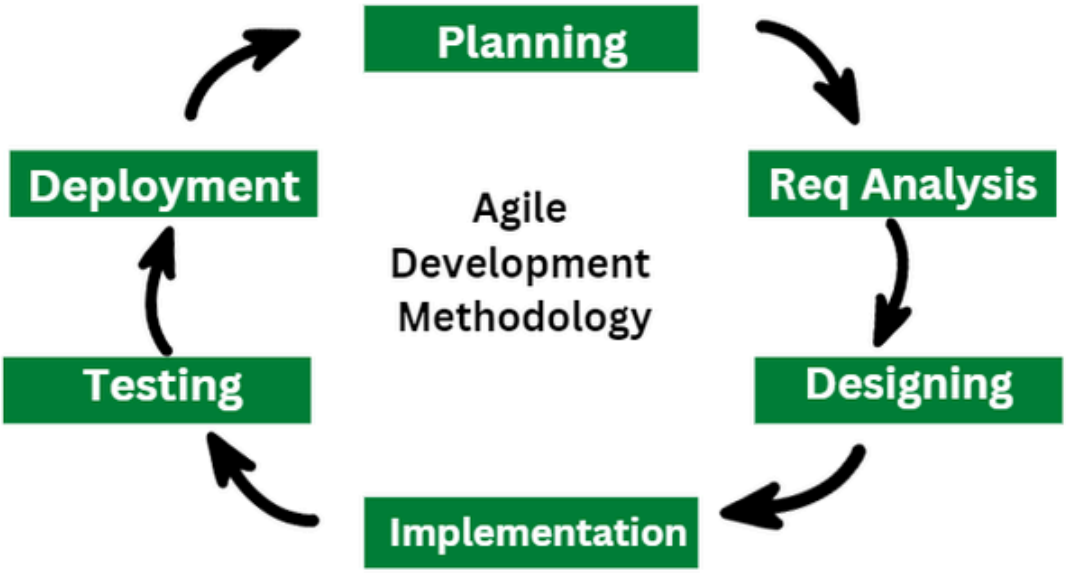


Diagram 1.3.2.1 Shows Life cycle of Agile Methodology

a. Requirements Analysis and Gathering

During this phase, stakeholders, such as clients are regularly engaged in order to obtain comprehensive information about the project's requirements. This stage makes sure that the project specifications are understood and specified precisely. The development process is guided by the creation of user stories and backlog items.

b. Planning and Design

The system architecture, user interface designs, and database schema are produced during the design and planning stage. Ensuring an engaging and user-friendly interface while ensuring that the system's functionalities are clear is the aim. To organize and prioritize the work for each sprint, planning meetings are held.

c. Implementation

The project is split up into sprints during the implementation process, with each sprint focusing on creating a particular set of features and functions. Collaboratively, cross-functional teams execute the user stories and backlog items according to the priorities established in the planning sessions. The goal of continuous integration is to test changes in a common environment and integrate them often.

d. Testing

Each sprint in the Agile approach includes testing as a necessary component. Every release of an increment is made sure to be completely tested by continuous testing. User acceptability testing, integration testing, and unit testing are all included in this. Early on in the development process, defects are found and addressed to ensure high standards.

e. Deployment

During the deployment phase, the system is gradually released, with each version being tested for stability and functionality. In order to enhance the system in later versions, user feedback is gathered. This stage makes sure that the system satisfies user demands and is prepared for production usage.

f. Maintenance

Continuous support is provided during the maintenance phase to guarantee that the system continues to be secure, operational, and compliant with changing needs. This entails resolving any problems that crop up and putting improvements into action in response to user input and new specifications.

1.3.3 Software Architecture

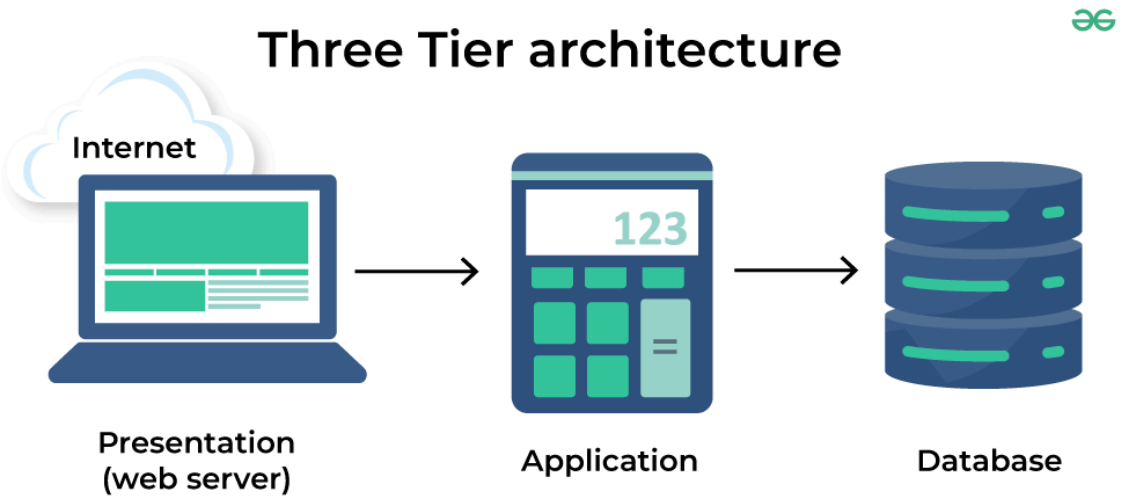


Diagram 1.3.3.1 shows Three-Tier Architecture

The 3-tier architecture, which consists of the presentation tier, application tier, and data tier as its three logical processing levels, is used by the Bus Ticket Reservation System. The user interface, business logic, and data storage layers are divided in this client-server design, which has several benefits for both development and production contexts. This design improves scalability, maintainability, and overall system effectiveness by giving development teams the ability to update individual application components independently of the others.

The Bus Ticket Reservation System's user interface is part of the presentation tier, sometimes referred to as the front-end layer. It will be made to be accessed through a web browser or web application, providing users with a graphical user interface for system interaction. By displaying relevant data and content, this layer will make the system accessible and easy to use. The technologies like HTML5, JavaScript, CSS, and front-end frameworks like React.js are used to build an interactive and responsive display layer. Through API calls, this tier will interface with the other levels of the system to facilitate data interchange and smooth integration.

The operational business logic of the system is handled by the application tier, forming its core. It includes all of the primary features of the Bus Ticket Reservation System, such as payment processing, routing management, user authentication, and booking administration. This layer will include strong and scalable RESTful web services built using Java framework. The application layer serves as a bridge between the presentation and data tiers, ensuring system performance and integrity by managing all processing.

Every relevant information is stored and accessed by the system on the data layer, also known as the database tier. This contains details about user profiles, bus routes, reservations, payments, and reviews. The database and the data access layer, which enable effective data retrieval and manipulation, form the data tier. The relational database management system of choice is MySQL. Data security, integrity, and prompt access to information are guaranteed by this configuration.

There are several benefits to using the 3-tier architecture for the Bus Ticket Reservation System. By modularizing the system, it makes development more flexible and enables independent part updates. By clearly separating responsibilities, this design improves overall effectiveness, scalability, and maintainability. It also increases performance and facilitates debugging because modifications made to one layer do not always affect the others.

1.3.4 Project Management Plan

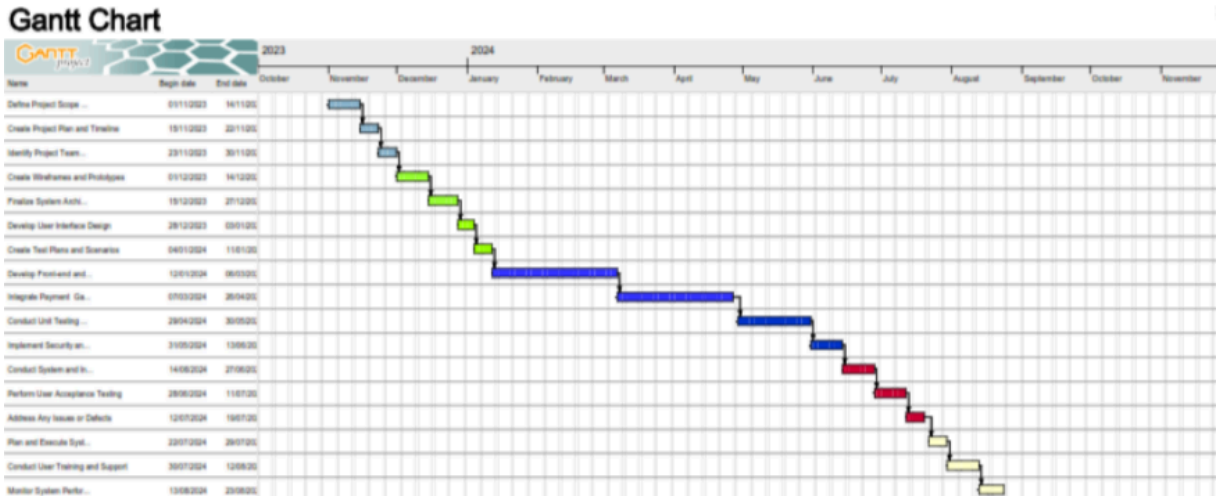


Diagram 1.3.4.1 shows Project Scheduling

The project schedule for the Bus Ticket Reservation System spans from November 2023 to October 2024 and includes key phases such as defining the project scope, creating a detailed project plan, and identifying the project team. Initial phases include creating wireframes and prototypes, followed by finalizing system requirements and developing the user interface design. The project then moves into creating test plans, integrating the payment gateway, and system implementation. Subsequent phases involve system testing, user acceptance testing, addressing any issues or defects, and conducting user training and support. Finally, the project will be reviewed and monitored to ensure it meets all objectives and requirements, with the system ready for deployment by the end of October 2024.

1.4 EXPECTED OUTCOMES

After completion, the current manual booking procedures will be replaced by Tiny Travel Company's Bus Ticket Reservation System. With the ease of online booking, real-time updates, and secure payment methods, this new technology will speed up the process of reserving tickets.

The system is expected to improve operating efficiency, reduce overbooking, and optimize resource management for bus service providers. All things considered, the system will offer a cutting-edge, effective, and dependable solution, significantly improving the booking experience for users and enabling the development of bus services.

1.5 ENTREPRENEURSHIP

1.5.1 Qualification

A highly qualified team developed the Bus Ticket Reservation System with remarkable knowledge and commitment. By employing an effective and flexible three-tier design, the system ensures maximum efficiency and maintenance. The user interface, application, and data layers are successfully divided by this structure, which improves flexibility and maintenance simplicity.

The development team includes professionals with extensive experience in Java, JavaScript, and web development frameworks, known for delivering high-quality, user-friendly applications.

Agile methodology expertise guarantees on-time project completion while upholding the highest standards of quality. The system has extensive features, including secure payment processing, real-time seat availability, and user-friendly interfaces, all of which are intended to give clients a seamless and dependable booking experience.

1.5.2 Pricing options

The Bus Ticket Reservation System offers flexible pricing options to cater to various client needs, ensuring both ownership and comprehensive support services.

- Complete System Package:
    - Includes all features and integrations with hosting services.
    - Database provision included.
    - Price: RM35,000
- Monthly Maintenance:
    - Regular system updates, bug fixes, and essential support services.
    - Price: RM2,000 per month
- Bi-Annual Comprehensive Maintenance:
    - All standard maintenance services, priority support, custom feature requests, and system health reviews.
    - Price: RM10,000 every six months

1.5.3 Clarify terms and conditions

a) Data Security and Privacy:

The Bus Ticket Reservation System places the highest priority on protecting client data. To prevent unauthorized access, data will be stored securely using innovative encryption methods together with additional strong security measures. In addition, a strict privacy policy is implemented, ensuring that clients' information cannot be disclosed to third parties without prior authorization. This ensures that private and sensitive data is kept private and utilized only for whatever it was intended.

b) Payment Terms:

The Bus Ticket Reservation System's payment conditions are intended to be simple and adaptable. Clients that purchase a monthly maintenance subscription, which offers regular updates and crucial support services, will be invoiced at the start of each payment cycle. The convenience of transaction is ensured by the acceptance of many payment methods, such as bank transfer and credit card. By using a structured invoicing approach, clients receive constant service and support, which preserves the dependability and functioning of the system.

c) Cancellation Policy:

Clients are free to terminate their subscription whenever they choose. Clients will still be able to use the system after cancellation, but maintenance services will no longer be provided. This policy achieves a compromise between service continuance and client autonomy by allowing customers to keep using the system while choosing not to receive ongoing support services.

d) Licensing and Usage:

Clients are granted a non-exclusive, non-transferable license to use the Bus Ticket Reservation System under the licensing conditions. It also means that although users may use the system for their operations, they are not permitted to sublicense, sell, or distribute it without the provider's prior written authorization. This ensures the system's independence while enabling clients to utilize its features according to the terms and conditions set out.

e) Modifications and Updates:

Subscription clients will receive free upgrades and new features on a regular

basis, guaranteeing the system's continued effectiveness and relevance. This includes additions that bring new features and boost system efficiency. However, any customized changes that the customer requests come with extra fees in keeping with the proposal that was agreed upon. This strategy guarantees that certain customisations are handled independently to satisfy particular client demands, even while the system maintains current for all users.

1.6 CONCLUSION

In conclusion, the Bus Ticket Reservation System from Tiny Travel Company is a robust solution designed to enhance and streamline the bus ticket booking process. Inspired by successful businesses like Sani Express Sdn Bhd, the system mission is to provide convenient, efficient, and comfortable travel experiences while embracing technology and fostering a culture of excellence.

The system is designed to exceed customer expectations with a vision to become the preferred choice for travelers seeking reliable and innovative transportation solutions. Its objectives include automating the manual procedures of bus ticket reservation, providing real-time updates, implementing secure payment gateways, and offering flexibility in reservation management.

The system's scope covers a user-friendly online platform, real-time information availability, secure payment processing, reservation management, admin privileges, scalability, and customer-centric features. It aspires to create a strong online presence that aligns with Tiny Travel Company's brand objectives, making it a leader in the industry driven by a commitment to quality service, customer satisfaction, and sustainable business practices.

2. SOFTWARE PROJECT SPECIFICATION (BY HAZIQAH)

2.1 DEVELOPMENT REQUIREMENT

2.1.1 Hardware Specification

1. Servers:
    - Processor: Intel Xeon or AMD EPYC
    - RAM: 32 GB or higher
    - Storage: SSDs with at least 1 TB of capacity
    - Network: Gigabit Ethernet adapters
    - Operating System: Linux (Ubuntu, CentOS) or Windows Server
2. Client Devices:
    - Desktop/Laptops:
        - Processor: Intel Core i5 or higher
        - RAM: 8 GB or higher
        - Storage: 256 GB SSD or higher
        - Network: Wi-Fi or Ethernet capability
    - Tablets/Smartphones:
        - Processor: Quad-core or higher
        - RAM: 2 GB or higher
        - Storage: 32 GB or higher
        - Network: Wi-Fi/4G/5G capability

2.1.2 Software Specification

1. Server Software:
    - Operating System: Windows Server
    - Web Server: Apache HTTP Server
    - Database Server: SQL Server
    - Application Server: Tomcat

2. Client Software:
    - Operating System: Windows, macOS, Linux, iOS, or Android
    - Web Browser: Chrome, Firefox, Safari, or Edge (latest versions)
3. Application Software:
    - Programming Languages: Java
    - Mobile Application Development: Native iOS/Android development
4. Development Tools:
    - Code Editors/IDEs: Visual Studio Code, Apache NetBeans IDE

2.2 REQUIREMENTS

2.2.1 Requirements Gathering Technique

To gather a comprehensive and clear set of requirements for the Bus Ticket Reservation System, a combination of effective elicitation techniques is utilized.

One-on-one interviews are conducted with key stakeholders who hold significant influence over the system, including representatives from bus companies, and system administrators. These interviews aim to ensure the completeness and clarity of the requirements through in-depth discussions and direct clarification with stakeholders. This technique focuses on identifying conscious requirements specific to the bus ticket reservation process. Surveys are distributed to a diverse group of stakeholders such as passengers. These surveys aim to collect both quantitative and qualitative data, opinions, and preferences related to the bus ticket reservation system. The gathered information is then statistically analyzed to derive insights and formulate new requirements. Surveys provide a broader perspective on system requirements and allow for anonymous feedback, including insights on the likelihood of users recommending the system to others.

By employing these techniques, a comprehensive and well-rounded set of requirements can be gathered, ensuring the Bus Ticket Reservation System meets stakeholder expectations and delivers superior user experience.

NO.	REQUIREMENT DESCRIPTION	REQUESTED BY	PRIORITY
1	• The system shall allow employers and job applicants to register as a user • The system shall allow users to have the ability to manage and update their profiles, including contact information and seating preferences.	Dahlia (Operator of Sani Express)	High
2	• The system shall allow all users to log in or log out from the system • The system should allow users to search for available routes based on preferred departure and destination cities, dates, and times. • Users should be able to book tickets for selected routes, specifying seating preferences.	Syazwani (Operator of Sani Express)	High

3	• The system should support multiple payment methods, including credit cards, debit cards, and cash. • Users should receive a clear and detailed receipt after successful payment	Naqiyah (Passenger)	High
4	• The system should support multiple payment methods, including credit cards, debit cards, and cash. • Users should receive a clear and detailed receipt after successful payment.	Shafiqah (Passengers)	Moderate
5	• The user interface should be intuitive and easy to navigate to ensure a positive user experience. • User training requirements should be minimal, with clear instructions provided where necessary.	Maheer (Passenger)	Low

Table 1 Requirement Description by requested by the stakeholders

2.2.2 Functional Requirements

Functional requirements specify the precise tasks carried out by the system and the data that these tasks operate on. The scenarios that show an operational system from the viewpoint of its end users also include the functional requirements. All the system's features are shown in one or more ways, and all the requirements related to each feature are listed.

- The system shall provide a mechanism for user authentication
- The system shall evaluate and verify all user input
- The system shall allow user to make payment
- The system shall let user to book bus ticket reservations
- The system shall manage different routes for the customers
- The system shall allow operators to view the list of reservation made by customer
- The system shall have a user-friendly mobile app that mirrors the functionality of the web interface.

2.2.3 Non-Functional Requirements

Non-functional requirements deal with features of the system that are not related to the tasks it completes. These factors include system performance, expenses, and general attributes like portability, security, and dependability. Aspects of operational staff and the system development process are also covered by the non-functional requirements. It consists of the subsequent items:

- The system shall be consistent and easy to use
- The system shall target a certain client base
- The system shall offer an appealing graphical user interface
- The system shall allow developers to access the installed environment

2.3 SYSTEM MODELING

2.3.1 Business Process and Functional Modeling

2.3.1.1 Use Case Diagram

Use-case diagram model is the behavior of a system and helps to capture the requirements of the system. These diagrams also identify the interactions between the system and its actors.

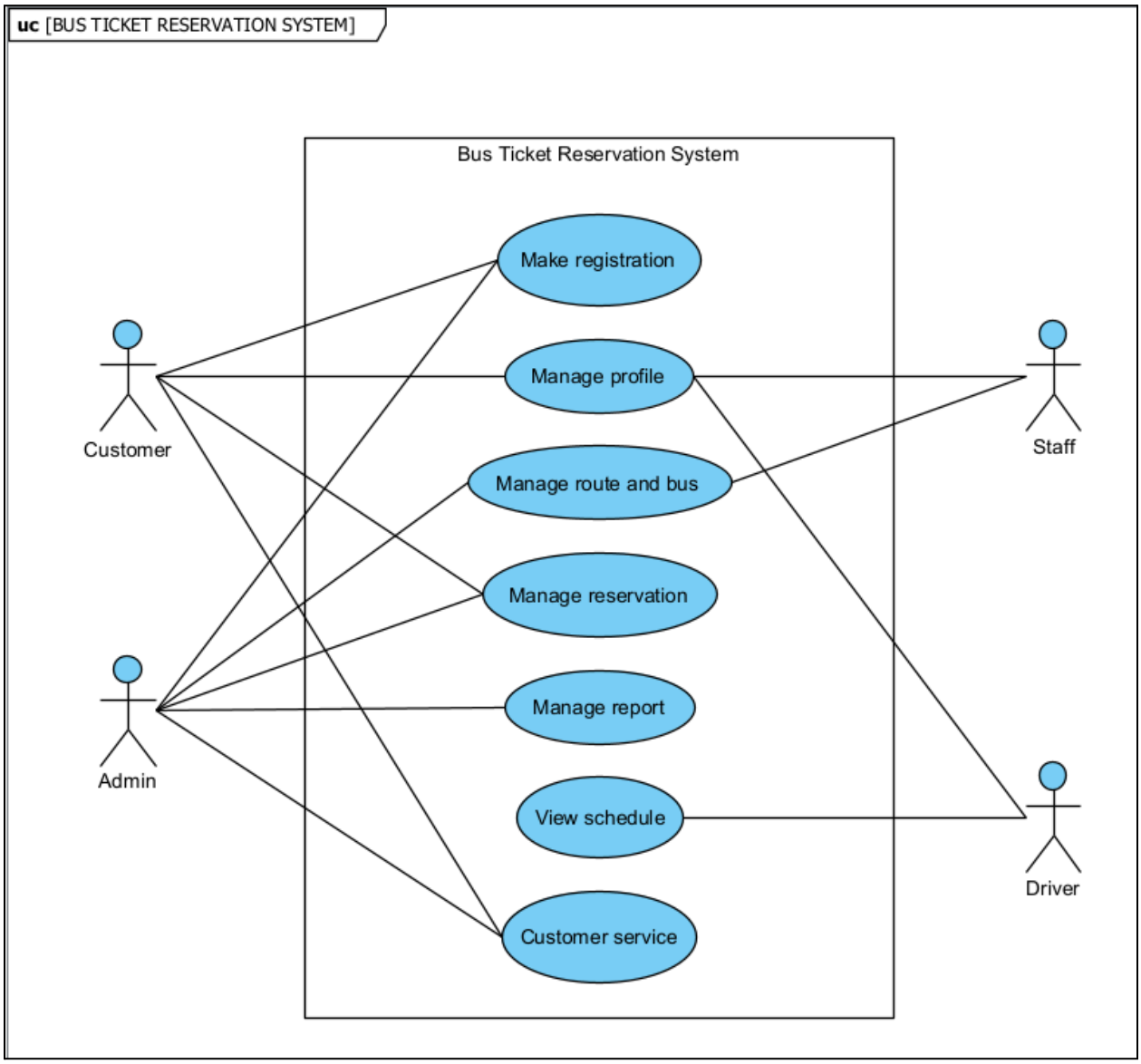


Diagram 2.3.1.1.1 shows a use case diagram for Bus Ticket Reservation System

2.3.1.2 Use Case Description

Use Case	Actors	Description
Make Registration	Admin, Customer	1. Customer opens the registration page. 2. Customer fills in the required details (name, email, password, ic number.). 3. Customer submits the registration form. 4. System validates the information. 5. System creates a new account for the customer. 6. Customer receives a confirmation message. 7. Admin open the registration page. 8. Admin fills in the required details of staff and driver (name, email, password.). 9. Admin submits the registration form. 10. System validates the information. 11. System creates a new account for the staff and driver. 12. Staff and Driver receives a confirmation message.
Manage Profile	Customer, Driver, Staff	1. Customer/Driver/Staff opens the profile management page. 2. Customer/Driver/Staff views or edits the profile details. 3. Customer/Driver/Staff submits the changes. 4. System validates the updated information. 5. System updates the profile in the database. 6. Customer/Driver/Staff receives a confirmation message.

Manage Route and Bus	Staff, Admin	1. Staff/Admin opens the route and bus management page. 2. Staff/Admin views, adds, updates, or deletes route and bus details. 3. Staff/Admin submits the changes. 4. System validates the information. 5. System updates the route and bus details in the database. 6. Staff/Admin receives a confirmation message.
Manage Resevation	Customer, Admin	1. Customer/Admin opens the reservation management page. 2. Customer makes a new reservation or views/cancels an existing reservation. 3. Admin manages reservations. 4. Customer/Admin submits the changes. 5. System validates the information. 6. System updates the reservation details in the database. 7. Customer/Admin receives a confirmation message.
Manage Report	Admin	1. Admin opens the report management page. 2. Admin assign driver to the bus. 3. Admin submits the changes. 4. System validates the information. 5. System updates the assigned driver report in the table. 6. Admin views the report.

View Schedule	Driver	1. Driver opens the schedule viewing page. 2. System retrieves and displays the schedule information. 3. Driver views the schedule.
Customer Service	Customer, Admin	1. Customer opens the customer service page. 2. Customer enters the feedback. 3. Customer submits the feedback. 4. System validates the information. 5. Admin views and reply the feedback. 6. Admin submits the reply feedback. 7. System validates the information. 8. Admin/Customer receives a notification message.

Table 2 shows a use case description for Bus Ticket Reservation System

2.3.1.3 Activity Diagram

An activity diagram is basically a flowchart to represent the flow from one activity to another activity, and that activity can be described as an operation of the system.

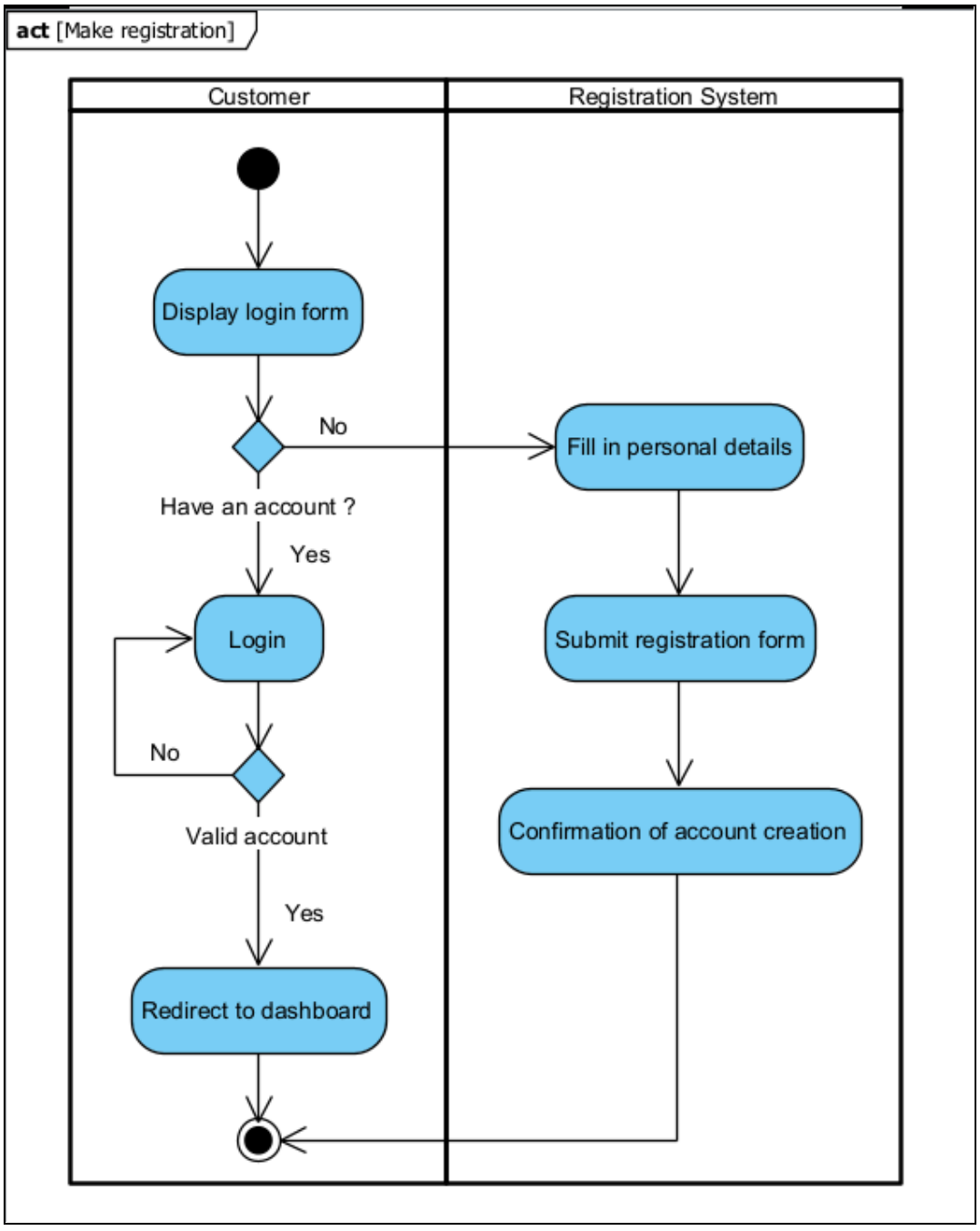


Diagram 2.3.1.3.1 shows an activity diagram of Customer make a registration

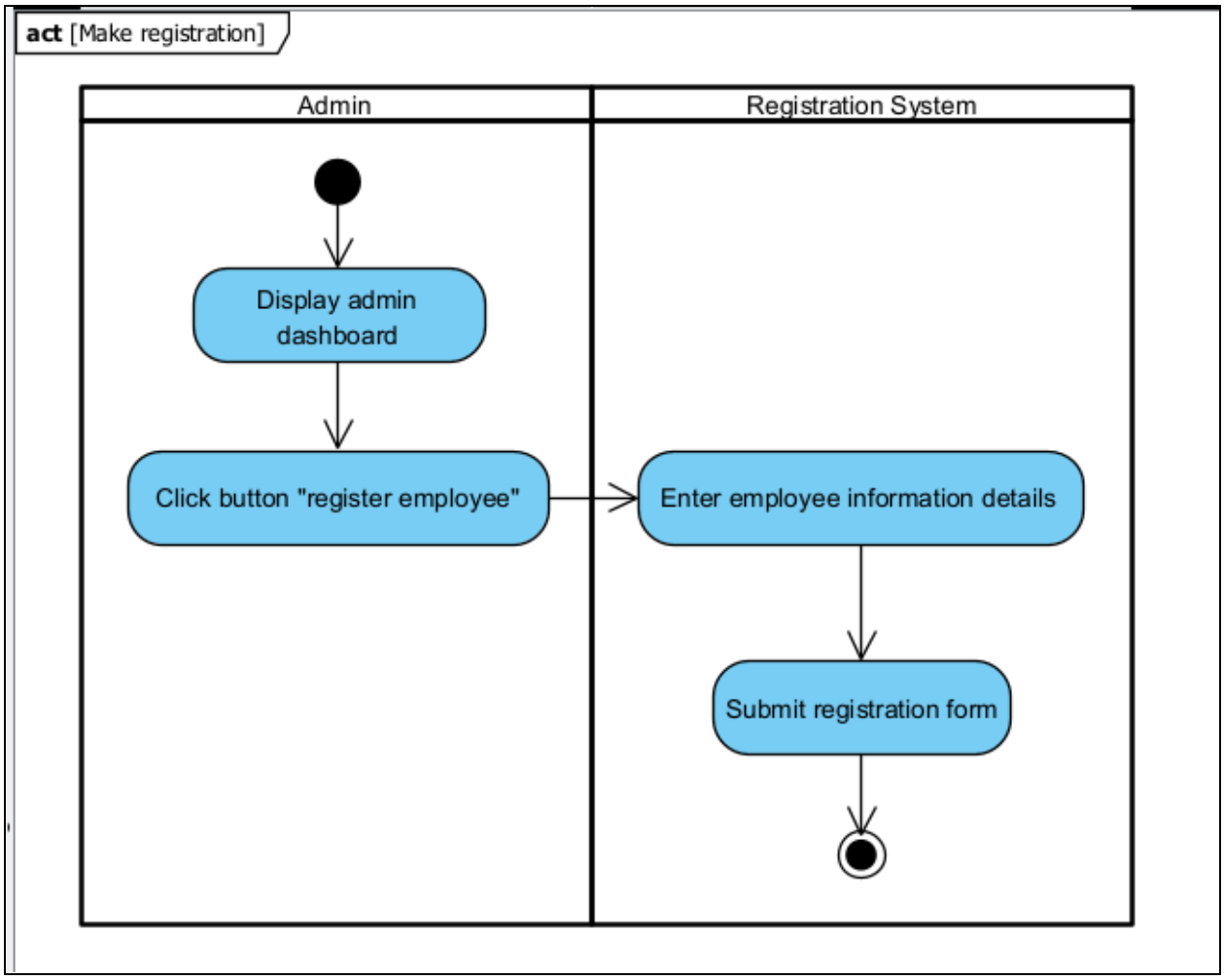


Diagram 2.3.1.3.2 shows an activity diagram of Admin make a registration for their employees

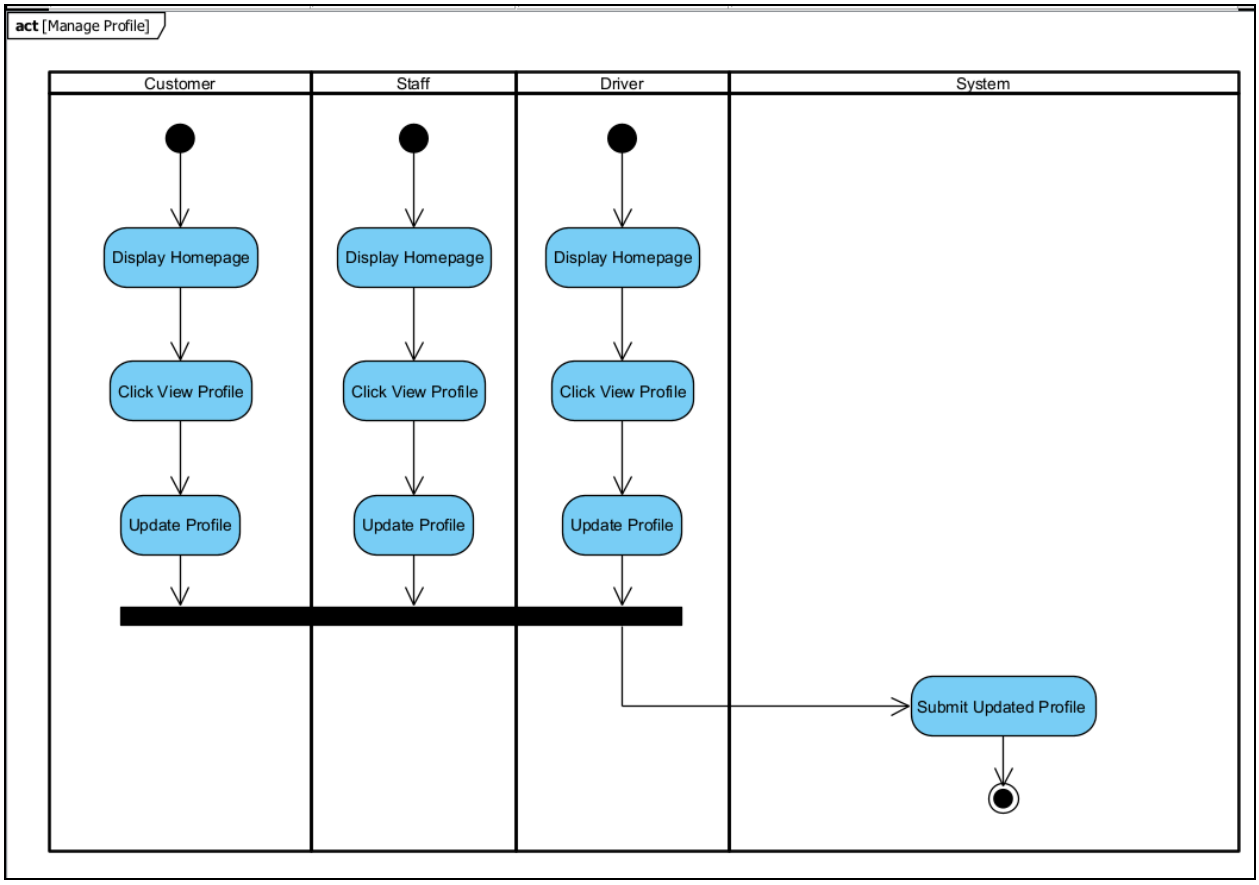


Diagram 2.3.1.3.3 shows an activity diagram of Customer, Staff and Driver manage their profile

