



WOLLO UNIVERSITY
KOMBOLCHA INSTITUTE OF TECHNOLOGY
COLLAGE OF INFORMATICS
DEPARTMENT OF SOFTWARE ENGINEERING

Requirement Engineering SRS Document
Smart Bus Ticketing System (SBTS)

Group_Members

Number	Name	Id
1	Wubet lema	2027/16
2	Hanamarim sebsbew	0977/16
3	Mahlet Belete	4187/16
4	Ezedin muhamed	0772/16
5	Mubarek ali	1447/16

DECLARATION

We, the undersigned, members of the project group, declare that this Software Requirements Specification (SRS) for the Smart Bus Ticketing System (SBTS) is our original work and has been carried out as part of the Requirement Engineering assignment. We confirm that this document has not been submitted, in whole or in part, for any other course or degree at this or any other institution.

All materials, ideas, and data used from external sources have been properly acknowledged and cited in accordance with standard academic practices.

Project Title: Smart Bus Ticketing System (SBTS)

Department: Software Engineering

Institution: Kombolcha Institute of Technology, Wollo University

GROUP MEMBERS' PARTICIPATION

Name	ID	Date	Signature
1 Hanamariam sebsbew	0977/16	_____	_____
2 mahlet belete	4187/16	_____	_____
3 wubet lema	2027/16	_____	_____
4 mubarek ali	1447/16	_____	_____
5 ezedin muhamed	0772/16	_____	_____

CERTIFICATION

This is to certify that the above-named students have performed this Requirement Engineering project under my supervision and guidance. This document is now submitted for evaluation by the Department of Software Engineering.

Instructor's Name: _____

Signature: _____

Date: _____

ACKNOWLEDGEMENT

First and foremost, we would like to express our deepest gratitude to the Almighty for providing us with the wisdom, strength, and perseverance required to complete this Software Requirements Specification (SRS) document successfully.

We extend our sincere thanks to our course instructor, for his invaluable guidance, professional suggestions, and continuous encouragement throughout the duration of this project. His insights into the field of Requirement Engineering and Software Engineering standards have been instrumental in shaping the technical quality of this work.

We would also like to thank Wollo University, Kombolcha Institute of Technology (KIOT), and specifically the Department of Software Engineering, for providing us with the academic environment and resources necessary to carry out our studies.

Finally, we acknowledge the collective effort of our group members—Wubet, Hanamariam, Ezedin, Mubarek, and Mahlet. The completion of this Smart Bus Ticketing System (SBTS) document is a result of our shared dedication and teamwork.

Table of Contents

List of figure.....	1
Abstract.....	3
Introduction.....	7
1.1 Purpose.....	8
1.1.1 Scope.....	9
1.1.2 Benefits.....	9
1.1.3 Objectives.....	9
1.1.4 Goals.....	9
1.2 Document Conventions.....	9
1.3 Intended Audience and Reading Suggestions.....	9
1.3.1 Intended Audience.....	9
1.3.2 Suggested Reading Order.....	9
1.4 Definitions, Acronyms, and Abbreviations.....	9
1.5 Overview of the Document.....	9
2. Overall Description.....	9
2.1 Product Perspective.....	9
2.2 Product Features and Functions.....	9
2.2.1 Core User Functions.....	16
2.2.2 Extended and Value-Added Features.....	18
2.2.3 Administrative and Backend Tools.....	19
2.3 User Classes and Characteristics.....	19
2.3.1 Primary User Classes(Favored user classes).....	20
2.3.2 Secondary User Classes.....	20
2.3.3 Indirect User Classes.....	20
2.4 General Constraints.....	20
2.5 User Documentation.....	23
2.6 Assumptions and Dependencies.....	23
3. Specific Requirements.....	36
3.1 User Requirements.....	3
3.1.1 Functional Requirements.....	37
3.1.2 Non-Functional Requirements.....	37
3.2 Detailed System Requirements.....	37
4. External Interface Requirements.....	40
4.1 User Interfaces.....	40
4.2 Hardware Interfaces.....	42
4.3 Software Interfaces.....	43
4.4 Communications Interfaces.....	44
5. Analysis Models.....	45
5.1 System Use Case Model.....	45
5.1.1 Use Case Descriptions.....	59

5.2 Use Case Diagrams.....	64
5.3 Sequence Diagrams.....	64
5.4 Activity Diagrams.....	64
6 Conclusion	70
7 Reference	70
Appendix.....	72

List of figure

- Figure 2.1 Major components of the overall system.....
- Figure 5.0.1 Essential use case.....
- Figure 5.0.2 use case diagram of the system.....
- Figure 5.3 Sequence diagram of login.....
- Figure 5.4 sequence diagram of seat booking.....
- Figure 5.5 sequence diagram of Manage bus.....
- Figure 5.6 sequence diagram of register.....
- Figure 5.7 sequence diagram of make payment.....
- Figure 5.8 Activity diagram of login
- Figure 5.9 Activity diagram of make payment.....
- Figure 5.10 Activity diagram of register.....
- Figure 5.11 Activity diagram of manage bus.....
- Figure 5.12 Activity diagram of seat booking.....
- Figure 5.13 Activity diagram of validate ticket.....

ABSTRACT

In Ethiopia, the traditional intercity bus transportation sector is largely dominated by manual ticketing processes, which are characterized by long queues, lack of real-time schedule information, and physical presence requirements at bus stations. This Software Requirements Specification (SRS) document presents the requirements for the Smart Bus Ticketing System (SBTS), a web-based platform designed to digitize and modernize bus travel across the country.

The primary goal of the SBTS is to provide a seamless, secure, and efficient environment for passengers to search for routes, select specific seats, and make electronic payments via integrated local gateways such as Telebirr and CBE Birr. The system also incorporates advanced features including QR-code-based ticket verification, real-time vehicle tracking via GPS, and an AI-driven chatbot for automated customer support.

From an administrative perspective, the platform provides bus operators with robust tools for fleet management, route scheduling, and financial reporting. The development follows a structured Requirement Engineering approach to ensure the system is scalable, secure, and user-friendly for individuals with varying levels of digital literacy. By automating the booking lifecycle, the SBTS aims to reduce operational errors, eliminate physical bottlenecks, and significantly enhance the overall travel experience for passengers in Ethiopia

INTRODUCTION

The Online Bus Ticketing System is designed to provide a convenient, efficient, and reliable platform for customers to search buses, check schedules, select seats, and purchase tickets electronically. The system allows users to book tickets anytime and from anywhere, eliminating the need to visit physical ticket counters and reducing waiting time. The system includes user registration and authentication, real-time bus availability, secure online payment, and electronic ticket generation. This system improves customer experience by offering faster service, accurate booking information, and enhanced transparency, while also ensuring data security and reducing operational errors. Overall, the Online Bus Ticketing System plays an important role in modernizing transportation services and increasing customer satisfaction in Ethiopia.

1.1 Purpose

This Software Requirements Specification (SRS) document defines the software requirements for the Bus System Website – Version 1.0. The purpose of the document is to collect and analyze all assorted ideas that have come up to define the system. This document lays out everything you need to know about what the system should do and how it should work, both the features you'll see and the technical stuff behind the scenes. It's the go-to guide for developers, project managers, testers, stakeholders, and academic reviewers.

The system is a web-based bus booking and management platform. It lets passengers plan and book direct trips, pick their seats, choose bus classes, pay online, get tickets with QR codes, and receive real-time updates about their travel. This SRS covers the whole system: what users see, dashboards for operators, admin controls, AI-chatbot, and notification services. This isn't just a piece of the puzzle—it's the full picture.

1.1.1 Scope

The Bus System Website aims to design, build, and launch a platform that streamlines bus travel bookings across Ethiopia. The main idea is to give travelers an easy, smart, and trustworthy way to search, compare, book, and manage bus trips, even those tricky multi-stop journeys. Meanwhile, operators and admins get solid tools for running their services and digging into performance data.

The system allows passengers to register and log in, search for available buses based on route and travel date, view schedules and seat availability, select seats, make secure online payments, and receive electronic tickets. It also provides administrative features for managing buses, routes, schedules, fares, and bookings.

The scope of this system is limited to online ticket reservation and management and does not include physical ticket sales at bus stations or vehicle tracking functionalities.

1.1.2 Benefits

The primary benefit is to make bus ticket booking in Ethiopia digital, straightforward, and more efficient for travelers, while bringing bus operator's day-to-day operations into the modern age.

