“AMOBUS”
BUS RESERVATION SYSTEM

Project Documentation Submitted to the Faculty of the School of Engineering and Technology National University, Fairview Quezon City

In Partial Fulfillment of the Requirements for Mobile Programming MOBPROG

Submitted By:

Project Manager: Gabriel Mendoza

Developer: Kyle Quintos

Documentation: Rhealyn Nabual

I. **Project Description**

The Bus Reservation System Application called "AMOBUS" is a digital solution designed to address the challenges and inefficiencies faced by both passengers and bus operators in the traditional bus ticketing process. We brainstormed about creating this because in the past, booking bus tickets involved long queues, manual paperwork, and limited access to information for passengers. Bus operators also faced difficulties in managing schedules, seat availability, and ticket sales efficiently.

With advancements in technology and the increasing demand for online services, there is a growing need for a comprehensive and user-friendly system that can revolutionize the bus reservation process. The application aims to fill this gap by providing a convenient and efficient platform for booking and managing bus tickets.

The core concept of the "AMOBUS" Bus Reservation System Application is to create a seamless and user-friendly experience for passengers. The application will allow users to search for available bus routes, view real-time seat availability, select preferred seats, and make secure online payments. We are offering a user-friendly interface and 24/7 accessibility; passengers will have the flexibility to plan and book their bus journeys at their convenience.
For the bus operators, the application will streamline the ticketing process and provide valuable tools for route management, schedule optimization, and revenue generation. Bus operators will be able to update and manage their schedules, define seat availability, and set dynamic pricing based on demand and seasonality. This will optimize resources, increase operational efficiency, and allow bus operators to cater to a wider audience. AMOBUS will also integrate with secure payment gateways, ensuring the security and convenience of online transactions. This will reduce the reliance on cash transactions and improve financial transparency for both passengers and bus operators.

All in all, the "AMOBUS" Bus Reservation System Applications overall goal is for transforming the bus ticketing industry by improving both the passenger booking experience and the operational effectiveness of bus operators. Bus travel will be easier and more accessible thanks to the application's user-friendly interface, real-time data, and secure payment gateway integration, which will help the transportation industry grow and experience a technological shift.

II. **System Documentation**

A. Product Requirement Documentation

1. Objectives

- Automate the bus reservation process: Implement a system that eliminates manual booking processes, reducing errors and enhancing efficiency.
- Provide a user-friendly booking interface: Design an intuitive and accessible platform for users, ensuring a seamless and pleasant booking experience.
- Ensure accurate seat allocation and availability tracking: Implement a robust seat management system to prevent overbooking, with real-time updates on seat availability.
- Enable real-time communication between passengers and administrators: Incorporate features such as instant confirmation notifications via email and SMS to enhance communication.
- Generate comprehensive reports for system analysis and improvement: Develop reporting functionalities to analyze user behavior, revenue, and system performance for continuous enhancement.

- Enhance Mobile Compatibility: Optimize the system for mobile devices, providing a user-friendly experience on smartphones and tablets.
- Integrate Payment Gateways: Enable secure and seamless online payments through integration with popular payment gateways.
- Enhance Security Measures: Implement multi-factor authentication and encryption to safeguard user data and transactions.

2. User Module

2.1 Registration and Login:
- Users can register and create accounts.
- Secure login with authentication mechanisms.

2.2 Bus Search and Selection:
- Users can search for buses based on origin, destination, and date.
- View available buses and their schedules.

2.3 Seat Selection:
- Users can choose seats from an interactive seat map.
- Real-time seat availability updates.

2.4 Booking Confirmation:
- Users receive instant booking confirmation via email and SMS.
- Access to e-tickets for reference.

2.5 Profile Management:
- Users can manage personal information and view booking history.
- Option to cancel or modify reservations.

3. Admin Module

3.1 Dashboard:
- Overview of current reservations, and bus schedules

3.2 Bus Management:
- Add, edit, or remove buses from the system.
- Define routes, schedules, and seating capacity.

3.3 Reservation Management:
- View and manage user reservations.
- Process cancellations and refunds.

3.4 Reporting:
- Generate reports on revenue, popular routes, and user demographics.
- Analyze system performance.

4. System Architecture

4.1 Frontend:
- Responsive application for users.
- Intuitive interface for seamless booking.

4.2 Backend:
- Database to store user data, reservations, and bus information.
- Server for processing reservations and communicating with users.