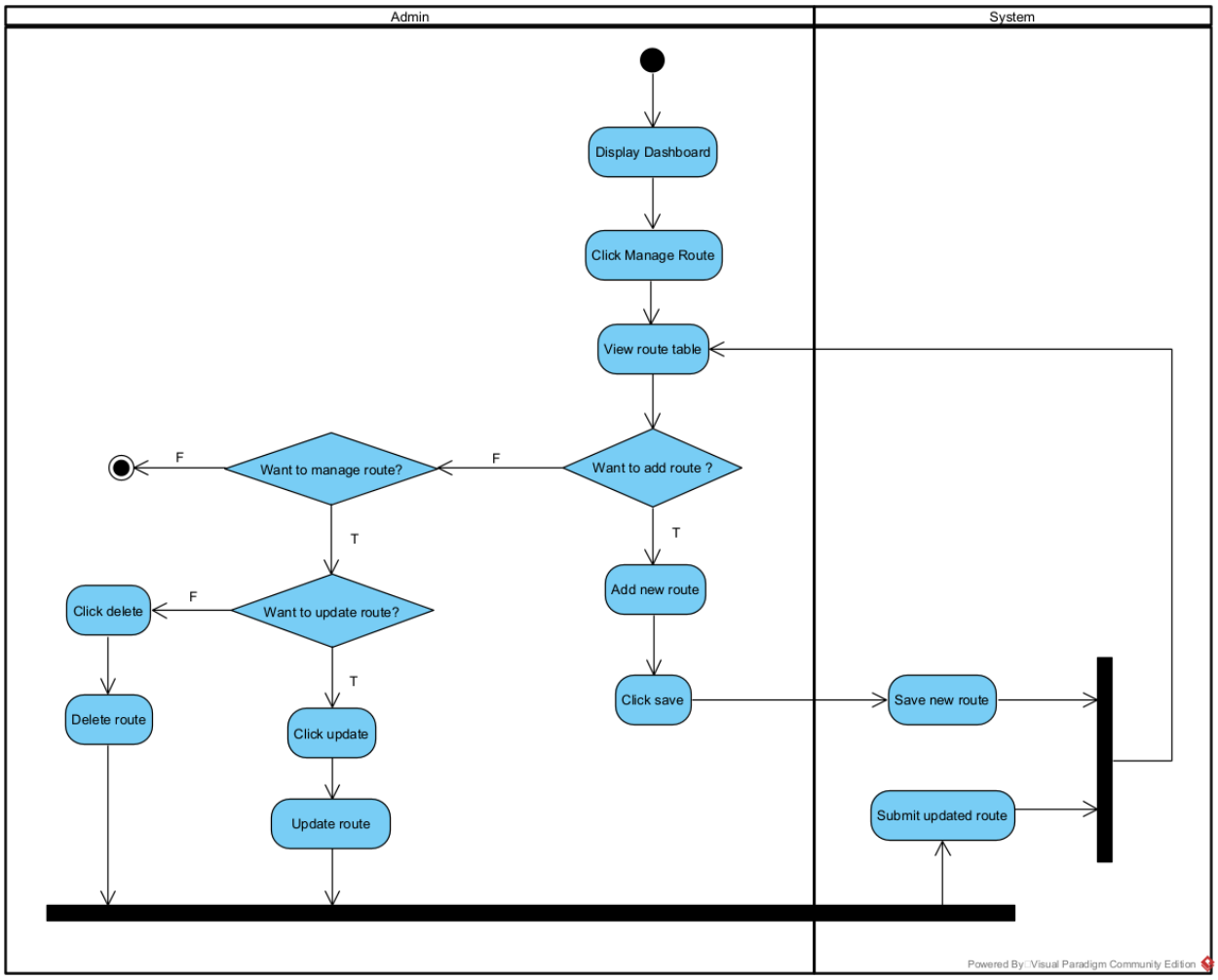


Diagram 2.3.1.3.4 shows an activity diagram of Admin manage the route

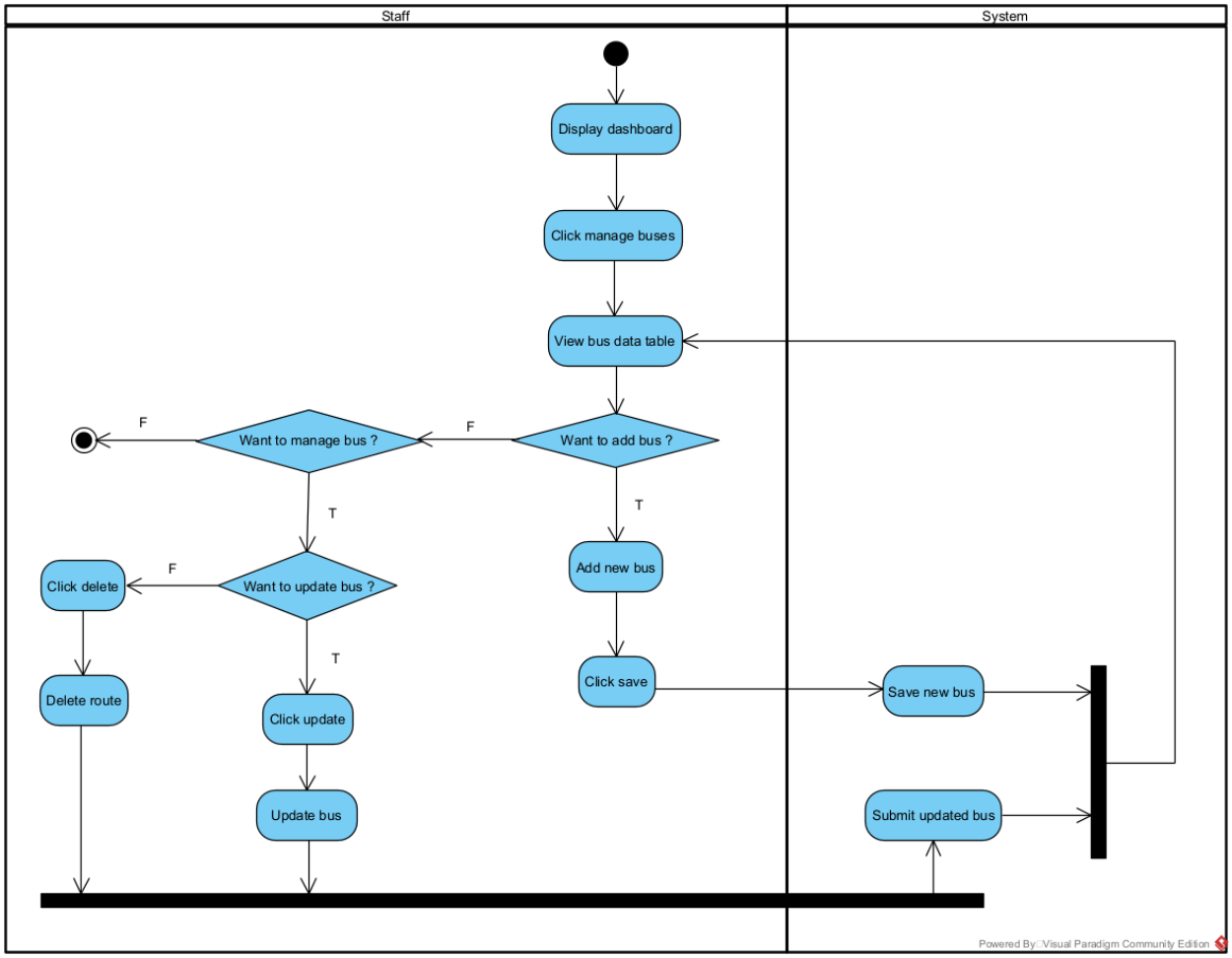


Diagram 2.3.1.3.5 shows an activity diagram of Staff manage the bus

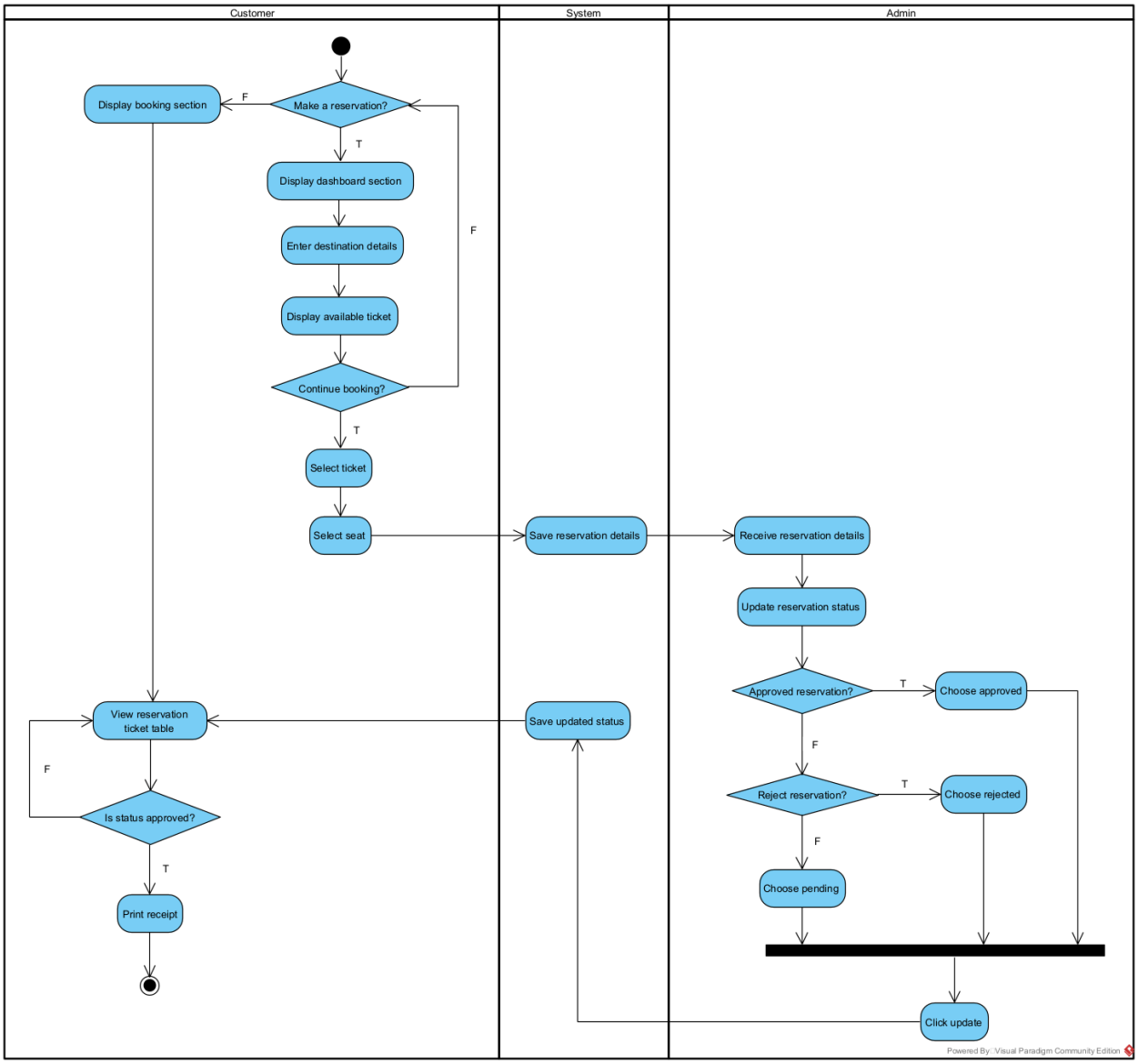


Diagram 2.3.1.3.6 shows an activity diagram of Customer and Admin manage the reservation

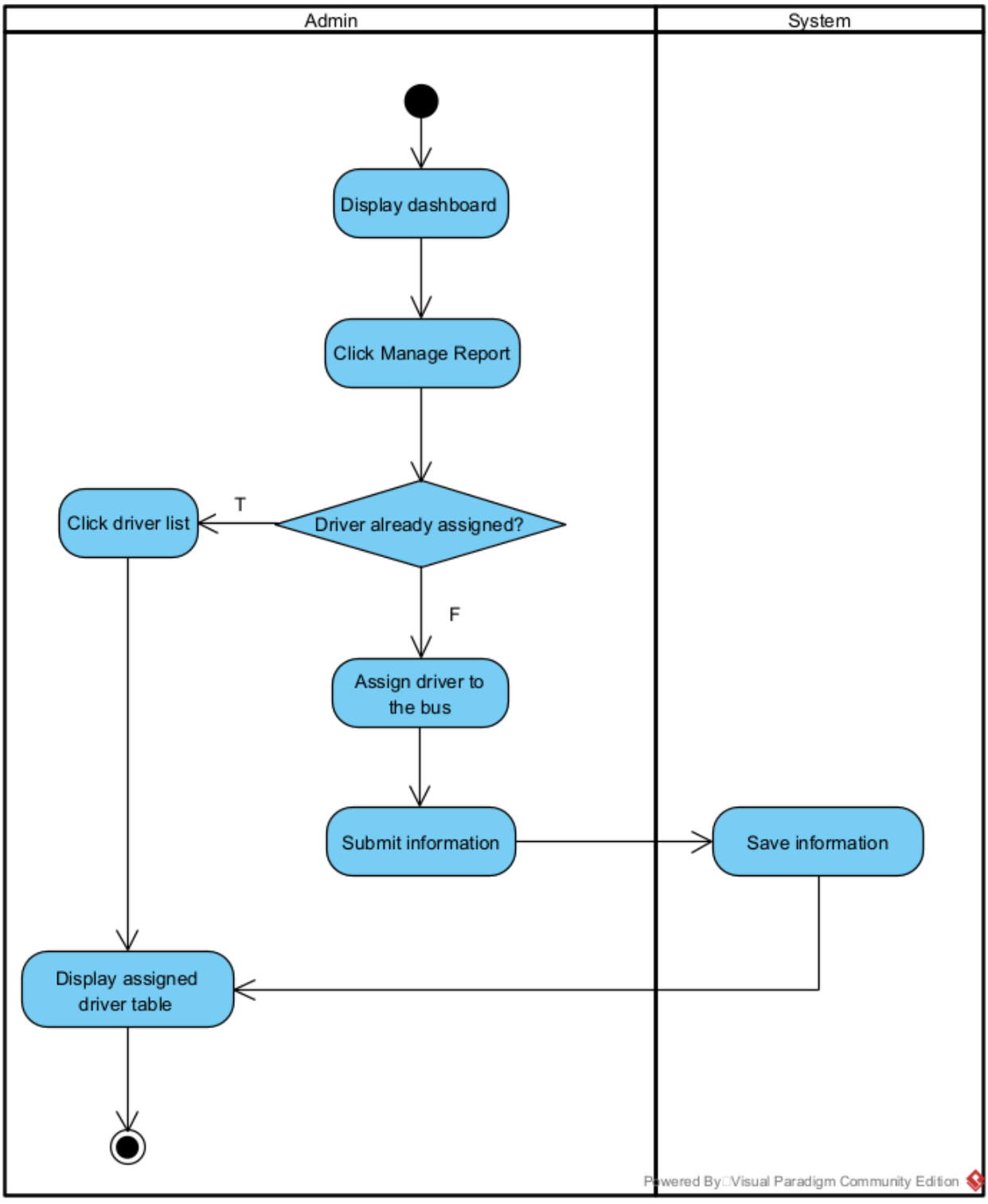


Diagram 2.3.1.3.7 shows an activity diagram of Admin manage the report

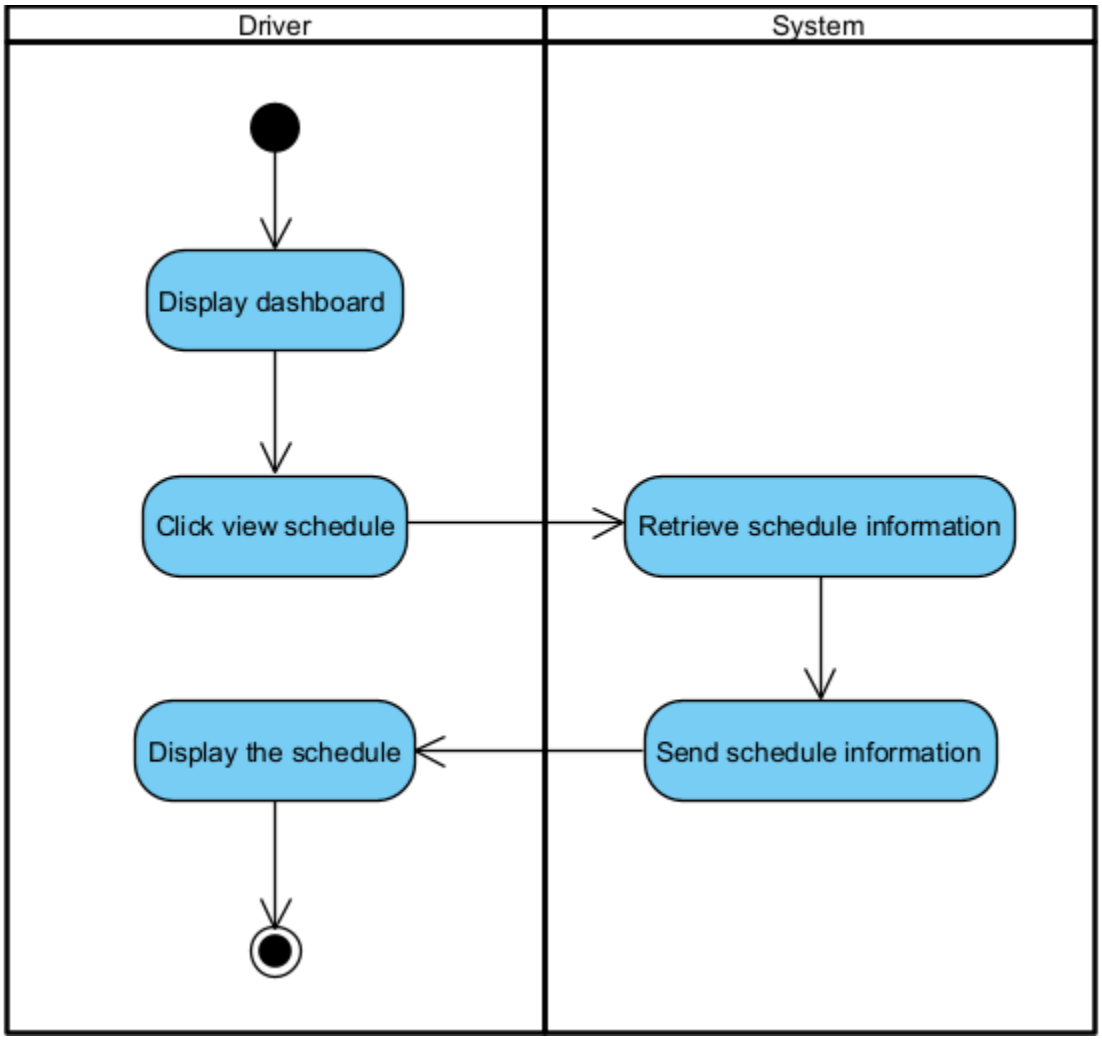


Diagram 2.3.1.3.8 shows an activity diagram of Driver view their schedule

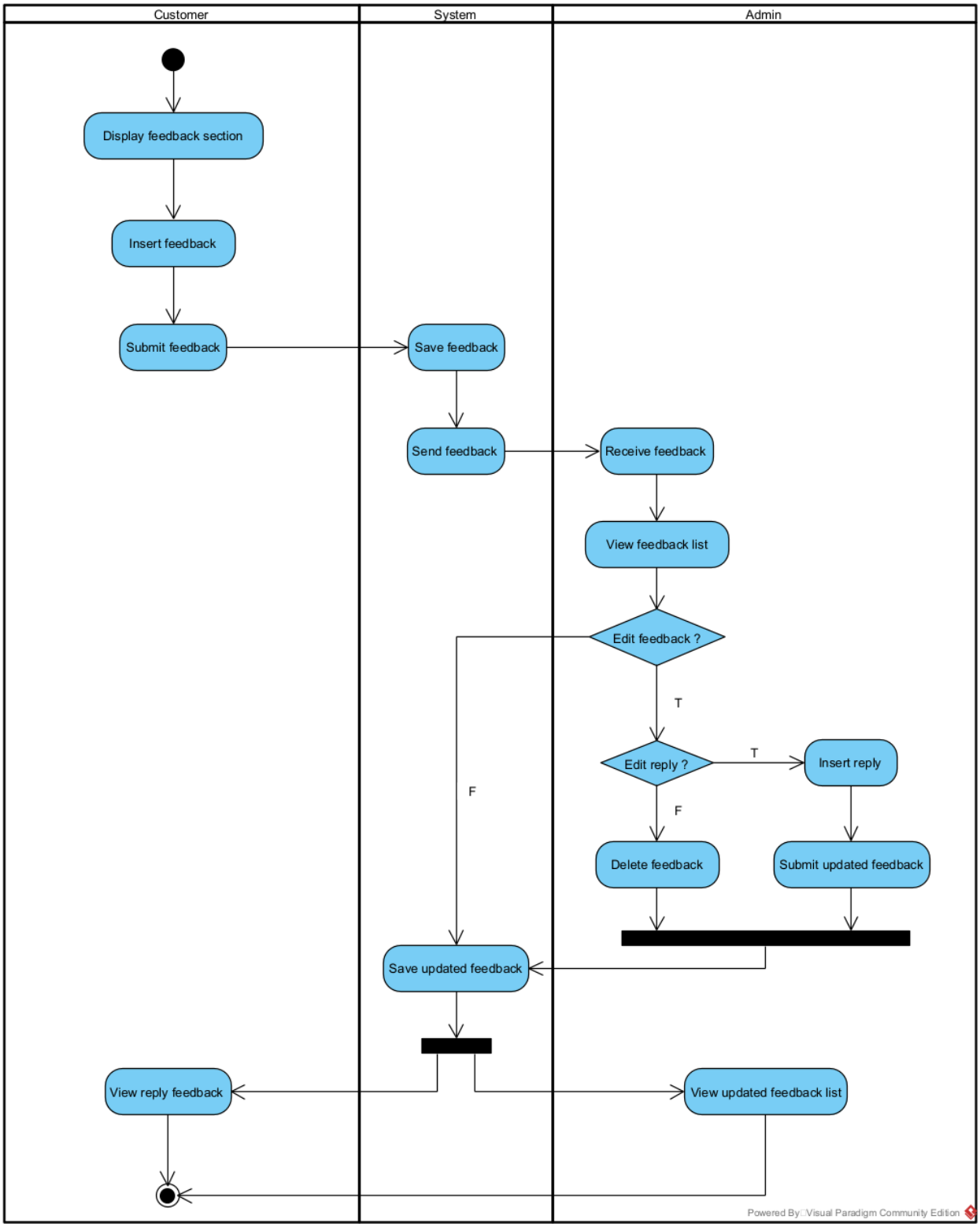


Diagram 2.3.1.3.9 shows an activity diagram for customer service of Customer and Admin

2.3.2 Structural Modeling

Structural models of software display the organization of a system in terms of the components that make up that system and their relationships.

2.3.2.1 CRUD Diagram

The CRUD (Create, Read, Update, Delete) diagram helps to understand which actors can perform which operations on the system entities.

Entity	Customer	Staff	Admin	Driver
Customer Profile	C, R, U, D		R, D	
Staff Profile		R, U, D	C, R, D	
Driver Profile			C, R, D	R, U, D
Admin Profile			C, R, U, D	
Route and Bus	R	C, R, U, D	C, R, U, D	
Reservation	C, R, D	R	R, U, D	R
Report			C, R, U, D	R
Schedule				R

Customer Service	C, R, U, D		C, R, U, D	

Diagram 2.2.2.1.1 shows a CRUD Diagram for Bus Ticket Reservation System

2.3.2.2 Class Diagram

The class diagram is a static view of an application. It is used to visualize, describe, document various different aspects of the system, and also construct executable software code.

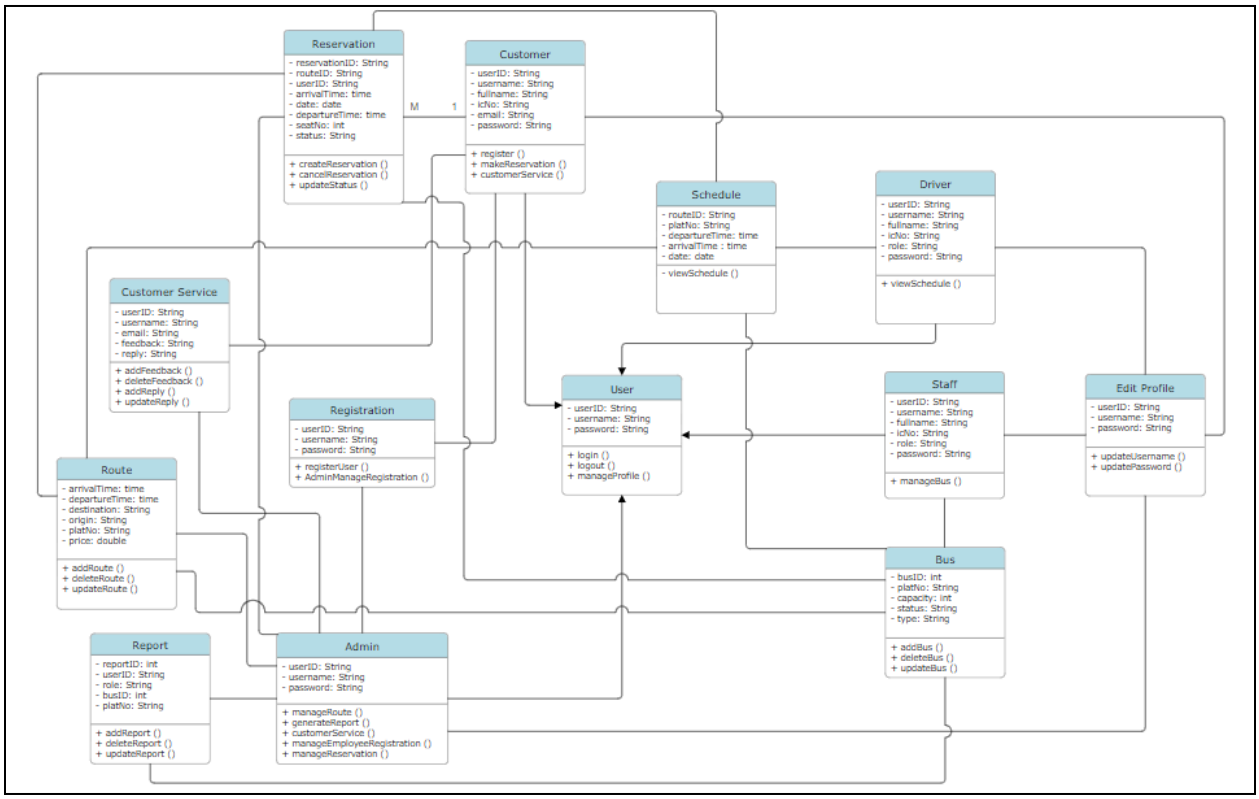


Diagram 2.2.2.2.1 shows a class diagram for Bus Ticket Reservation System

2.3.3 Behavioral Modeling

2.3.3.1 Sequence Diagram

The sequence diagram is to shows the sequence of messages that pass between objects

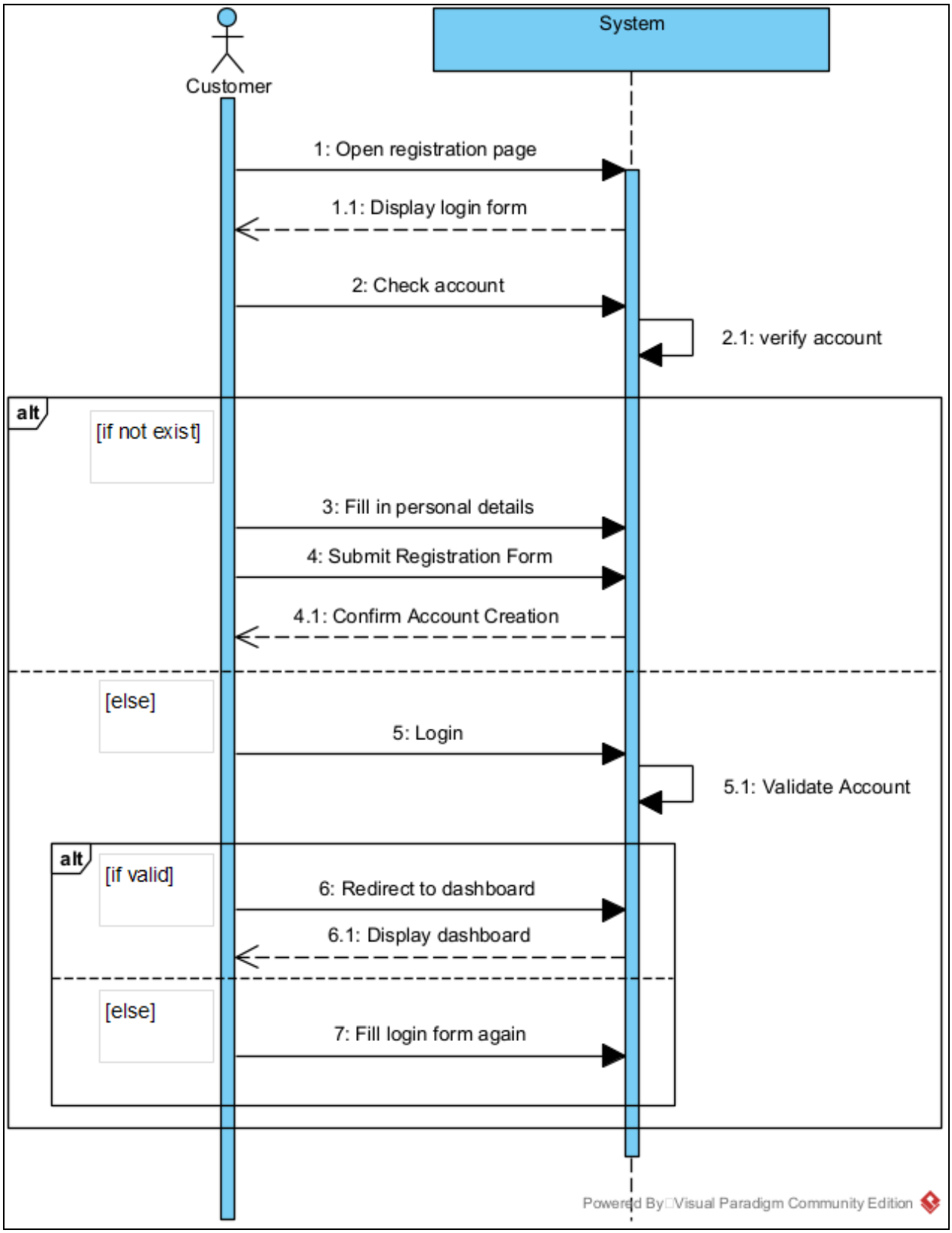


Diagram 2.2.3.1.1 shows a sequence diagram of Customer make a registration

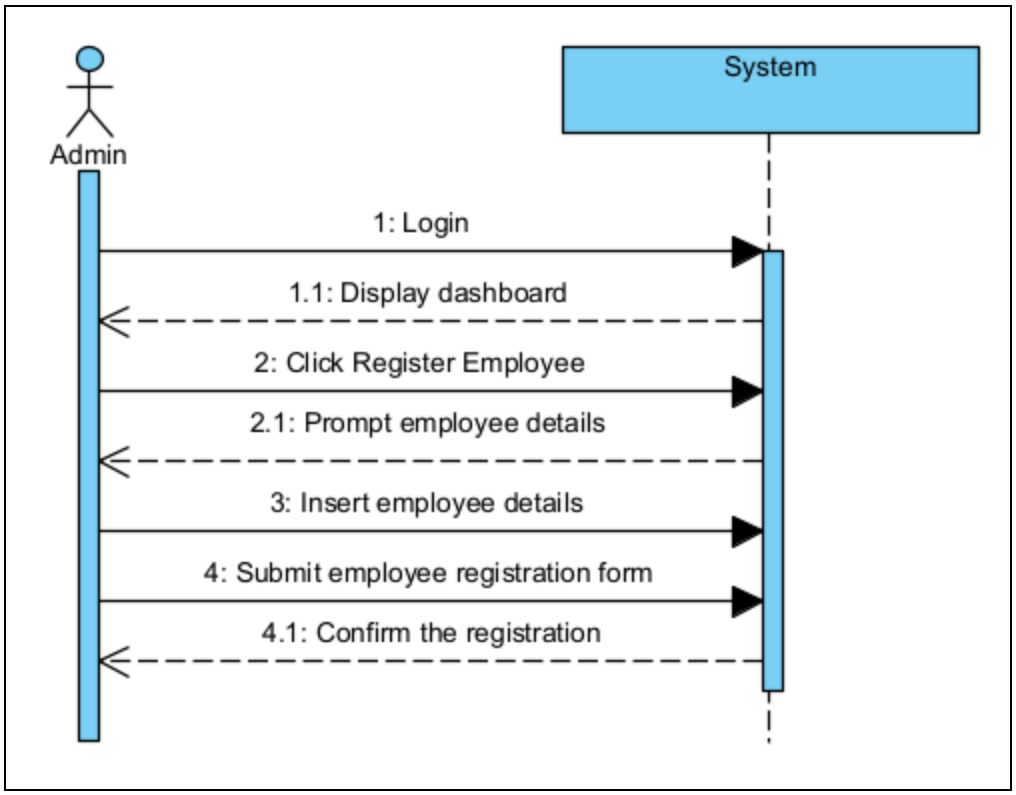


Diagram 2.2.3.1.2 shows a sequence diagram of Admin make a registration for their employees

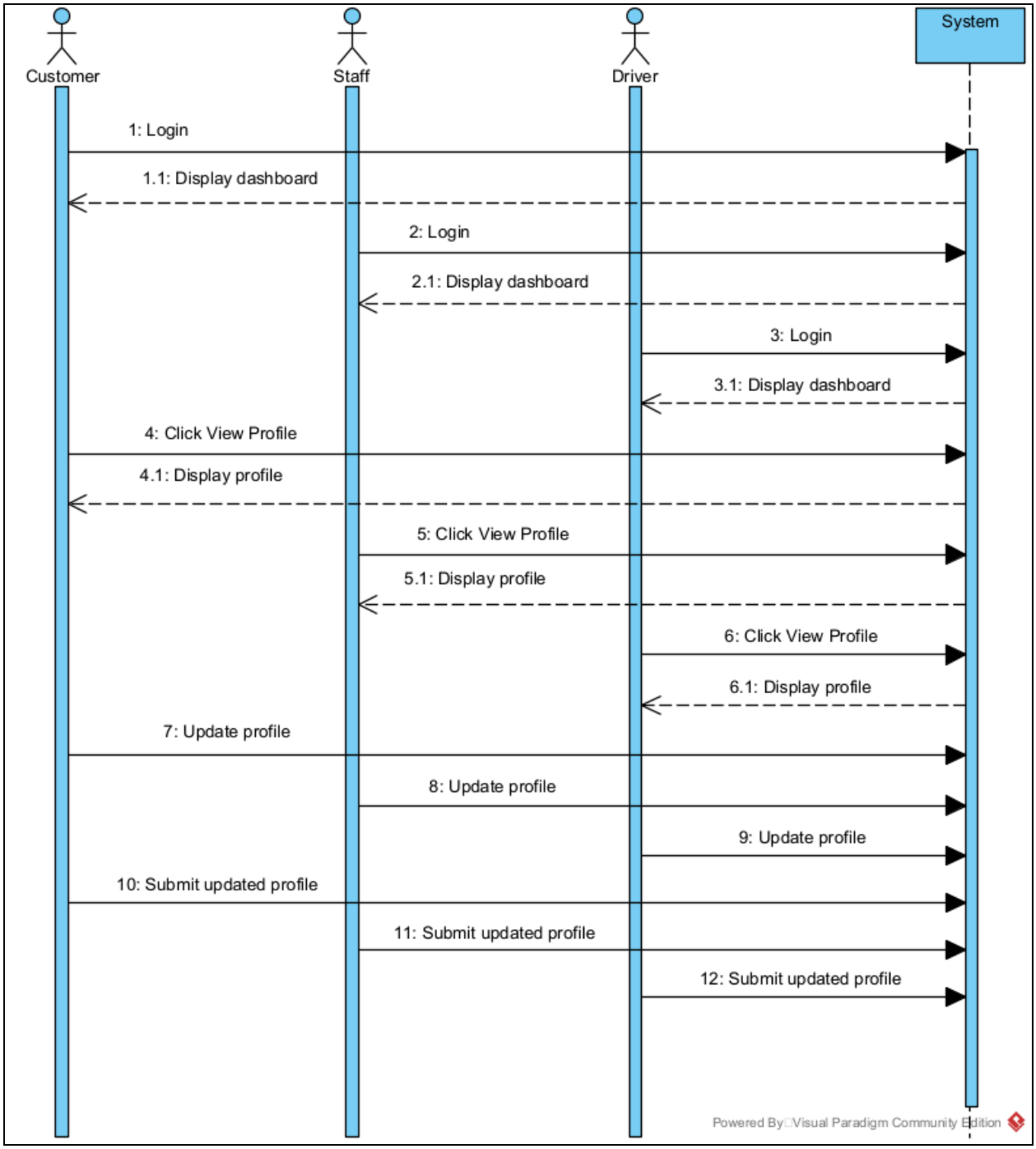


Diagram 2.2.3.1.3 shows a sequence diagram of Customer, Staff and Driver manage their profile

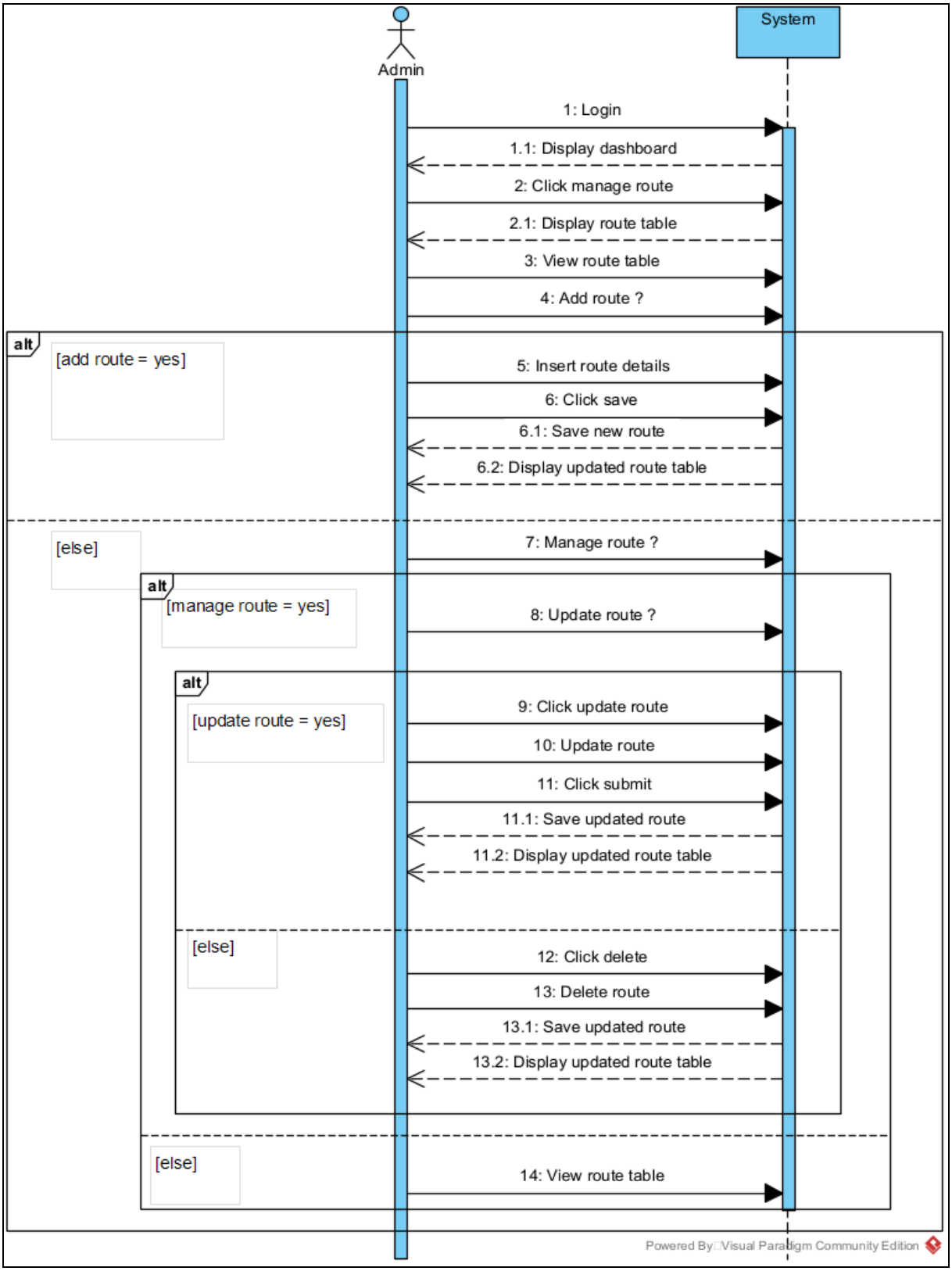


Diagram 2.2.3.1.4 shows a sequence diagram of Admin manage the route

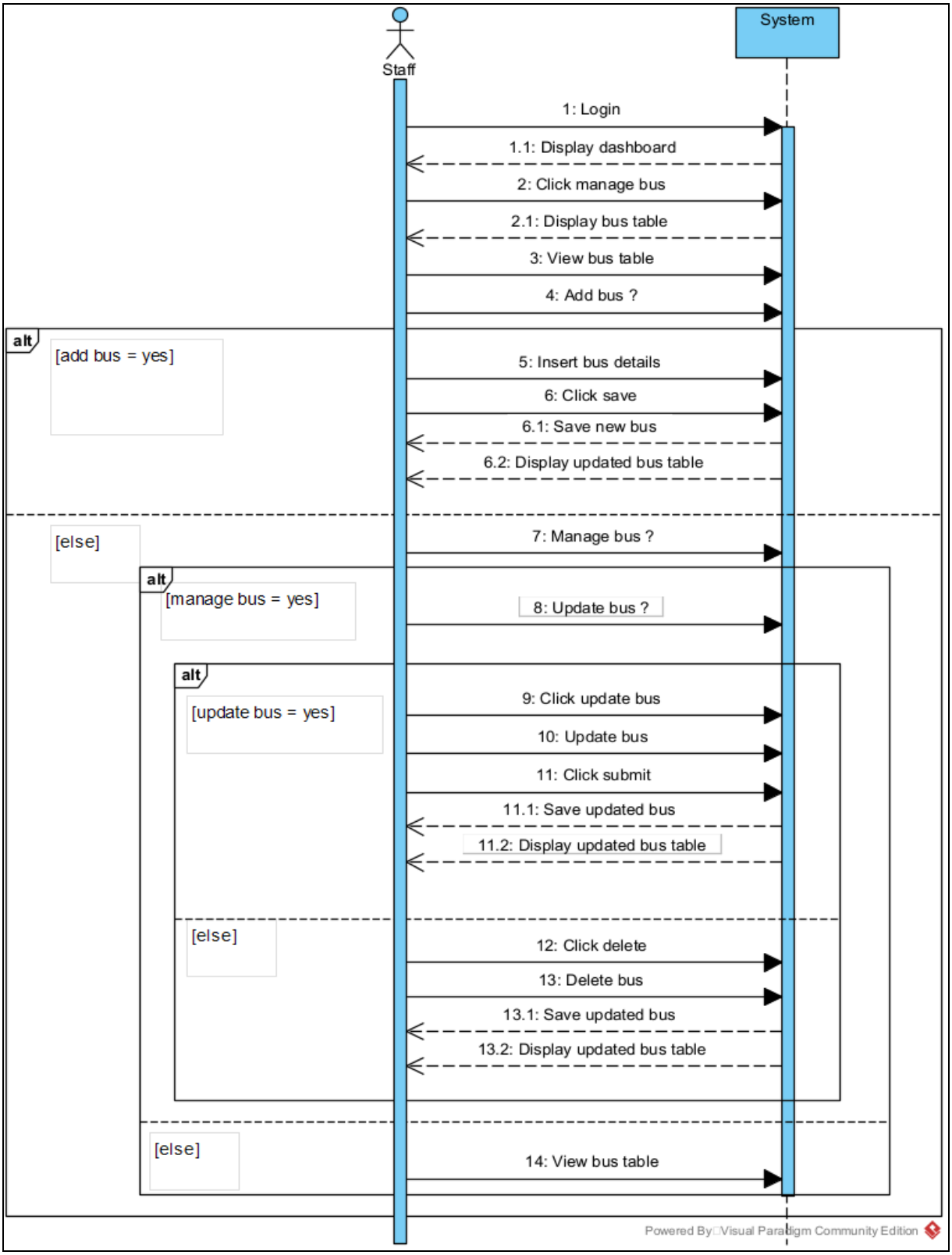


Diagram 2.2.3.1.5 shows a sequence diagram of Staff manage the bus

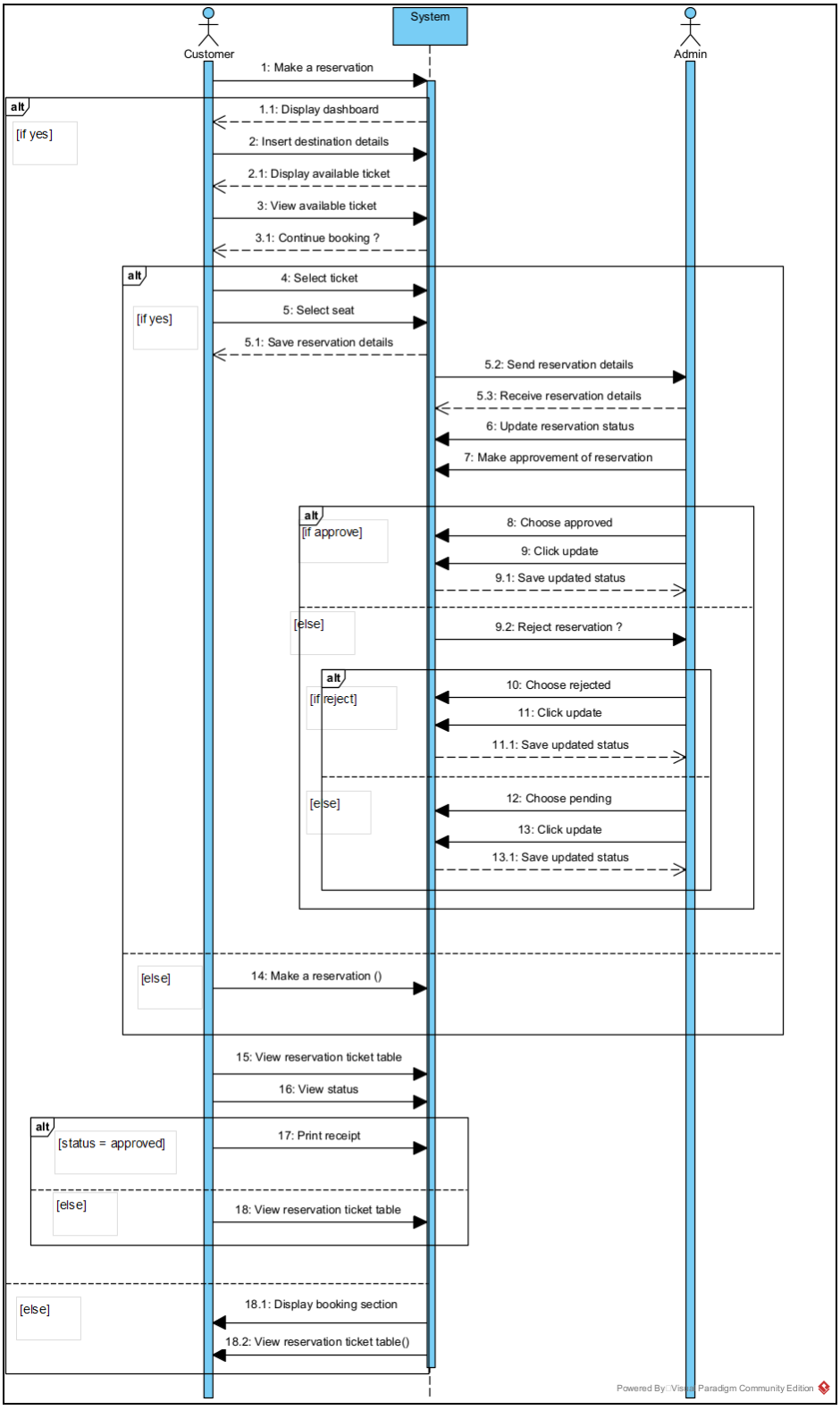


Diagram 2.2.3.1.6 shows a sequence diagram of Customer and Admin manage the reservation