Customers and service providers can profit from the Online Bus Ticketing System in a number of ways. Customers can save time and effort by simply searching for buses, checking timetables, choosing seats, and purchasing tickets online rather than going to ticket booths. By automating seat assignment and payment processing, the technology lowers booking errors. It increases operational effectiveness, offers precise booking records, and streamlines route, schedule, and bus administration for bus operators and administrators. In general, the approach improves customer satisfaction and service quality and cuts down manual ticketing systems .

1.1.3 Objectives

The main objectives of the Online Bus Ticketing System are:

- Build a secure, scalable online bus booking platform
- To guarantee dependable and safe payment processing
- To effectively manage bus routes, schedules, and seat availability
- To cut down on physical labor and human error
- To give users access to real-time booking information

1.1.4 Goals

The Online Bus Ticketing System's long-term objectives are:

- ➔ To update the conventional methods for purchasing bus tickets

- To improve accessibility by enabling ticket purchases at any time and from any location
- To increase openness and confidence between clients and service providers
- To facilitate upcoming improvements like third-party integrations and mobile apps

1.2 Document Conventions

When creating the SRS document, the type of font used for the headers is Times New Roman (bold) and. The document's readability and navigation are enhanced by the constant use of headings and numbering.

- ❖ The usual IEEE documentation rules are used in this Software Requirements Specification (SRS) document.
- ❖ High Priority requirements are marked with (H), Medium Priority with (M) and Low Priority with (L).
- ❖ Requirement IDs, such as FR-1 and NFR-1, are used to individually identify each functional and non-functional requirement for traceability.
- ❖ All requirements are written in a clear and uniform manner.
- ❖ The usual IEEE documentation rules are used in this Software Requirements Specification (SRS) document.
- ❖ Requirements that are required, advised, or optional are denoted by the terms "shall," "should," and "may." Depending on how crucial they are to the operation of the system,
- ❖ requirements are categorized as High, Medium, or Low.

1.3 Intended Audience and Reading Suggestions

1.3.1 Intended Audience

This SRS document is intended for the following stakeholders involved in the development, management, testing, and deployment and evaluation of the Online Bus Ticketing System.:.

- ★ **Developers & Software Architects:** Will use this document to understand system functionality, technical constraints, and interface requirements for design and implementation.
- ★ **Project Managers:** Will refer to this document for scope definition, requirement priorities, and to plan sprints, allocate resources, and track deliverables.

- ★ **Quality Assurance (QA) Engineers & Testers:** Will use the specific requirements to create test plans, test cases, and validation criteria to ensure the system meets all specified functions and performance standards.
- ★ **UI/UX Designers:** Will rely on the user requirements, interface specifications, and user class descriptions to create intuitive and effective user interfaces.
- ★ **System Administrators & DevOps Engineers:** Will use the non-functional requirements (performance, security, deployment) to plan the infrastructure, deployment pipeline, and operational monitoring.
- ★ **Product Owners & Business Analysts:** Will use this document as the authoritative source for agreed-upon features to guide development and validate the final product.
- ★ **Technical writers:** will use this document to write user manuals, admin guides, and help documentation.

This SRS document is organized in the following way:

Section 1: Introduction: Explains why this document exists, who it's for, and how to use it.

Section 2: Overall Description: Covers the bigger picture—what the product is, main features, user types, system boundaries, needed documentation, and assumptions.

Section 3: Specific Requirements: Lists all features, functions, constraints, and what the software needs to deliver.

Section 4: External Interface Requirements: Details user interactions, system interfaces, and communications.

Section 5: Analysis Models: Shows use cases, system behaviors, and diagrams.

Section 6: References: Lists all supporting documents used to prepare this SRS.

Appendix: Extra diagrams, questionnaires, and supporting info.

1.3.2 Suggested Reading Order:

Passengers, or End Users

To comprehend the goal, scope, and key features of the Online Bus Ticketing System, end users should mainly read the Introduction and Overall Description parts. These sections give a high-level summary of how the system facilitates the booking, payment, and ticket finding processes. Without the need for technical expertise, reading these parts enables users to comprehend the advantages of the system and the services that are offered.

System Administrators

The Overall Description, Specific Requirements, and External Interface Requirements sections should be the primary emphasis of system administrators. These sections describe system limitations and interfaces, as well as administrative features like bus, route, timetable, and user account management. Administrators can better grasp their roles, access levels, and the system's operational boundaries with the use of this information

Software Developers

It is recommended that software developers thoroughly examine the sections on Analysis Models, External Interface Requirements, and Specific Requirements. The functional logic, interfaces, interaction flows, and system behavior necessary for implementation are covered in these sections. Before starting development, developers can also consult the Overall Description to comprehend the architectural expectations and system context.

System Analysts

The Introduction, Overall Description, and Specific Requirements sections should receive special attention from system analysts as they go through the full SRS document. This enables analysts to confirm that the specifications are comprehensive, coherent, and in line with stakeholder expectations. The Analysis Models can also be used by analysts to guarantee traceability between system behavior and requirements..

Testers and the Quality Assurance Group

The Specific Requirements and Analysis Models sections should be the primary focus of testers and quality assurance staff. Detailed functional and non-functional requirements that can be turned into test cases are provided in these sections. By going over these parts, you can make sure that every aspect of the system has been adequately evaluated and that it operates as intended both in error and normal situations.

Projects Managers

To comprehend the project's scope, goals, limitations, and possible hazards, project managers should read the Introduction, Overall Description, and Assumptions and Dependencies sections. Throughout the development lifecycle, this data facilitates efficient resource allocation, project planning, and progress tracking.

UI/UX Designers

To comprehend user interactions, interface limitations, and usability requirements of the Online Bus Ticketing System, UI/UX designers should go over the Overall Description, External Interface Requirements (User Interfaces), and Specific Requirements sections. These sections assist designers construct intuitive, consistent, and user-friendly interfaces by offering recommendations on screen flows, user actions, input requirements, and error handling. For a deeper understanding of user journeys and interaction sequences, designers can also consult the Analysis Models, which include use case and activity diagrams.

1.4 Definitions, Acronyms, & Abbreviations

- ★ SRS: Software Requirements Specification
- ★ SBTS: Smart Bus Ticketing Systems
- ★ MVP: Minimum Viable Product
- ★ UI: User Interface
- ★ API: Application Programming Interface
- ★ GPS: Global Positioning System

- ★ RBAC: Role-Based Access Control
- ★ PCI DSS: Payment Card Industry Data Security Standard
- ★ OTP: One-Time Password
- ★ QR Code: Quick Response Code
- ★ WCAG: Web Content Accessibility Guidelines

1.5 Overview of the Document

This Software Requirements Specification (SRS) document contains a complete description of the functional and non-functional requirements of the Online Bus Ticketing System. It defines the system's features, user requirements, system behavior, constraints, interfaces, and analysis models needed to guide the design, development, testing, and maintenance of the software.

A . What this SRS Contains

This document contains the following key components:

- **A Formal Definition of the Product:** It precisely identifies the software system, its release version (1.0), and its boundaries by clearly stating what is included in and excluded from the current project scope.
- **Comprehensive Requirement Specifications:** It details all functional capabilities the software must provide to its end-users (travelers, operators, administrators) and the non-functional qualities it must possess (such as performance, security, and usability).
- **System Context and Interfaces:** It describes the environment in which the software will operate, including interactions with users, other software systems (e.g., payment gateways), and hardware considerations.
- **Supporting Models and References:** It provides visual analysis models (like use case diagrams) to illustrate complex processes and includes references to related project documentation.

B . How this SRS is Organized

The SRS is organized into structured sections that present the system from a general overview to detailed specifications. The document begins with an introduction that outlines the purpose, scope, and intended

audience. It then describes the overall system context, features, user classes, and constraints. Detailed user and system requirements are presented next, followed by external interface requirements. The document also includes analysis models such as use case, sequence, and activity diagrams, and concludes with references and appendices for supporting information.

2 . Overall Description

2.1 Product Perspective

The SmarttBus Ticketing System is conceived as a comprehensive, web-based platform that modernizes the way intercity bus transportation is accessed and managed in Ethiopia. Although the system functions as an independent product, it operates within a broader digital transportation environment and interacts with several external services that are essential to its operation. These include mobile money and banking platforms (such as Telebirr ,CBE....)

The primary motivation behind SmartBus ticketing system is the need to replace the predominantly manual approach used in most bus stations today. Traditional ticketing processes often dependent on physical presence, handwritten logs, and limited communication channels create inefficiencies for both passengers and bus companies. SmartBus ticketing system aims to centralize these fragmented activities into a single, organized, and digitally accessible system.