5. Non-functional Requirements:

5.1 Performance:

- System should handle a minimum of 1000 simultaneous users.
- Response time for actions should be under 2 seconds.

5.2 Security:

- Use secure encryption for user data and transactions.
- Regular security audits and updates.

5.3 Scalability:

- Design should accommodate future expansion in terms of buses and users.

6. Maintenance and Support

- Regular updates and bug fixes.
- 24/7 support for users and administrators.
- Documentation for system maintenance and troubleshooting.

7. Business Rules

7.1 Bus Seat Allocation Rule:

- Business Rule: The system must ensure that no two users can book the same seat for the same bus at the same time.
- Rationale: Prevents conflicts and confusion regarding seat reservations.

7.2 Cancellation Refund Policy:

- Business Rule: Users can cancel a reservation and receive a full refund if the cancellation is made at least 24 hours before the scheduled departure time.
- Rationale: Encourages users to make cancellations in a timely manner and helps in optimizing seat availability.

8. User and Admin Management Flowchart

Essential tools to document, visualize, and understand the processes involved in managing users and administrators within a bus reservation system. They contribute to a seamless user experience, enhance system security, and facilitate efficient system management.

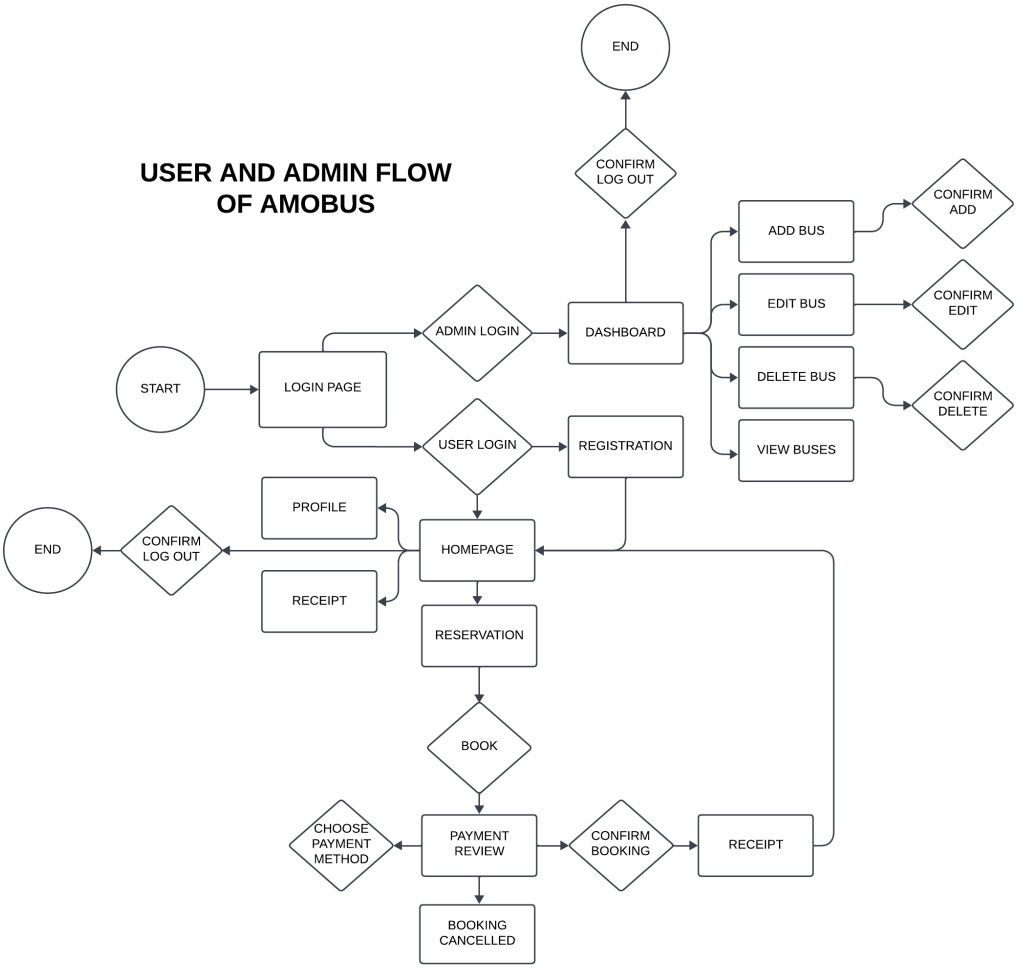

Figure 1. User and Admin flow

9. User Stories

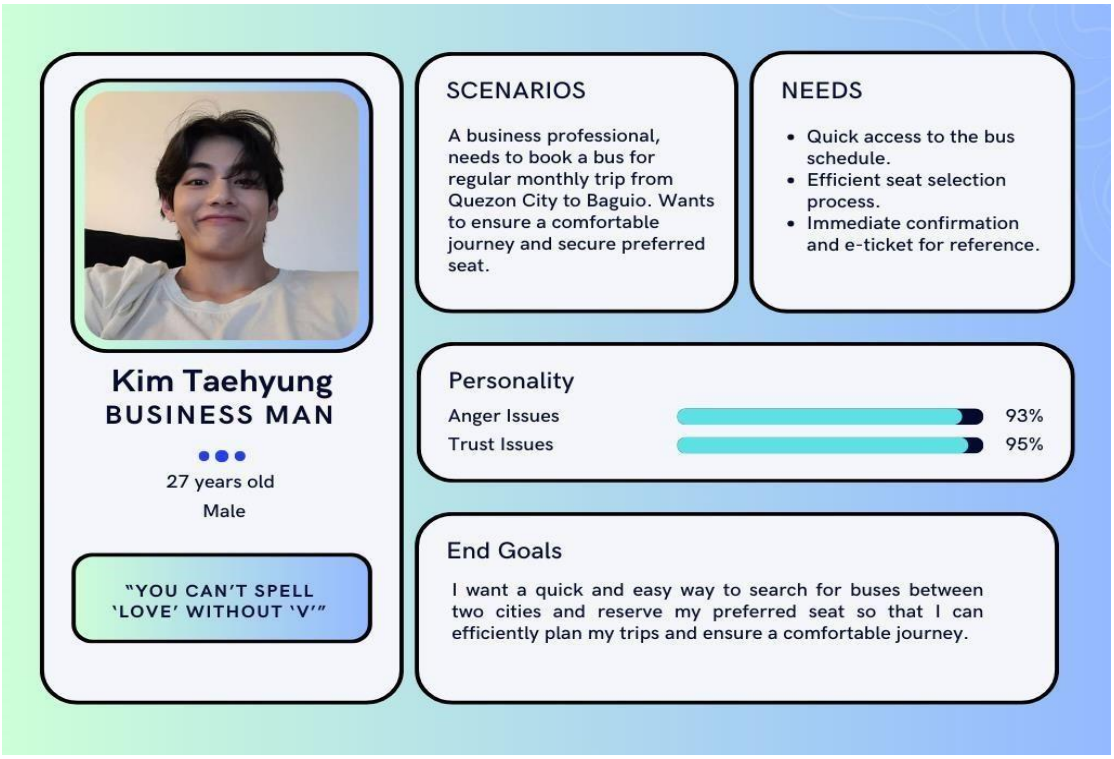

Figure 2. User Story no.1

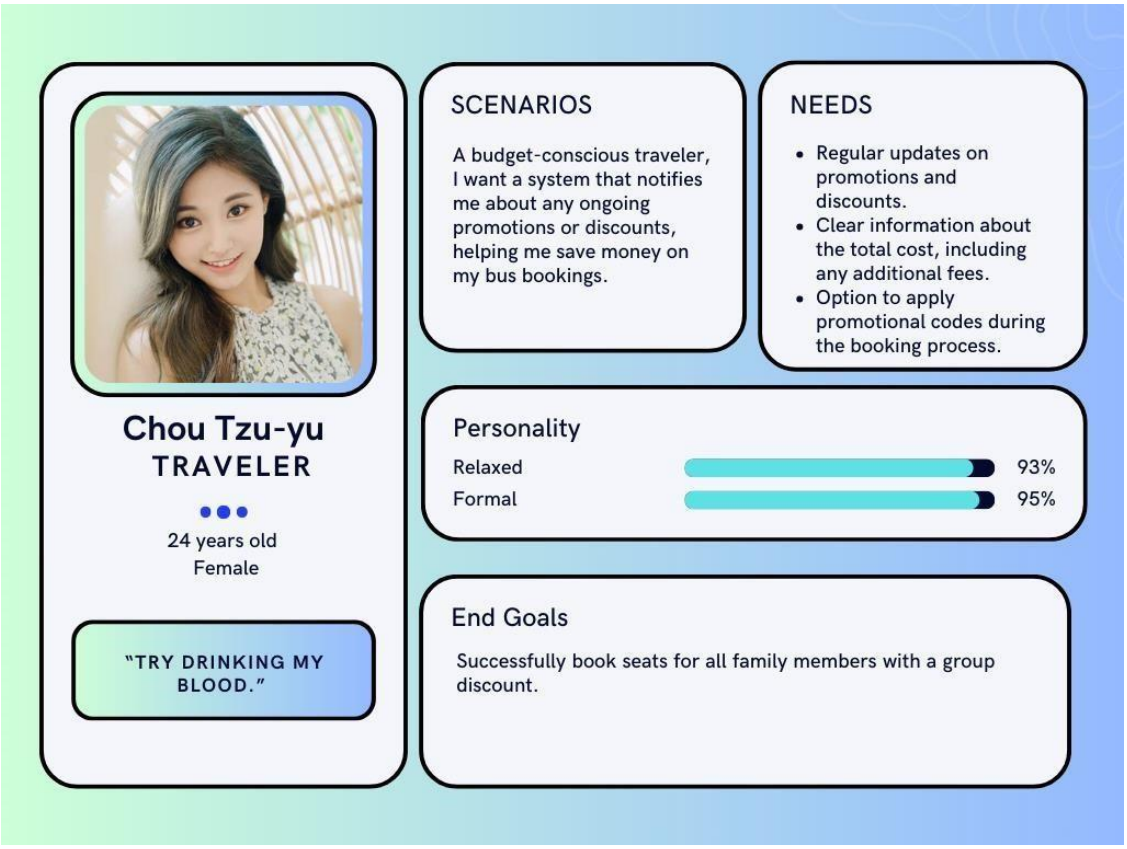