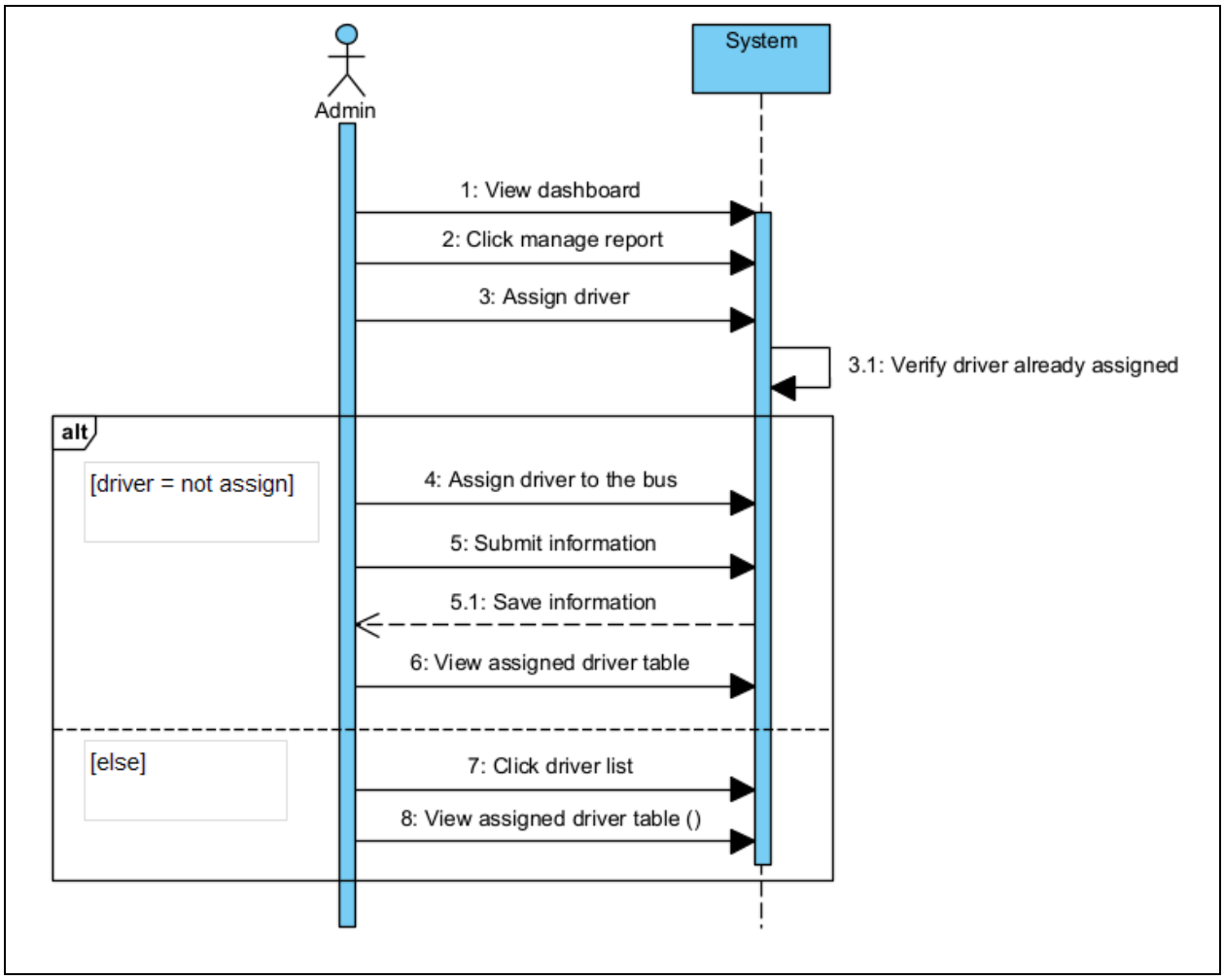


Diagram 2.2.3.1.7 shows a sequence diagram of Admin manage the report

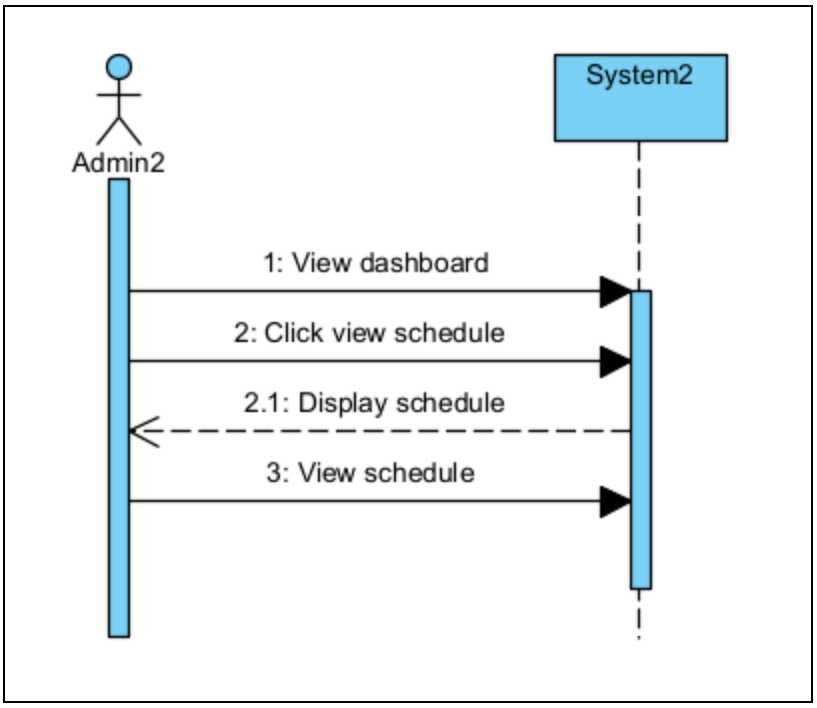


Diagram 2.2.3.1.8 shows a sequence diagram of Driver view their schedule

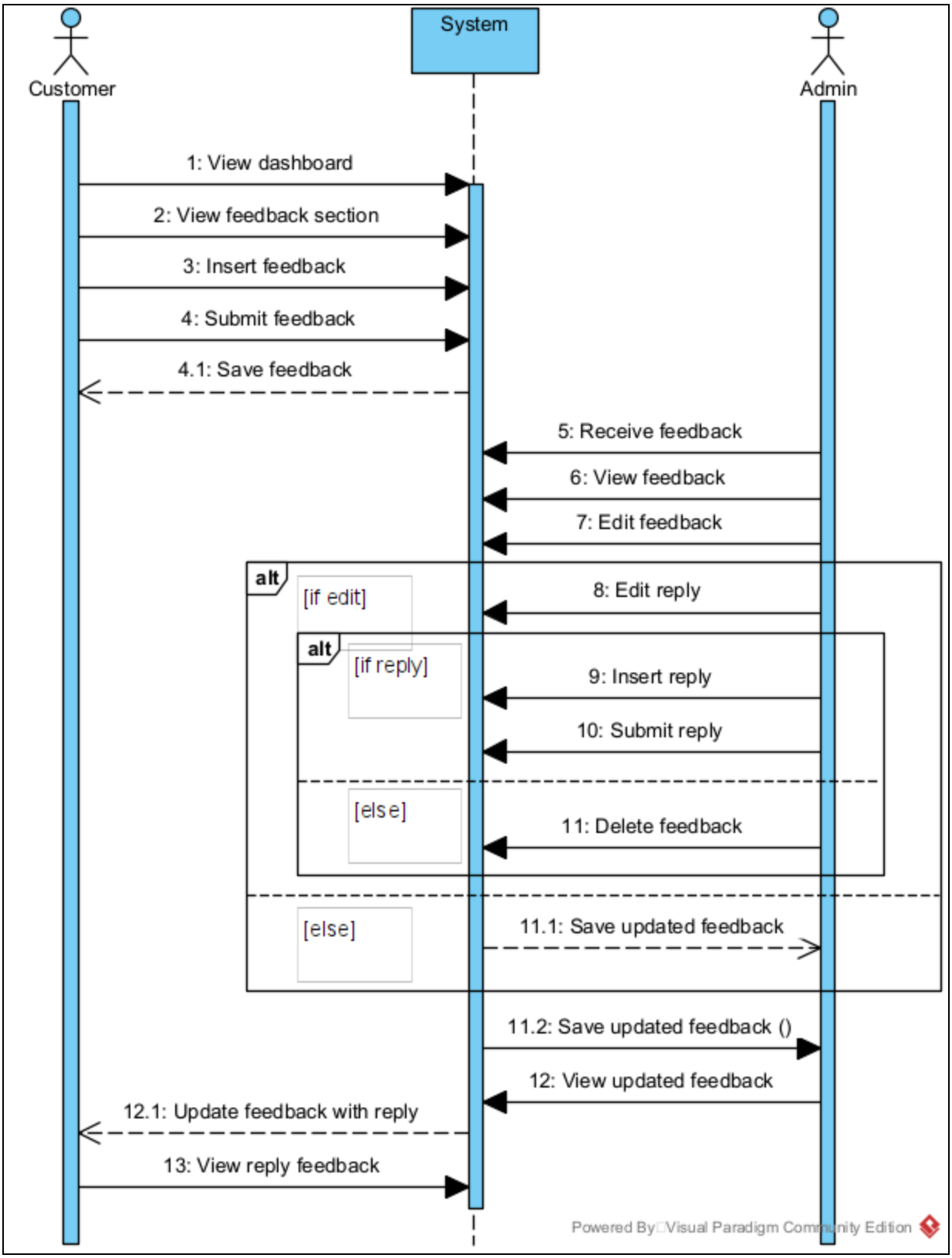


Diagram 2.2.3.1.9 shows a sequence diagram for customer service of Customer and Admin

2.4 CONCLUSION

In conclusion, the bus ticket reservation system aims to streamline the process of booking, managing, and maintaining bus reservations through a comprehensive software solution. This specification document has outlined the critical components, functionalities, and requirements needed to develop a robust, efficient, and user-friendly system.

Key Highlights:

1. **Functionality Overview:**
    - The system supports various essential functionalities such as user registration, profile management, route and bus management, reservation handling, report generation, schedule viewing, and customer service.
    - Different actors, including customers, staff, admin, and drivers, are provided with specific roles and access levels to ensure smooth operation and security.
2. **Hardware and Software Requirements:**
    - The hardware requirements specify the necessary components to support the system's performance, including servers, client devices, and network infrastructure.
    - The software requirements detail the operating systems, databases, application servers, development frameworks, and security measures needed for system development and deployment.
3. **Use Case and CRUD Analysis:**
    - Detailed use case descriptions provide a clear understanding of each system function, including the interactions between actors and the system.
    - The CRUD diagram clarifies which operations (Create, Read, Update, Delete) each actor can perform on various system entities, ensuring a clear understanding of data manipulation capabilities.
4. **Performance and Security:**
    - Emphasis on performance through high-specification hardware and optimized software solutions ensures that the system can handle peak loads and provide a seamless user experience.

- Robust security measures, including authentication, authorization, and data encryption, protect user data and maintain system integrity.

5. **Development and Maintenance:**
    - The specification highlights the importance of using modern development tools and methodologies, such as version control, continuous integration, and automated testing, to ensure efficient and high-quality software development.
    - Regular monitoring, logging, and maintenance practices are recommended to keep the system running smoothly and to quickly address any issues that may arise.

3. SOFTWARE DESIGN DOCUMENT (BY IRFAN HAKIMI)

3.1 INTRODUCTION

This document lays out the complete system design for our system , an online bus ticket booking system. Our system makes it easy for users to book seats on available bus routes through a convenient online platform. Here, you'll find the architecture, database schema, and the different parts of the system explained in detail. Our goal is to give you a clear and organized guide to how we plan to develop and implement this system

3.2 NORMALIZATION

First Normal Form

(user_id , route_id) → reservationID, date, departuretime, arrivaltime, status

Second Normal Form

(user_id) → feedback_id, content, reply, paymentID, amount, staff_id, ic_number, password, role, staff_fullname, username, fullname, email, ic_number, password

(route_id) → id, platno, type, capacity, status, origin, destination, departuretime, arrivaltime, price,

Third Normal Form

(staff_id) →, ic_number, password, role, staff_fullname

(id) → platno, type, capacity, status

(feedback_id) → content, user_id, reply

(paymentID) → reservationID, amount

Tables:

bus (id, platno, type, capacity, status)

feedback (feedback_id, content, user_id, reply)

payment (paymentID, reservationID, amount)

reservation (reservationID, user_id, routeid, date, departuretime, arrivaltime, status)

route (routeid, platno, origin, destination, departuretime, arrivaltime, price)

staff (staff_id, ic_number, password, role, staff_fullname)

users (user_id, username, fullname, email, ic_number, password)

3.3 ENTITY RELATIONSHIP DIAGRAM

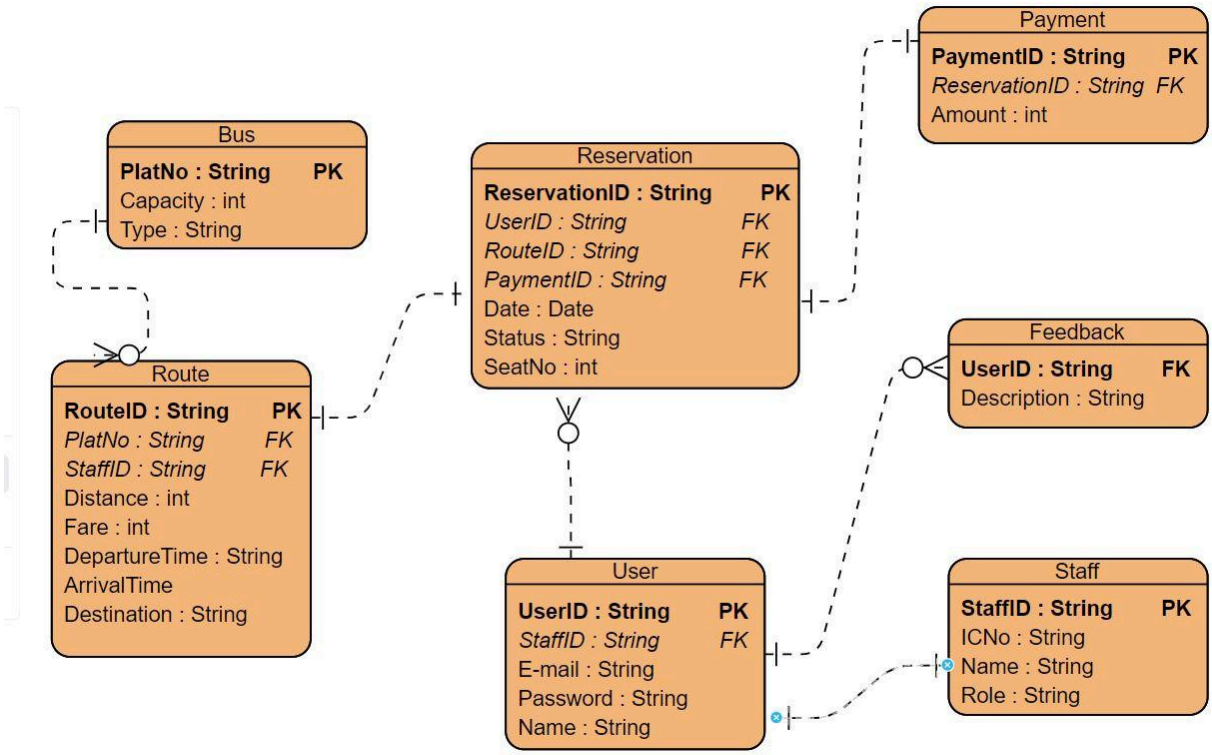


Diagram 3.3 shows the Entity Relationship Diagram for the Bus Booking system

3.4 DATA DICTIONARY

Table User

Attribute	Description	Data Type	Field Length	Constraints	Key	Foreign Key From
user_id	User ID	INT	11	AUTO INCREMENT, NOT NULL	Primary Key	
username	Username	VARCHAR	255	NOT NULL		
fullname	Fullname	VARCHAR	255	NOT NULL		
email	e-mail	VARCHAR	255	NOT NULL		
ic_number	IC Number	VARCHAR	14	NOT NULL		
password	Password	VARCHAR	255	NOT NULL		

Table Staff

Attribute	Description	Data Type	Field Length	Constraints	Key	Foreign Key From
staff_id	Staff ID	INT	11	AUTO INCREMENT, NOT NULL	Primary Key	
ic_number	IC Number	VARCHAR	14	NOT NULL		
password	Password	VARCHAR	255	NOT NULL		
role	Role	ENUM		NOT NULL		
staff_fullname	Staff Fullname	VARCHAR	255	NOT NULL		

Table Route

Attribute	Description	Data Type	Field Length	Constraints	Key	Foreign Key From
routeid	Route ID	VARCHAR	25	NOT NULL	Primary Key	
platno	Plat Number	VARCHAR	25	NOT NULL		
origin	Origin	VARCHAR	100	NOT NULL		
destination	Destination	VARCHAR	100	NOT NULL		
departuretime	Departure Time	TIME		NOT NULL		
arrivaltime	Arrival Time	TIME		NOT NULL		
price	Price	DECIMAL	7,2	NOT NULL		

Table Reservation

Attribute	Description	Data Type	Field Length	Constraints	Key	Foreign Key From
reservationID	Reservation ID	INT	4	AUTO INCREMENT, NOT NULL	Primary Key	
user_id	User ID	INT	11	NOT NULL	Foreign Key	Table User
routeid	Route ID	VARCHAR	25	NOT NULL	Foreign Key	Table Route
date	Date	VARCHAR	255	NOT NULL		
departuretime	Departure Time	TIME		NOT NULL		
arrivaltime	Arrival Time	TIME		NOT NULL		
status	Status	ENUM		NOT NULL		
seatno	Seat Number	VARCHAR	255	NOT NULL		

Table Payment

Attribute	Description	Data Type	Field Length	Constraints	Key	Foreign Key From
paymentID	Payment ID	INT	4	AUTO INCREMENT, NOT NULL	Primary Key	
reservationID	Reservation ID	INT	4	NOT NULL	Foreign Key	Table Reservation
amount	Amount	DECIMAL	9,2	NOT NULL		

Table Feedback

Attribute	Description	Data Type	Field Length	Constraints	Key	Foreign Key From
feedback_id	Feedback ID	INT	11	AUTO INCREMENT, NOT NULL	Primary Key	
content	Content	TEXT		NOT NULL		
user_id	User ID	INT	11	NOT NULL	Foreign Key	Table User
reply	Reply	INT		NOT NULL		

Table Bus

Attribute	Description	Data Type	Field Length	Constraints	Key	Foreign Key From
id	Bus ID	INT	5	AUTO INCREMENT, NOT NULL	Primary Key	
platno	Plat Number	VARCHAR	25	NOT NULL		
type	Bus Type	VARCHAR	100	NOT NULL		
capacity	Capacity	INT	5	NOT NULL		
Status	Status	VARCHAR	100	NOT NULL		