The system consists of several major components, including a user interface, application server, database, and external service interfaces. The system is accessible through standard web browsers and is intended to support both customers and administrators in performing ticket booking and management activities efficiently. Although the current version is designed for intercity bus ticketing, the platform is intentionally structured to grow into a broader, integrated mobility system. Future development could include mobile apps, integration with rail and airline platforms, and tools for governmental transport agencies. This positions SwiftBus not simply as a booking platform but as a foundational component in Ethiopia's evolving digital transport ecosystem.

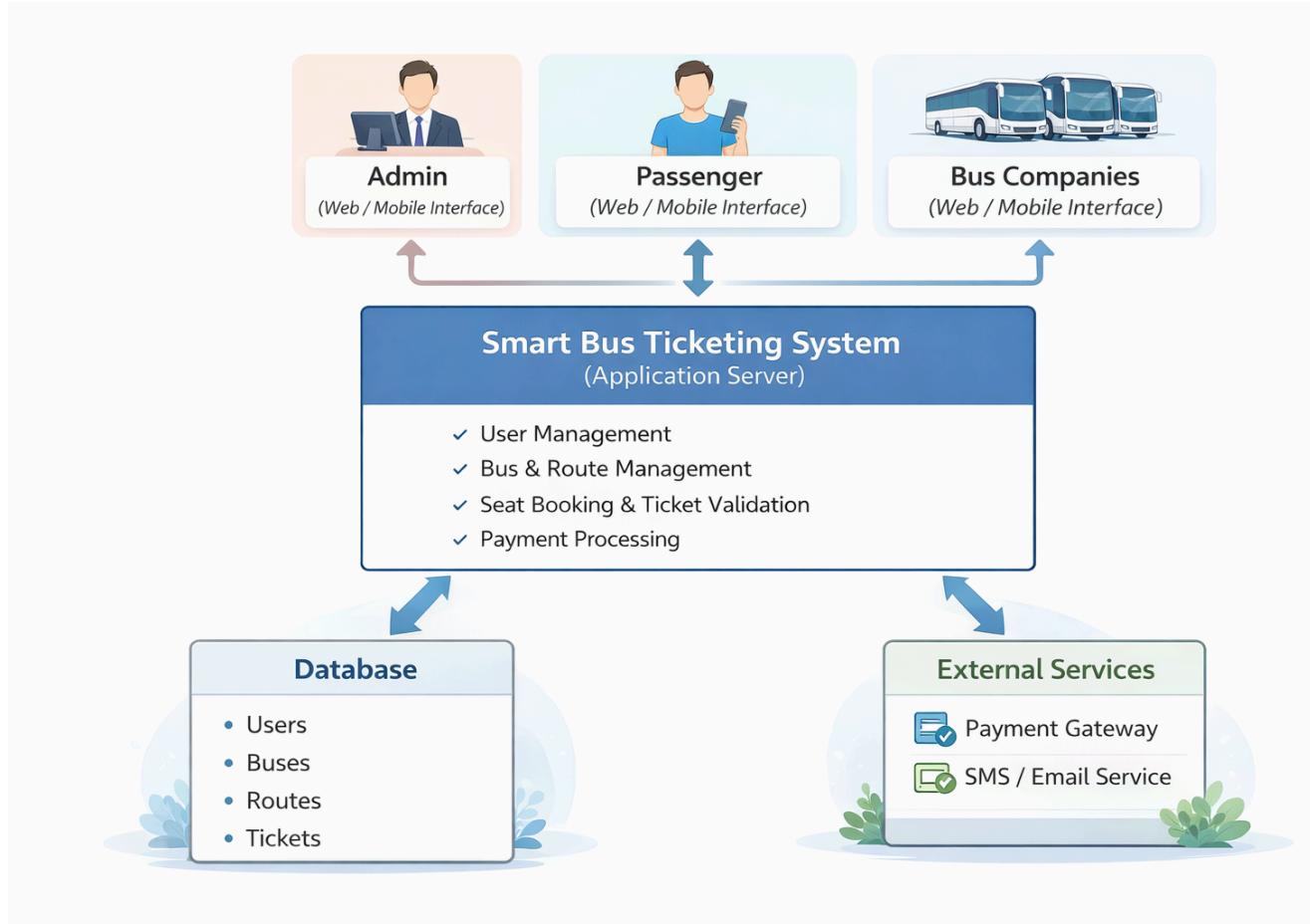


Figure 2.1 Major components of the overall system

2.2 Product Features and Functions

2.2.1 Core User Functions

→ Route Search and Bus Comparison

Users can search for available buses based on origin, destination, and travel date. The system allows filtering by price, travel duration, bus company, and additional services. Real-time seat availability is displayed, and users may compare multiple options before making a decision.

→ Registration and Login

Travelers may create accounts using email or authenticate using external providers such as Google or Facebook. Accounts store personal details, booking history, and preferences. Password recovery mechanisms are included to ensure accessibility.

→ Viewing Trip and Bus Information

Before making a reservation, users may check comprehensive details such as the bus type, departure and arrival times, route, seat arrangement, and ticket cost.

→ Processing Payments Online

Passengers can use integrated digital payment methods to purchase tickets thanks to this feature. Before completing the reservation, the system verifies the payment.

→ Choosing a seat and purchasing a ticket

Passengers can purchase tickets and choose their chosen seats from the available alternatives. In order to prevent double booking, the system temporarily reserves seats throughout the booking process.

→ Generation of Digital Tickets

The technology creates a digital ticket with booking information and a unique identifier, like a QR code, following a successful booking and payment.

→ Validation of Tickets

To guarantee that every passenger has a valid ticket for the journey, conductors or authorized personnel validate tickets during boarding.

2.2.2 Extended and Value-Added Features

→ AI-Driven Assistance

A conversational chatbot is available to support users by answering questions, guiding them through the booking process, and providing quick access to frequently requested information.

→ **Promotional and Loyalty Features**

SwiftBus incorporates discounts, seasonal offers, group fare reductions, and potentially a points-based reward system to encourage long-term user engagement.

→ **Language Options**

The interface is available in both English and Amharic, enabling accessibility for a wide group of users across Ethiopia.

→ **Bus Company Profiles and Ratings**

Each operator has a dedicated profile with information about services, amenities, and user-submitted ratings and reviews.

→ **Ticketing via QR Codes**

Digital tickets come with QR codes that can be scanned for fast and precise verification, which minimizes errors and manual checks.

→ **Ticket Management and Booking History**

Through their accounts, travelers may check travel details, download tickets, and examine previous and future reservations.

→ **Updates on Seats in Real Time**

Following each reservation or cancellation, seat availability is promptly updated, guaranteeing reliable information for all users.

→ **Access on Multiple Platforms**

Users can access the system via mobile and online interfaces, giving them flexibility and convenience.

2.2.3 Administrative and Backend Tools

→ **Administrative Dashboard**

Bus companies and system administrators have access to dashboards for managing routes, schedules, prices, user complaints, and financial reports. Administrators can also audit platform activity and manage system-level configurations.

→ Bus Management:

Administrators and bus companies can add, modify, or remove bus information, such as capacity, type, and registration details.

→ Route and Schedule Management:

This feature enables administrators to set up routes, assign schedules, and control departure and arrival times.

→ User and Role Management:

Administrators can manage user accounts, assign roles (admin, bus company staff, conductor, passenger), and control access permissions.

→ Payment and Transaction Monitoring:

The system keeps track of all payment transactions, allowing administrators to monitor revenue and make decisions.

→ Reporting and Analytics:

The system generates reports on bookings, seat occupancy, revenue, and system usage.

2.3 User Classes and Characteristics

2.3.1 Primary User Classes(Favored user classes)

Travelers

This group represents the largest portion of system users and includes individuals traveling for personal, educational, or business reasons. Their digital literacy varies widely, which necessitates an interface that is

simple, predictable, and mobile-friendly. Many will rely on smartphones, making responsiveness a critical design requirement.

Bus Company Operators

Operators interact with the system daily to update routes, verify bookings, monitor seat occupancy, and generate reports. Their needs center on efficiency, accuracy, and reliability, as they depend on the system to manage significant volumes of passengers.

2.3.2 Secondary User Classes

System Administrators

These users maintain the platform's technical health. They address user issues, monitor performance, configure security protocols, and ensure integration with external APIs.

Customer Support Personnel

Support agents assist travelers with difficulties related to booking, refunds, or payment issues. Their tools require access to booking details but must remain secure and privacy-compliant.

2.3.3 Indirect User Classes

Financial Institutions

These systems interact with our system continuously through APIs to validate and process payments.

Government Authorities

Transport regulators may use the platform periodically to access reports, check compliance, and analyze travel trends.