Figure 2.2 User Story no.2

10. Use Case Diagram

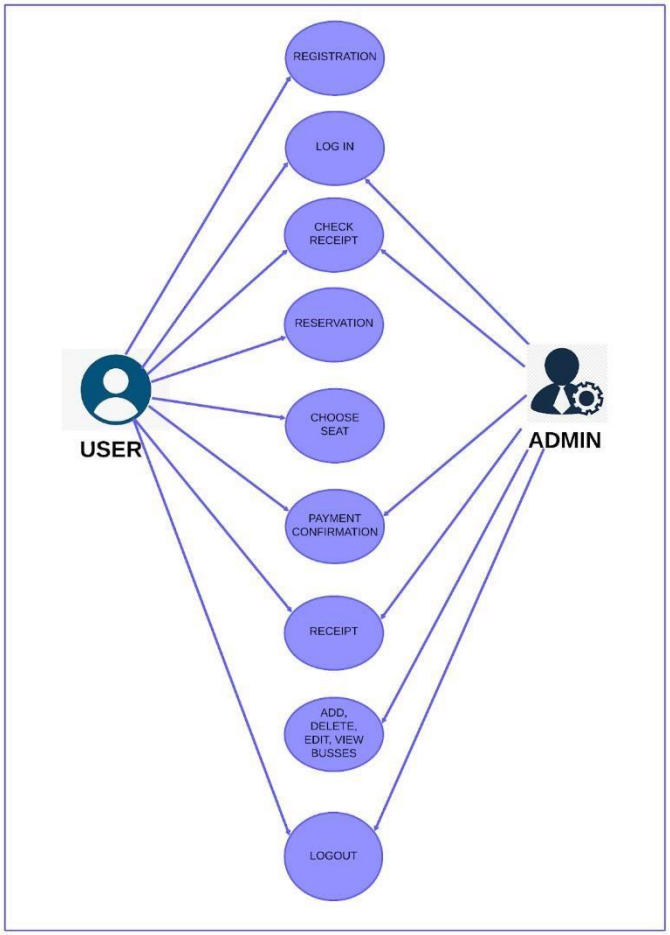

Figure 3. Use Case Diagram of User and Admin

B. **UX Design Documentation**

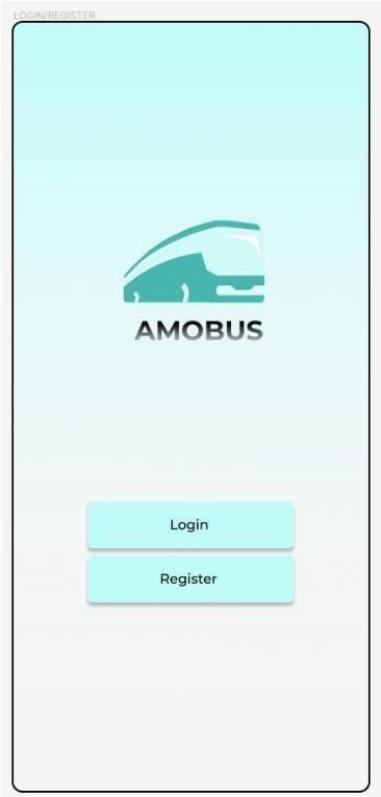

Figure 4. Login/registpage

Figure 4.1. Login

4.2. Register Page

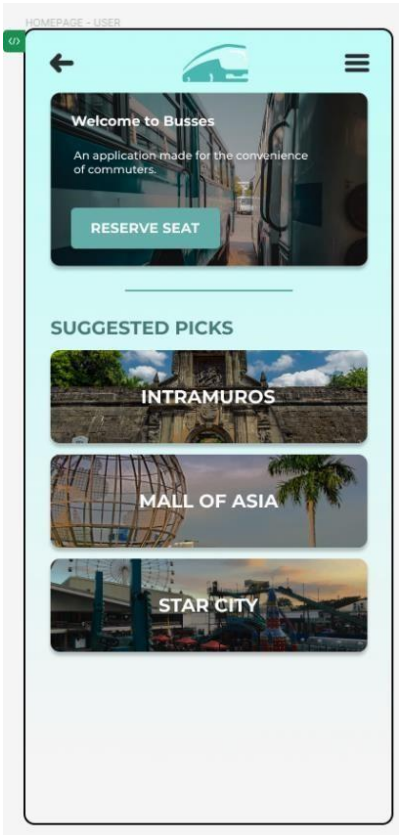

Figure 4.3 Homepage - User

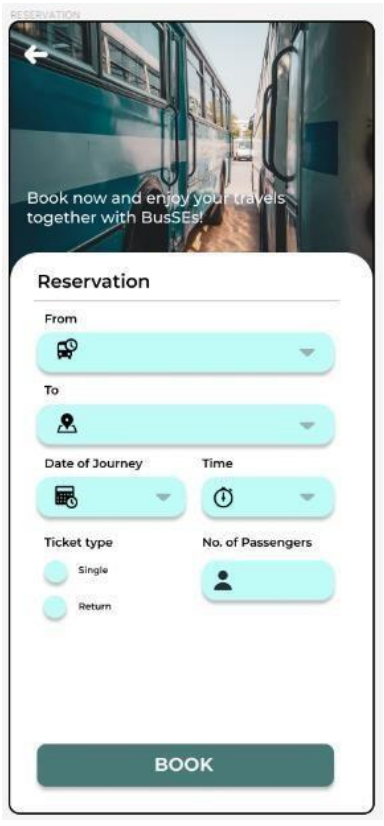

Figure 4.5 Reservation Page

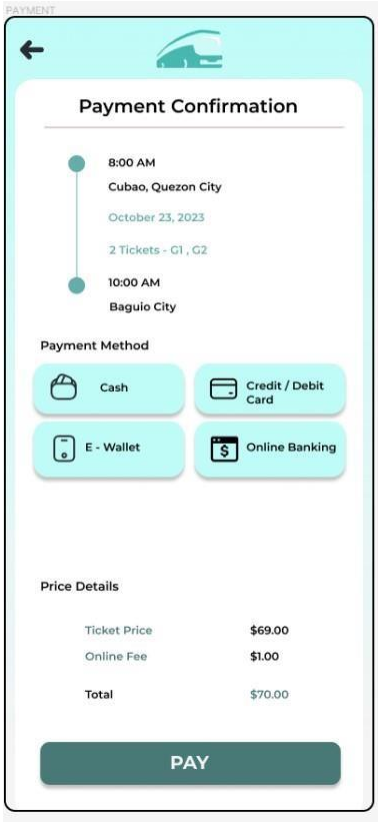

Figure 4.6 Payment Confirmation

Figure 4.7 Receipt Page

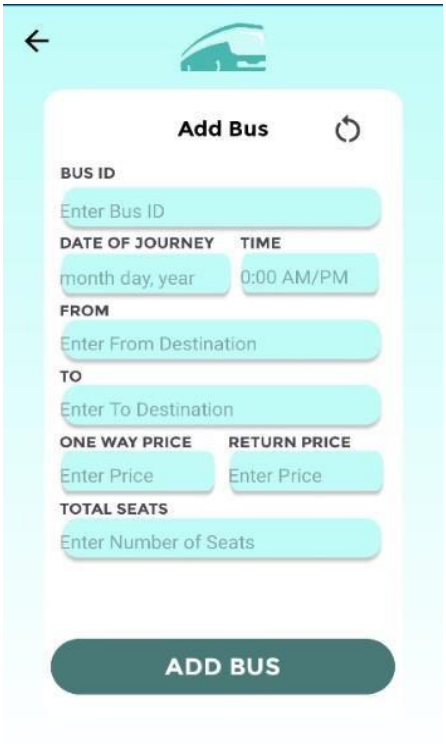

Figure 4.8 Admin – Add Bus

Figure 4.9 Admin-Del Bus

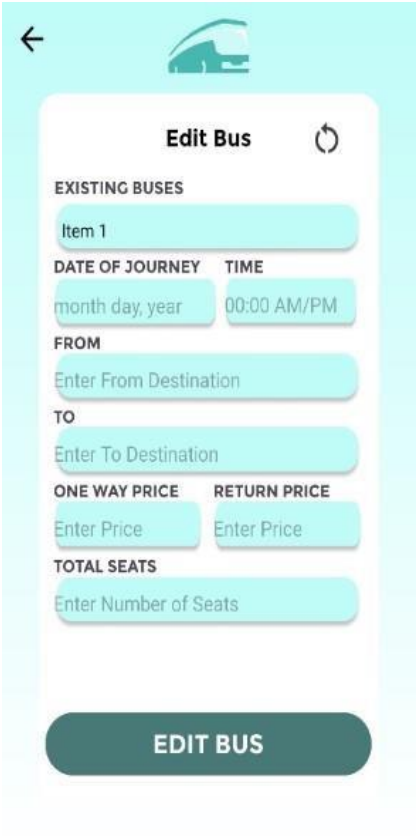

Figure 4.10 Admin-Edit bus

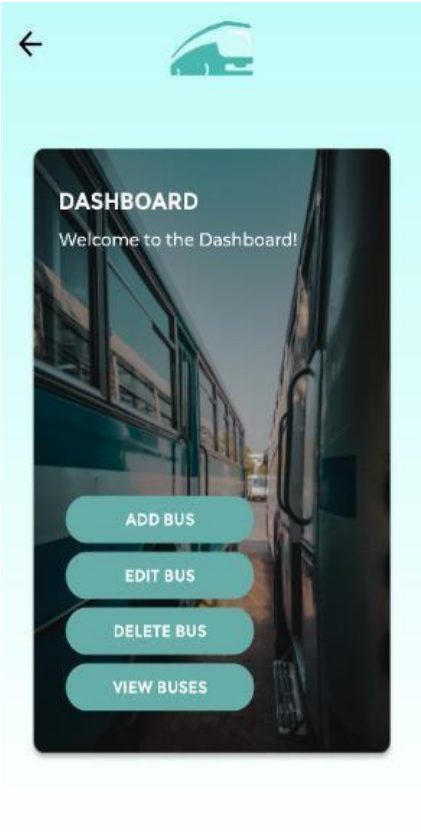

Figure 4.11 Admin-Dashboard

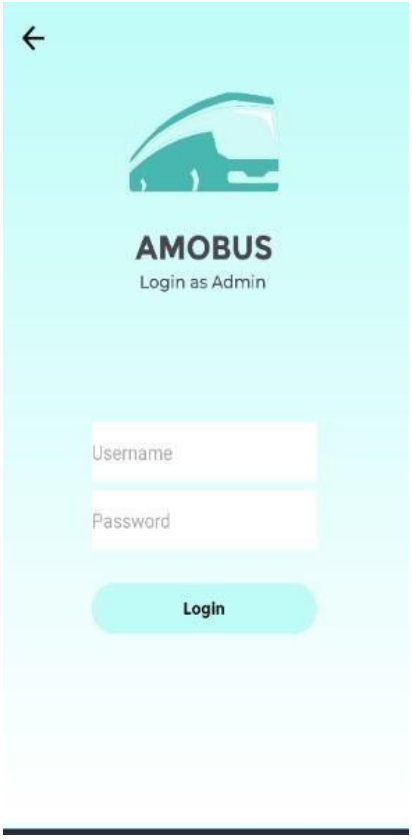