3.5 USER INTERFACE DESIGN

USER MODULE

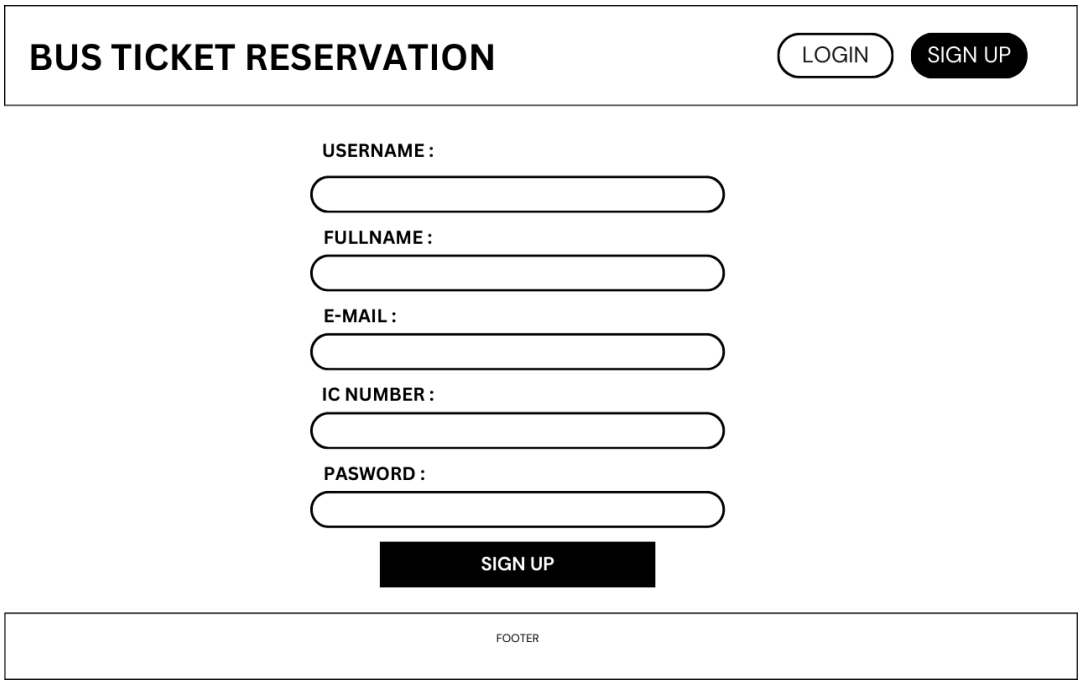


Diagram 3.5.1 shows the registration page for user

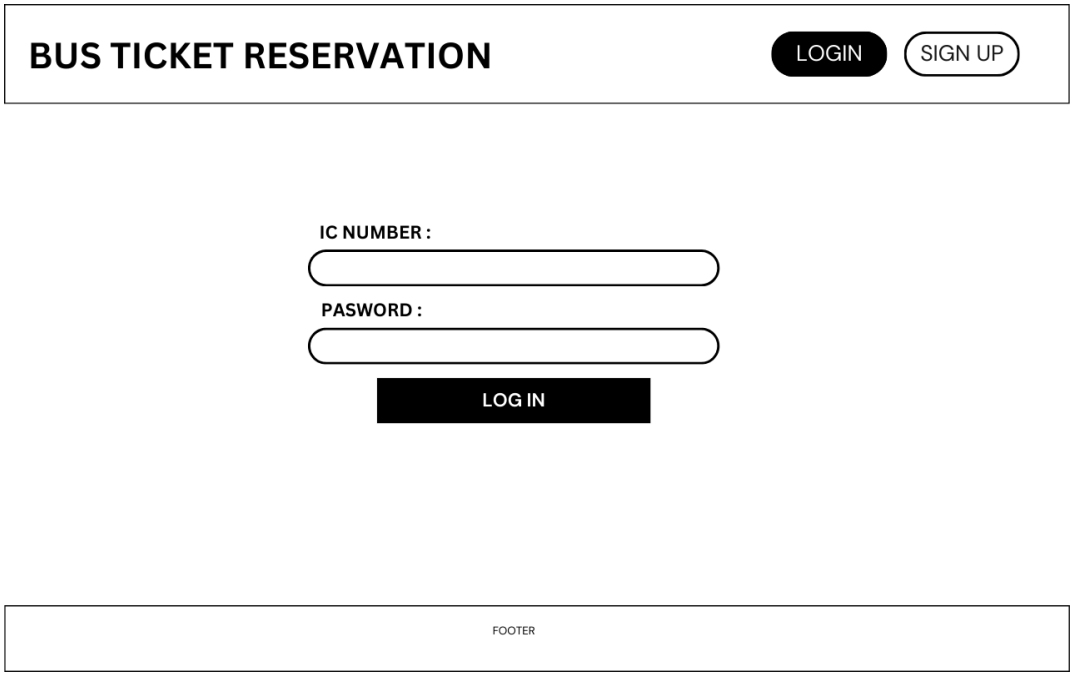


Diagram 3.5.2 shows the login page for user

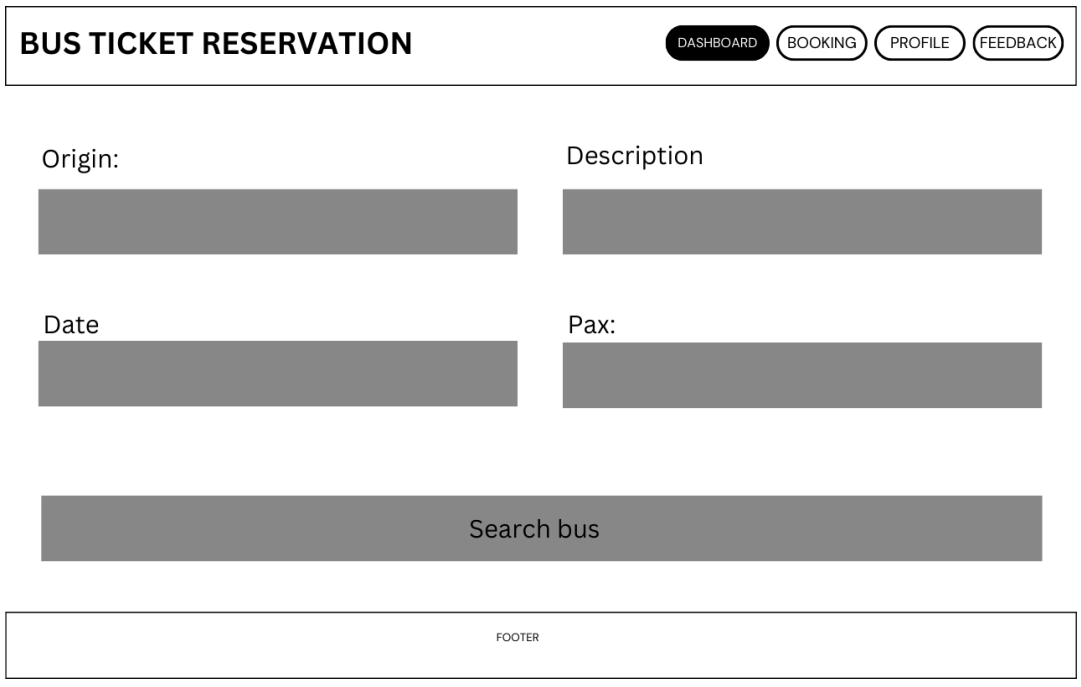


Diagram 3.5.3 shows the dashboard page for user

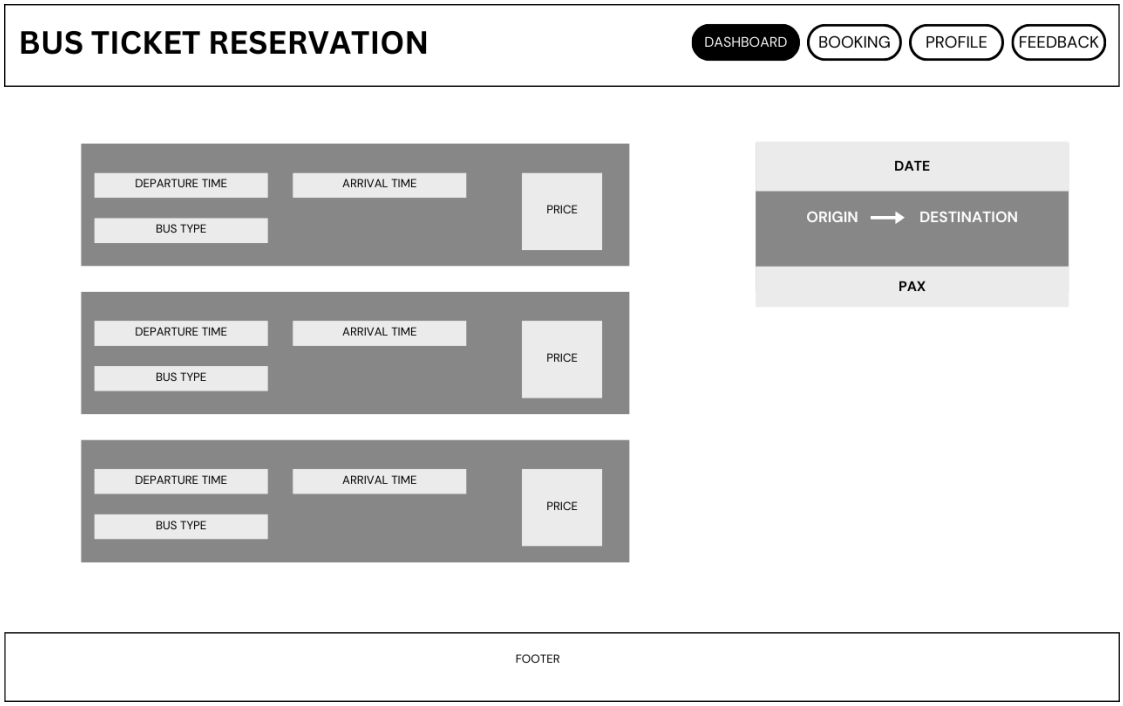


Diagram 3.5.4 shows the bus list page for user

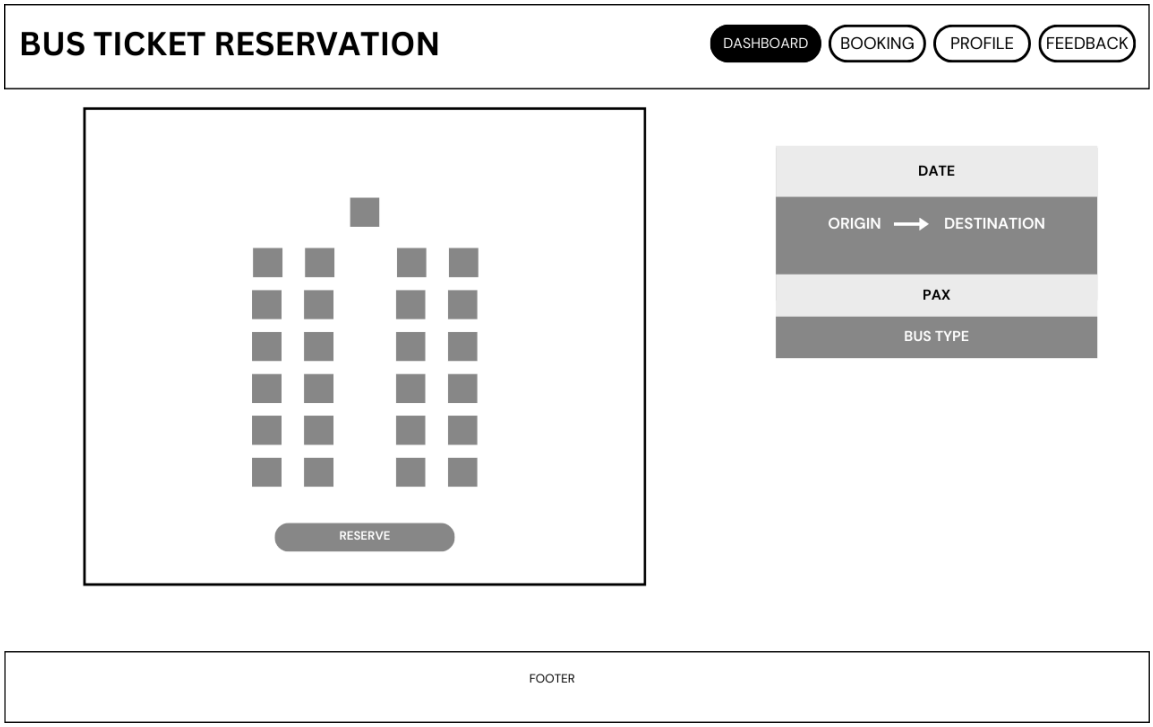


Diagram 3.5.5 shows the bus seat layout page for user

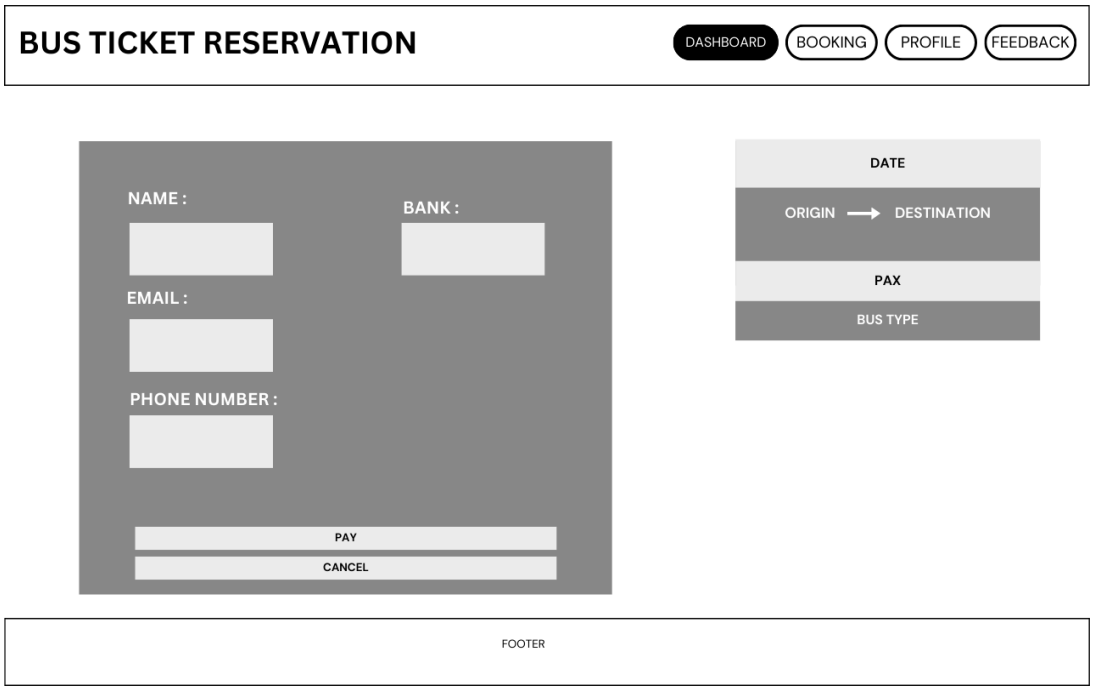


Diagram 3.5.6 shows the payment page for user

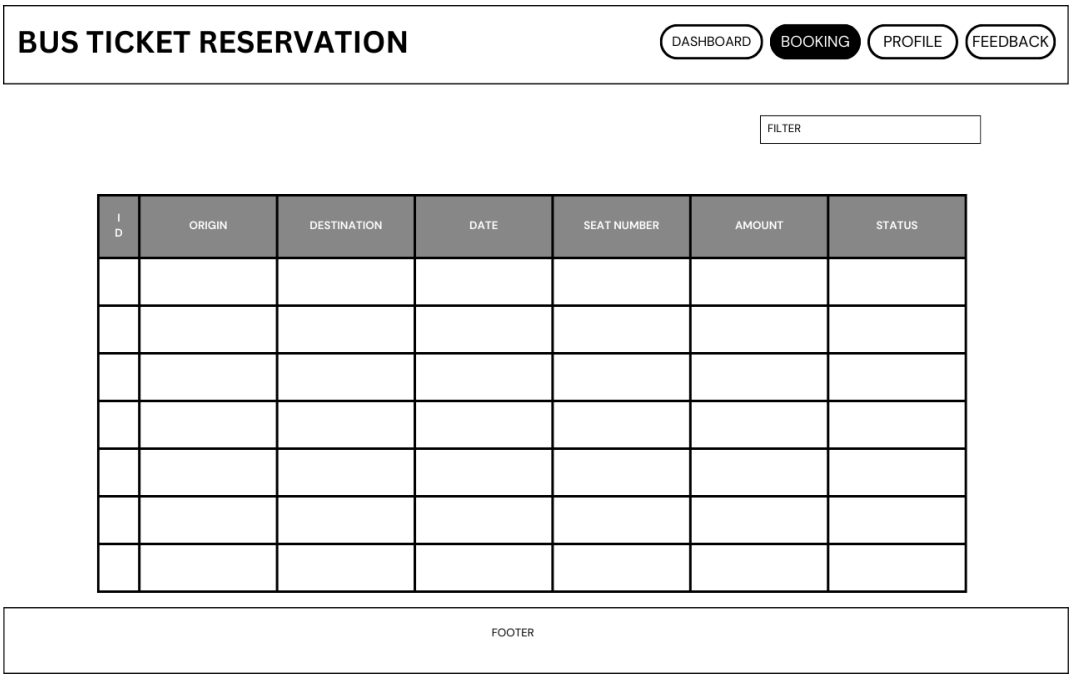


Diagram 3.5.7 shows the booking list page for user

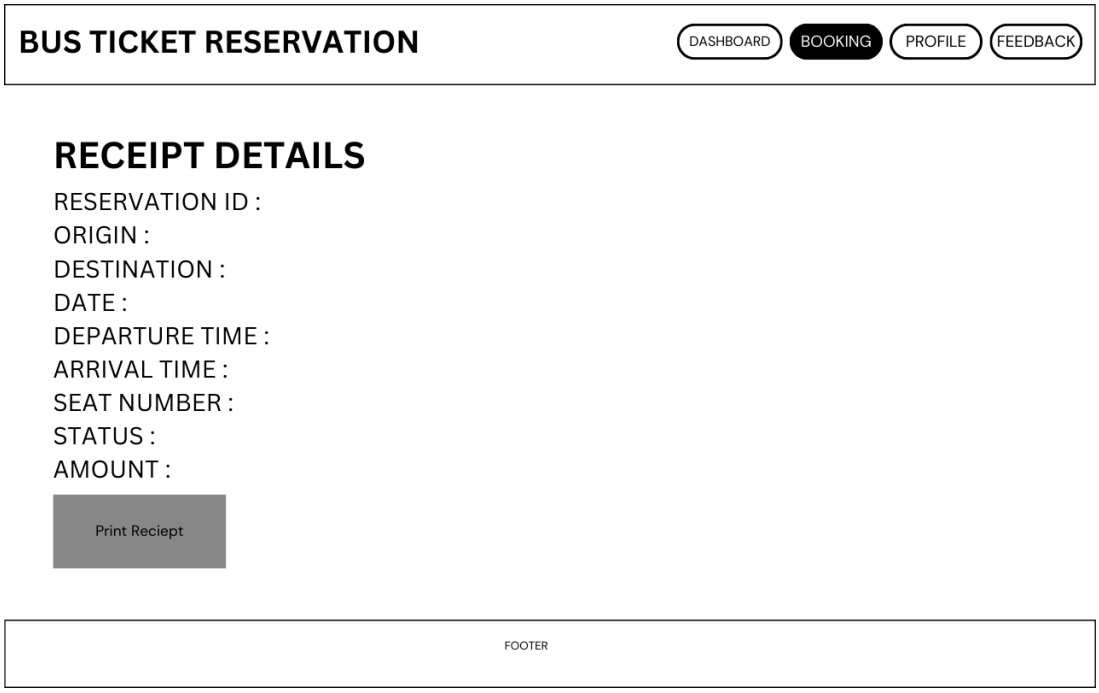


Diagram 3.5.8 shows the receipt page for user

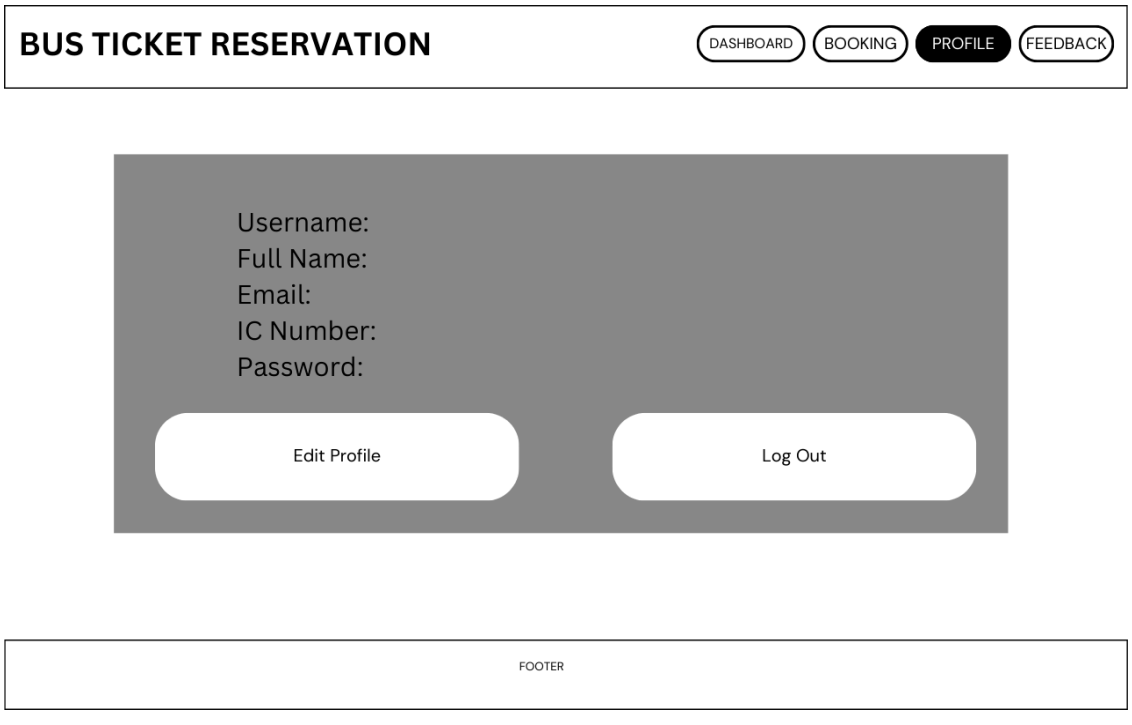


Diagram 3.5.9 shows the profile page for user

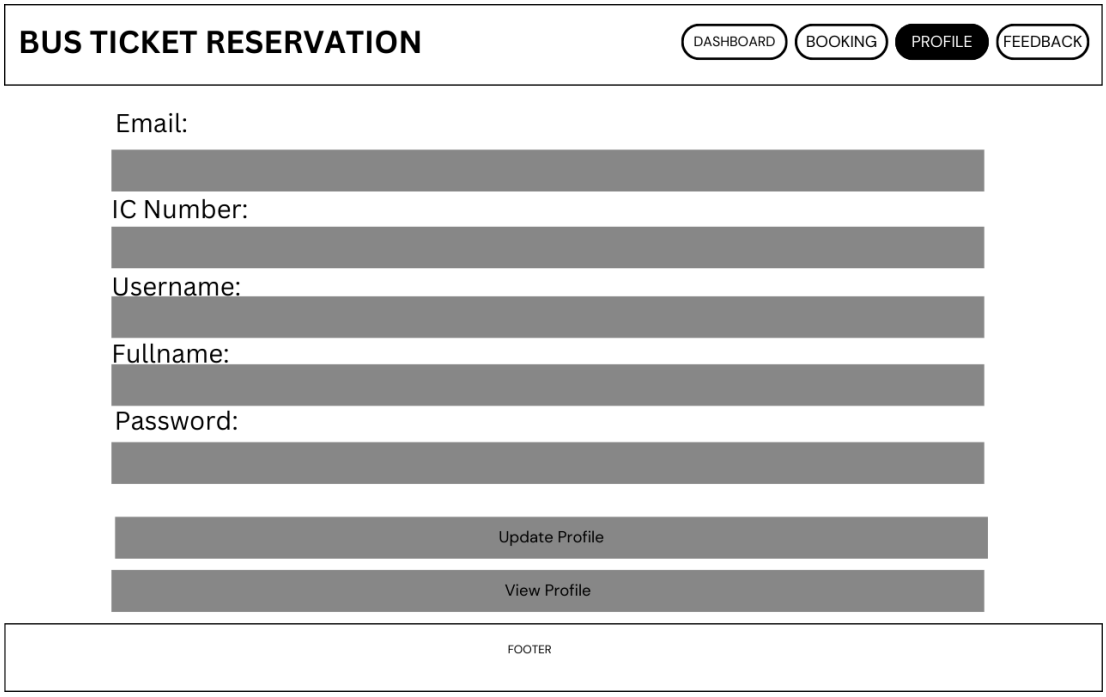


Diagram 3.5.10 shows the profile edit page for user

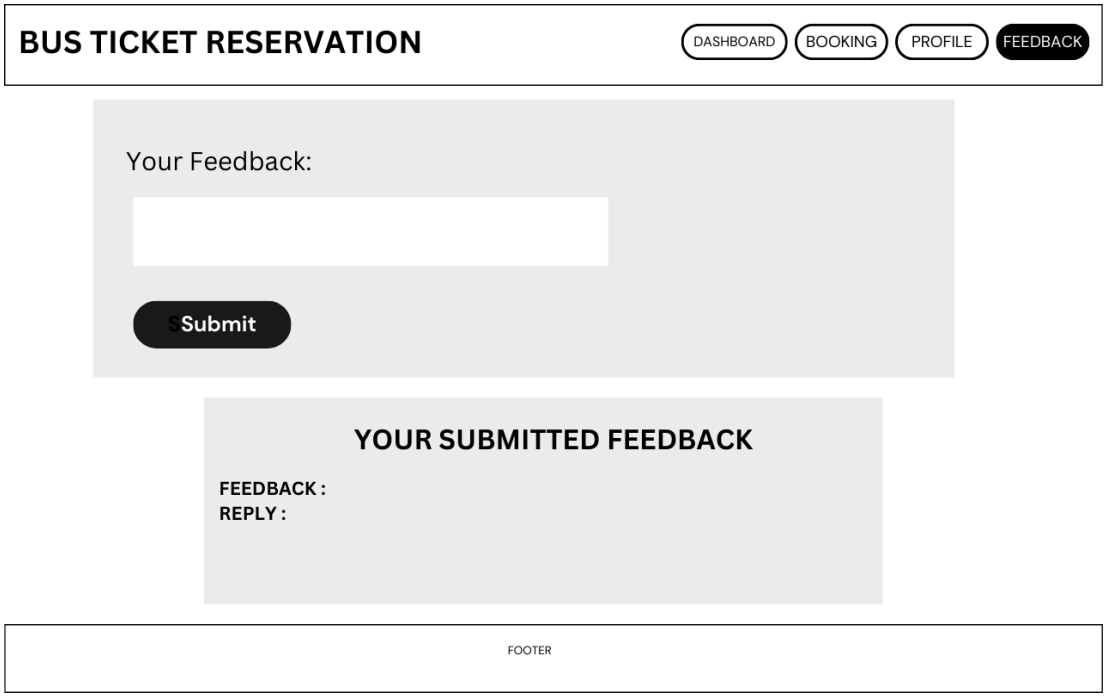


Diagram 3.5.11 shows the feedback page for user

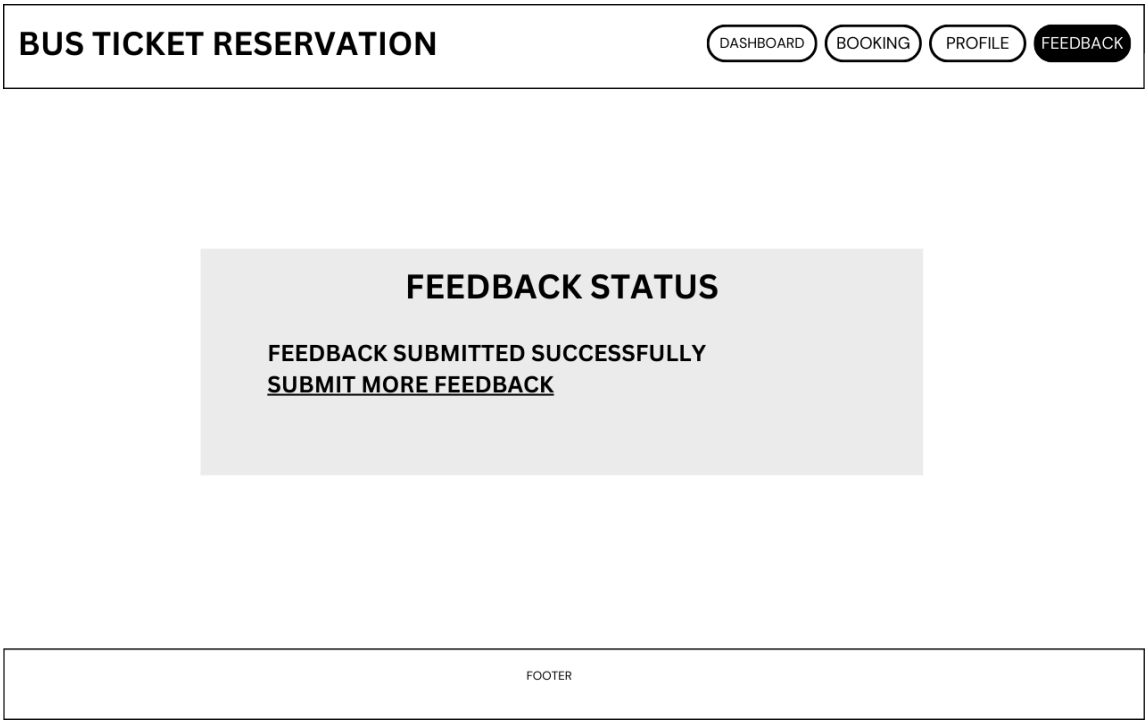


Diagram 3.5.12 shows the feedback sent successfully page for user

ADMIN MODULE

Diagram 3.5.13 shows the login page for admin and staff

Diagram 3.5.14 shows the login page for admin

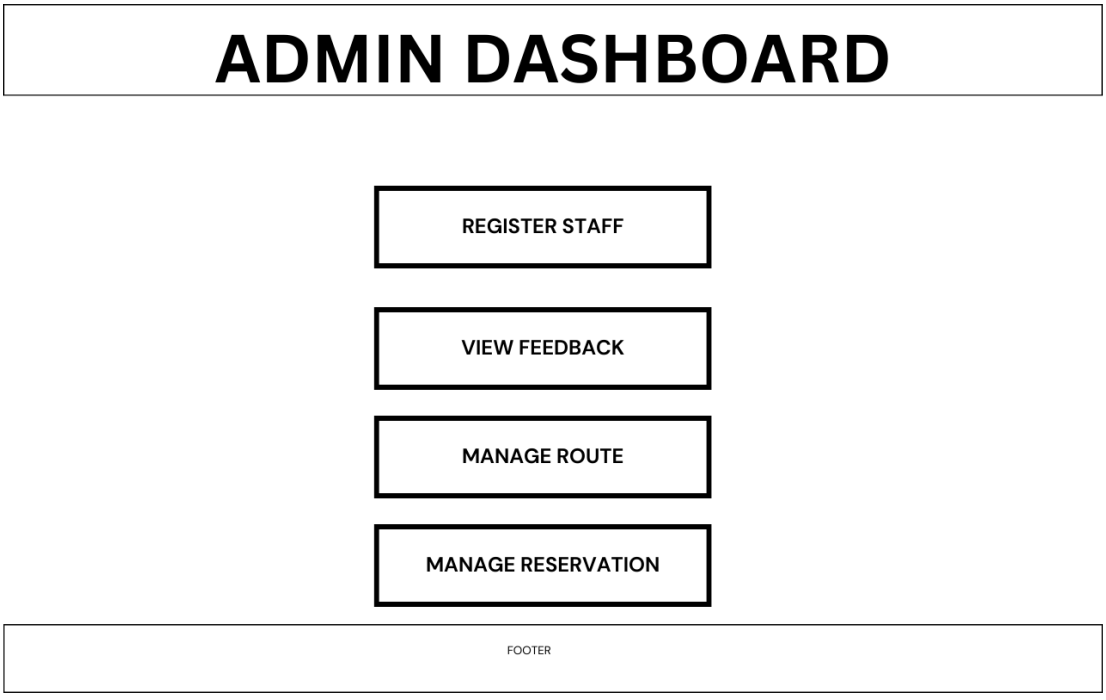


Diagram 3.5.15 shows the dashboard page for admin

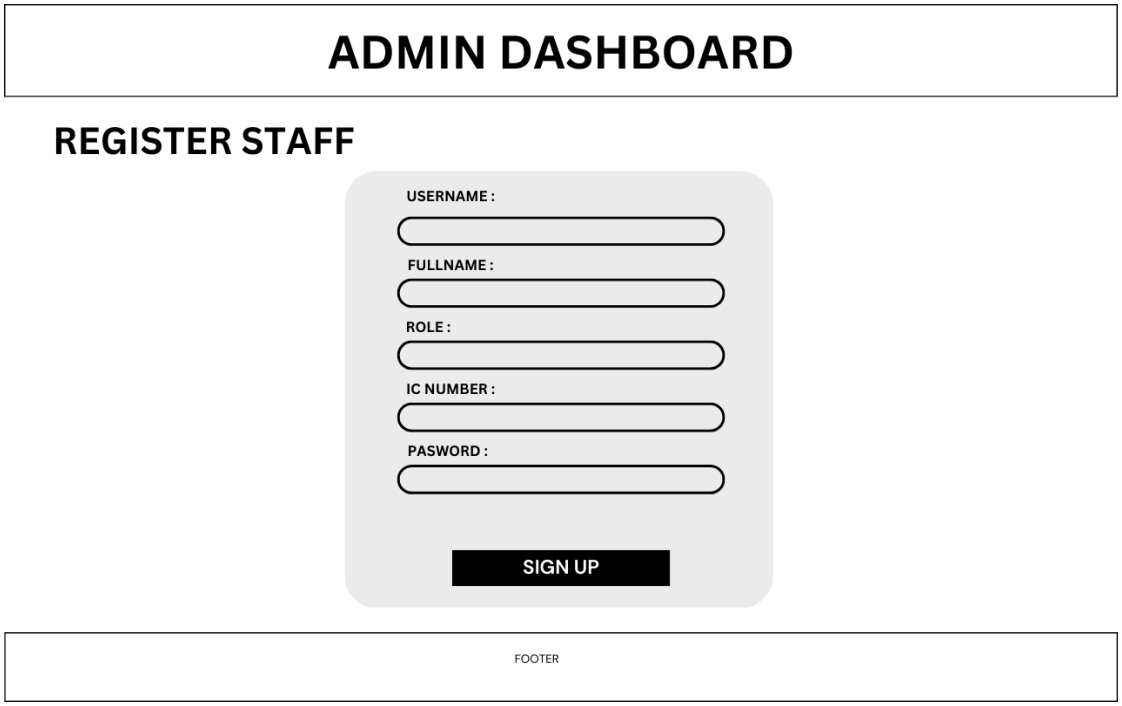


Diagram 3.5.16 shows the register staff page for admin

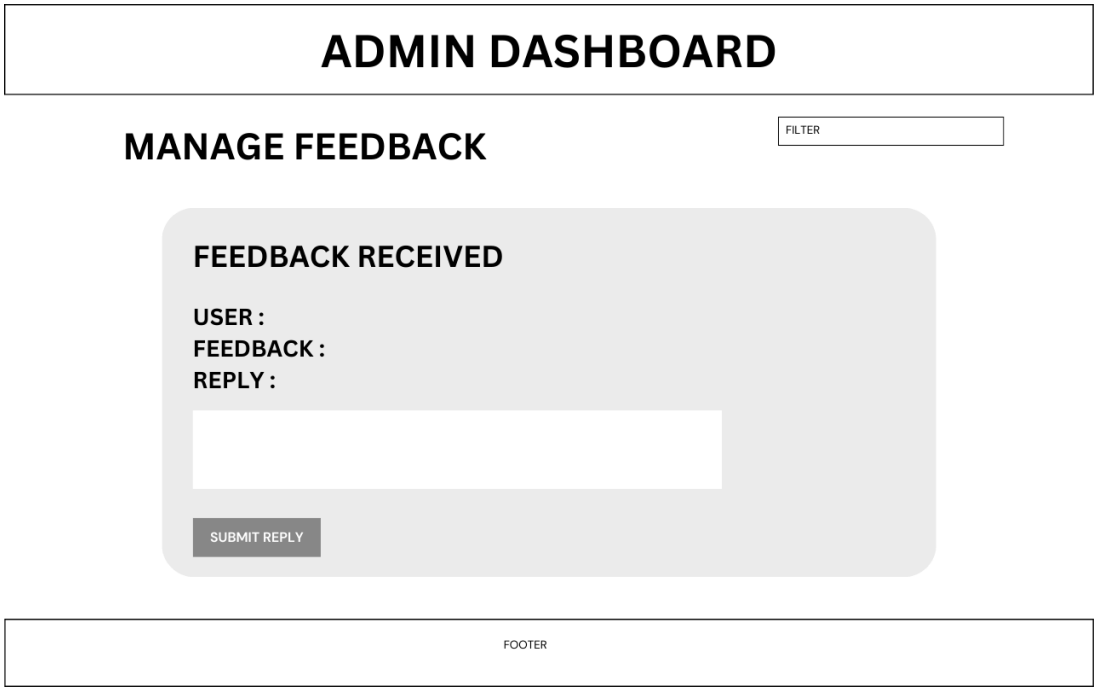


Diagram 3.5.17 shows the manage feedback page for admin

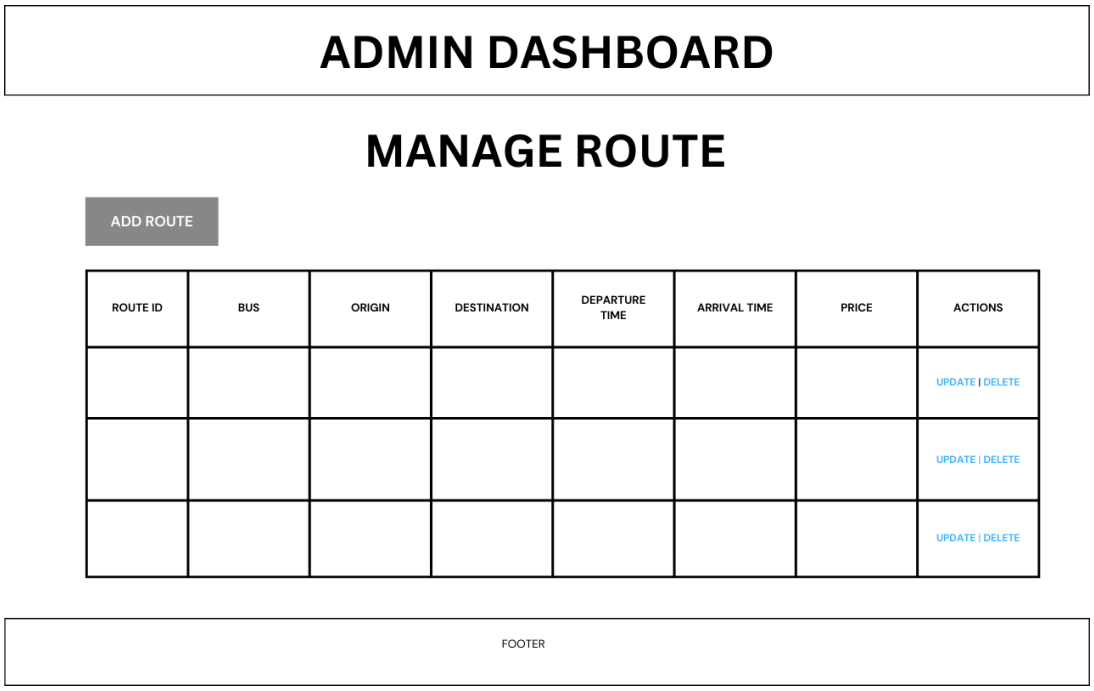


Diagram 3.5.18 shows the manage route page for admin

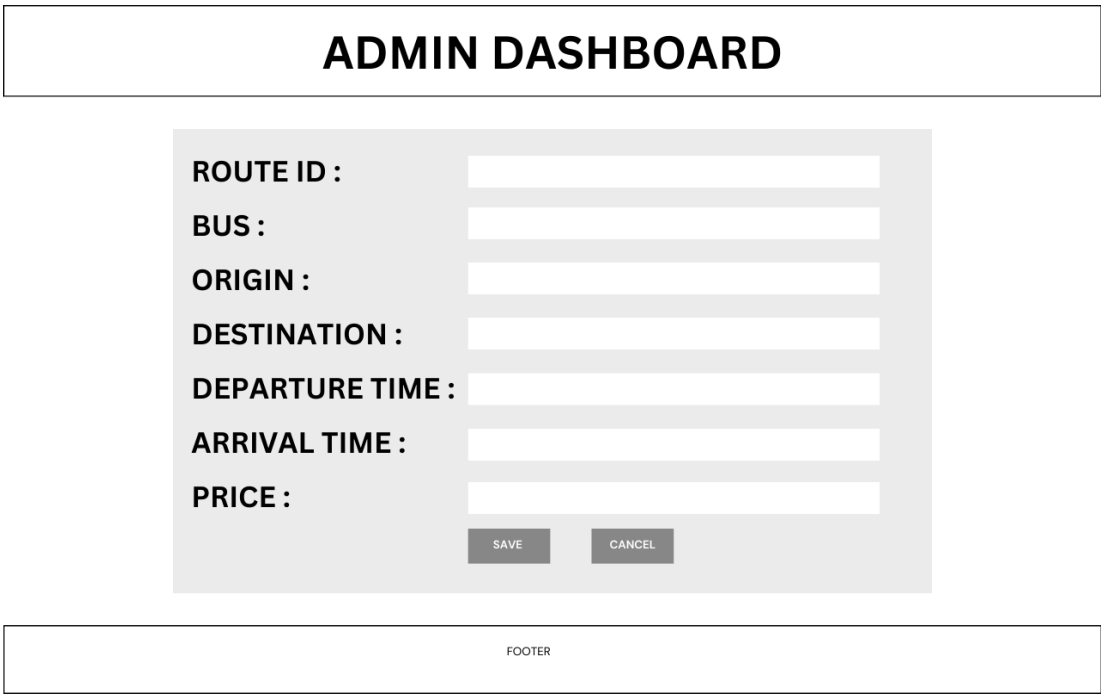


Diagram 3.5.19 shows the add route page for admin

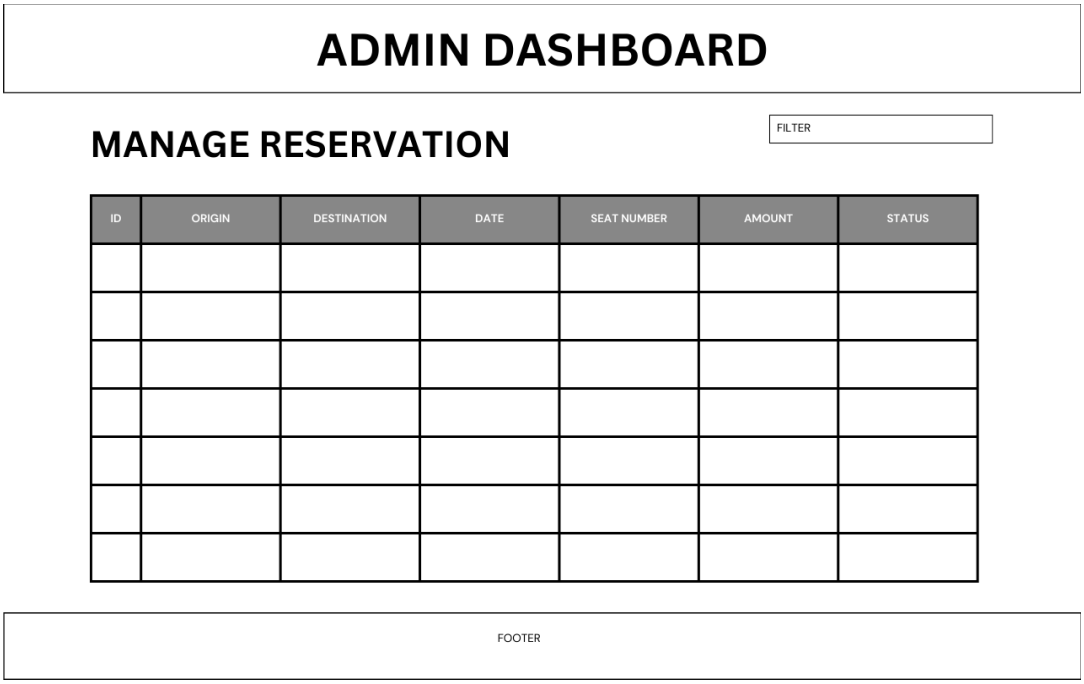


Diagram 3.5.20 shows the manage reservation page for admin

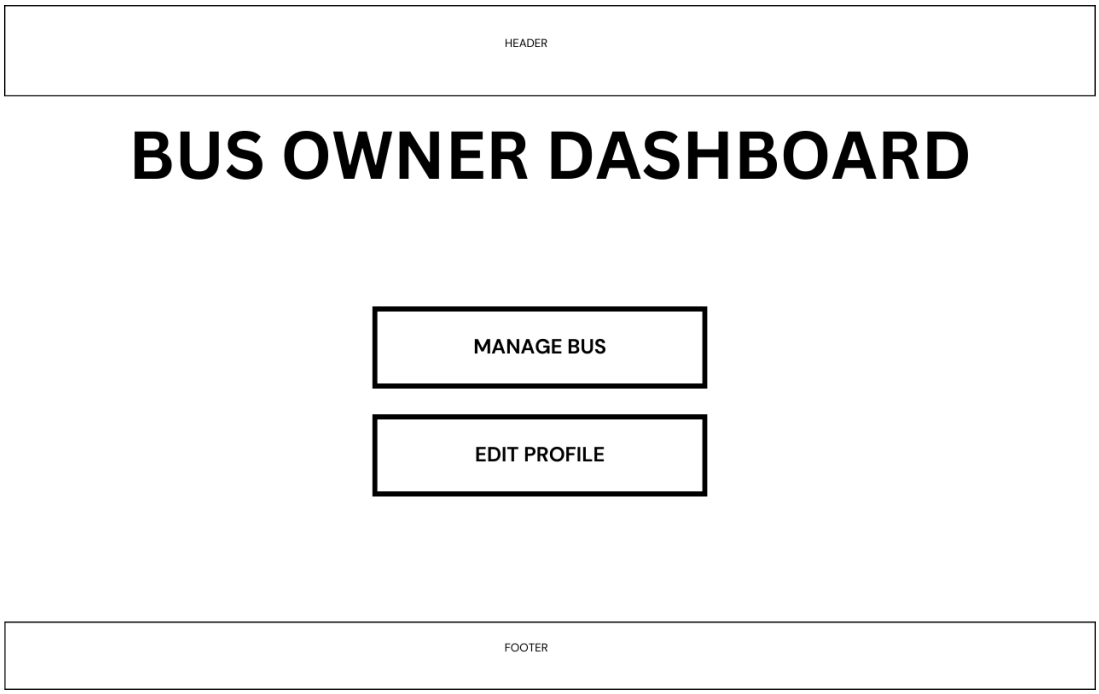


Diagram 3.5.21 shows the dashboard page for staff

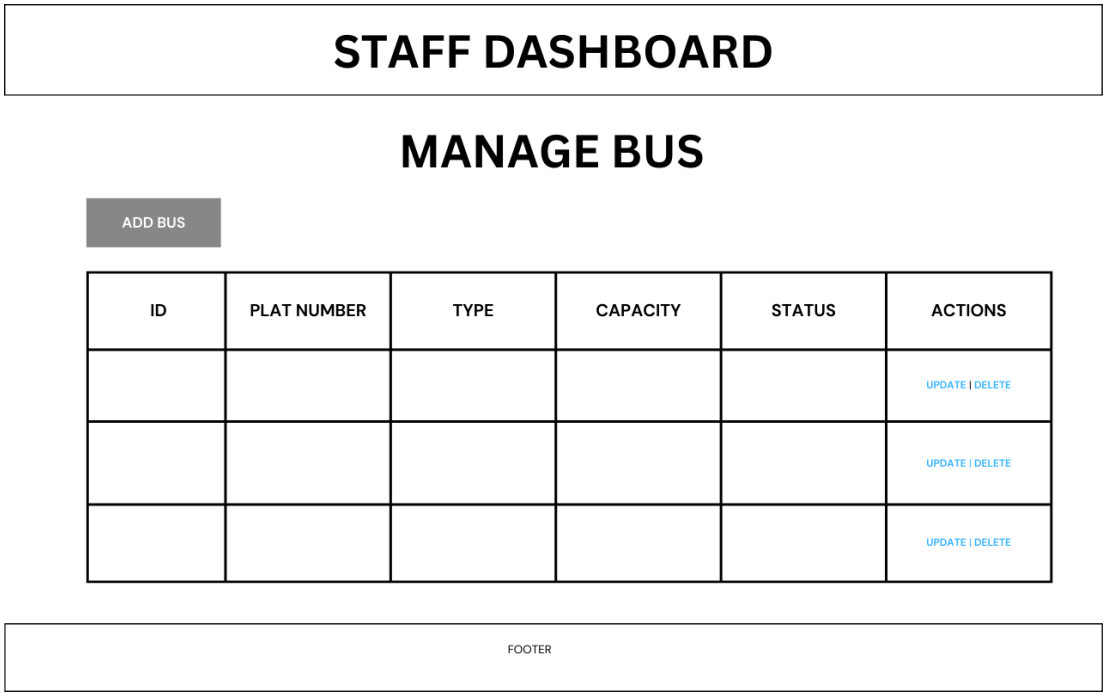


Diagram 3.5.22 shows the manage bus page for staff

Diagram 3.5.23 shows the add bus page for staff

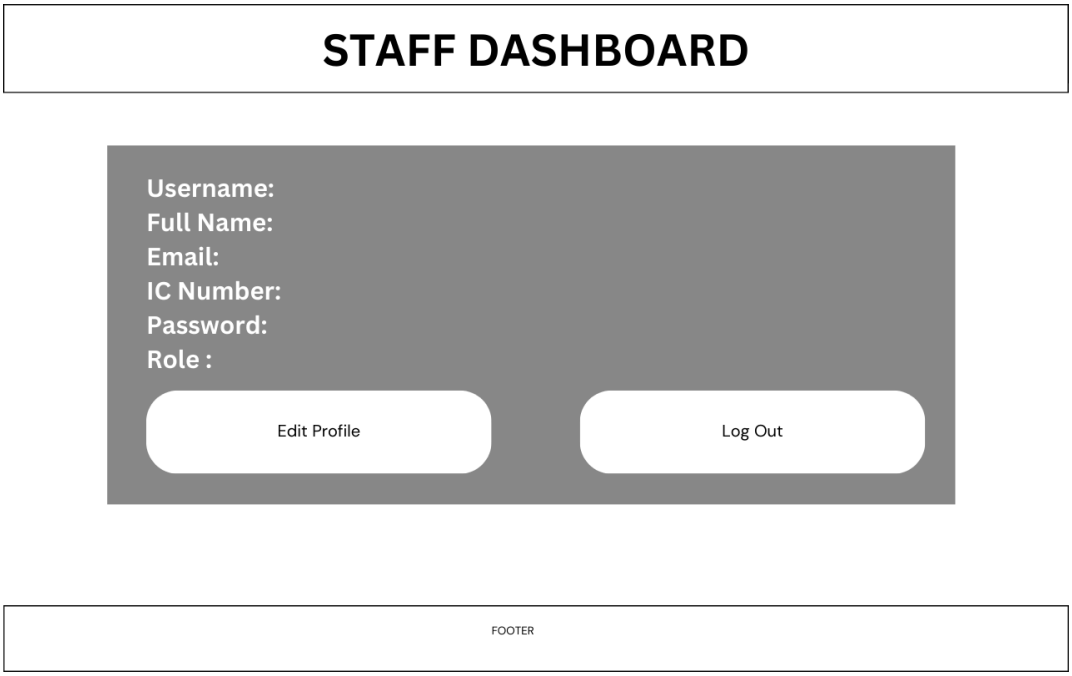


Diagram 3.5.24 shows the edit profile page for staff

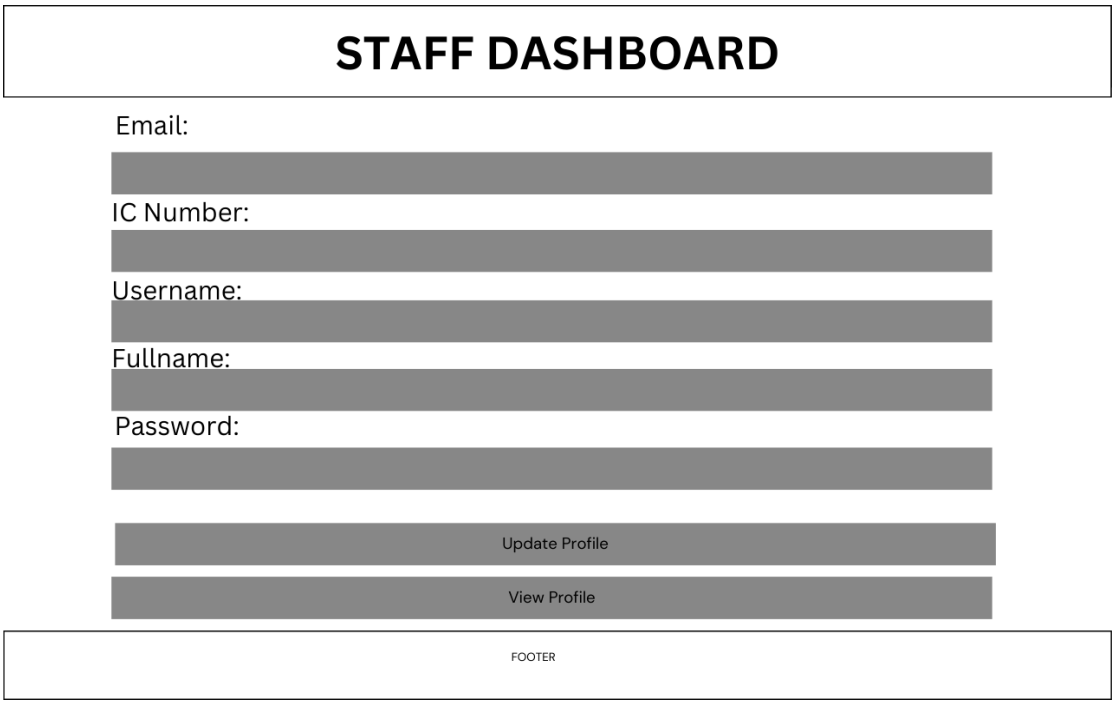


Diagram 3.5.25 shows the edit profile page for staff

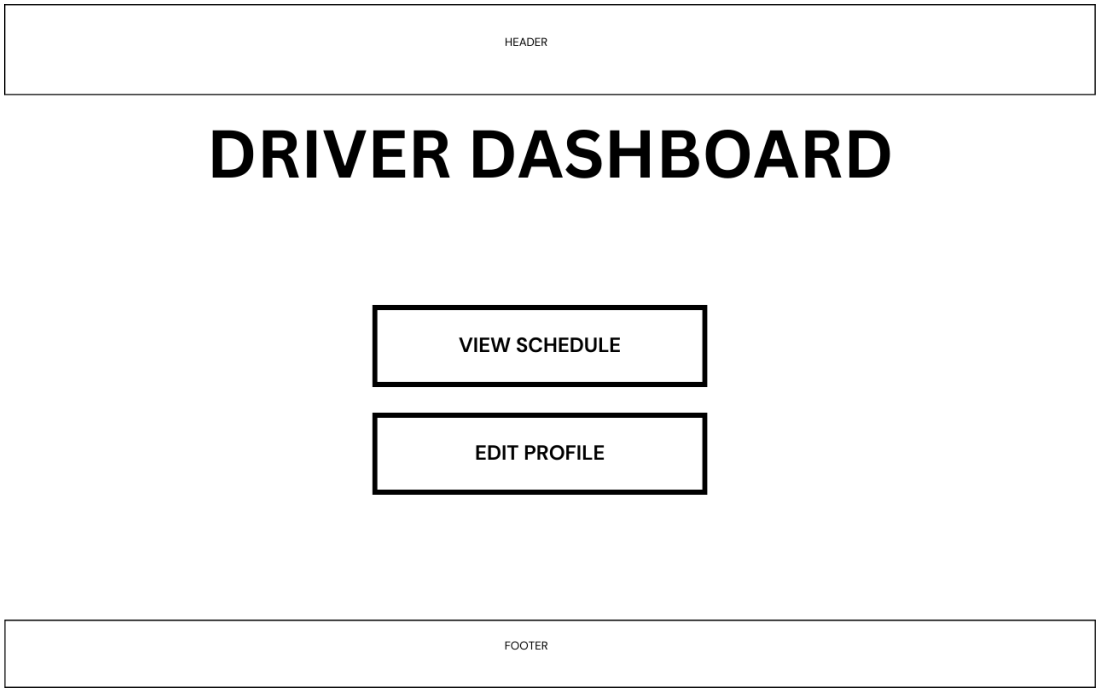


Diagram 3.5.26 shows the dashboard page for driver

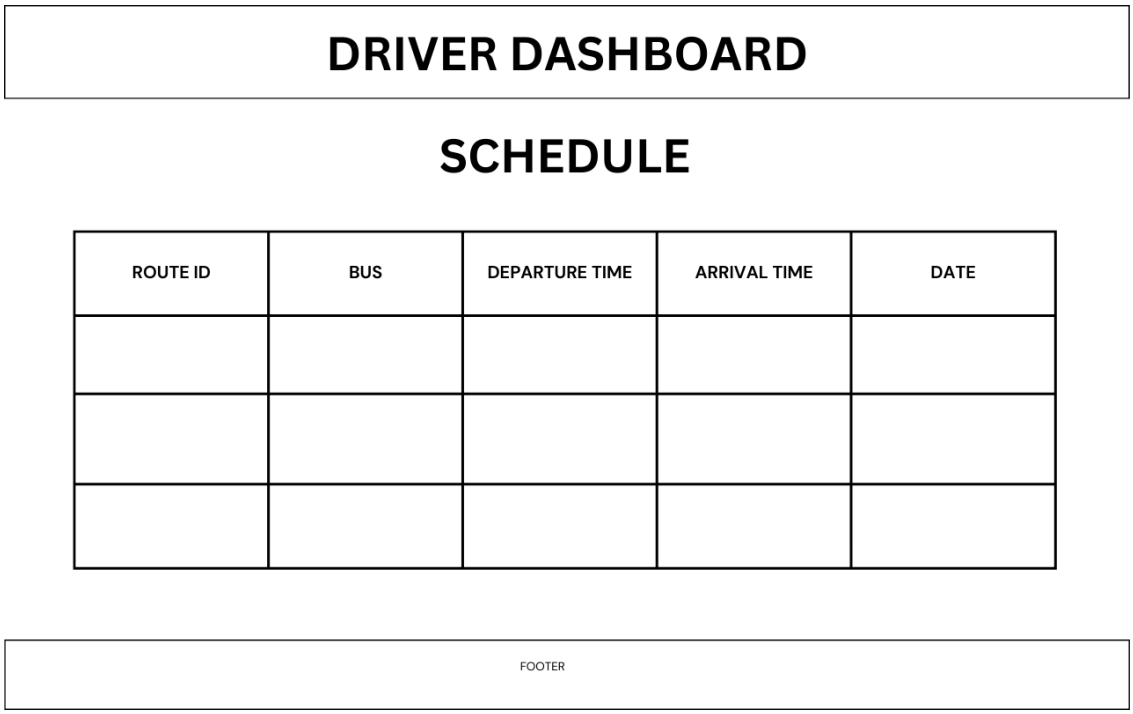


Diagram 3.5.27 shows the schedule page for driver

4. USER MANUAL (BY HARITH)

4.1 INTRODUCTION

The Tiny Travel Company's Bus Ticket Reservation System is intended to completely transform the way consumers make travel arrangements and purchase bus tickets. Travelers may quickly search for routes, check the availability of seats in real time, and securely make payments online with this user-friendly web site. The system guarantees a customized booking experience with features designed to fit different bus models, seating configurations, and amenities. The entire procedure is smooth and effective because of the administrators' ability to effectively manage bus data, schedules, and routes. Our system is designed to handle future development expectations by prioritizing customer happiness, reliability, and scalability, drawing inspiration from the success of organizations such as Sani Express Sdn Bhd.