2.4 General Constraints

1. Operating Environment Constraint

The system needs to function in a mobile and web-based setting. Both Android smartphones and contemporary web browsers should be able to access it. This limitation guarantees that users can access the system without installing specific software or hardware.

2. Hardware Platform Constraint

Standard desktop PCs, tablets, and low- to mid-range cellphones must all be able to use the system effectively. This restriction strives to provide adequate performance even on devices with limited memory and processing power because user devices vary widely.

3. Operating System Constraint

The program must work with widely used operating systems, such as Windows or Linux for backend servers and Android for mobile users. This forces developers to employ cross-platform-compatible technology and restricts the selection of platform-specific features.

4. Internet Connectivity Constraint

For real-time functions like ticket booking, payment processing, and ticket validation, the system depends on reliable internet access. Real-time feature deployment is limited in regions with shaky networks since the system must manage delays and avoid data discrepancies.

5. Integration Constraint

External services including payment gateways, SMS/email notification services, and QR code scanning tools must be integrated with the system. These third-party services impose limits, security requirements, availability, and APIs on developers.

6. Regulatory and Policy Constraint

Ethiopian data protection policies, digital payment laws, and national transportation restrictions must all be followed by the system. Options for transaction handling, data storage, and system design are limited by these regulatory restrictions.

7. Performance Constraint

Multiple users carrying out tasks at once, such purchasing tickets for the same bus, must be supported by the system. To prevent seat overbooking or data conflicts, this requirement necessitates concurrency management and streamlined database operations.

8. Security Constraint

Strong security mechanisms, such as role-based access control, encrypted communication, and secure authentication, must be enforced by the system. This necessitates adherence to secure development principles and restricts the usage of unsecure technologies.

9. Technology and Tool Constraint

To guarantee maintainability and scalability, development must make use of widely supported online and mobile technologies, databases, and frameworks. Compatibility and long-term support considerations limit the selection of databases, tools, and programming languages.

10. Language and Localization Restrictions

Both English and regional languages must be supported by the system. Database architecture, content management, and user interface design are all impacted by this limitation.

11. Constraint on Maintenance and Coding Standards

To enable future maintenance by the customer's organization, the program must adhere to established design rules and coding methods. This limits the usage of custom solutions that are undocumented or excessively complicated.

12. Constraint on Parallel Operations

The system must enable error-free simultaneous use by several passengers, bus firms, and administrators. System architecture, database locking techniques, and transaction management are all impacted by this limitation.

2.5 User Documentation

To support various user groups and guarantee efficient system usage, the Smart Bus Ticketing System will be supplied with the following user documentation components:

User Manual

Administrators, employees of the bus firm, and passengers will all receive an extensive user manual. It will contain detailed instructions for system management, ticket booking, payment, and ticket validation. The handbook will be accessible online and in PDF format.

Online Help System

Both the web and mobile platforms will incorporate an in-application online assistance option. Frequently asked questions (FAQs), context-sensitive help, and brief instructions for routine operations will all be provided.

Tutorials and User Guides

To assist new users in immediately grasping the system, brief lessons and quick-start guidelines will be offered. These could include walkthroughs and visual aids that are delivered in text and picture formats.

Administrative Guide

For system administrators and bus company operators, a different administrative handbook will be created. System configuration, route management, seat assignment, reporting, and user administration processes will all be covered in this paper.

Technical Documentation

Developers and maintenance teams will have access to technical documentation. System architecture, database design, API documentation, and deployment recommendations will all be included.

2.6 Assumptions and Dependencies

Assumptions

When creating the specifications for the Smart Bus Ticketing System, the following Assumptions were made. Modifications to these presumptions could impact the system's functionality, timeline, or scope:

- Smartphones and computers with internet connectivity will be available to users (passengers, administrators, and bus business employees).
- Route, schedule, and seat availability information shall be provided by bus companies in a timely and reliable manner.
- Third-party payment services will be accessible, dependable, and adhere to regional financial laws.
- In order to use web-based or mobile applications, users must possess a basic level of digital literacy.
- Users of the system will adhere to established security and usage policies.
- Travelers are willing to adopt digital booking practices.
- Bus companies and financial institutions will cooperate fully in providing the necessary data and access.

Dependencies

The successful operation of the Smart Bus Ticketing System depends on the following external factors:

- Users possess basic smartphone or internet access.
- Payment gateways (Telebirr, CBE Birr) remain operational and accessible.
- SMS and email services are available for notifications.
- Bus operators regularly update route and schedule data.

3. SPECIFIC REQUIREMENTS

3.1 User Requirements

3.1.1 Functional User Requirements

User Account Management

Requirement ID	FR-01
Requirement	User Registration
Description	The system shall allow users to create accounts using phone numbers or email addresses.
Source	Passenger / Stakeholders
Priority	High
Related Requirements	FR-02, FR-03

Requirement ID	FR-02
Requirement	User Login
Description	The system shall allow users, admins, and staff to log in securely using passwords.
Source	Passenger / Admin
Priority	High
Related Requirements	FR-01, NFR-02

Requirement ID	FR -03
Requirement	Profile Management
Description	The system shall allow users to view and update their personal profile information.
Source	Passenger
Priority	Medium
Related Requirements	FR-01

Requirement ID	FR-04
Requirement	Concession Application
Description	The system shall allow eligible users to apply for fare concessions such as student or special-category discounts.
Source	Passenger / Admin
Priority	Medium
Related Requirements	FR-07, FR-20

Bus, Route, and Schedule Management

Requirement ID	FR-05
Requirement	Bus Registration
Description	The system shall allow administrators to register buses with bus ID, capacity, route, and driver information..
Source	Admin
Priority	High
Related Requirements	FR-06

Requirement ID	FR-06
Requirement	Bus Update and Deletion
Description	The system shall allow administrators to update or delete existing bus records..
Source	Admin
Priority	High
Related Requirements	FR-05

Requirement ID	FR-07
Requirement	Route and Schedule Management
Description	The system shall allow administrators to create and manage routes, schedules, departure times, arrival times, and stop locations.
Source	Admin
Priority	High

Requirement ID	FR-07
Requirement	Route and Schedule Management
Related Requirements	FR-08

Requirement ID	FR-08
Requirement	Active Tickets Display
Description	The system shall display active tickets for each bus and route to authorized staff in real time.
Source	Admin
Priority	Medium
Related Requirements	FR-12

Requirement ID	FR-09
Requirement	Route Selection
Description	The system shall allow users to select routes based on origin, destination, and travel date.
Source	Passenger
Priority	High
Related Requirements	FR-07

Ticket booking and Payment

Requirement ID	FR-10
Requirement	Fare Calculation
Description	The system shall automatically calculate fares based on route, distance, seat type, and concessions.
Source	Passenger / Finance
Priority	High
Related Requirements	FR-04, FR-20

Requirement ID	FR -11
Requirement	Seat Booking
Description	The system shall allow users to view real-time seat availability and reserve seats without double booking
Source	Passenger
Priority	High
Related Requirements	FR-10
Requirement ID	FR-11

Requirement ID	FR-12
Requirement	Payment Processing
Description	The system shall support payments through Telebirr and CBE Birr and

Requirement ID	FR-12
Requirement	Payment Processing
	verify payment before booking confirmation.
Source	Passenger / Finance
Priority	High
Related Requirements	FR-13, NFR-02

Requirement ID	FR-13
Requirement	Ticket Generation
Description	The system shall generate downloadable digital tickets containing ticket ID, seat number, route details, and QR code..
Source	Passenger
Priority	High
Related Requirements	FR-12, FR-14

Administration and Operations

Requirement ID	FR-14
Requirement	Ticket Verification
Description	The system shall allow conductors to scan QR codes to verify ticket validity.
Source	Admin
Priority	High
Related Requirements	FR-13

Requirement ID	FR-15
Requirement	Driver Portal
Description	The system shall allow drivers to view assigned routes, passenger lists, and update departure or delay status.
Source	Driver
Priority	High
Related Requirements	FR-07

Requirement ID	FR-16
Requirement	Admin Dashboard
Description	The system shall provide an admin dashboard to manage buses, routes, bookings, drivers, and complaints
Source	Admin
Priority	High
Related Requirements	FR-05, FR-07

Requirement ID	FR-17
Requirement	Real-Time System Monitoring
Description	The system shall allow administrators to monitor system performance, bus status, and operational alerts in real time.
Source	Admin / Operations Team
Priority	High
Related Requirements	NFR-01

Requirement ID	FR-18
Requirement	Financial Reporting
Description	The system shall generate financial reports including daily revenue, payment summaries, and transaction history.
Source	Finance Department
Priority	High
Related Requirements	FR-12