Figure 4.12 Admin-login

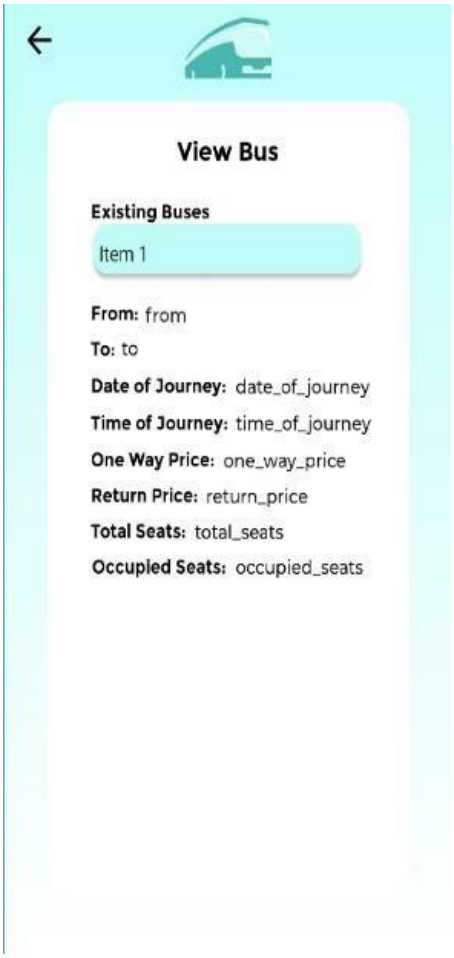

Fig. 4.13 - Admin View bus

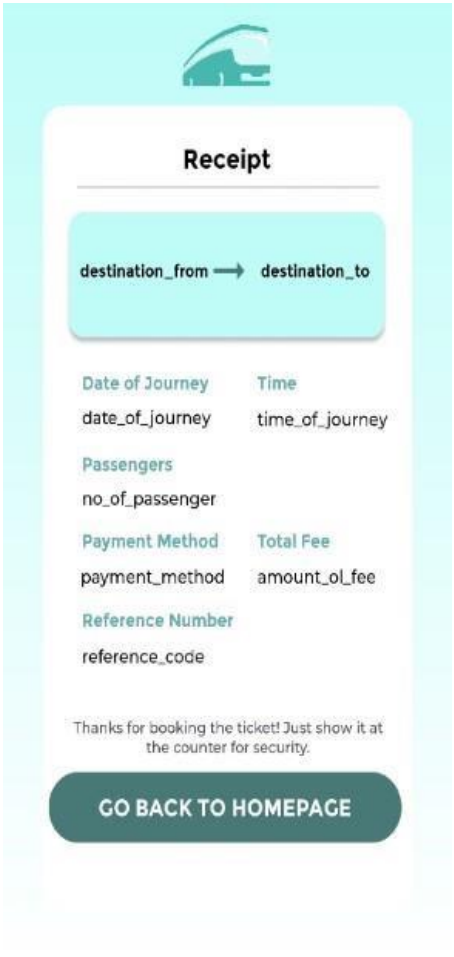

Fig.4.14 - Receipt

C. **Software Architecture**

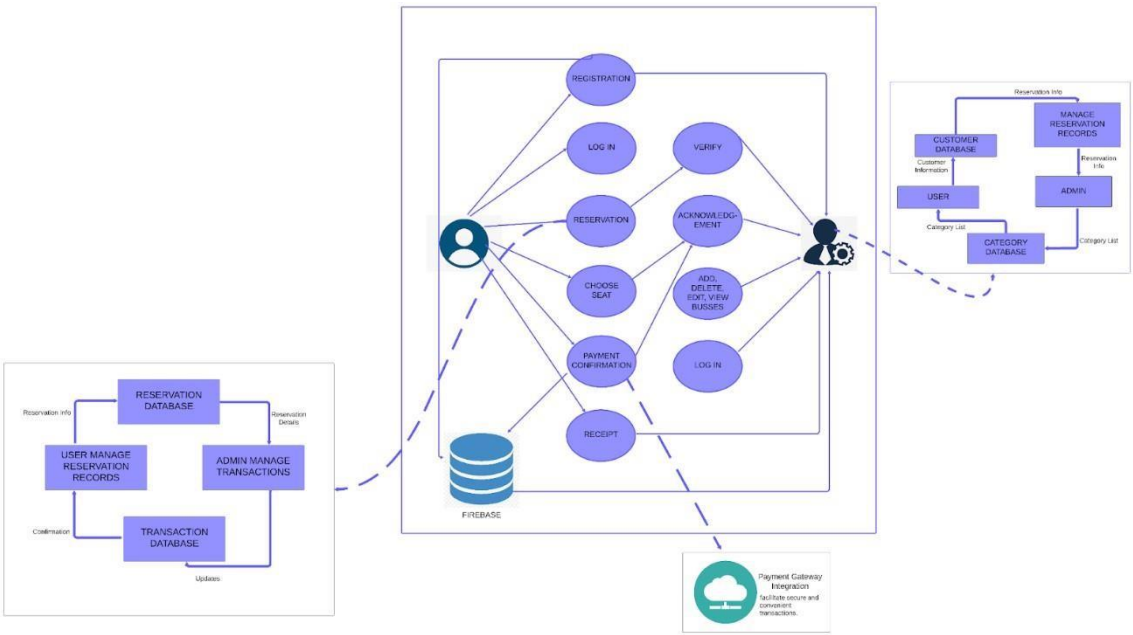

Figure 5. Software Architecture of AMOBUS

D. **Source Code Documentation**

Our source Code has all been saved to a cloud drive. For documentation purposes.

Class Files

XML FIles

III. **User Documentation**

A. **End-User Documentation**

This user guide is designed to help users understand how to utilize “AMOBUS” to book bus tickets, manage bookings, and make payments.

- To get started with AMOBUS, you first need to register for an account. Follow the steps below to register:

1. Open the AMOBUS application on your device.
2. Click on the "Register" button.