4.1.1 System Overview

Tiny Travel Company's Bus Ticket Reservation System is intended to make the bus ticket booking procedure easier for administrators and customers alike. Through an easy-to-use web interface, users may search for bus routes that are available, monitor seat availability in real-time, make secure payments, and manage their bookings. The system makes sure that clients may choose solutions that best suit their needs by supporting a variety of bus models, and seat arrangements. Routes, schedules, and bus specifics may all be managed by administrators, resulting in a smooth and effective experience for all parties.

4.1.2 System Requirement

4.1.2.1 Hardware Requirements

- Server Requirements:
    - Processor: Intel Xeon or AMD equivalent, minimum 2 GHz
    - RAM: 8 GB or higher
    - Storage: SSD with at least 500 GB free space
    - Network: Reliable internet connection with a minimum speed of 100 Mbps
- Client Requirements:
    - Processor: Intel i3 or AMD equivalent, minimum 1.6 GHz
    - RAM: 4 GB or higher
    - Storage: 100 GB free space
    - Display: 1024 x 768 resolution or higher
    - Network: Stable internet connection

4.1.2.2 Software Requirements

- Server-Side:
    - Operating System: Linux (Ubuntu, CentOS) or Windows Server
    - Web Server: Apache or Nginx
    - Database: MySQL or PostgreSQL
    - Scripting Language: PHP 7.x or higher
    - Frameworks: Laravel or CodeIgniter (for PHP)
    - SSL Certificate for secure transactions
- Client-Side:
    - Operating System: Windows, macOS, Linux, iOS, or Android
    - Web Browser: Latest versions of Google Chrome, Mozilla Firefox, Safari, or Microsoft Edge
    - JavaScript enabled

4.1.3 System Setting

Registration and Login

- Users must register with a valid email address and create a secure password.
- Set user roles and permissions (e.g., admin, staff, driver, user)

Payment Gateway Integration

- Secure payment gateways are integrated to process payments.
- Payment confirmation and e-tickets are sent to the user's registered email upon successful transaction.

Administrator Functions

- Administrators can manage bus routes, schedules, and bus details through an admin panel.
- Administrators can generate reports

Search and Booking

- Users can search for bus routes by entering departure and destination locations along with travel dates.
- Available routes and schedules are displayed, including details on bus models, and seat availability.

Booking Management

- Users can view and manage their bookings through a personal dashboard.

4.2 SYSTEM ACCESSING SYSTEM MODULE

4.2.1 Welcome Page

Diagram 4.2.1.1 shows the welcome page when admin or staff enters the system

The admin and staff need to choose the respective login page. The page shown to login as admin or as staff.

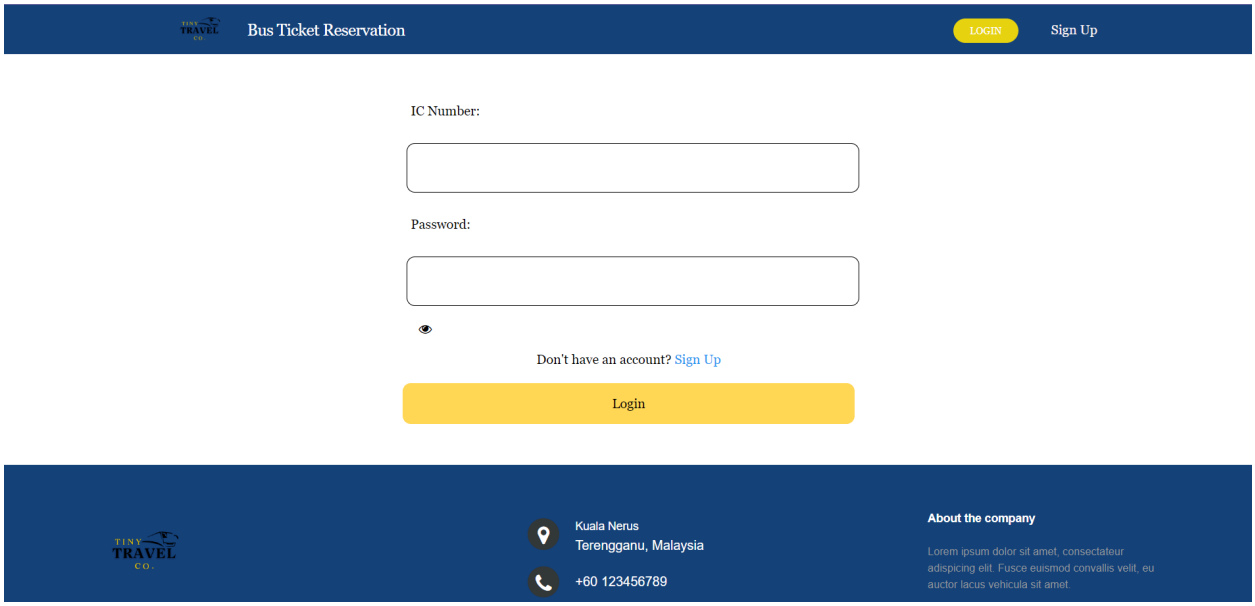


Diagram 4.2.1.2 shows the welcome page when user enters the system

The user shows this as their welcome page when they enter the system.

4.2.2 Registration, Login and Authentication Module (By Harith)

4.2.2.1 Admin login

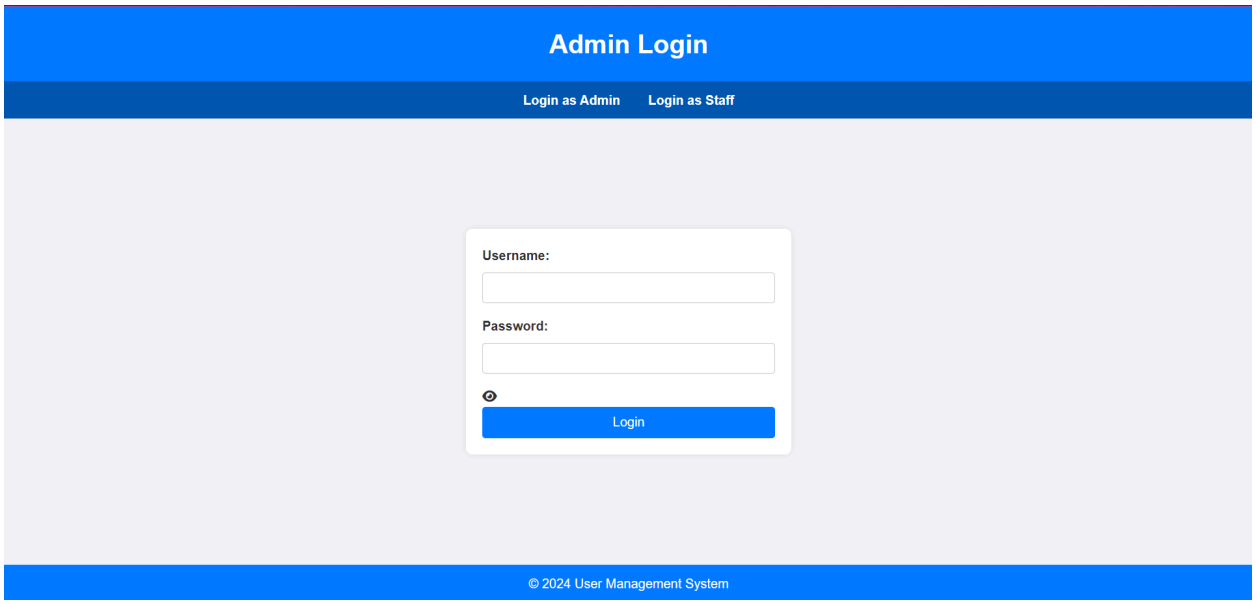


Diagram 4.2.2.1.1 shows the login page for admin

1. Enter the correct **Username**.
2. Enter the correct **Password**.
3. Click the **Login** button.

4.2.2.2 Wrong Password Or Username for Admin

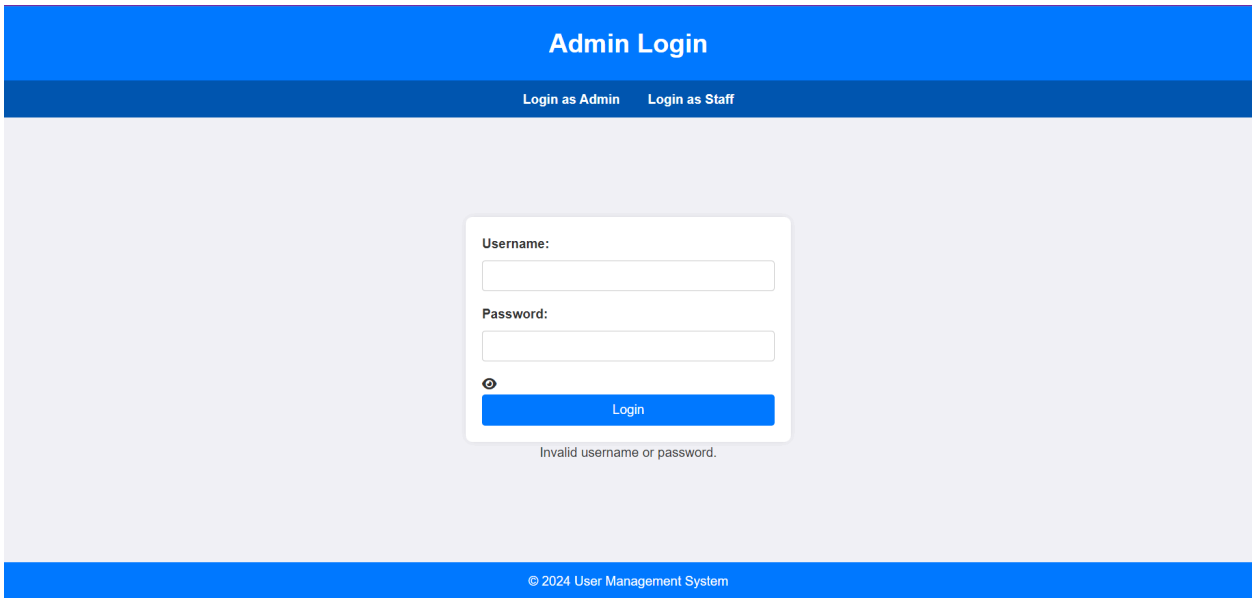


Diagram 4.2.2.2.2 shows the error notification if enter invalid username or password

1. Admin have entered either invalid password or username or both username and password. The system will give the notification “Invalid username or password”. More attempts can be made after each failure.

4.2.2.3 Admin Dashboard

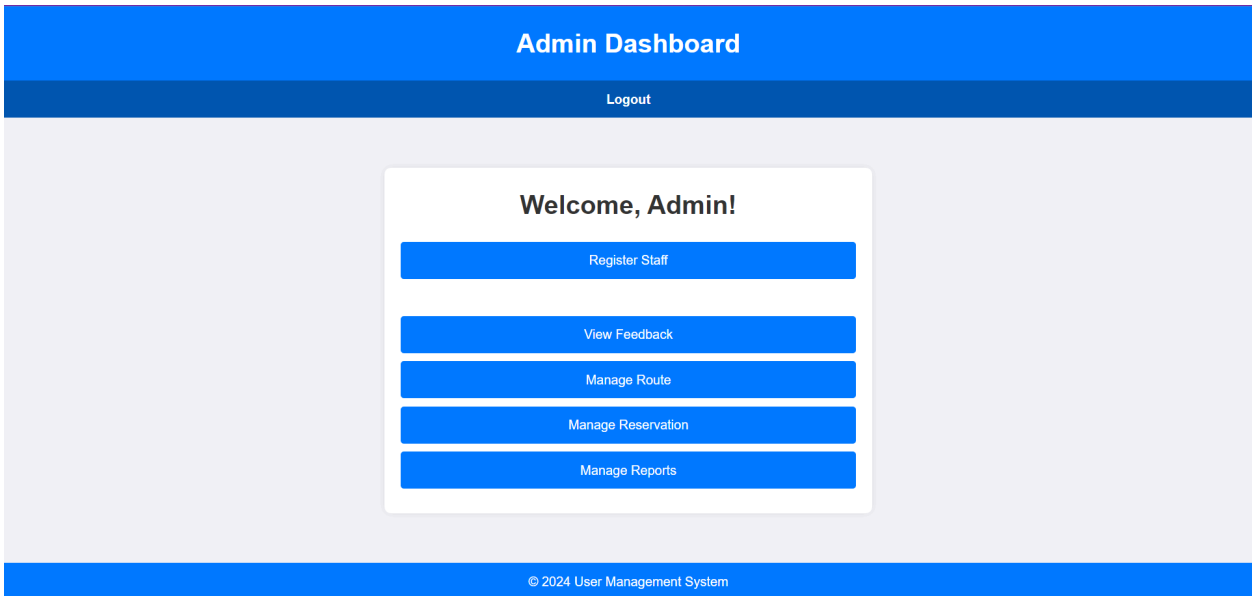


Diagram 4.2.2.3.1 shows the Admin Dashboard

1. When admin successfully login, admin will go to the Admin Dashboard.
2. Admin may click the button they want that being shown in the page
3. Admin may logout by clicking the **Logout** button.

4.2.2.4 Register Staff

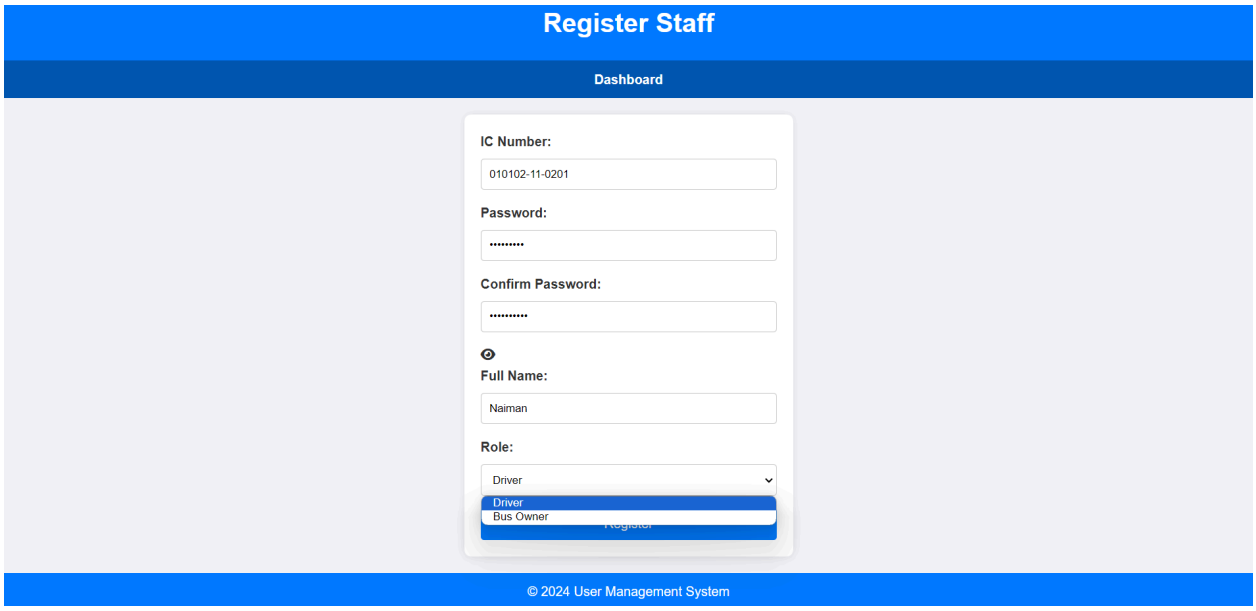


Diagram 4.2.2.4.1 shows the registration page for admin to register staffs

1. Enter valid **IC Number** that follows format “021122-11-0033”.
2. Enter valid **Password** that follow the format of at least one capital letter, one number, one special character, and be at least 8 characters long.
3. Enter valid **Fullname**.
4. Choose the **Role**.
    - Driver
    - Staff
5. Click the **Register** button.

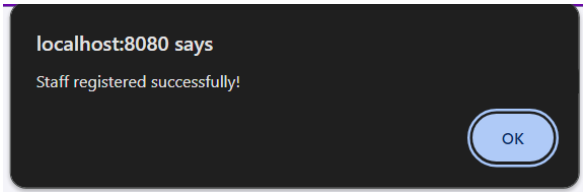


Diagram 4.2.2.4.2 shows the registration page for admin to register staffs

1. System will notify on successfully register the 'Driver' or 'Staff'.
2. Next, click the **OK** button.

4.2.2.5 IC Number Already Exist

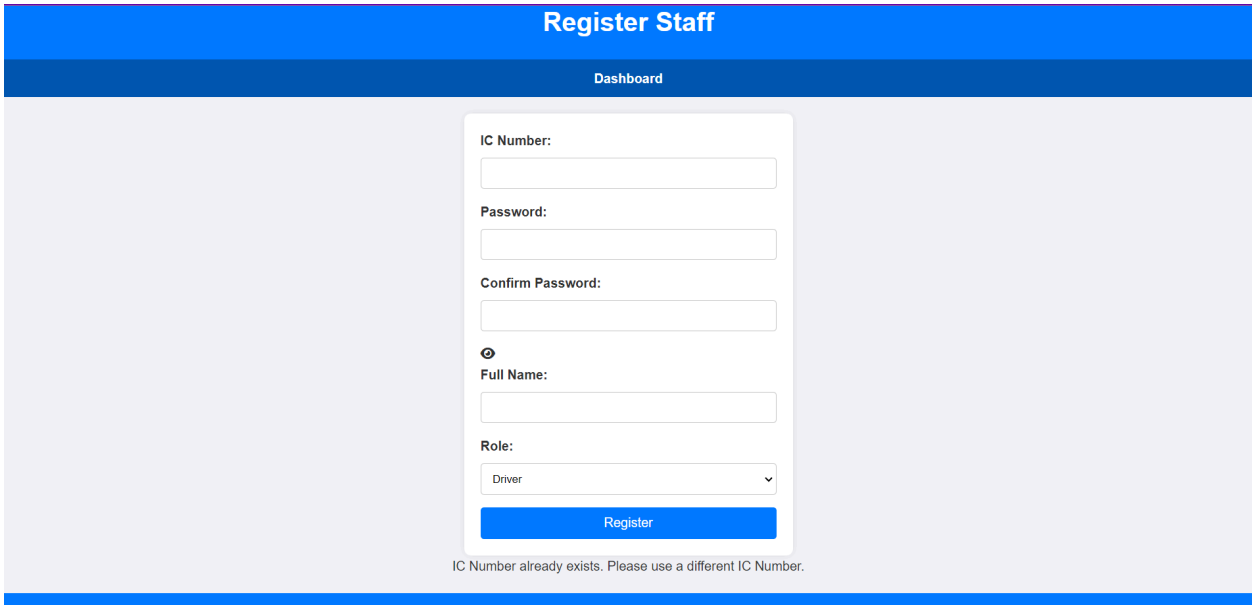


Diagram 4.2.2.5.1 shows the notification if the IC number already exist in the system

1. Admin enter the IC Number that already exists in the database that the system will give notification "IC Number already exists. Please use a different IC Number". Admin must check back the IC Number entered.

4.2.2.6 Staff Login

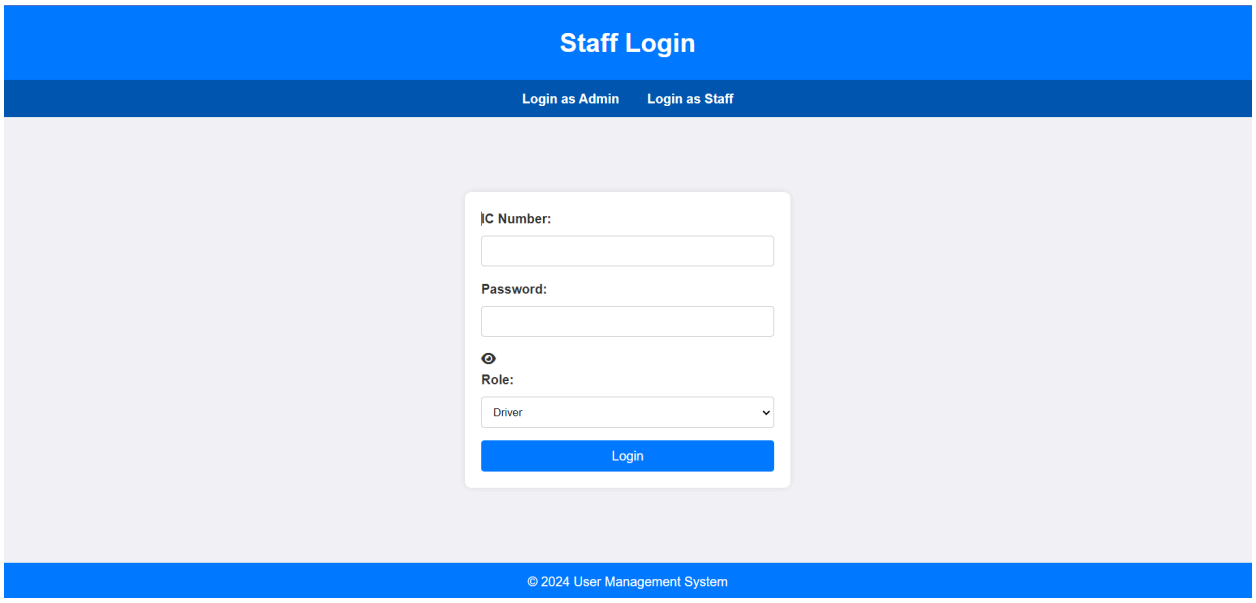


Diagram 4.2.2.6.1 shows the login page for staff

1. Enter the correct **IC Number**.
2. Enter the correct **Password**.
3. Choose respective **Role.**
    - Driver
    - Staff
4. Click the **Login** button.

4.2.2.7 Wrong Password Or Username for Staff

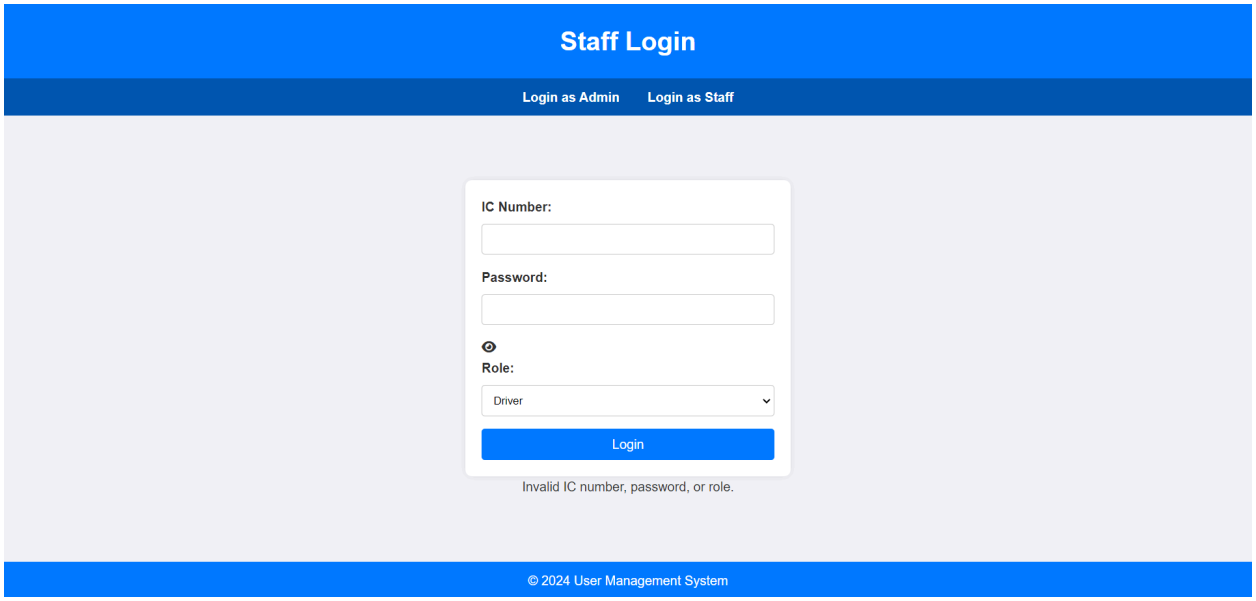


Diagram 4.2.2.7.1 shows the error notification if enter invalid username or password

1. Admin has entered either invalid password or IC Number or Role or all of it. The system will give the notification “Invalid IC number,password, role”. More attempts can be made after each failure.

4.2.2.8 Driver Dashboard

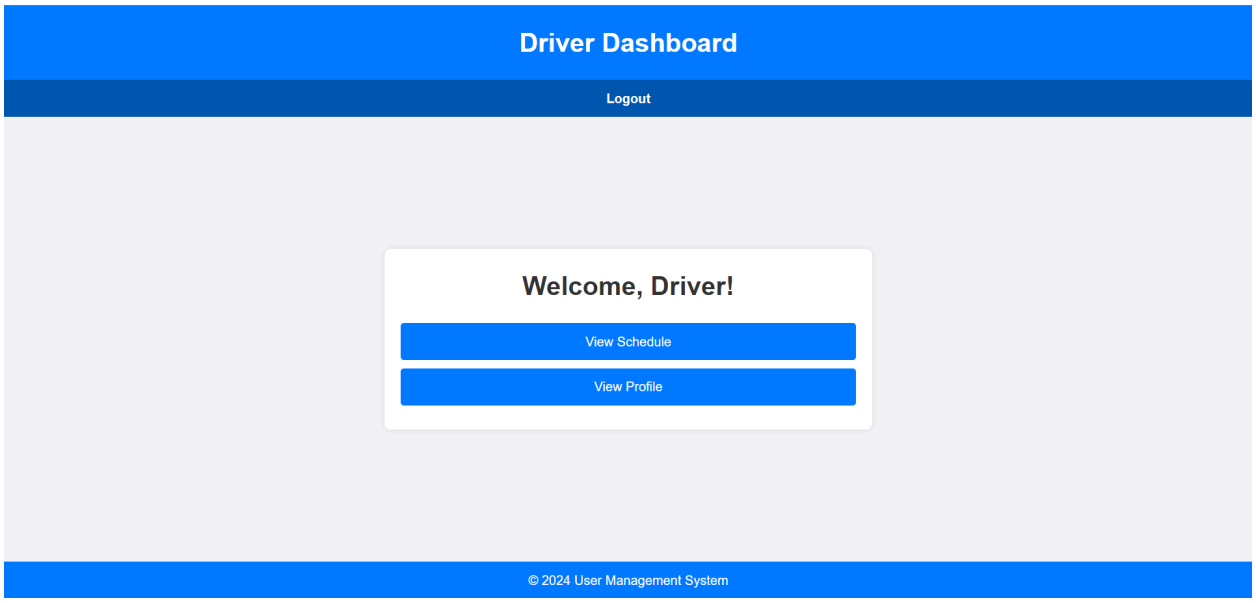


Diagram 4.2.2.8.1 shows the Admin Dashboard

1. When ‘Driver’ successfully logs in, the Driver will go to the Driver Dashboard.
2. Driver may click the button they want that being shown in the page
3. Driver may logout by clicking the **Logout** button.

4.2.2.9 Staff Dashboard

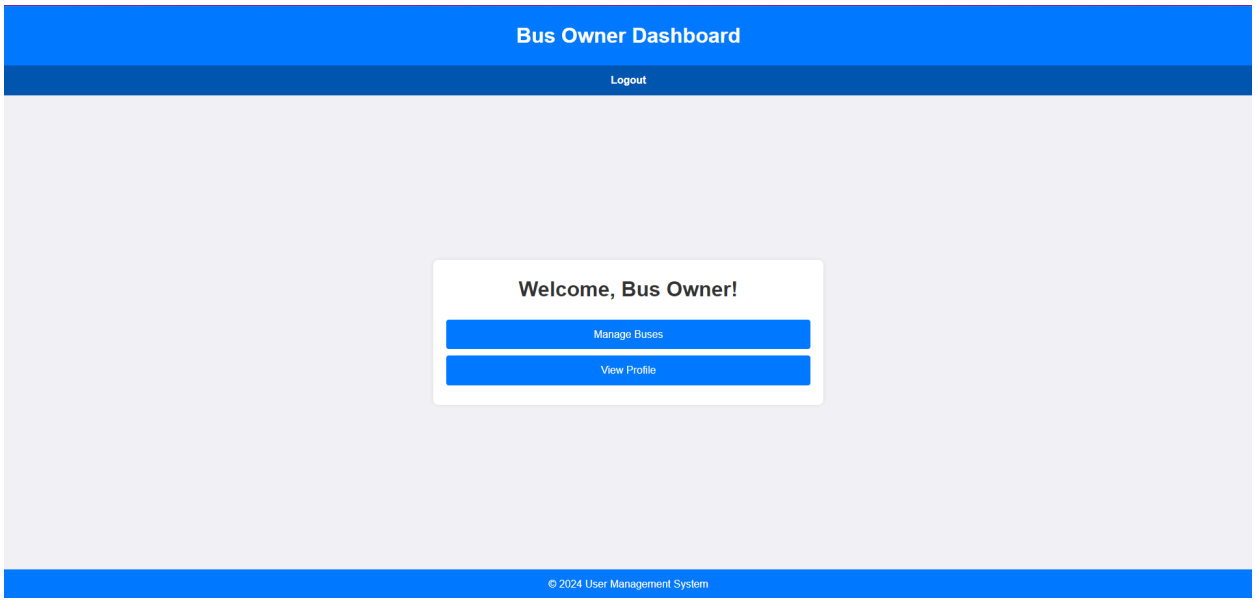


Diagram 4.2.2.9.1 shows the Admin Dashboard

1. When ‘Staff’ successfully logs in, the Driver will go to the Staff Dashboard.
2. Staff may click the button they want that being shown in the page
3. Staff may logout by clicking the **Logout** button.

4.2.2.10 User Sign Up

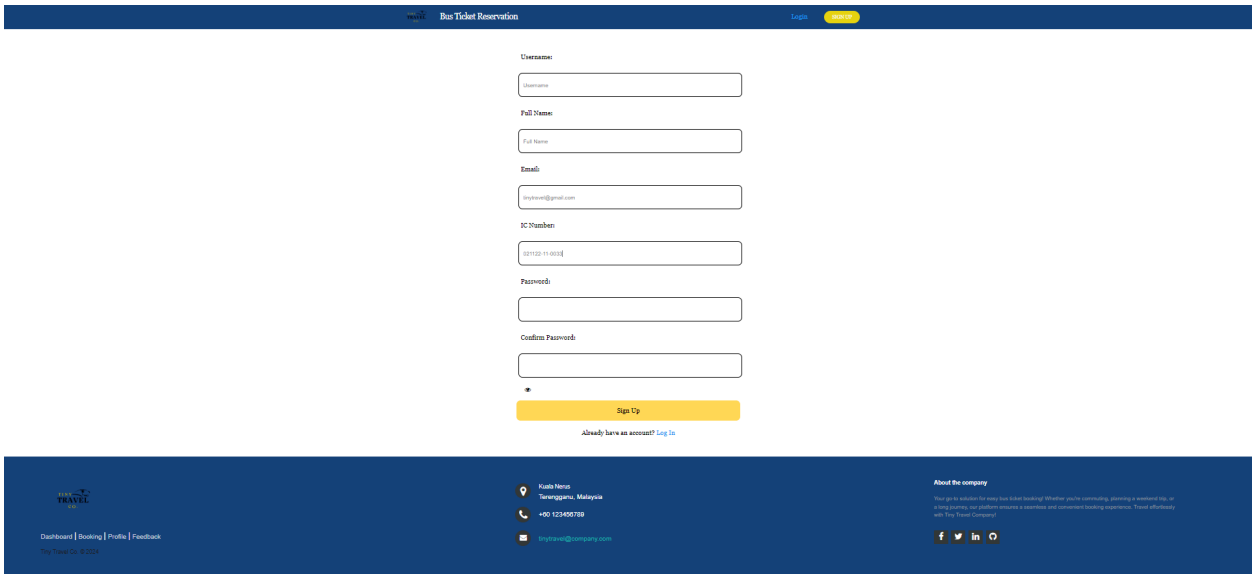


Diagram 4.2.2.10.1 shows the Sign Up page for User

1. Enter the valid **Username**.
2. Enter the valid **Fullname**.
3. Enter the valid **Email** that follows format 'tinytravel@gmail.com'.
4. Enter the valid **IC Number** that follows format '021122-11-0033'
5. Enter valid **Password** that follow the format of at least one capital letter, one number, one special character, and be at least 8 characters long.
6. Re-enter the password to match the password entered.
7. Click the **Sign Up** button.

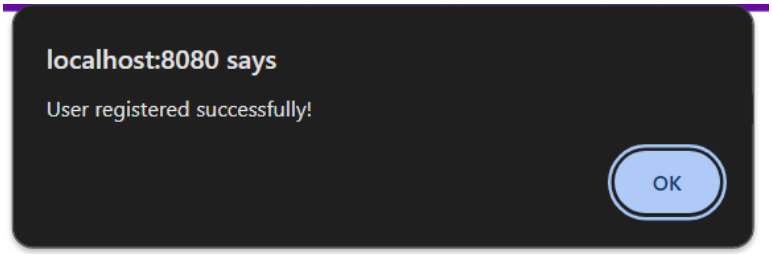


Diagram 4.2.2.10.2 shows the notification if the user successfully registered

1. The system will give the notification to the user upon successfully register their account
2. Next, click the **OK** button.

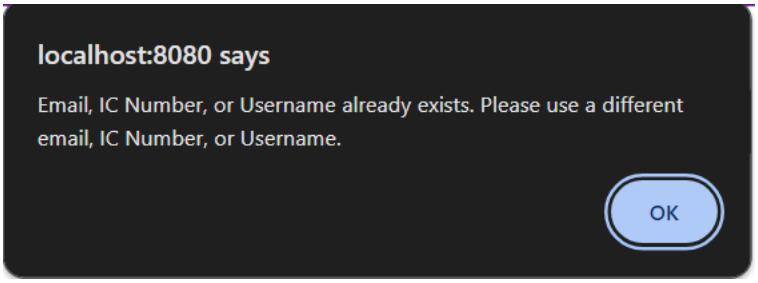


Diagram 4.2.2.10.3 shows the notification if Email,IC Number, or Username already exist

1. User has entered either Email, IC Number, or Username that already exists in the system. The system will give the notification “Email, IC Number, or Username already exists. Please use a different email, IC Number, or Username”. More attempts can be made after each failure. User must check back the input entered.
2. Next,click the **OK** button.

4.2.2.11 User Login

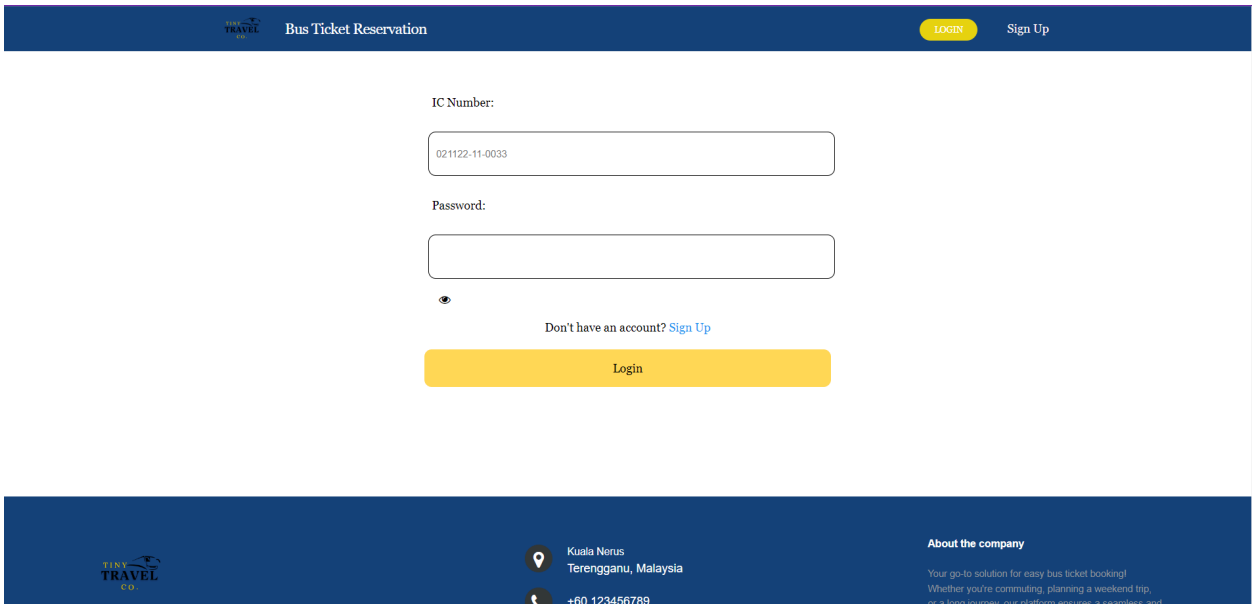


Diagram 4.2.2.11.1 shows the login page for User

1. Enter the correct **IC Number**.
2. Enter the correct **Password**.
3. Click the **Login** button.

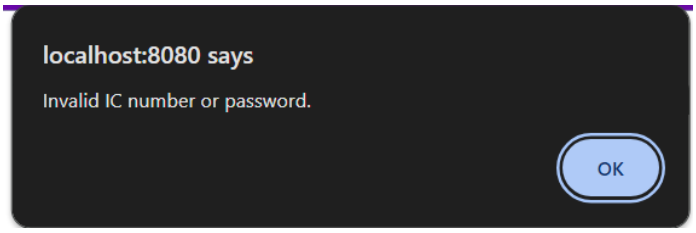


Diagram 4.2.2.11.2 shows the login page for User

1. The system will give the notification either wrong IC Number or wrong password ot both IC Number and password.
2. Next, click the **OK** button.