Requirement ID	FR-19
Requirement	Fare Management
Description	The system shall allow administrators to define, update, and manage fare structures for different routes and user categories.
Source	Admin / Finance
Priority	High
Related Requirements	FR-10

3.1.2 Non-Functional Requirements

Performance Requirements

Requirement ID	NFR-01
Requirement	System Response Time
Description	The system shall respond to user requests within 3 seconds under normal operating conditions.
Source	Stakeholders
Priority	High
Related Requirements	FR-09, FR-11

Requirement ID	NFR-02
Requirement	Ticket Validation Speed
Description	The system shall validate tickets via QR code scanning within 2 seconds.
Source	Operations Team
Priority	High
Related Requirements	FR-14

Requirement ID	NFR-03
Requirement	Concurrent User Support
Description	The system shall support at least 10,000 concurrent users without performance degradation.
Source	Business Requirement
Priority	High
Related Requirements	FR-11, FR-12

Security Requirements

Requirement ID	NFR-04
Requirement	Data Encryption
Description	The system shall encrypt sensitive user data both at rest and in transit using industry-standard encryption methods.
Source	Security Policy
Priority	High
Related Requirements	FR-01, FR-02

Requirement ID	NFR-05
Requirement	Authentication Security
Description	The system shall implement secure authentication mechanisms, including password hashing and role-based access control.
Source	Security Policy
Priority	High
Related Requirements	FR-02, FR-16

Requirement ID	NFR-06
Requirement	PCI DSS Compliance
Description	The system shall comply with PCI DSS standards for handling and processing payment transactions.
Source	Regulatory Requirement
Priority	High
Related Requirements	FR-12

Requirement ID	NFR-07
Requirement	System Availability
Description	The system shall maintain an availability of 99.9% excluding scheduled maintenance.
Source	Stakeholders
Priority	High
Related Requirements	FR-17

Requirement ID	NFR-08
Requirement	Data Backup and Recovery
Description	The system shall perform automated daily backups and support data recovery in case of system failure.
Source	Operations Team
Priority	High
Related Requirements	FR-18

Usability Requirements

Requirement ID	NFR-09
Requirement	Web Accessibility
Description	The system shall comply with WCAG accessibility guidelines to ensure usability for users with disabilities.
Source	Accessibility Standards
Priority	Medium
Related Requirements	FR-01

Requirement ID	NFR-10
Requirement	Mobile Responsiveness
Description	The system shall be fully responsive and usable across mobile, tablet, and desktop devices.
Source	User Experience Requirement
Priority	High
Related Requirements	FR-09

Requirement ID	NFR-11
Requirement	Multilingual Support
Description	The system shall support multiple languages, including Amharic and English, with localized formats.
Source	Business Requirement
Priority	High
Related Requirements	FR-01

Scalability Requirements

Requirement ID	NFR-12
Requirement	User Scalability
Description	The system shall scale to support increasing numbers of registered and active users.
Source	Business Strategy
Priority	High
Related Requirements	FR-03, FR-11

Requirement ID	NFR-13
Requirement	Geographic Expansion
Description	The system shall support expansion to multiple cities and regions without major architectural changes.
Source	Business Strategy
Priority	Medium
Related Requirements	FR-07, FR-19

System Requirements

Detailed Requirements for Selected Functional Requirements

1. User Registration (FR-1)

- The system shall allow users to register by providing an email, mobile number, password, and acceptance of terms and conditions.
- The system shall validate email format, mobile number format, and password strength.
- The system shall prevent registration using a duplicate email address.
- The system shall send a one-time password (OTP) to the user's mobile number for verification.
- The system shall verify the OTP within a limited validity period.
- The system shall create a user account only after successful OTP verification.
- The system shall activate the user account after email verification.
- The system shall display appropriate error messages for invalid input, expired OTP, or verification failure.
- The system shall securely store user credentials.
- The system shall complete the registration process within acceptable response time limits.

2. User Login(FR-2)

- ➔ The system shall allow registered users to log in using valid credentials.
- ➔ The system shall authenticate users before granting access.
- ➔ The system shall display an error message for invalid login attempts.
- ➔ The system shall maintain a secure user session after successful login

3. Profile Management(FR-03)

- The system shall allow users to view their profile information.
- The system shall allow users to update editable profile details.
- The system shall validate profile updates before saving changes.
- The system shall prevent unauthorized access to user profiles.

4. Concession Application(FR-4)

- The system shall allow eligible users to apply for travel concessions.

- The system shall allow submission of required supporting documents.
- The system shall allow administrators to approve or reject concession applications.
- The system shall notify users of the application status

5. Bus Registration(FR-5)

- ➔ The system shall allow authorized bus companies to register buses.
- ➔ The system shall store bus details securely.
- ➔ The system shall validate bus information before registration.

6. Bus Update and Deletion(FR-6)

- The system shall allow authorized users to update registered bus details.
- The system shall allow authorized users to deactivate or delete buses.
- The system shall prevent deletion of buses with active trips.

7. Route and Schedule Management(FR-7)

- ➔ The system shall allow administrators to create and manage routes.
- ➔ The system shall allow scheduling of bus trips with date and time.
- ➔ The system shall prevent schedule conflicts.

8. Active Tickets Display(FR-8)

- The system shall display all active tickets for logged-in users.
- The system shall update ticket status in real time.
- The system shall restrict access to ticket data based on user roles.

9. Route Selection(FR-9)

- ➔ The system shall allow passengers to select routes based on source and destination.
- ➔ The system shall display available buses and schedules for the selected route. The system shall notify users when no routes are available.

10. Fare Calculation(FR-10)

- The system shall allow passengers to view fare details before booking.
- The system shall record all payment transactions accurately.
- The system shall provide passengers with payment confirmation details.

11. Seat Booking(FR-11)

- ➔ The system shall allow passengers to select available seats.
- ➔ The system shall prevent double booking of the same seat.
- ➔ The system shall update seat availability in real time.

12. Ticket Generation(FR-12)

- The system shall generate an electronic ticket after successful booking.
- The system shall assign a unique ticket identifier.
- The system shall deliver the ticket to the passenger digitally.

13. Payment Processing(FR-13)

- ➔ The system shall process payments using approved payment services.
- ➔ The system shall confirm payment success before issuing tickets.
- ➔ The system shall handle payment failures and notify users accordingly

14. Ticket Verification(FR-14)

- The system shall allow authorized staff to verify tickets.
- The system shall validate ticket authenticity and usage status.
- The system shall reject invalid or already-used tickets.

15. Driver Portal(FR-15)

- ➔ The system shall provide drivers access to assigned trip details.
- ➔ The system shall display passenger and route information to drivers.

- The system shall restrict driver access to authorized functions only.

16. Admin Dashboard(FR-16)

- The system shall provide administrators with a centralized dashboard.
- The system shall allow management of users, buses, routes, and fares.
- The system shall restrict dashboard access based on admin privileges.

17. Real-Time System Monitoring(FR-17)

- The system shall monitor active bookings and ticket usage in real time.
- The system shall detect and log system errors or anomalies.
- The system shall display system status information to administrators.

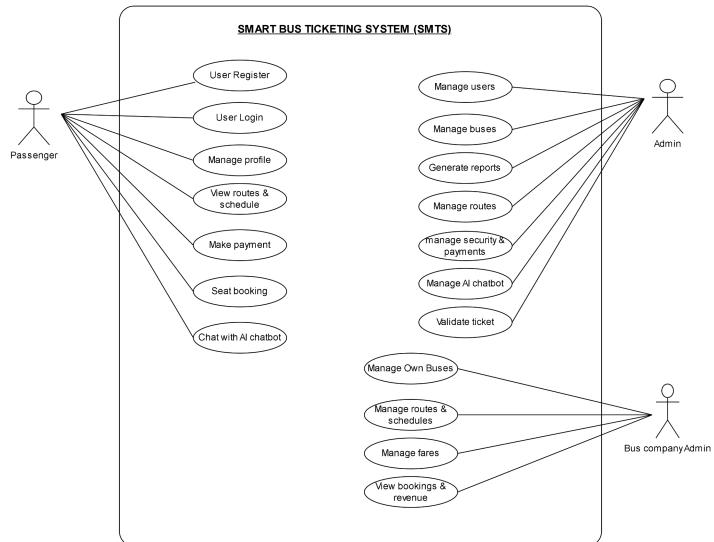
18. Financial Reporting(FR-18)

- The system shall generate reports on ticket sales and revenue.
- The system shall allow administrators to filter financial reports by date and route.
- The system shall ensure accuracy and consistency of financial data

19.Fare Management(FR-19)

- The system shall allow administrators to define and update fare prices.
- The system shall apply fares automatically during booking.