3. Fill in the required fields such as your name, email address, phone number, and password.
4. Submit your registration form.

Once you have successfully registered for an account, you can now log in using the register information that you filled up and start browsing the homepage and make reservations.

- To book a bus ticket on AMOBUS, users should follow these steps:

1. They must first log in to their account.
2. Select the desired travel route and date.
3. Choose a preferred bus and seat(s).

Confirm the booking details for accuracy and safety before clicking on the "Choose Seat" button.

- To confirm the payment AMOBUS supports various payment options, including E-wallet, credit/debit cards, online banking and cash. In the Payment Confirmation page, the information that the user chooses will show and also the total amount fee.
- To know that the booking is successful an official receipt and the information there will show up in the last page of the application. After that, the user can go back to the homepage.

B. **System Admin Documentation**

- **Functional description** – Our project focuses on allowing users to book via their smartphones. The system is designed to be easy to use and efficient, and it provides a number of features that make it a valuable tool for bus operators and passengers alike.

It allows the user to:

- Register to the application
- Reserve their seats in the bus
- Choose where they are traveling from and traveling to
- Manage their reservations
- Know if buses are available for the desired date/location

The application is composed of a Software application itself, Database.

- **System admin guide** – explains different types of system behaviors in different environments and with other systems. It also should provide instructions on how to deal with malfunction situations.

AMOBUS is a mobile application software that allows users to book their bus travels through their smartphones. This application uses a database to store the information about buses, booked travels, routes, schedules, and available reservations.

System Monitoring

For maintaining and monitoring the system it is good practice to

- Monitor the database server
- Monitor the application
- Monitor the logs on the database server

- Backup and Recovery Always create a backup of the database and the application files to make sure to recover from possible faults