4.2.3 Feedback Module(By Harith)

4.2.3.1 View Feedback for Admin

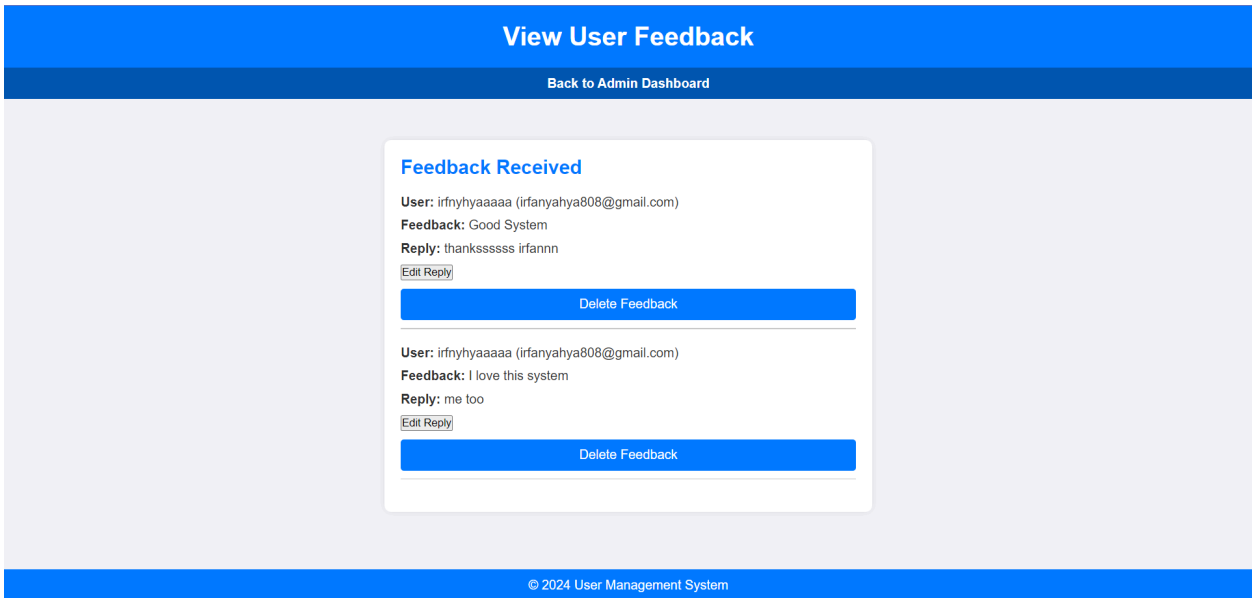


Diagram 4.2.3.1.1 shows the User Feedback in for admin to view

1. The admin may view all the feedback that users send to the system.

4.2.3.2 Reply,Edit or Delete Feedback for Admin

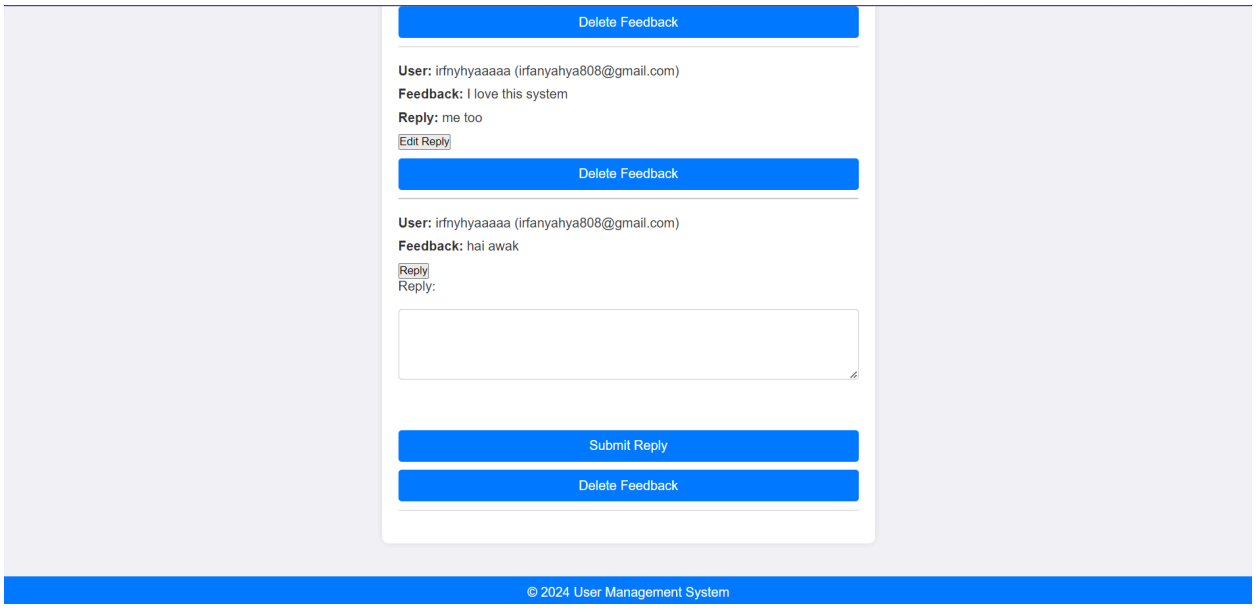


Diagram 4.2.3.2.1 shows the reply box for admin to reply the feedback

1. Click the **Reply** button.
2. Write the reply message.
3. Click the **Submit Reply** button.
4. Click the **Delete Feedback** button to delete the feedback.

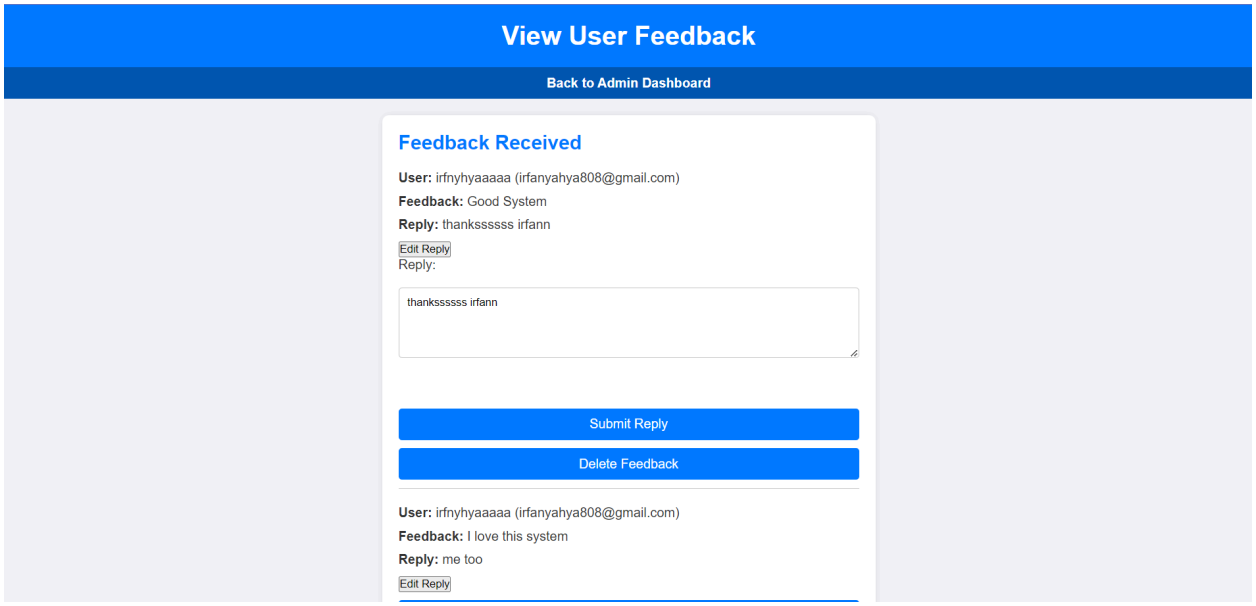


Diagram 4.2.3.2.2 shows the reply box for admin to edit replied feedback

1. Click **Edit Reply** to edit submitted reply
2. Click the **Submit Reply** button.

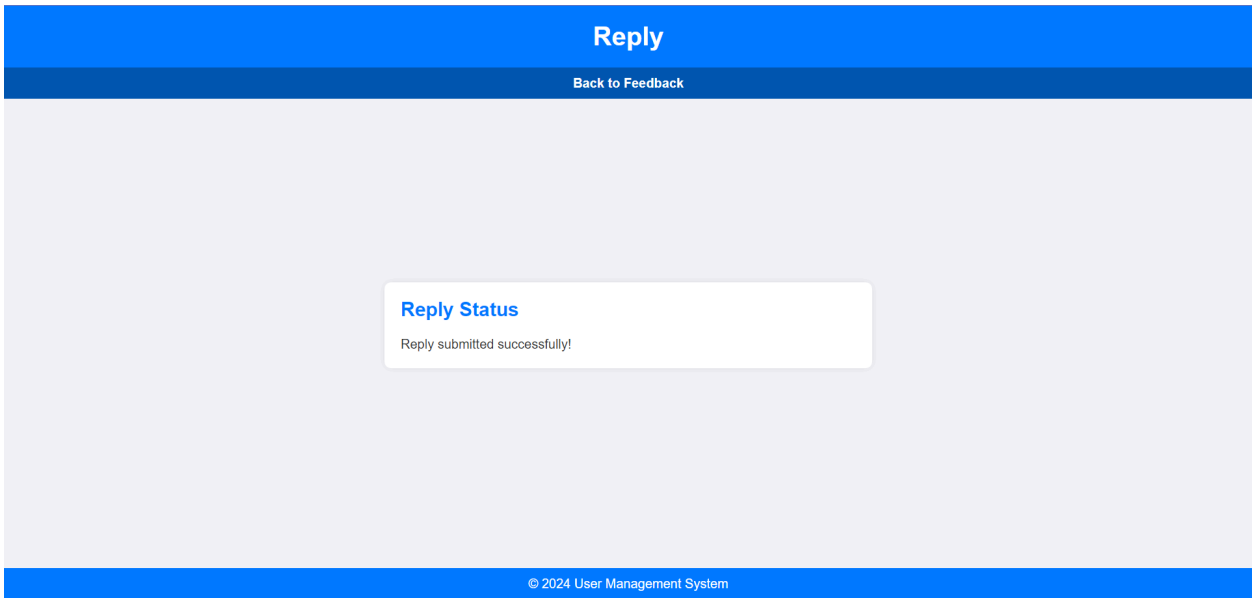


Diagram 4.2.3.2.3 shows the notification for reply successfully submitted

1. The system will give notification for every reply submitted.

4.2.3.3 Give Feedback for User

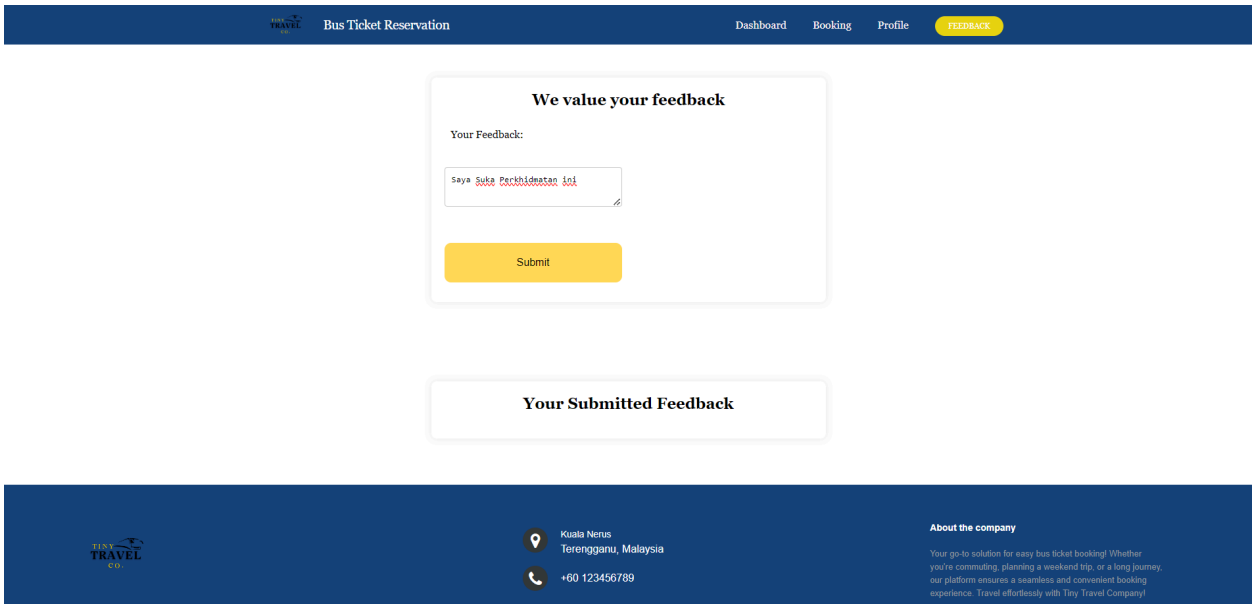


Diagram 4.2.3.3.1 shows the interface for user to give feedback

1. Write feedback in the box that is being provided.
2. Click the **Submit** button.

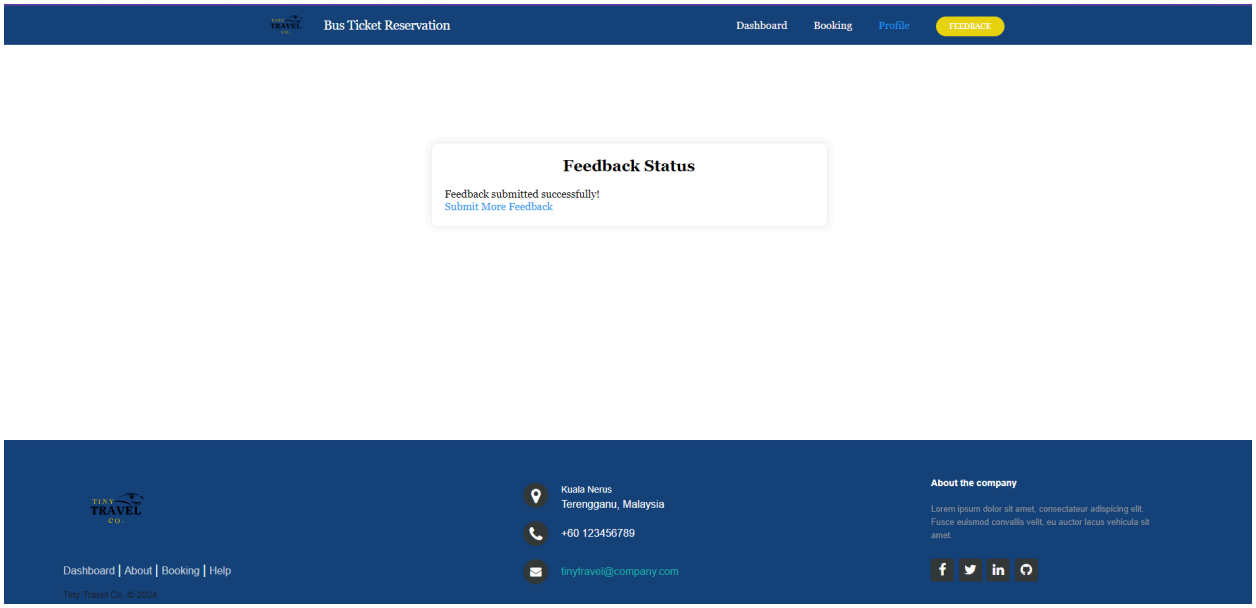


Diagram 4.2.3.3.1 shows the interface for user to give feedback

1. The system will give the notification on successfully submitting the feedback.
2. Click the **Submit More Feedback** button to submit more feedback.

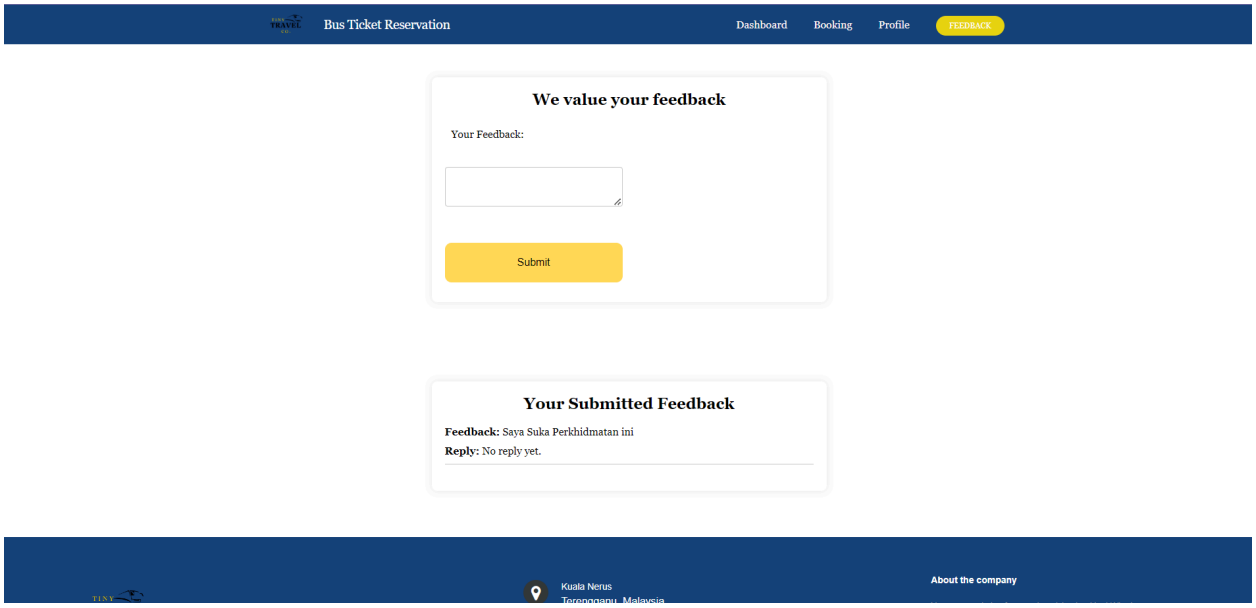


Diagram 4.2.3.3.2 shows the submitted feedback

1. The system shows to the user the feedback that they submit and the reply from the admin.

4.2.4 Manage Route Module(By Irfan)

4.2.4.1 Add, Update Or Delete Route

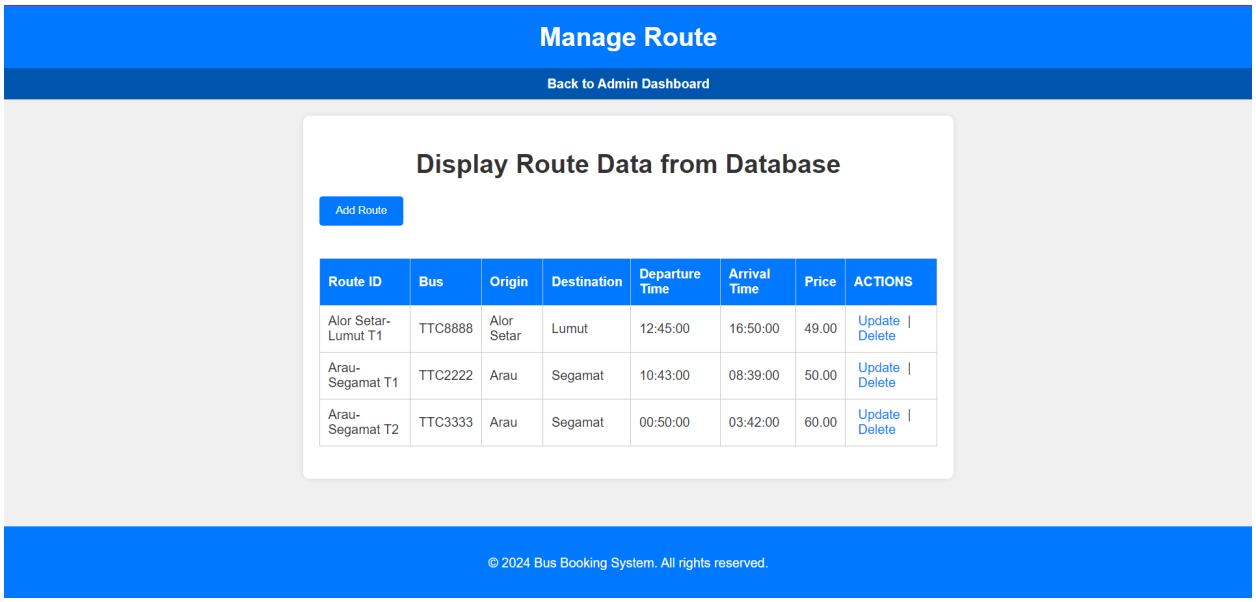


Diagram 4.2.4.1.1 shows the interface for admin to manage route

1. Click the **Add Route** button to add a new route.
2. Click the **Update** button to edit the existing route.
3. Click the **Delete** button to delete the existing route.

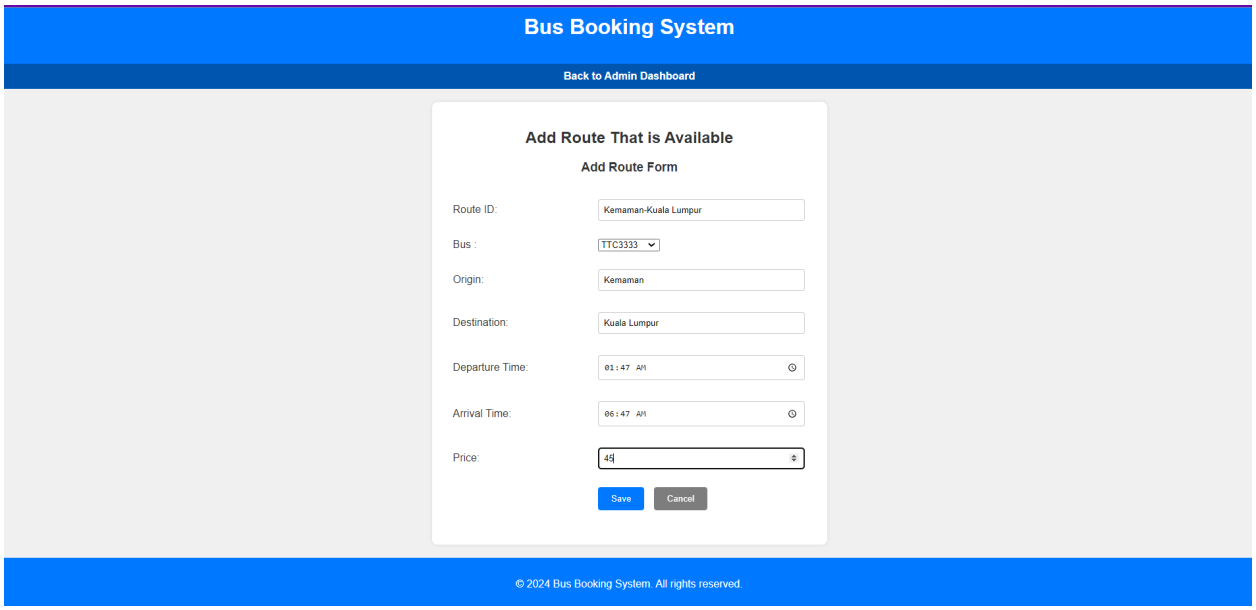


Diagram 4.2.4.1.2 shows the interface for admin to add route

1. Enter valid **Route ID**. Example : Kemaman-Kuala Lumpur.
2. Choose Bus that are already being registered into the system.
3. Enter the **Origin**. Example : Kemaman.
4. Enter the **Destination**. Example : Kuala Lumpur.
5. Select **Departure Time**.
6. Select **Arrival Time**.
7. Enter the **Price**. Example : 40.50
8. Click the **Save** button to save the route or **Cancel** button to terminate the process.

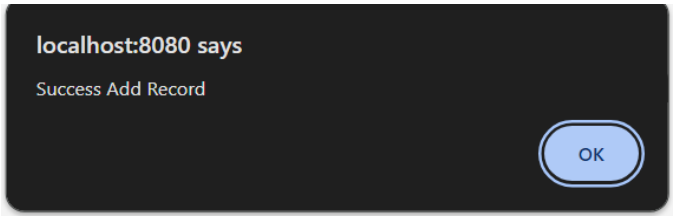


Diagram 4.2.4.1.3 shows the notification if route successfully added

1. System will give notification “Success Add Record” upon successfully adding the route.
2. Next, click the **OK** button.

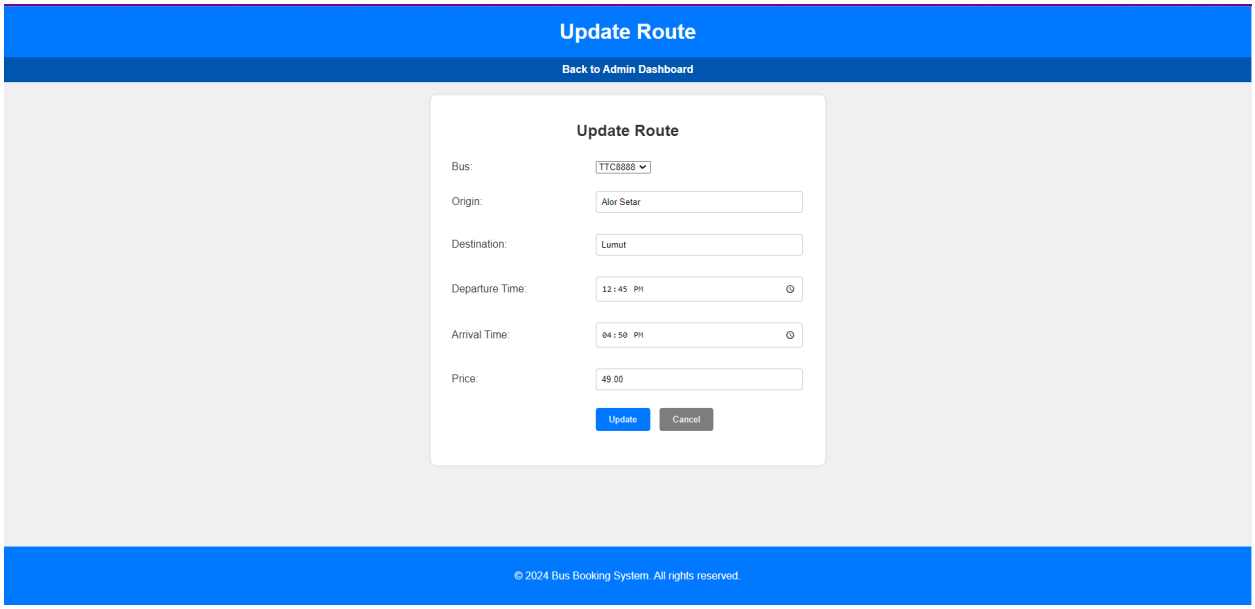


Diagram 4.2.4.1.4 shows the interface to edit route

1. Change the thing that must be changed. Example : **Price** from 49.00 to 49.50
2. Click the **Save** button to save the edited route or **Cancel** button terminate the process.

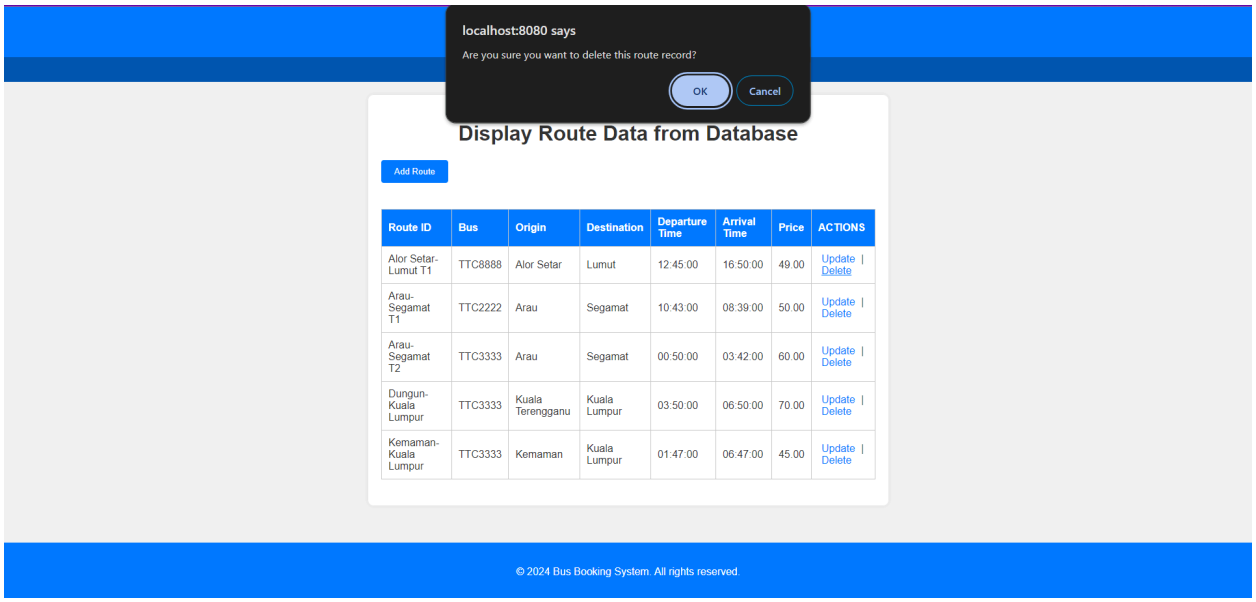


Diagram 4.2.4.1.5 shows the notification for confirmation on deletion of the route

1. The system will give notification “Are you sure you want to delete this route record?” to make confirmation on the deletion.
2. Next, click the **OK** button if confirm or **Cancel** button to terminate the process.

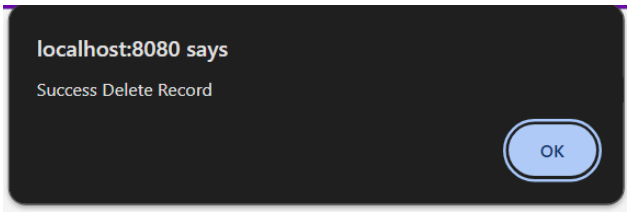
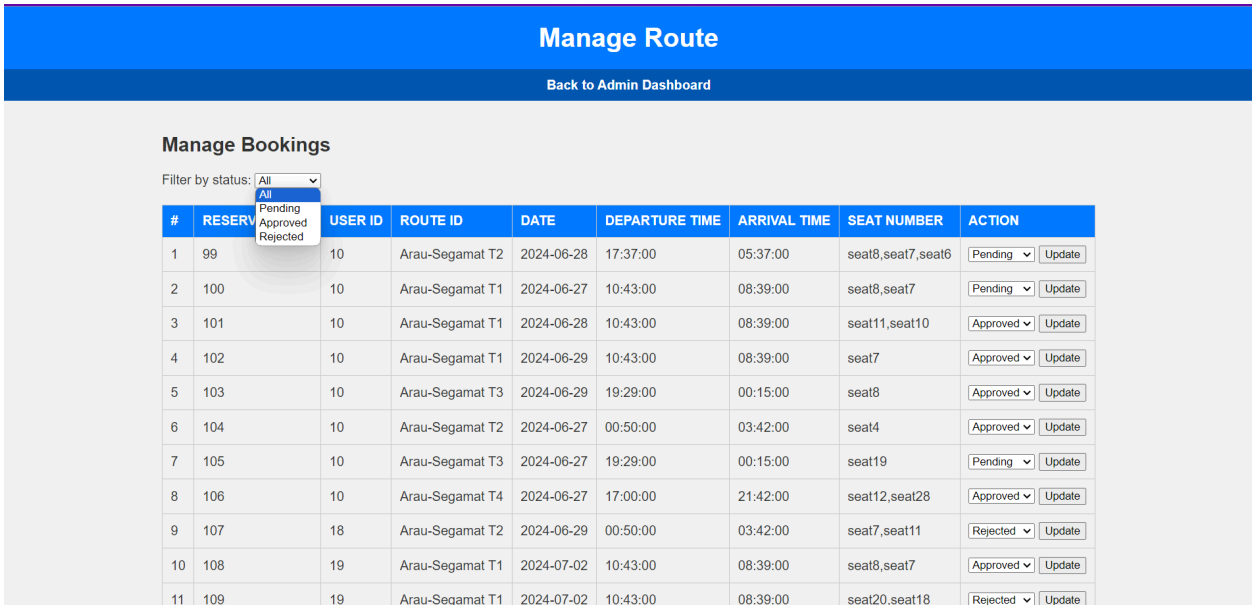


Diagram 4.2.4.1.6 shows the notification for successfully delete the route

1. The system will give notification “Success Delete Record” upon successfully deleting the route.
2. Next, click the **OK** button.

4.2.5 Manage Reservation Module Admin(By Aesya)

4.2.5.1 View Reservation



#	RESERV	USER ID	ROUTE ID	DATE	DEPARTURE TIME	ARRIVAL TIME	SEAT NUMBER	ACTION
1	99	10	Arau-Segamat T2	2024-06-28	17:37:00	05:37:00	seat8,seat7,seat6	Pending Update
2	100	10	Arau-Segamat T1	2024-06-27	10:43:00	08:39:00	seat8,seat7	Pending Update
3	101	10	Arau-Segamat T1	2024-06-28	10:43:00	08:39:00	seat11,seat10	Approved Update
4	102	10	Arau-Segamat T1	2024-06-29	10:43:00	08:39:00	seat7	Approved Update
5	103	10	Arau-Segamat T3	2024-06-29	19:29:00	00:15:00	seat8	Approved Update
6	104	10	Arau-Segamat T2	2024-06-27	00:50:00	03:42:00	seat4	Approved Update
7	105	10	Arau-Segamat T3	2024-06-27	19:29:00	00:15:00	seat19	Pending Update
8	106	10	Arau-Segamat T4	2024-06-27	17:00:00	21:42:00	seat12,seat28	Approved Update
9	107	18	Arau-Segamat T2	2024-06-29	00:50:00	03:42:00	seat7,seat11	Rejected Update
10	108	19	Arau-Segamat T1	2024-07-02	10:43:00	08:39:00	seat8,seat7	Approved Update
11	109	19	Arau-Segamat T1	2024-07-02	10:43:00	08:39:00	seat20,seat18	Rejected Update

Diagram 4.2.5.1.1 shows all the reservation made by the users

1. System shows all the reservations made by the users with the status of approved, pending and rejected.
2. The reservations can be filter to three categories that admin may choose:
    - Pending
    - Approved
    - Rejected

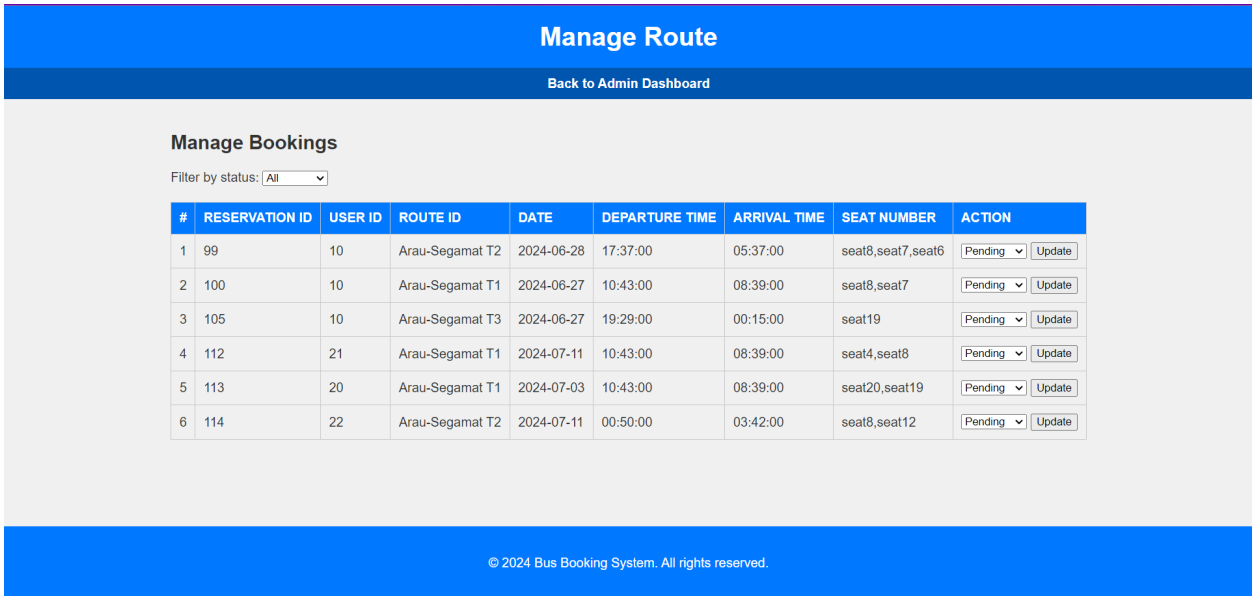


Diagram 4.2.5.1.2 shows all the reservation made by the users that 'Pending'

1. The system shows all the reservations that are 'Pending'.

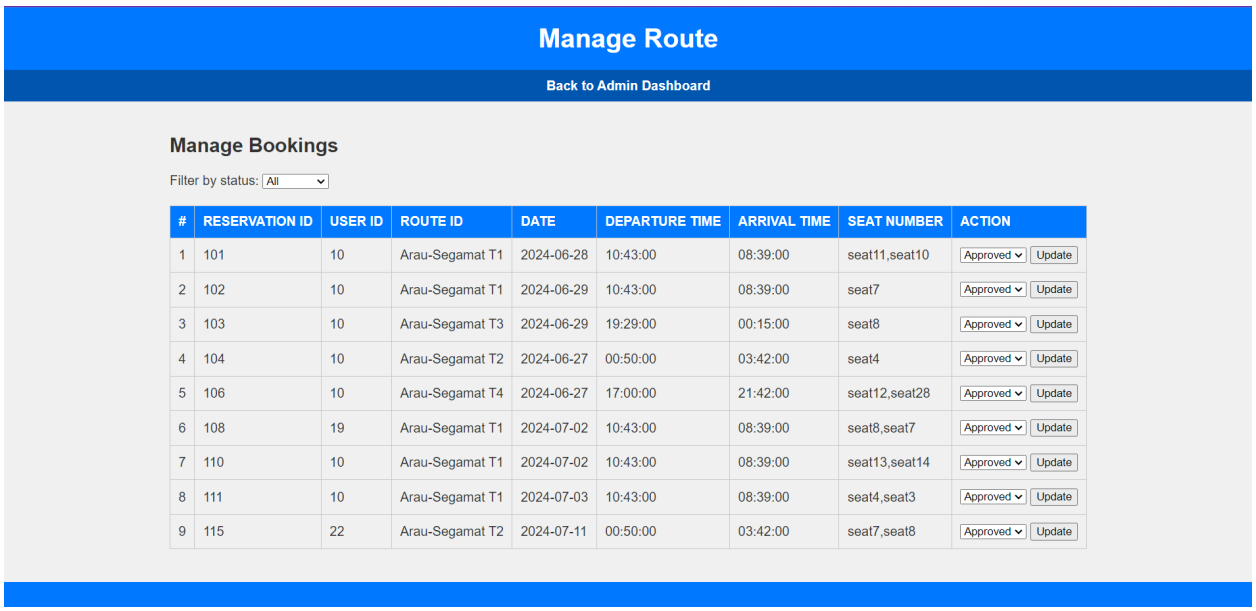


Diagram 4.2.5.1.3 shows all the reservation made by the users that 'Approved'

1. The system shows all the reservations that are 'Approved'.

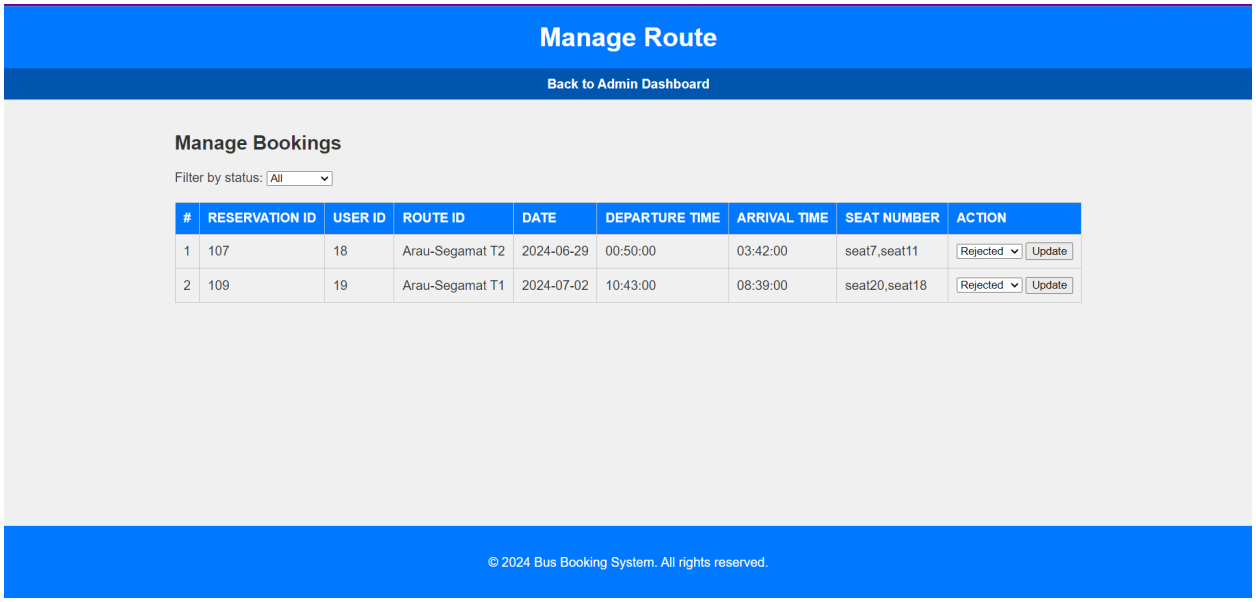


Diagram 4.2.5.1.4 shows all the reservation made by the users that 'Rejected'

1. The system shows all the reservations that are 'Rejected'.

4.2.5.2 Update Reservation Status

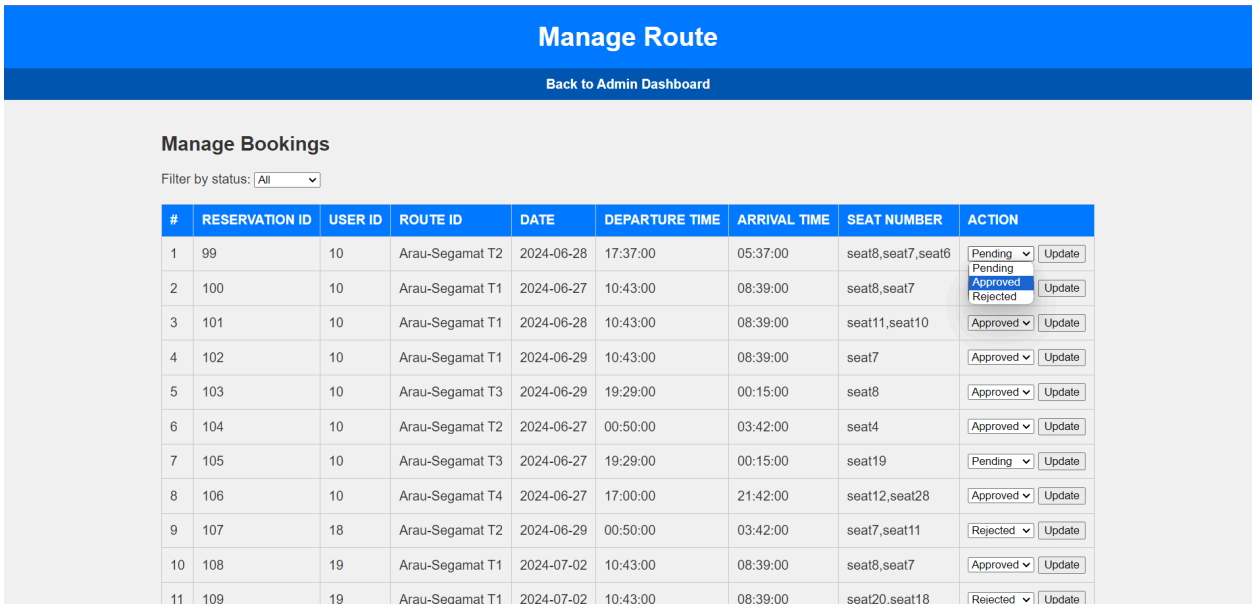


Diagram 4.2.5.2.1 shows the status change of the reservation

1. Admin choose between Pending, Approved and Rejected based on the data from the user.
2. Click the **Update** button.

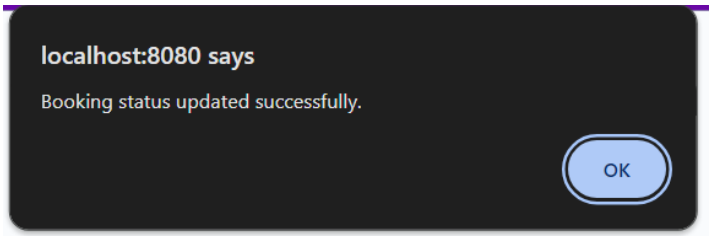


Diagram 4.2.5.2.2 shows the notification of successfully update the status

1. The system will give notification “Booking status updated successfully” upon successfully updating the status.
2. Next, click the **OK** button.