- The system shall ensure fare changes affect only future bookings



4. External Interface Requirements

4.1 User Interfaces

The system shall provide a user-friendly graphical interface accessible via web browsers and mobile devices. The user interface requirements define the logical characteristics of all interfaces between the Smart Bus Ticketing System and its users, including passengers, administrators, and bus company staff. These requirements ensure usability, accessibility, consistency, and responsiveness across supported platforms.

REQ-UI-001: Interface Responsiveness

- ❖ The system shall provide a web-based user interface that adapts to screen resolutions from 320×480 pixels (mobile) to 3840×2160 pixels (4K displays)
- ❖ All interface elements shall remain functional and legible at all supported resolutions
- ❖ Touch targets shall be at least 44×44 pixels on touch-enabled devices

REQ-UI-002: Navigation Consistency

- ❖ Primary navigation shall be consistently positioned across all pages

- ❖ Breadcrumb navigation shall be provided for multi-step processes
- ❖ All pages shall provide access to help documentation

REQ-UI-003: Form Validation and Feedback

- ❖ Form validation shall occur in real-time as users complete fields
- ❖ Error messages shall appear adjacent to the problematic field
- ❖ Success confirmation shall be displayed upon completion of actions
- ❖ All user inputs shall be preserved when validation errors occur

REQ-UI-004: Accessibility Compliance

- ❖ The interface shall comply with WCAG 2.1 Level AA accessibility standards
- ❖ All functionality shall be operable via keyboard-only navigation
- ❖ Text alternatives shall be provided for non-text content
- ❖ Color shall not be the sole means of conveying information

REQ-UI-005: Multi-language Support

- ❖ The interface shall support English as the primary language
- ❖ The system architecture shall support additional language packs without code changes
- ❖ Date, time, and currency formats shall adapt to user's locale settings

REQ-UI-006: Booking Interface Requirements

- ❖ The seat selection interface shall display available seats in a graphical layout
- ❖ Selected seats shall be visually distinguished from available and booked seats
- ❖ Seat selection shall require explicit confirmation before proceeding
- ❖ Fare calculation shall update in real-time as selections change

REQ-UI-007: Administrative Interface Requirements

- ❖ Administrative functions shall be separated from user functions
- ❖ Critical administrative actions shall require confirmation
- ❖ Bulk operations shall provide progress indication
- ❖ Audit trails shall be accessible through the administrative interface

4.2 Hardware Interfaces

This section describes the hardware interfaces between the Smart Bus Ticketing System and the physical devices it interacts with. It defines how the system communicates with client devices, servers, and optional peripheral equipment to ensure reliable and efficient operation. QR-code scanners or mobile camera devices will be used for ticket validation.

REQ-HW-001: Client Hardware Requirements

- ❖ The system shall operate on devices with minimum 4GB RAM and modern processors
- ❖ Display support shall include resolutions from 320×480 to 3840×2160 pixels
- ❖ The system shall support standard input devices (mouse, keyboard, touch)

REQ-HW-002: Server Hardware Requirements

- ❖ Application server shall require minimum 8GB RAM and 4 CPU cores
- ❖ Database server shall require minimum 16GB RAM and 8 CPU cores
- ❖ Storage shall support minimum 100GB with expansion capability
- ❖ Network interface shall support minimum 100Mbps throughput

REQ-HW-003: Printing Support

- ❖ The system shall generate print-ready tickets in standard A4 and letter formats
- ❖ Printed tickets shall include scannable QR codes with minimum 300 DPI resolution
- ❖ Print output shall be compatible with common inkjet and laser printers

REQ-HW-004: Peripheral Device Compatibility

- ❖ The system shall support USB barcode scanners using HID keyboard emulation
- ❖ The system shall be compatible with standard thermal printers for ticket printing
- ❖ Future integration shall support GPS tracking devices via standard APIs

4.3 Software Interfaces

This section outlines the software interfaces between the Smart Bus Ticketing System and other software components. It identifies the external and internal software systems that interact with the system and describes the nature of data exchange and integration requirements.

REQ-SW-001: Operating System Compatibility

- ❖ Server operating systems: Ubuntu 20.04 LTS or Windows Server 2019+
- ❖ Client browsers: Chrome 80+, Firefox 75+, Safari 13+, Edge 80+
- ❖ Mobile browsers: Safari 13+ (iOS), Chrome 80+ (Android)

REQ-SW-002: Database Interface

- ❖ Database system: MySQL 8.0+ or PostgreSQL 12+
- ❖ Connection protocol: JDBC 4.2+ (Java) or equivalent for other platforms
- ❖ Connection pooling: Minimum 10, maximum 100 concurrent connections
- ❖ Transaction support: ACID compliance with configurable isolation levels

REQ-SW-003: External Service Interfaces

- ❖ Payment gateway API: REST over HTTPS with JSON payload
- ❖ SMS gateway: HTTP/HTTPS POST with delivery status callbacks
- ❖ Email service: SMTP with TLS or REST API with template support
- ❖ All external APIs shall implement timeout and retry mechanisms

REQ-SW-004: Web Server Requirements

- ❖ Web server: Apache 2.4+ or Nginx 1.18+
- ❖ SSL/TLS support: TLS 1.2+ with strong cipher suites
- ❖ Compression: GZIP compression for text-based resources
- ❖ Caching: Appropriate HTTP cache headers for static resources

4.4 Communications Interfaces

This section defines the communication interfaces required by the Smart Bus Ticketing System. It specifies the communication mechanisms, data exchange methods, and security requirements used for transmitting information between system components and external services.

REQ-COM-001: Network Protocols

- ❖ Application protocol: HTTP/1.1 and HTTPS (TLS 1.2+)
- ❖ Email protocol: SMTP with STARTTLS (port 587)
- ❖ Real-time updates: WebSocket or Server-Sent Events (optional)

REQ-COM-002: Data Formats

- ❖ API data exchange: JSON (UTF-8 encoding)
- ❖ File exports: CSV, PDF, Excel formats
- ❖ Image formats: JPEG, PNG, SVG for graphical elements

REQ-COM-003: Security Protocols

- ❖ All external communications shall use TLS 1.2+ encryption
- ❖ API authentication shall use JWT tokens with 24-hour expiration
- ❖ Session management shall use secure, HttpOnly cookies
- ❖ Rate limiting shall be implemented for all public APIs (100 requests/minute)

REQ-COM-004: Performance Requirements

- ❖ API response time shall be under 500ms for 95% of requests
- ❖ Page load time shall be under 3 seconds for 95% of pages
- ❖ Concurrent connections shall be supported up to 1,000 users
- ❖ Database connection latency shall be under 100ms

REQ-COM-005: Error Handling

- ❖ Network timeouts shall be configurable (default: 30 seconds)
- ❖ Graceful degradation shall occur during service interruptions

- ❖ User-friendly error messages shall be provided for common failures
- ❖ All communication errors shall be logged with diagnostic information

5. Analysis Models

5.1 System Use Case Model

Actors:

1. **User (Passenger):** Books tickets and manages personal bookings
2. **Bus company :**Manages daily operations and schedules
3. **Admin (System Administrator):** Manages system configuration and users

Use Cases:

User (Passenger) Use Cases:

- User Registration
- User Login
- View Routes
- View Schedules
- Check Seat Availability
- Seat Booking

Buss company admin Use Cases:

1. Manage Company Profile - Update company information
2. Add/Edit Buses - Manage bus fleet
3. Define Routes - Create bus routes with stops
4. Set Fares - Configure pricing
5. Create Schedules - Set departure times

Admin Use Cases:

- User Login
- Manage Buses
- Manage Routes
- Generate Reports

Use Case Relationships:

- All actors require User Login to access system functions
- Seat Booking requires successful Check Seat Availability
- Generate Reports uses data from all booking and scheduling activities

5.1.1 Use Case Description

Use Case Name	User Register
Use Case ID	01
Actor	Passenger
Pre-condition	Passenger has not created an account
Flow of Events	<ol style="list-style-type: none"> 1. Passenger opens the registration page. 2. Passenger enters personal information (name, email/phone, password). 3. System validates the information. 4. System creates the account and confirms registration.
Alternative Flow	If email/phone is already registered, system asks to login or use a different email/phone.
Post Condition	Passenger account is created and can be used to login
Exceptions	Network failure during registration, invalid input format
Priority	High
Special Requirements	Input validation for email, phone, and password strength

Use Case Name	User Login
Use Case ID	02
Actor	Passenger
Pre-condition	Passenger has a registered account
Flow of Events	<ol style="list-style-type: none"> 1. Passenger opens login page. 2. Passenger enters username/email and password. 3. System validates credentials. 4. System grants access to the dashboard.
Alternative Flow	If credentials are incorrect, system prompts to retry or reset password..
Post Condition	Passenger is logged into the system
Exceptions	System downtime, network issues

Priority	High
Special Requirements	Secure password handling, account lockout after multiple failed attempts

Use Case Name	Manage Profile
Use Case ID	03
Actor	Passenger
Pre-condition	Passenger is logged in
Flow of Events	<ol style="list-style-type: none"> 1. Passenger opens profile settings. 2. Passenger updates personal info or password. 3. System validates input. 4. System saves changes and confirms update.
Alternative Flow	If input is invalid, system shows error messages.
Post Condition	Profile information updated successfully
Exceptions	Network failure, invalid input
Priority	Medium
Special Requirements	Password encryption, data validation

Use Case Name	View Routes & Schedule
Use Case ID	04
Actor	Passenger
Pre-condition	Passenger is logged in
Flow of Events	<ol style="list-style-type: none"> 1. Passenger selects “View Routes & Schedule” option. 2. System displays available buses, routes, and timings. 3. Passenger can filter by date or destination.
Alternative Flow	No routes available for selected date, system shows message.

Post Condition	Passenger sees all relevant bus schedules
Exceptions	Server downtime, network issues
Priority	High
Special Requirements	Real-time route updates

Use Case Name	Start Booking
Use Case ID	05
Actor	Passenger
Pre-condition	Passenger is logged in and viewing routes
Flow of Events	<ol style="list-style-type: none"> 1. Passenger selects a route and date. 2. Passenger chooses a seat. 3. System confirms seat availability.
Alternative Flow	Seat is no longer available, system prompts to select another seat or route.
Post Condition	Booking session is initiated and ready for payment
Exceptions	Seat selection conflict, network error
Priority	High
Special Requirements	Real-time seat availability check

Use Case Name	Make Payment
Use Case ID	06
Actor	Passenger
Pre-condition	Passenger has selected a route and seat
Flow of Events	<ol style="list-style-type: none"> 1. Passenger selects payment method. 2. Passenger enters payment details. 3. System processes payment. 4. System confirms payment and issues ticket.
Alternative Flow	Payment fails, the system prompts passengers to retry or use

	another method.
Post Condition	Ticket is confirmed and payment completed
Exceptions	Payment gateway failure, invalid payment info
Priority	High
Special Requirements	Secure payment integration, PCI compliance

Use Case Name	Chat with AI Chatbot
Use Case ID	07
Actor	Passenger
Pre-condition	Passenger is logged in
Flow of Events	<ol style="list-style-type: none"> 1. The passenger opens the chatbot window. 2. Passenger types a query or selects a suggested question. 3. The system processes queries and provides a response. 4. Passenger receives answer or guidance.
Alternative Flow	Chatbot cannot answer, system suggests contacting support.
Post Condition	Passenger query is answered or escalated
Exceptions	System downtime, network issues
Priority	Medium
Special Requirements	24/7 AI response, natural language processing

Use Case Name	Validate ticket
Use Case ID	08
Actor	Admin
Pre-condition	Admin has a registered account
Flow of Events	<ol style="list-style-type: none"> 1. Conductor opens the login page on the conductor app/device. 2. Conductor enters username/password or other

	<p>credentials.</p> <ol style="list-style-type: none"> 3. System verifies credentials. 4. System grants access to conductor functions.
Alternative Flow	If credentials are invalid, system prompts to retry.
Post Condition	Admin is logged in and can access the app
Exceptions	Network failure, server downtime
Priority	High
Special Requirements	Secure authentication, account lockout after multiple failed attempts

Use Case Name	Manage Users
Use Case ID	9
Actor	Admin
Pre-condition	Admin is logged in
Flow of Events	<ol style="list-style-type: none"> 1. Admin opens the “Manage Users” section. 2. Admin can add, edit, or delete user accounts (passengers, conductors) 3. The system validates the actions and confirms changes.
Alternative Flow	<ul style="list-style-type: none"> • If input is invalid, the system displays an error message. • Attempt to delete a non-existent user prompts a warning.
Post Condition	User accounts updated successfully
Exceptions	Network failure, server error
Priority	High
Special Requirements	Secure handling of user data, validation checks

Use Case Name	Manage Buses
Use Case ID	10

Actor	Admin
Pre-condition	Admin is logged in
Flow of Events	<ol style="list-style-type: none"> 1. Admin opens the “Manage Buses” section. 2. Admin can add, update, or remove buses (bus ID, capacity, driver info). 3. The system validates and saves the changes.
Alternative Flow	Adding a bus with an existing ID prompts an error.
Post Condition	Bus information updated successfully
Exceptions	Invalid input, network/server issues
Priority	High
Special Requirements	Real-time updates, validation of bus details

Use Case Name	Generate Reports
Use Case ID	11
Actor	Admin
Pre-condition	Admin is logged in
Flow of Events	<ol style="list-style-type: none"> 1. Admin selects the “Generate Reports” option. 2. Admin chooses report type (ticket sales, routes, users, payments). 3. The system generates the report and displays/downloads it.
Alternative Flow	If data is incomplete, the system warns the admin and allows partial reporting.
Post Condition	Report is generated and available for review
Exceptions	Network failure, insufficient data
Priority	Medium
Special Requirements	Exportable formats (PDF, Excel), secure ac

Use Case Name	Manage Routes
Use Case ID	12
Actor	Admin
Pre-condition	Admin is logged in
Flow of Events	<ol style="list-style-type: none"> 1. Admin opens the “Manage Routes” section. 2. Admin can add, edit, or remove bus routes (start, end, stops). 3. System validates changes and updates schedules accordingly.
Alternative Flow	Attempt to create overlapping routes prompts a warning.
Post Condition	Routes updated successfully
Exceptions	Invalid input, server/network failure
Priority	High
Special Requirements	Integration with schedule and booking system

Use Case Name	Manage Security & Payments
Use Case ID	13
Actor	Admin
Pre-condition	Admin is logged in
Flow of Events	<ol style="list-style-type: none"> 1. Admin opens “Security & Payments” settings. 2. Admin can manage user permissions, roles, and payment settings. 3. The system validates changes and confirms updates.
Alternative Flow	Invalid role assignment or payment setup triggers an error
Post Condition	Security settings and payment configurations updated
Exceptions	Network issues, invalid input
Priority	High
Special Requirements	Secure payment processing, role-based access control

Use Case Name	Manage AI Chatbot
Use Case ID	14
Actor	Admin
Pre-condition	Admin is logged in
Flow of Events	<ol style="list-style-type: none"> 1. Admin opens the “AI Chatbot” management section. 2. Admin can configure chatbot responses, update FAQs, and monitor queries. 3. The system validates changes and saves the configuration.
Alternative Flow	Invalid configuration prompts error or warning.
Post Condition	Chatbot updated and ready to interact with passengers
Exceptions	System downtime, network failure
Priority	Medium
Special Requirements	Easy content update, AI response monitoring, secure access

Use Case Name	Manage Own Buses
Use Case ID	15
Actor	Bus Company Admin
Pre-condition	Bus Company Admin is logged in
Flow of Events	<ol style="list-style-type: none"> 1. Admin opens the bus management section. 2. Admin adds, edits, or removes company buses. 3. The system validates bus ownership.
Alternative Flow	Attempt to modify another company’s bus → access denied.
Post Condition	Company bus data is updated
Exceptions	Invalid input, server error
Priority	High
Special Requirements	Company-level access control

Use Case Name	Manage Routes & Schedules
Use Case ID	16
Actor	Bus Company Admin
Pre-condition	Admin is logged in
Flow of Events	<ol style="list-style-type: none"> 1. Admin selects route & schedule management. 2. Admin creates or updates routes and schedules. 3. The system checks route conflicts.
Alternative Flow	Conflict detected → system requests correction.
Post Condition	Routes and schedules updated
Exceptions	Invalid route data, network failure
Priority	High
Special Requirements	Integration with booking system

Use Case Name	Manage Fares
Use Case ID	17
Actor	Bus Company Admin
Pre-condition	Admin is logged in
Flow of Events	<ol style="list-style-type: none"> 1. Admin opens fare management section. 2. Admin sets or updates ticket prices per route. 3. System validates pricing rules.
Alternative Flow	Invalid fare amount → system displays error.
Post Condition	Fare information updated
Exceptions	Data validation error
Priority	High
Special Requirements	Currency consistency and pricing rules