4.2.5.3 View Schedule

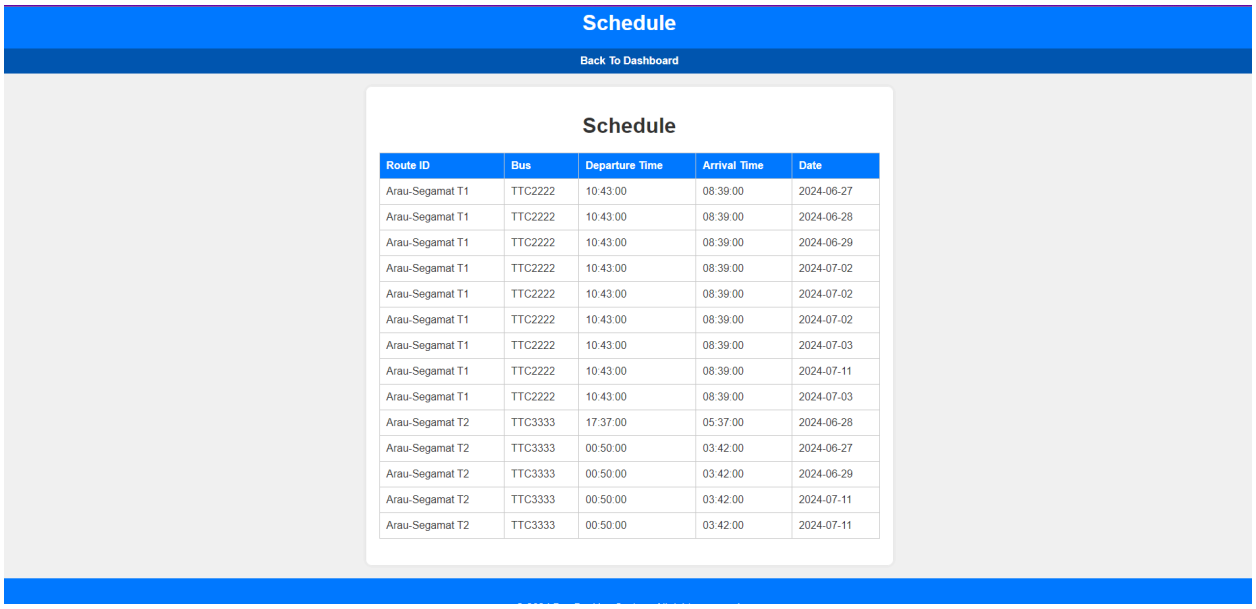


Diagram 4.2.6.1.1 shows the interface to view schedule for driver

1. Drivers may view their schedule on their trip of the day that has been assigned to them.

4.2.6 Report Module(By Haziqah)

4.2.6.1 Assign Driver to Bus

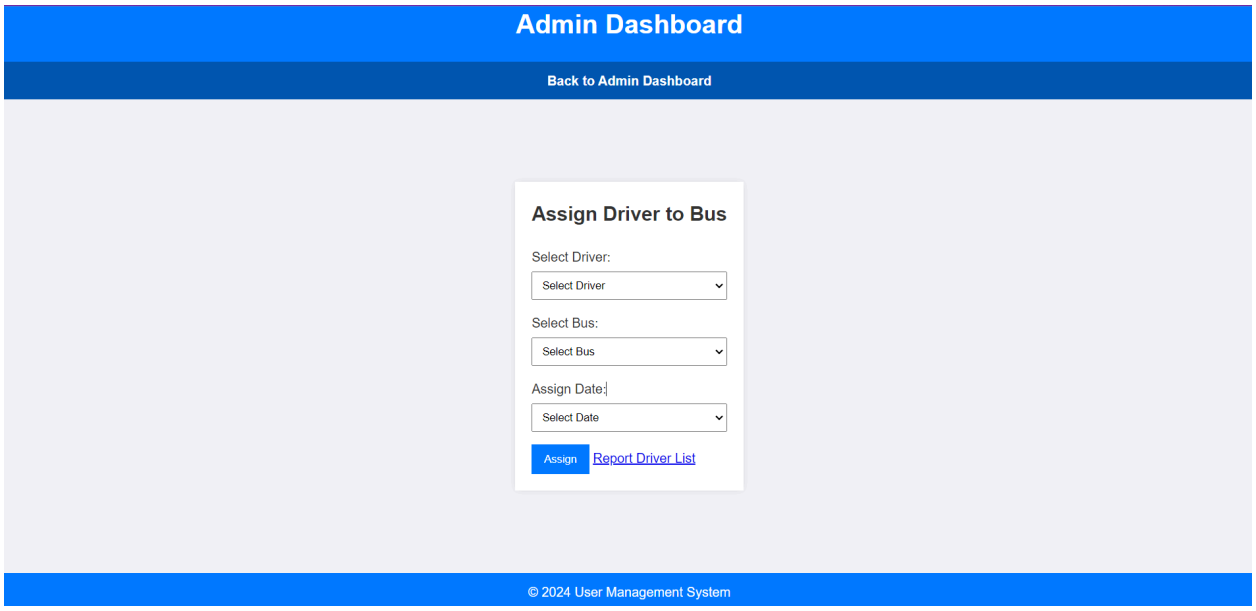


Diagram 4.2.6.1.1 shows the interface to Assign Driver to Bus

1. Select **Driver** that already registered into the system
2. Select **Bus** that already registered into the system
3. Assign **Date**.
4. Click the **Assign** button or click the **Report Driver List** button to view the report.

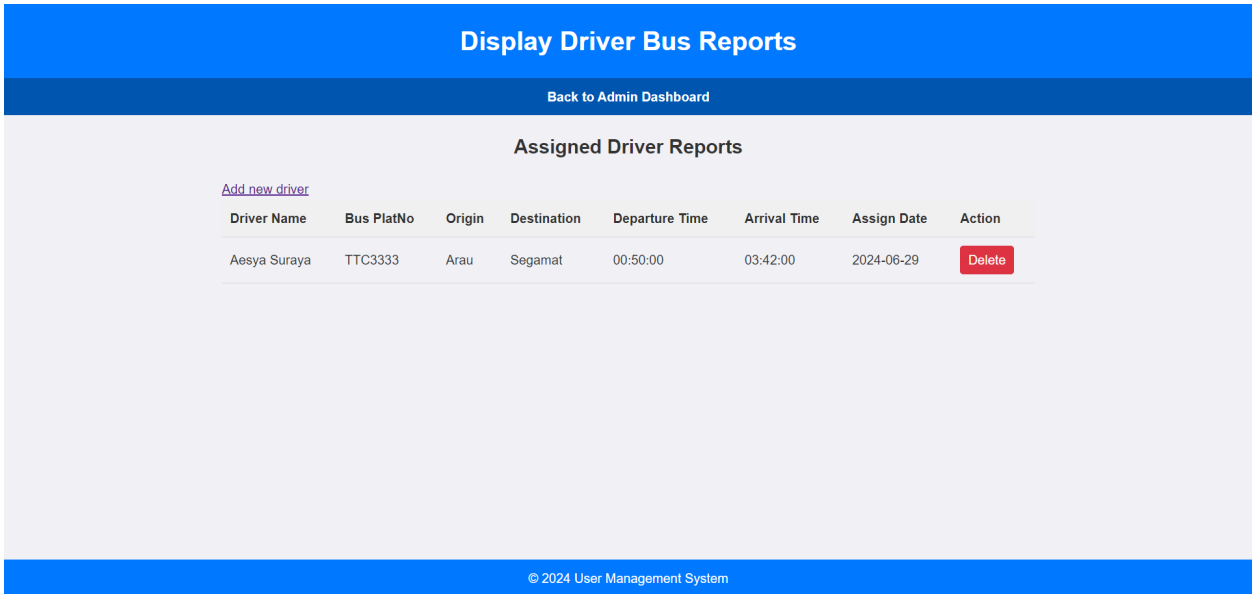


Diagram 4.2.6.1.2 shows the interface to view the report of the assigned driver

1. The system shows all the assigned driver to their specific bus and date
2. Click the **Delete** button to delete the record.

4.2.7 Manage Profile Module(By Haziqah)

4.2.7.1 View Profile

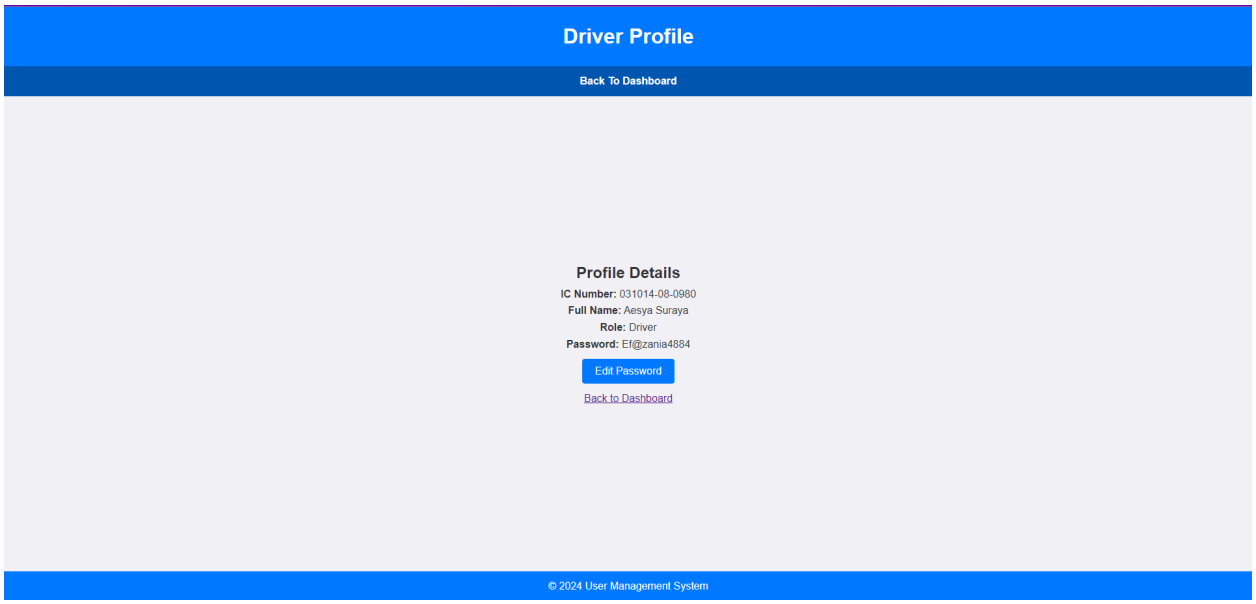


Diagram 4.2.7.1.1 shows the interface to view the profile of Driver

1. Driver login into their account
2. System shows the profile details of the Driver.
3. Click the **Edit Password** button to change password.

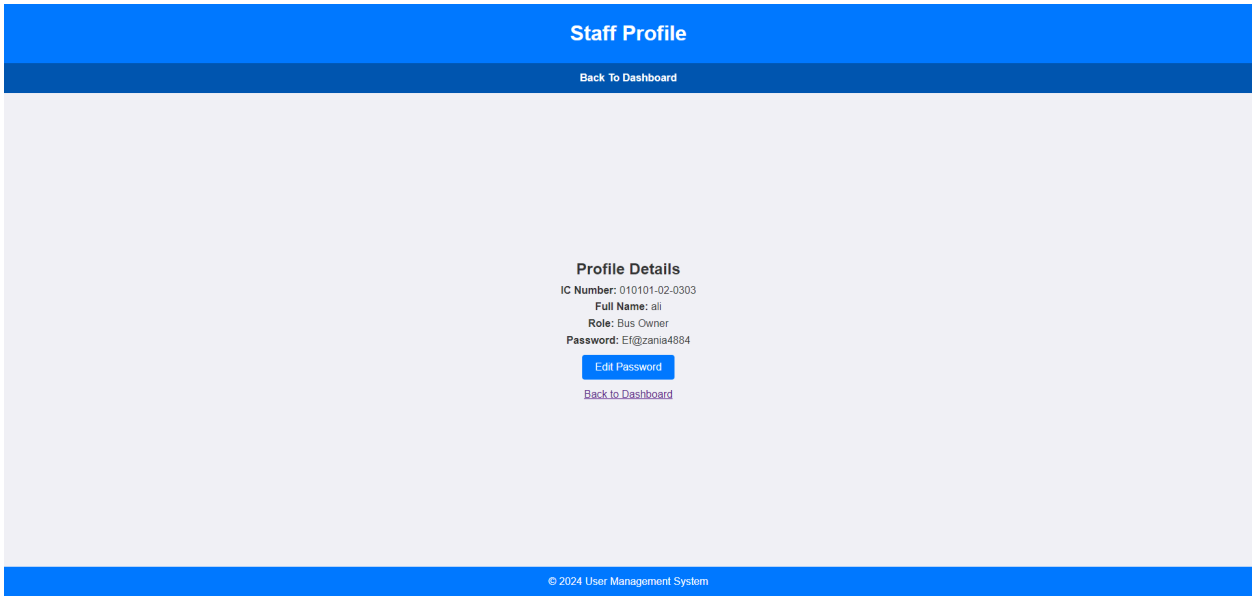


Diagram 4.2.7.1.2 shows the interface to view the profile of Staff

1. Staff login into their account.
2. System shows the details of the staff.
3. Click the **Edit Password** button to change password.

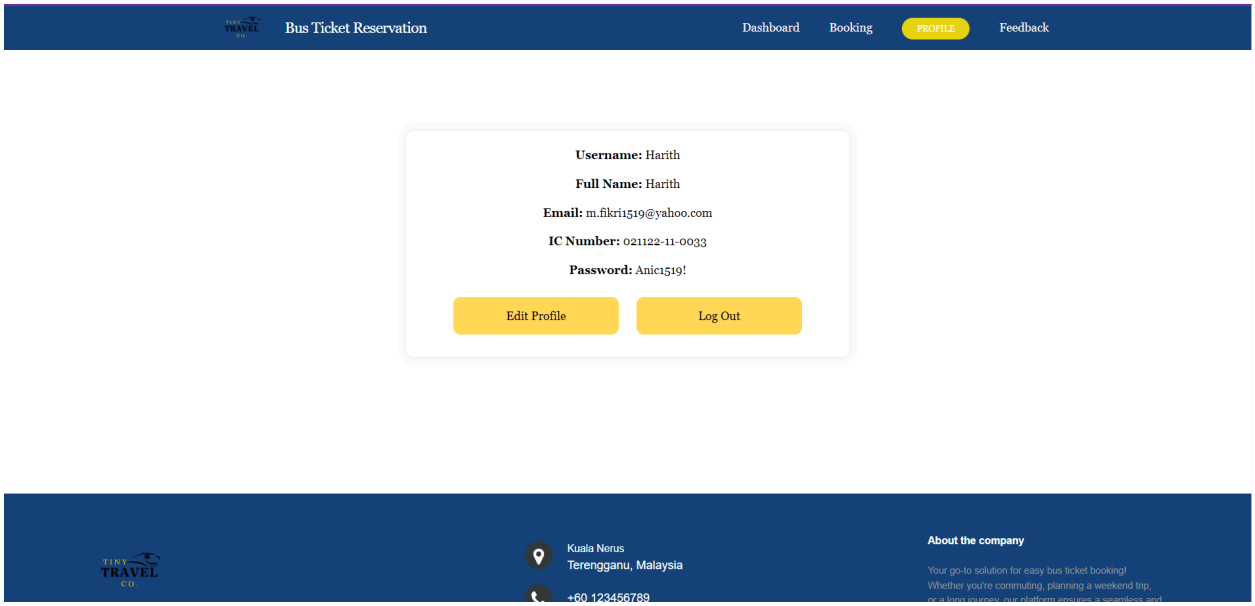


Diagram 4.2.7.1.3 shows the interface to view the profile of User

1. Users login into their account.
2. System shows the details of the User.
3. Click the **Edit Profile** button to change password.
4. Click the **Log Out** button to logout from the account.

4.2.7.2 Edit Profile

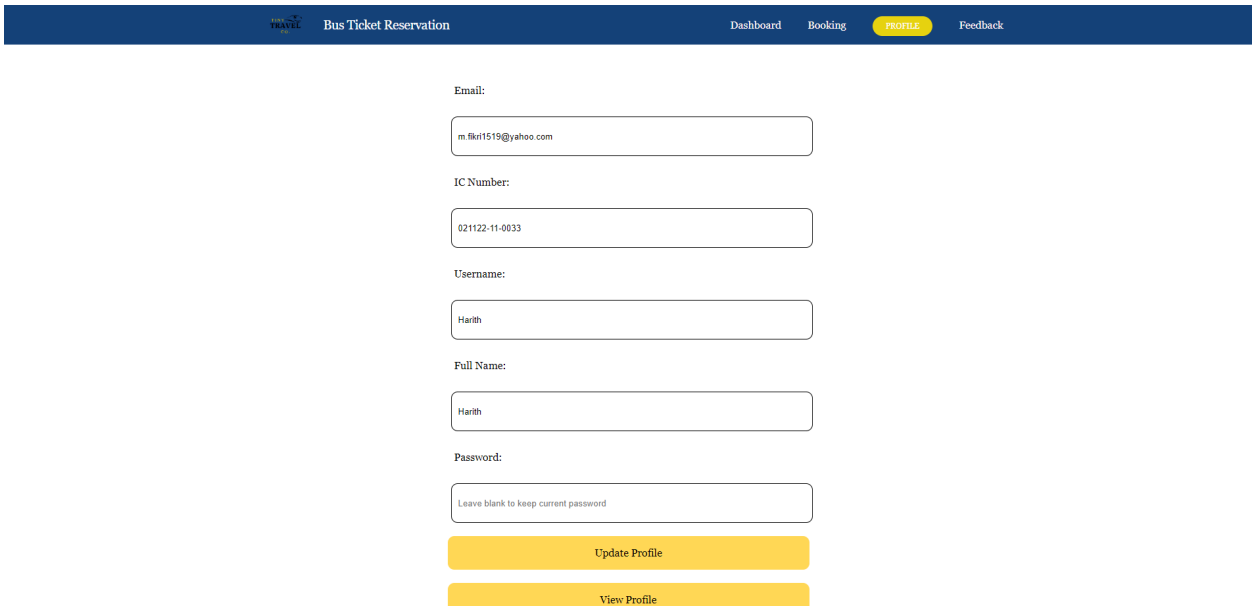


Diagram 4.2.7.2.1 shows the interface to edit user profile

1. Change the thing that wants to be changed. Example : **Full Name** from Harith to Harith Fikri.
2. Leave the password blank to keep the current password.
3. Click the **Update Profile** button to make changes.

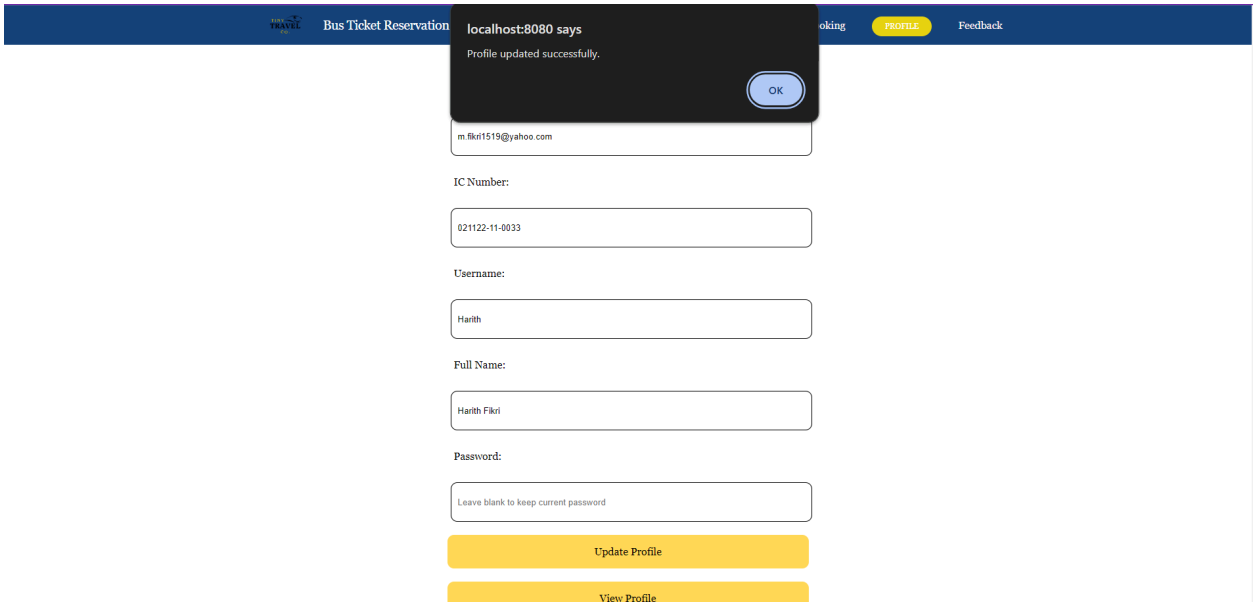


Diagram 4.2.7.2.2 shows the notification of the user profile successfully updated

1. System shows the notification “Profile updated successfully” upon successfully update the profile
2. Next, click the **OK** button.

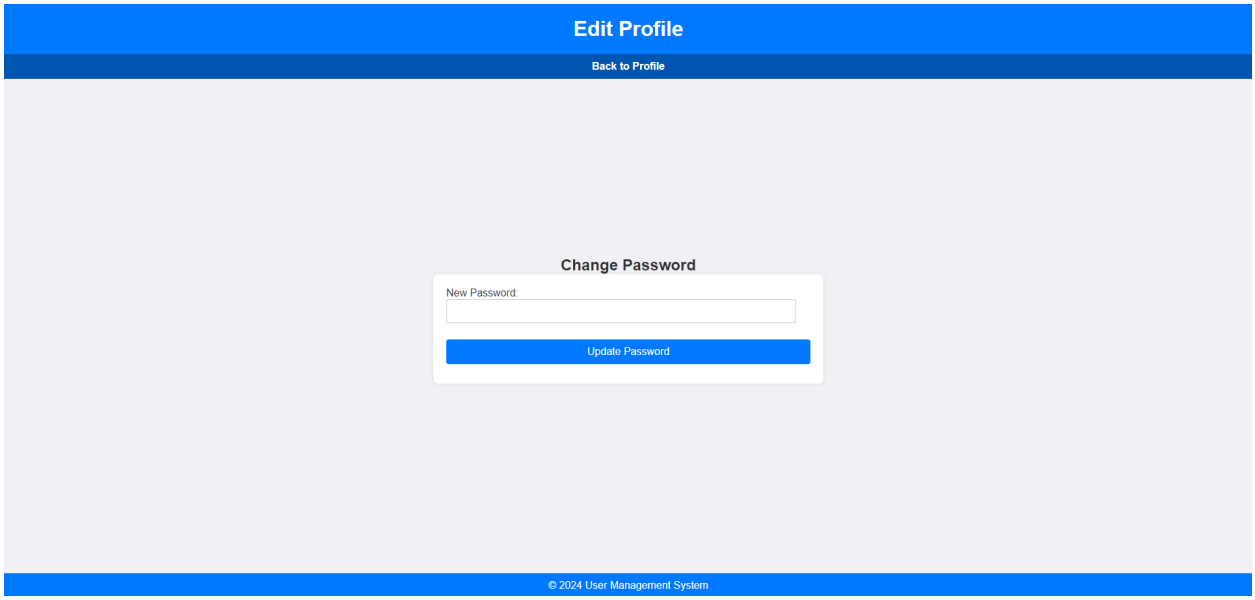


Diagram 4.2.7.2.3 shows the interface to edit driver and staff password

1. Enter valid **New Password**.
2. Click the **Update Password** button.

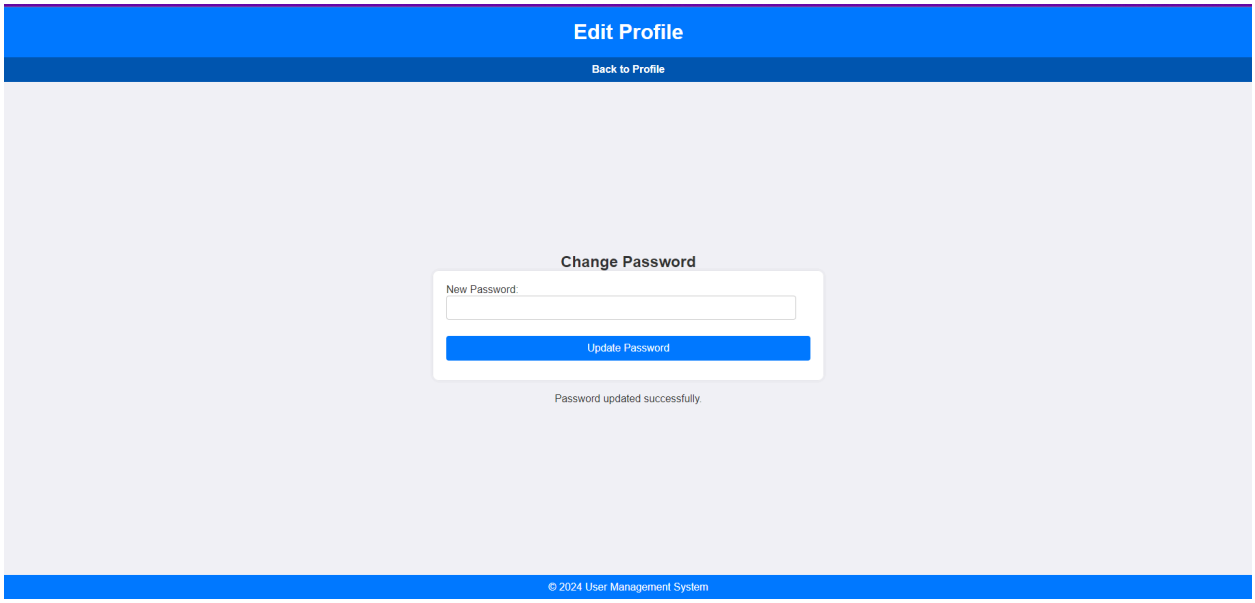


Diagram 4.2.7.2.4 shows the notification of the user profile successfully updated

1. System shows the notification “Password updated successfully” upon successfully updating the password.

4.2.8 Manage Bus Module(By Irfan)

4.2.8.1 Add, Update Or Delete Bus

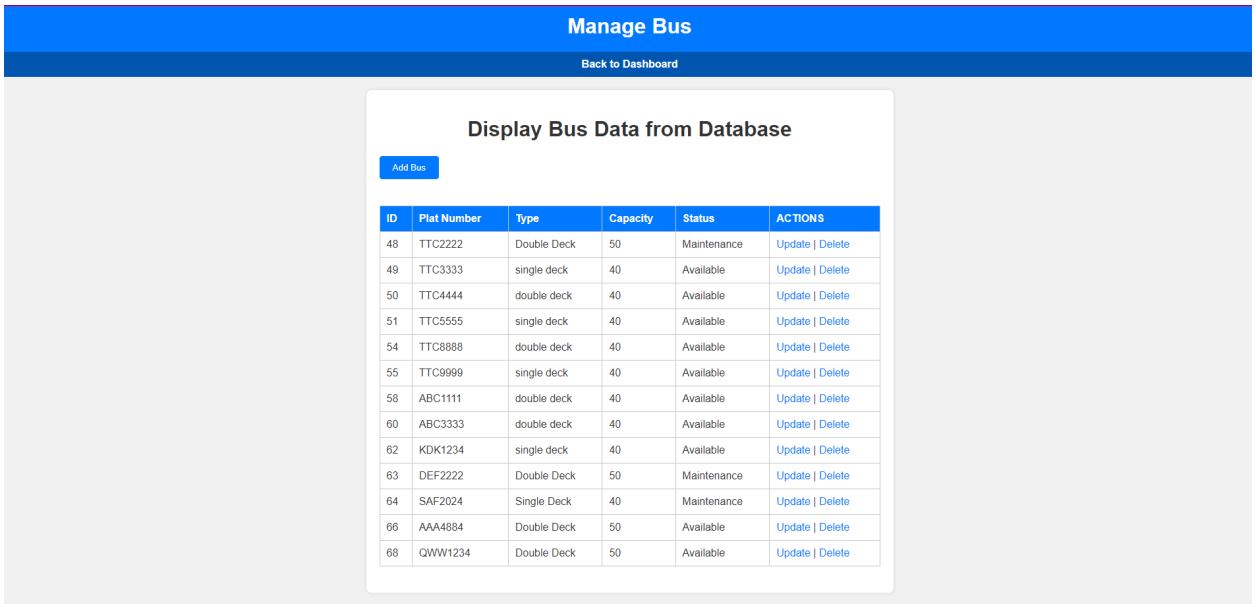


Diagram 4.2.8.1.1 shows the interface for staff to manage bus

1. Click the **Add Bus** button to add a new bus.
2. Click the **Update** button to edit the existing bus.
3. Click the **Delete** button to delete the existing bus.

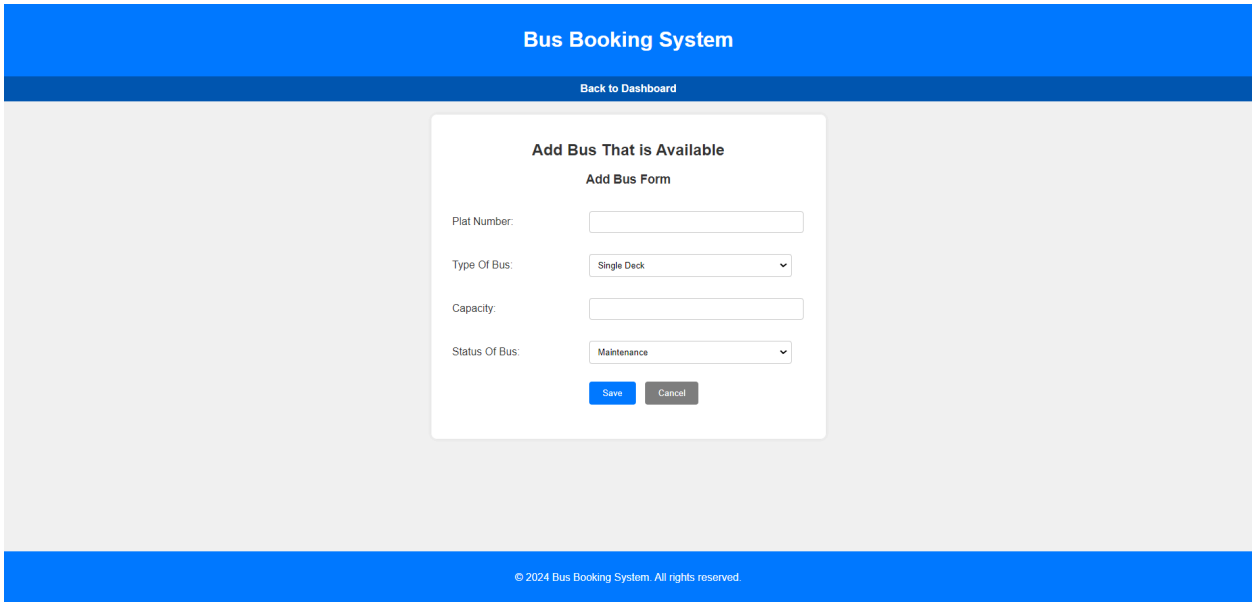


Diagram 4.2.8.1.2 shows the interface for staff to add bus

1. Enter valid **Plat Number**.
2. Choose **Type Of Bus**:
    - Single Deck
    - Double Deck
3. Enter the **Status Of Bus** :
    - Maintenance
    - Available
4. Click the **Save** button to save the bus or **Cancel** button to terminate the process.

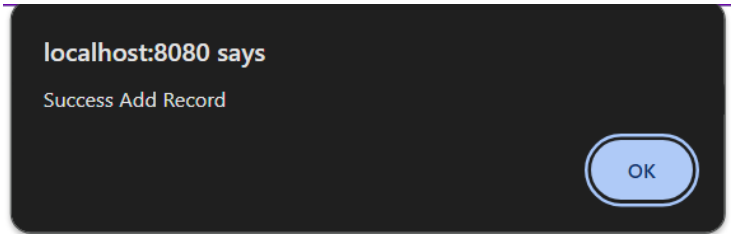


Diagram 4.2.8.1.3 shows the notification if bus successfully added

1. System will give notification “Success Add Record” upon successfully adding the bus.
2. Next, click the **OK** button.

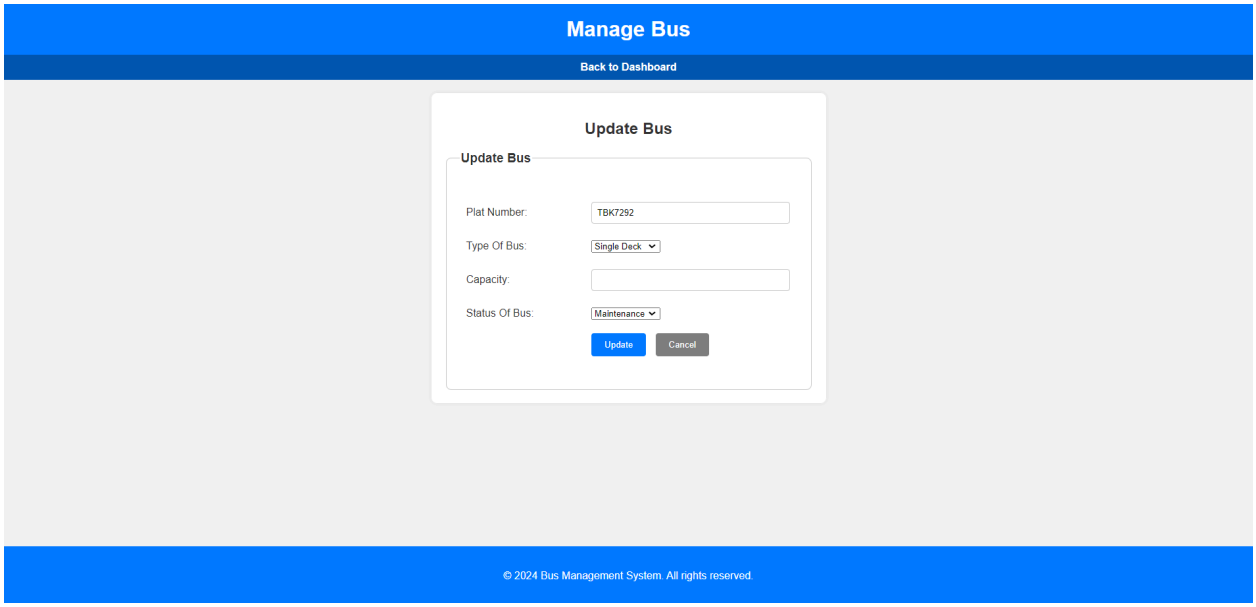


Diagram 4.2.8.1.4 shows the interface to edit route

1. Change the thing that must be changed. Example : **Status Of Bus** from ‘Maintenance’ to ‘Available’
2. Click the **Save** button to save the edited route or **Cancel** button terminate the process.

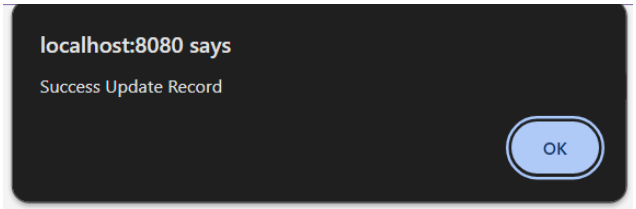


Diagram 4.2.8.1.5 shows the notification if bus successfully updated

1. System will give notification “Success Update Record” upon successfully updating the bus.
2. Next, click the **OK** button.

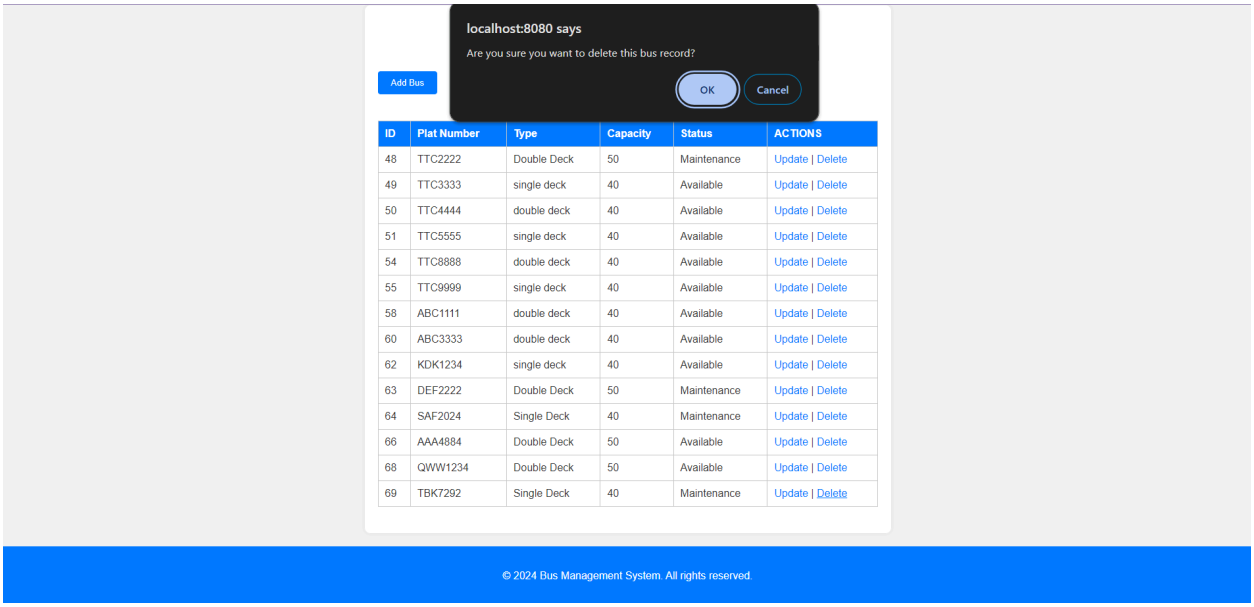


Diagram 4.2.8.1.6 shows the notification for confirmation on deletion of the route

1. The system will give notification “Are you sure you want to delete this route record?” to make confirmation on the deletion.
2. Next, click the **OK** button if confirm or **Cancel** button to terminate the process.

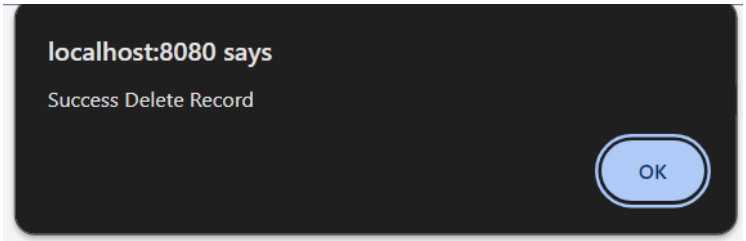


Diagram 4.2.8.1.7 shows the notification for successfully delete the route

1. The system will give notification “Success Delete Record” upon successfully deleting the bus.
2. Next, click the **OK** button.

4.2.9. Manage Reservation Module User(By Aesya)

4.2.9.1 Make Reservation

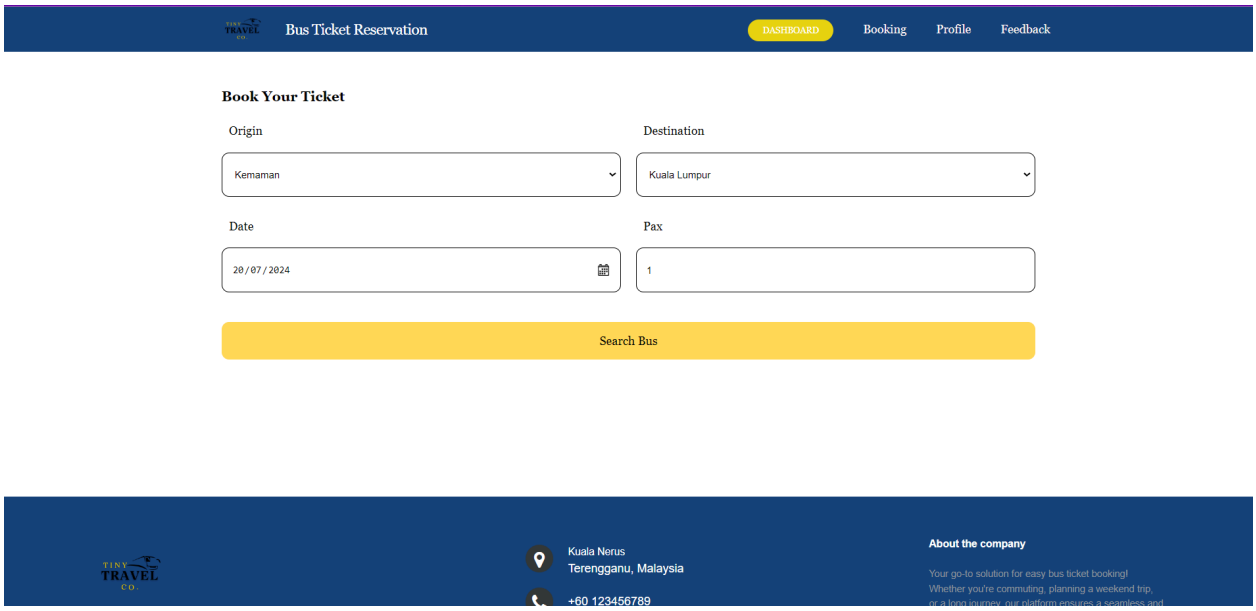


Diagram 4.2.9.1.1 shows the interface for user to make reservation

1. Choose the **Origin**.
2. Choose the **Destination**.
3. Choose **Date**.
4. Choose **Pax**.
5. Click the **Search Bus** button.

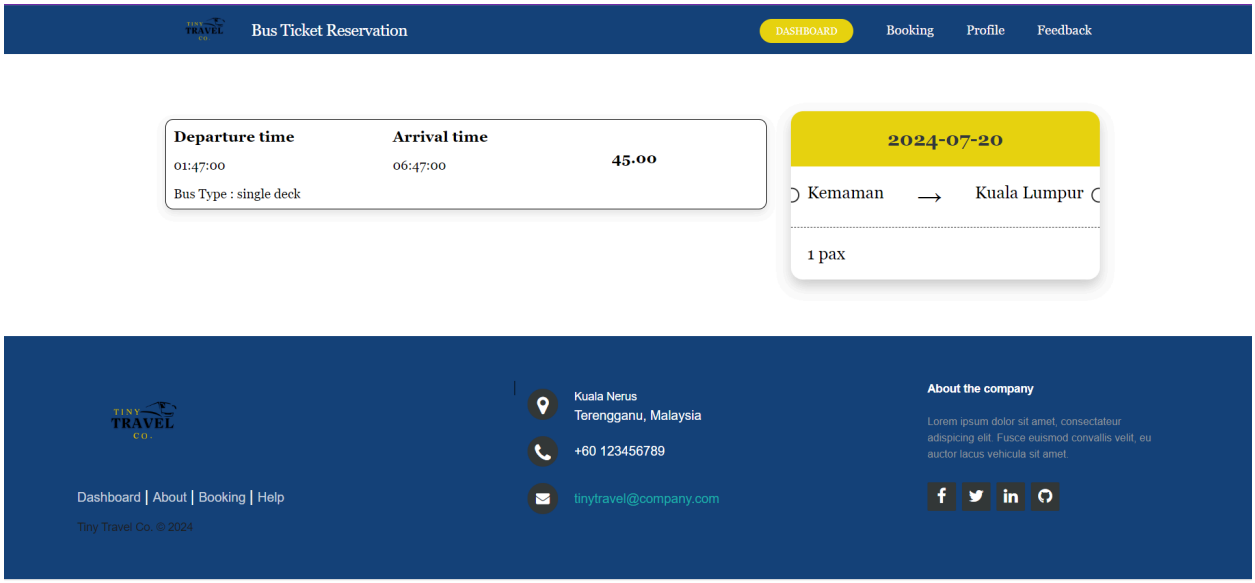


Diagram 4.2.9.1.2 shows all the bus that can be reserve

1. Click on the bus that is availables based on the searching criteria.

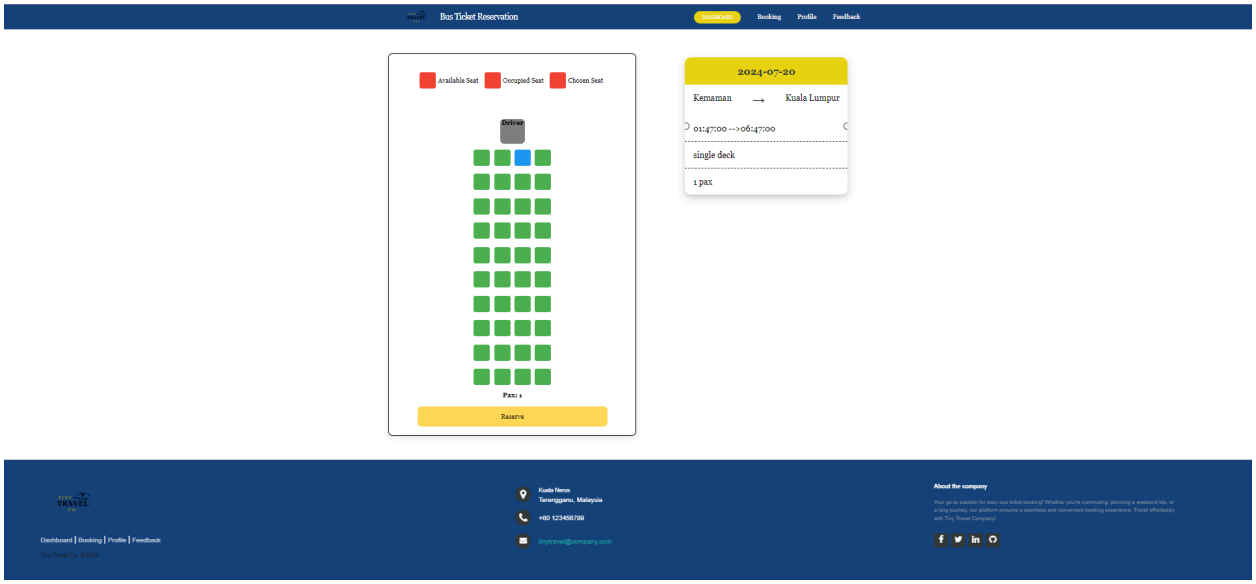


Diagram 4.2.9.1.3 shows the seat available on the chosen bus

1. Click on the seat that available
    - Green : Available Seat
    - Blue : Chosen Seat
    - Red : Occupied Seat
2. Click the **Reserve** button.

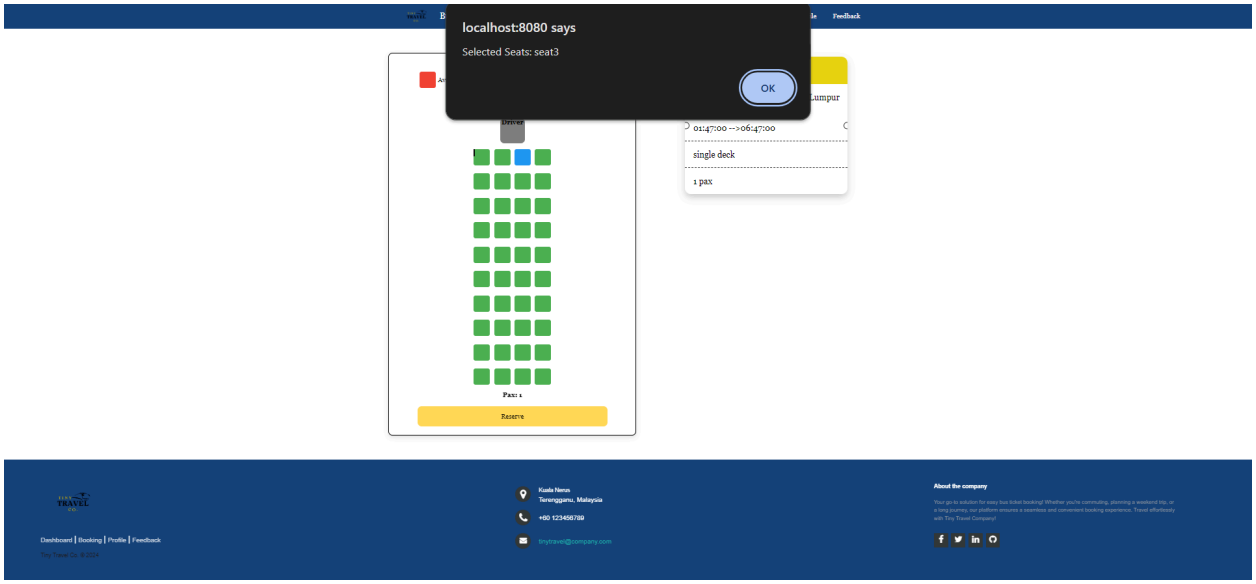


Diagram 4.2.9.1.4 shows the notification for confirmation on the seat selected

1. System gives notification “Selected Setas : seat3” to make confirmation on the selected seat.
2. Click the **OK** button.

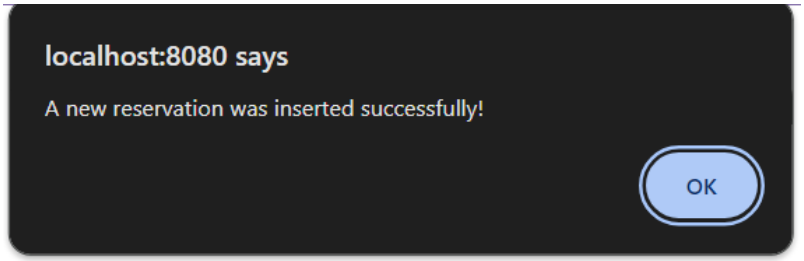


Diagram 4.2.9.1.4 shows the notification for successfully make reservation

1. System gives notification “A new reservation was inserted successfully” upon successfully make reservation
2. Click the **OK** button.

4.2.10 Payment Module (By Aesya)

4.2.10.1 Make Payment

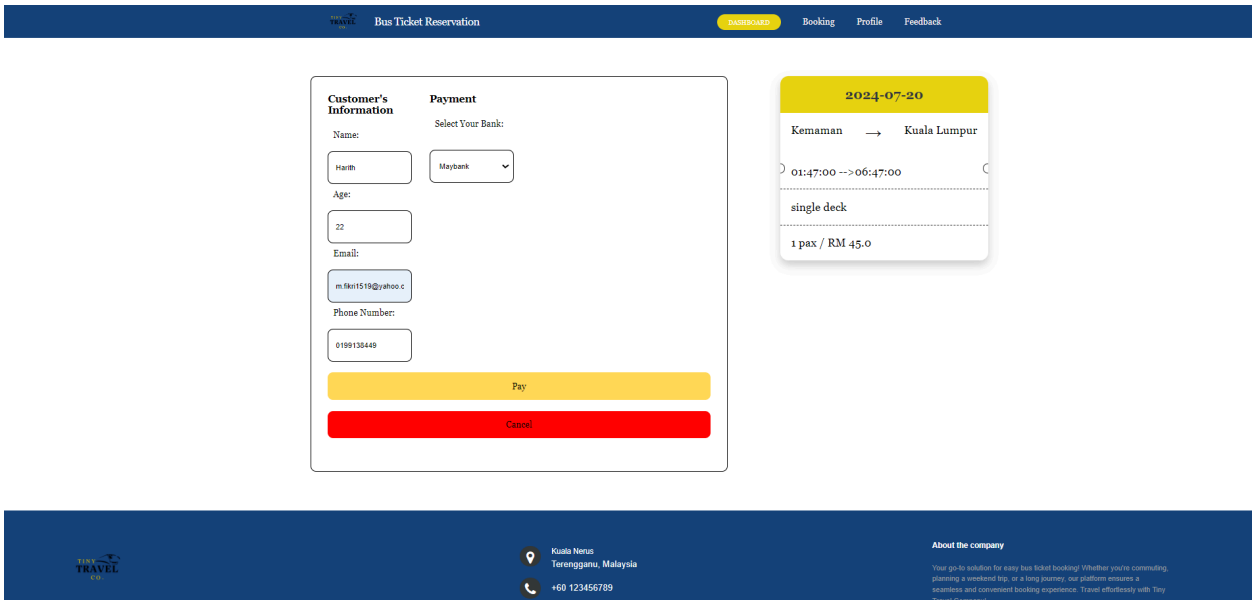


Diagram 4.2.10.1.1 shows the interface to make payment for the reservation

1. Enter valid **Name**.
2. Enter valid **Age**.
3. Enter valid **Email** that follows format 'tinytravel@gmail.com'.
4. Enter valid **Phone Number** that follows format '0199139449'.
5. Select **Bank**.
6. Click the **Pay** button to continue the payment or **Cancel** to terminate the process.

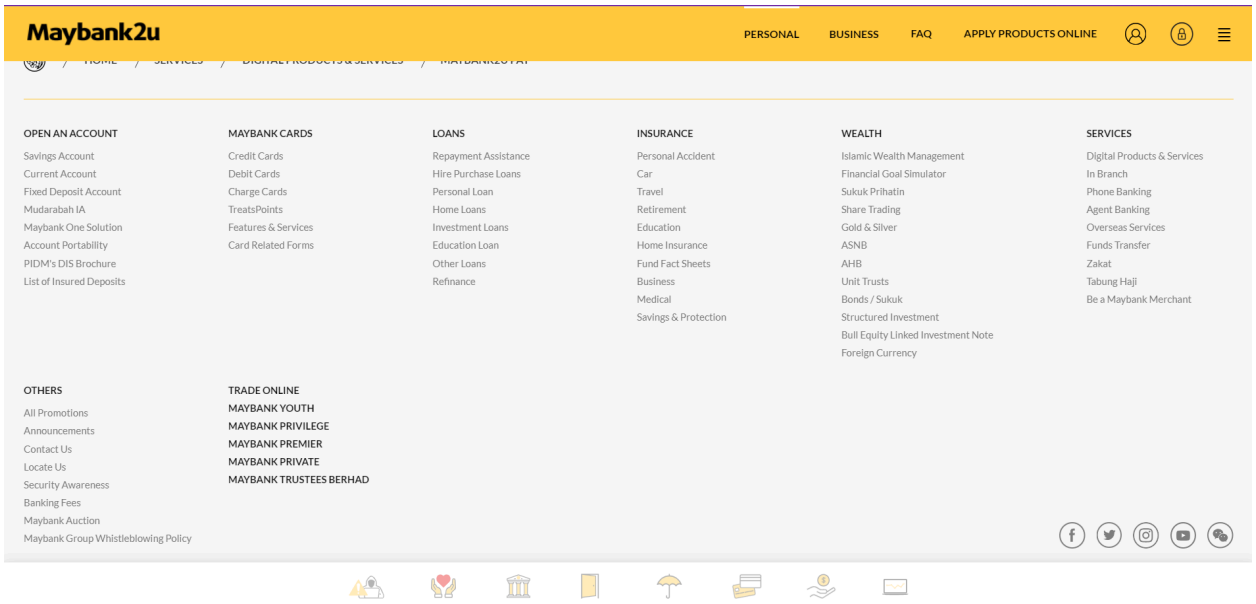


Diagram 4.2.10.1.2 shows the the bank chosen bank

1. Users will be redirected to the chosen bank to make payment on their reservation.

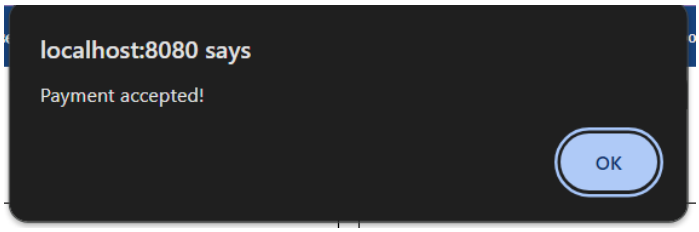


Diagram 4.2.10.1.3 shows the notification for successfully make payment

1. System gives notification “Payment accepted!” upon successfully making payment.
2. Click the **OK** button.

4.2.10.2 View Reservation

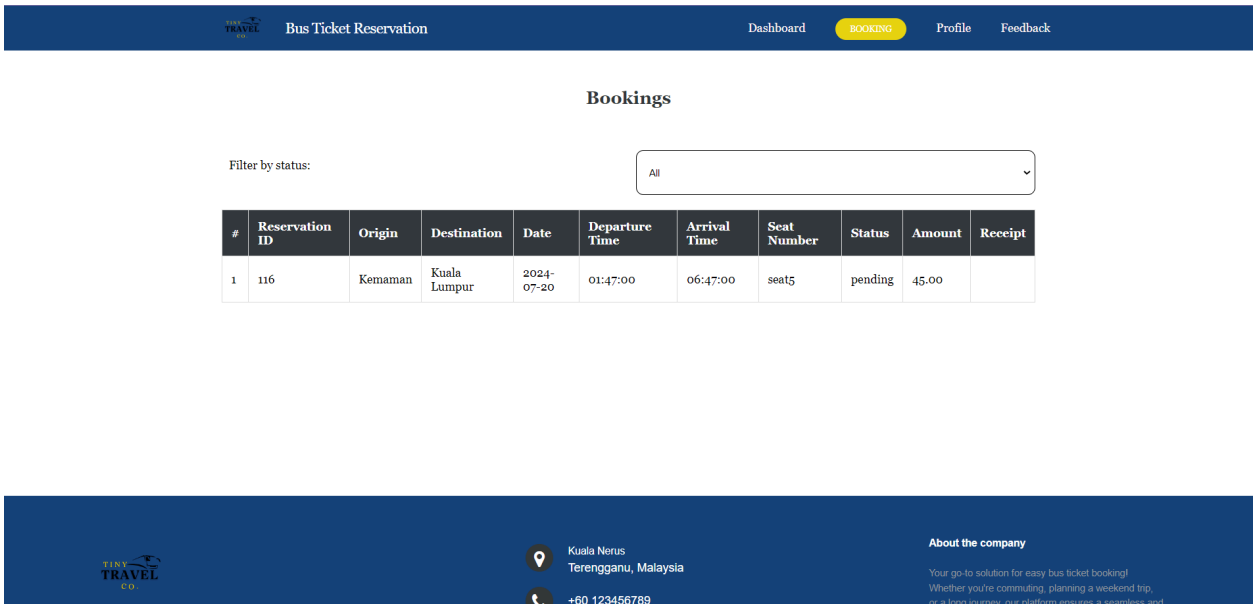


Diagram 4.2.10.2.1 shows the interface on the reservation make by a user

1. System shows the reservation that is being made by a user.

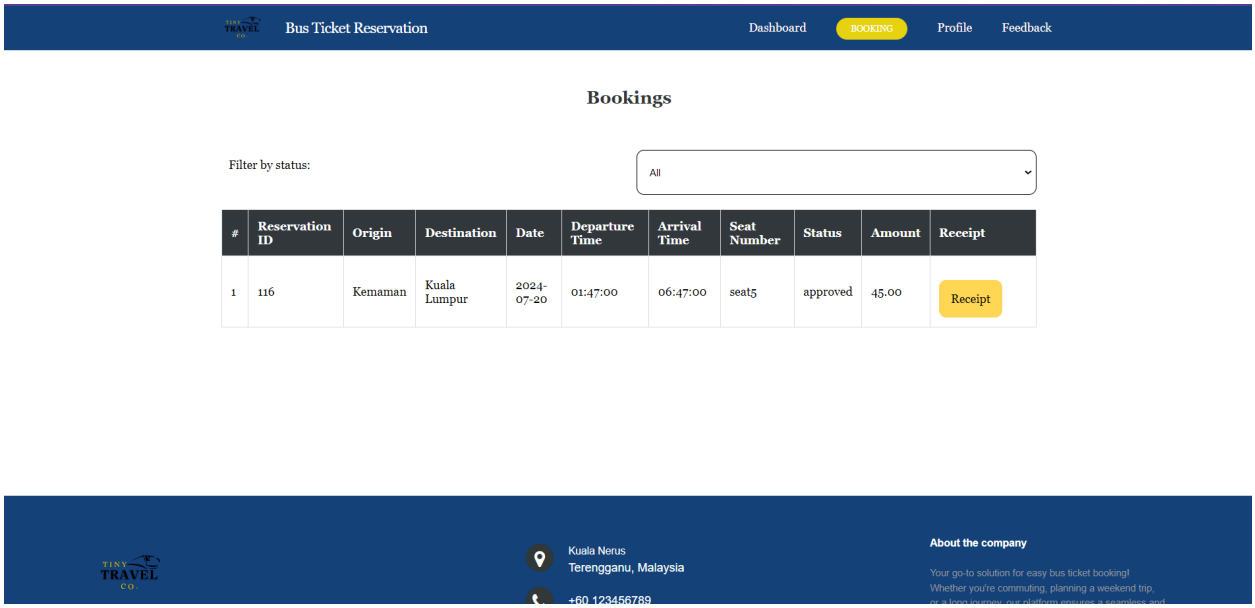


Diagram 4.2.10.2.2 shows the notification for successfully make payment

1. After the reservation has been approved. The system will give the user that makes the reservation a receipt.
2. Click the **Receipt** button.

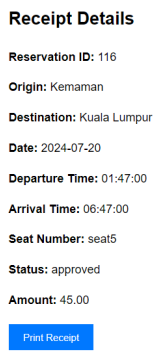


Diagram 4.2.10.2.3 shows the receipt of the reservation

1. System generated receipt was given to the user as proof of their reservation

5. REFERENCES

1. Padma, A., Gadde, S., Rao, B. S. P., & Ramachandran, G. (2021, July). Effective cleaning system management using JSP and Servlet Technology. In *2021 6th International Conference on Communication and Electronics Systems (ICCES)* (pp. 1472-1478). IEEE.
2. Smith, J. (2023, March 15). Building a web application with MVC architecture in Java. JavaWorld. https://www.javaworld.com/article/1234567/building-web-application-mvc-java.html
3. Doe, A. (2022, July 20). Implementing MVC in Java for web applications. CodeGuru. https://www.codeguru.com/java/implementing-mvc-java-web-applications
4. Brown, L. (2021, December 5). A guide to MVC architecture in Java web development. Developer.com. https://www.developer.com/java/guide-to-mvc-architecture-java-web-development
5. Project Geek. (2016, February 7). Retrieved from Bus Booking System Project: https://projectsgeek.com/2016/02/bus-booking-system.html
6. The Business Analysis Doctor. (2019, February 20). Retrieved from Use Case Description Basics: https://thebadoc.com/ba-techniques/f/use-case-description-basics
7. PHP Jabbers. (n.d.). Retrieved from Bus Reservation System: https://www.phpjabbers.com/bus-reservation-system/features.php
8. Chris, K. (2023, June 21). *Database Normalization – Normal Forms 1nf 2nf 3nf Table Examples*. freeCodeCamp.org. https://www.freecodecamp.org/news/database-normalization-1nf-2nf-3nf-table-examples/

6. <u>MEETING</u>

No	Date & Time	Location	Chairperson	Attendance	Agenda
1.	3/4/2024 9:00 PM - 11:00 PM	Google Meet	Aesya	Attendees: - Aesya - Irfan Hakimi - Harith - Haziqah Absent: None	I. Opening Remarks ● Welcome and Introduction ● Purpose of the Meeting II. Project Overview ● Discuss Project Objectives ● Review Key Milestones III. Team Roles and Responsibilities ● Assign Team Members to Tasks ● Clarify Responsibilities IV. Next Steps and Action Items ● Recap of Action Items ● Assign Responsible Parties ● Set Deadlines V. Closing Remarks ● Summary of Key Points

					• Confirm Next Meeting Date and Time
2.	17/4/2024 8:00 PM - 11:00 PM	PSNZ, UMT	Irfan, Harith	Attendees: - Aesya - Irfan Hakimi - Harith - Haziqah Absent: None	I. Opening Remarks • Welcome and Introduction • Purpose of the Meeting II. Review of Previous Meeting Minutes • Action Items Review • Status Updates on Action Items III. System Design Progress • Present System Architecture • Review User Interface Design • Gather Feedback IV. Development Plan • Discuss Development Phases • Allocate Tasks • Identify Risks and Mitigation Plans V. Next Steps and Action Items • Recap of Action Items

					• Assign Responsible Parties • Set Deadlines VI. Closing Remarks • Summary of Key Points • Confirm Next Meeting Date and Time
3.	2/7/2024	PSNZ, UMT	Haziqah	Attendees: - Aesya - Irfan Hakimi - Harith - Haziqah Absent: None	I. Opening Remarks • Welcome and Introduction • Purpose of the Meeting II. Review of Previous Meeting Minutes • Action Items Review • Status Updates on Action Items III. System Development Progress • Overview of Completed Tasks • Current Status of In-progress Tasks • Discuss Challenges and Mitigation Strategies IV. Testing and Integration Plan • Overview of Testing Phases

					• Assign Testing Tasks V. Closing Remarks • Summary of Key Points

7. APPENDIX

Appendix 7.1 shows the evidence interview with the stakeholders

Figure 7.1.1 shows the evidence of meeting with stakeholders

Appendix 7.1 shows the evidence questions given during the interview with the stakeholder

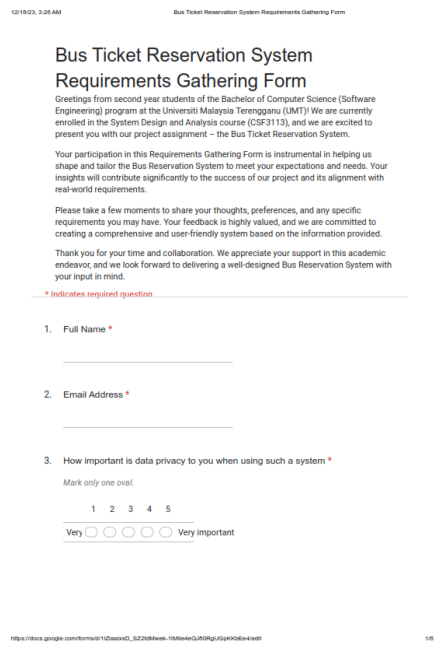
Appendix 7.2 shows list one-to-one questions to Stakeholder with High Priority (Operators)

Bil	Question
1.	Can you provide an overview of the current bus ticket reservation system you operate?
2.	What are the key features and functionalities of your system?
3.	How do users typically interact with the system?
4.	What is the typical volume of bookings your system handles on a daily basis?
5.	Are there specific times or seasons when booking activity experiences peaks?
6.	Have there been any challenges or issues related to payment transactions?

7.	What security measures are in place to protect user data and ensure transaction security?
8.	How do you train new users or staff members to use the reservation system?
9.	What feedback have you received regarding the ease of onboarding?

Appendix 7.2 One-to-one questions to Stakeholder with High Priority (Operators)

Appendix 7.3 shows Bus Ticket Reservation System Requirements Gathering Form and the responses



12/15/23, 3:26 AM Bus Ticket Reservation System Requirements Gathering Form

Bus Ticket Reservation System Requirements Gathering Form

Greetings from second year students of the Bachelor of Computer Science (Software Engineering) program at the Universiti Malaysia Terengganu (UMT)! We are currently enrolled in the System Design and Analysis course (CSF3113), and we are excited to present you with our project assignment – the Bus Ticket Reservation System.

Your participation in this Requirements Gathering Form is instrumental in helping us shape and tailor the Bus Reservation System to meet your expectations and needs. Your insights will contribute significantly to the success of our project and its alignment with real-world requirements.

Please take a few moments to share your thoughts, preferences, and any specific requirements you may have. Your feedback is highly valued, and we are committed to creating a comprehensive and user-friendly system based on the information provided.

Thank you for your time and collaboration. We appreciate your support in this academic endeavor, and we look forward to delivering a well-designed Bus Reservation System with your input in mind.

* Indicates required question

1. Full Name *

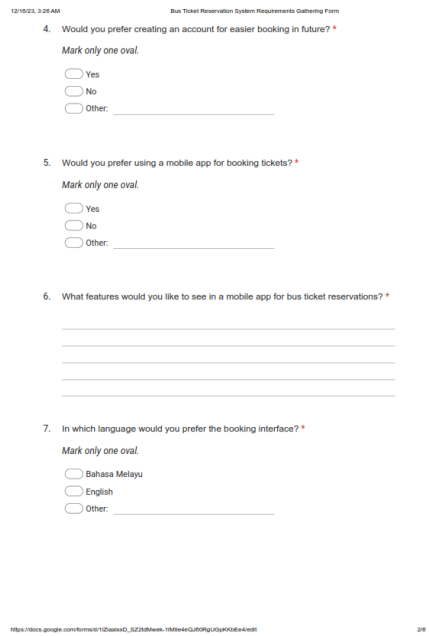
2. Email Address *

3. How important is data privacy to you when using such a system *

Mark only one oval.

	1	2	3	4	5	
Very	◯	◯	◯	◯	◯	Very important

https://docs.google.com/forms/d/1iZbaaixxD_SZ2tdMwek-1iMlie4eQJ60RgUGpKKbEe4/edit 1/6



12/15/23, 3:26 AM Bus Ticket Reservation System Requirements Gathering Form

4. Would you prefer creating an account for easier booking in future? *

Mark only one oval.

- Yes
- No
- Other: _____

5. Would you prefer using a mobile app for booking tickets? *

Mark only one oval.

- Yes
- No
- Other: _____

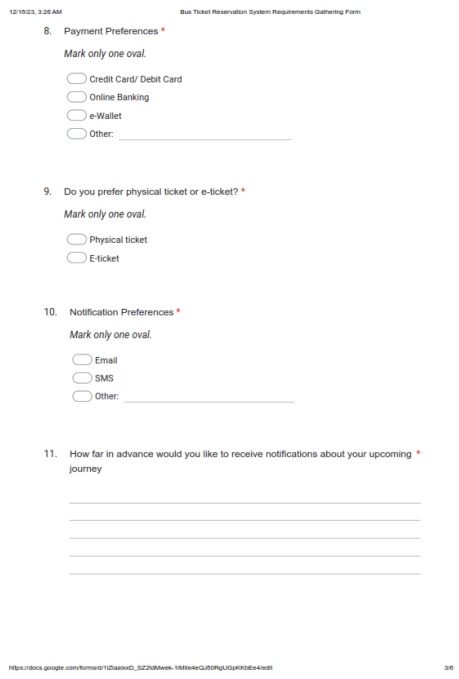
6. What features would you like to see in a mobile app for bus ticket reservations? *

7. In which language would you prefer the booking interface? *

Mark only one oval.

- Bahasa Melayu
- English
- Other: _____

https://docs.google.com/forms/d/1iZbaaixxD_SZ2tdMwek-1iMlie4eQJ60RgUGpKKbEe4/edit 2/6



12/15/23, 3:26 AM Bus Ticket Reservation System Requirements Gathering Form

8. Payment Preferences *

Mark only one oval.

- Credit Card/ Debit Card
- Online Banking
- e-Wallet
- Other: _____

9. Do you prefer physical ticket or e-ticket? *

Mark only one oval.

- Physical ticket
- E-ticket

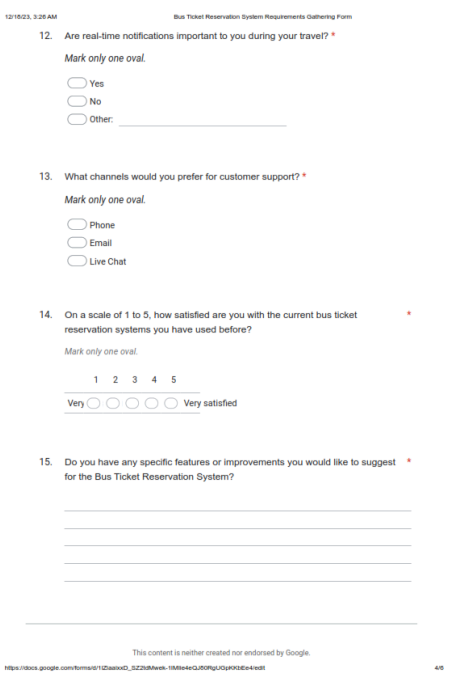
10. Notification Preferences *

Mark only one oval.

- Email
- SMS
- Other: _____

11. How far in advance would you like to receive notifications about your upcoming journey *

https://docs.google.com/forms/d/1iZiaaixxD_SZ2ldMwek-1iMlie4eQJ60RgUOpKKbEe4/edit 3/6



12/15/23, 3:26 AM Bus Ticket Reservation System Requirements Gathering Form

12. Are real-time notifications important to you during your travel? *

Mark only one oval.

- Yes
- No
- Other: _____

13. What channels would you prefer for customer support? *

Mark only one oval.

- Phone
- Email
- Live Chat

14. On a scale of 1 to 5, how satisfied are you with the current bus ticket reservation systems you have used before? *

Mark only one oval.

	1	2	3	4	5	
Very	◯	◯	◯	◯	◯	Very satisfied

15. Do you have any specific features or improvements you would like to suggest for the Bus Ticket Reservation System? *

This content is neither created nor endorsed by Google.

https://docs.google.com/forms/d/1iZiaaixxD_SZ2ldMwek-1iMlie4eQJ60RgUOpKKbEe4/edit 4/6

Responses:

[Responses to Bus Ticket Reservation System Requirements Gathering Form]

Appendix 7.3 Bus Ticket Reservation System Requirements Gathering Form and the responses

Appendix 7.4 shows evidence of meeting

Meeting 1

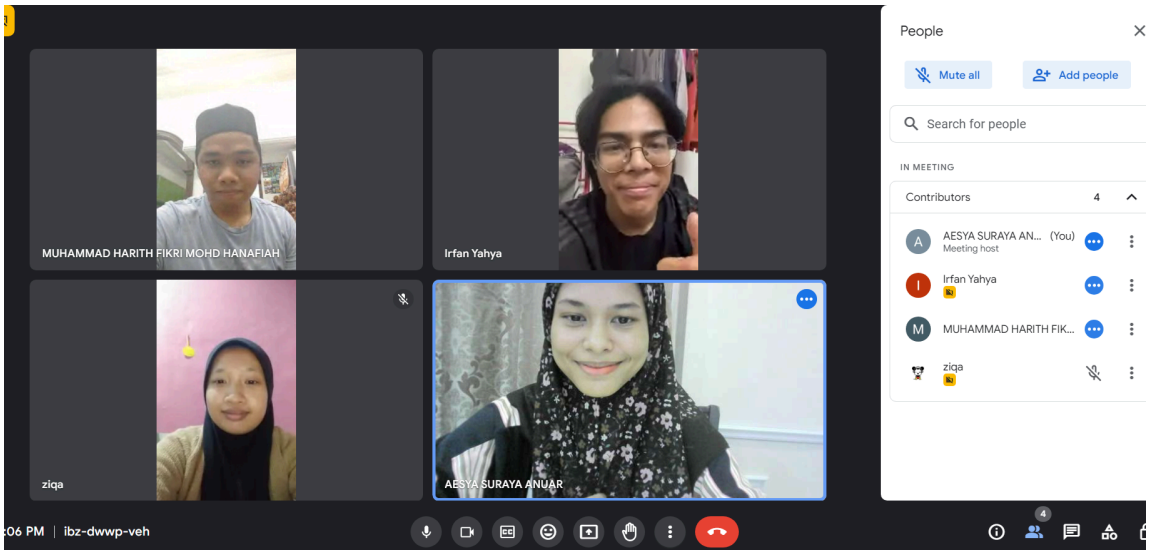


Figure 7.4.1 shows the the snapshot of Google Meet during Meeting 1

Meeting 2

Figure 7.4.1 shows the picture of Meeting 2

Meeting 3

Figure 7.4.1 shows the picture of Meeting 3