Use Case Name	View Booking & Revenue
Use Case ID	18
Actor	Bus Company Admin
Pre-condition	Admin is logged in
Flow of Events	<ol style="list-style-type: none"> 1. Admin opens booking & revenue dashboard. 2. System displays bookings and total revenue. 3. Admin filters by date, route, or bus.
Alternative Flow	No data available → system shows zero records.
Post Condition	Booking and revenue data displayed
Exceptions	Server/database error
Priority	Medium
Special Requirements	Graphs, export reports (PDF/Excel)

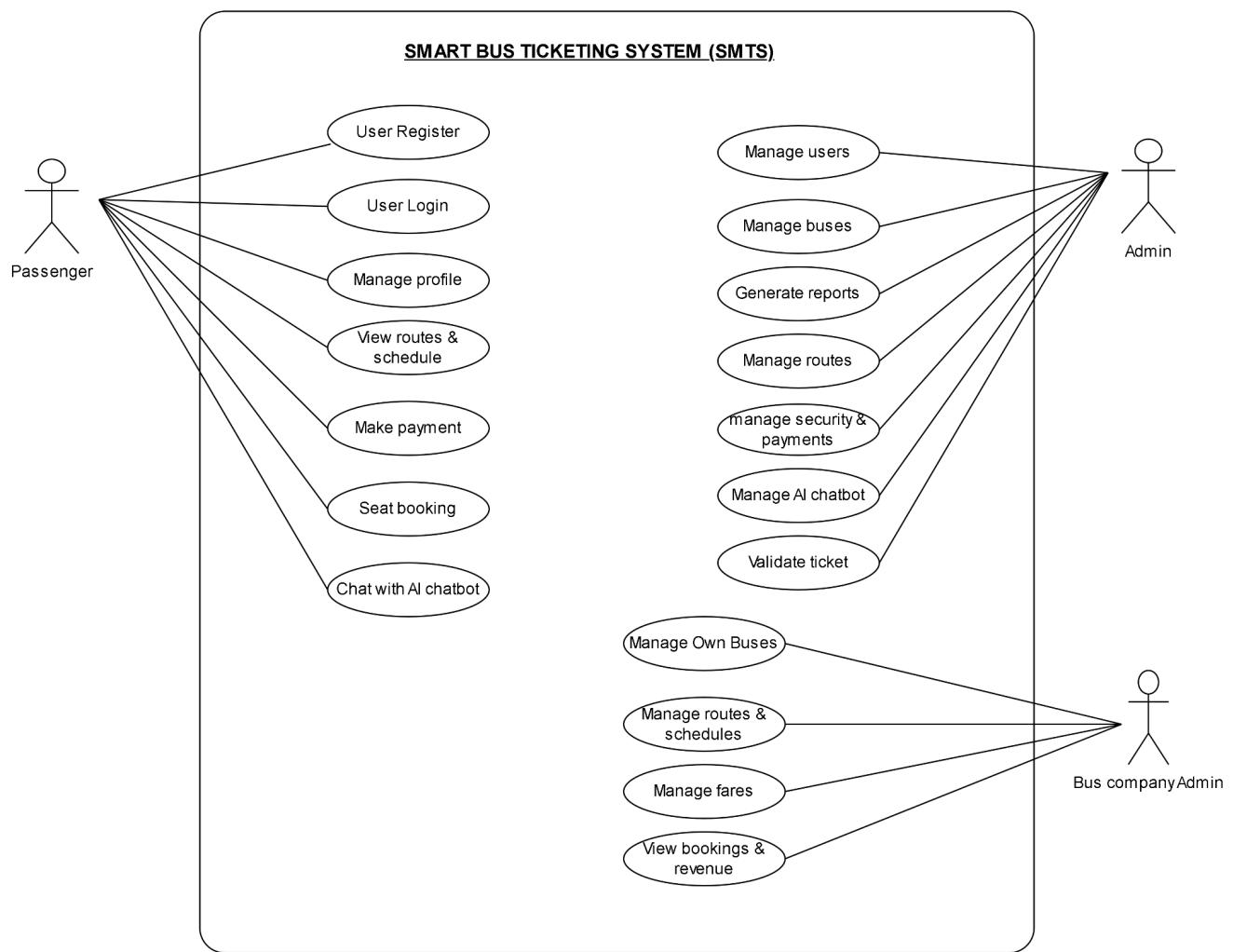


Figure 5.0.1 Essential use case

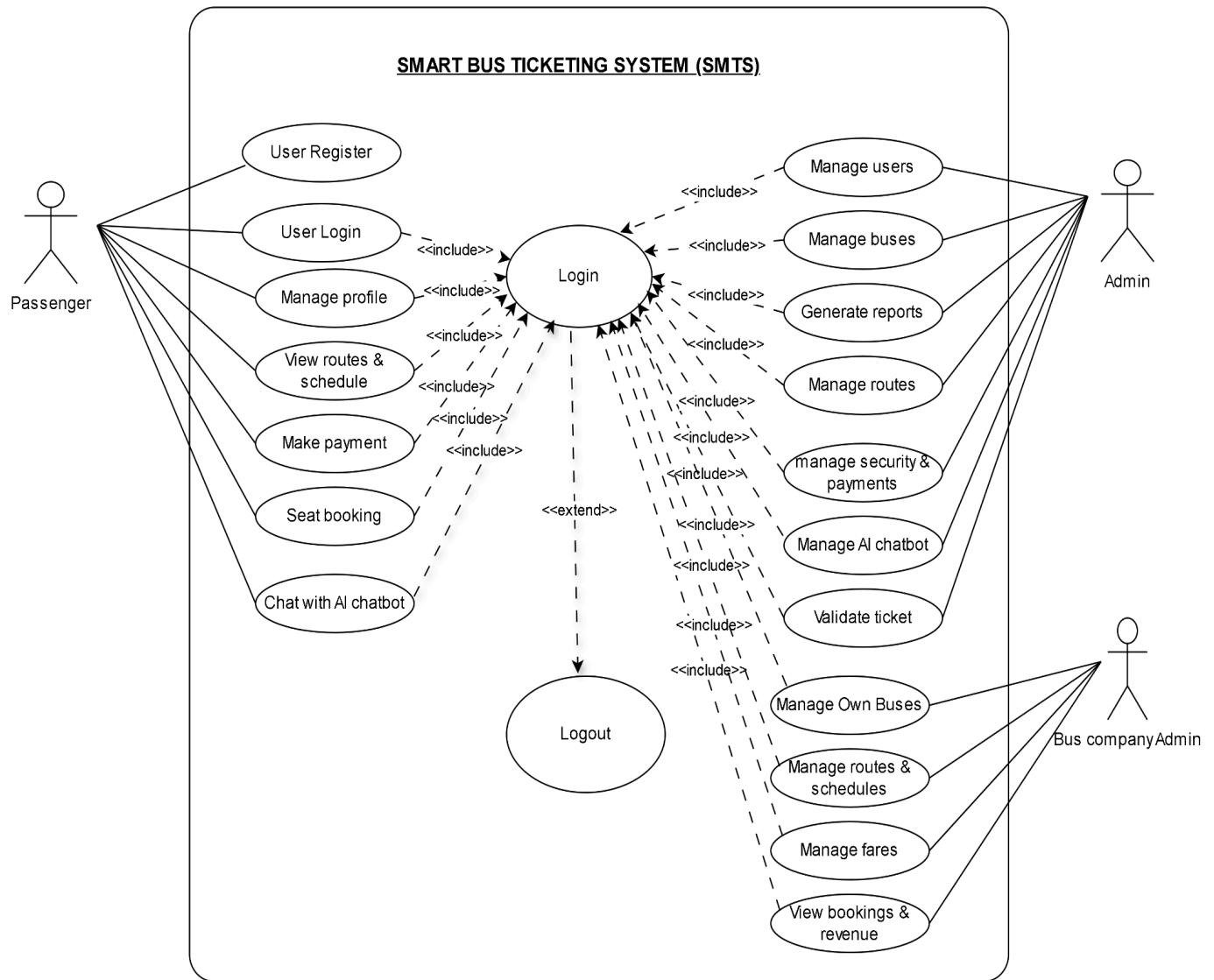


Figure 5.0.2 use case diagram of the system

1.1 Sequence Diagrams

Sequence diagrams represent the chronological order of interactions between system components and actors during the execution of specific use cases. They show how requests and responses are exchanged over time to complete a function such as user registration, ticket booking, or payment processing.

These diagrams help validate system logic, data flow, and interaction timing. Each sequence diagram is derived from the system requirements and supports verification of functional behavior defined in Section 3.1.

Here are some of the main sequences of the system represented by sequence diagrams.

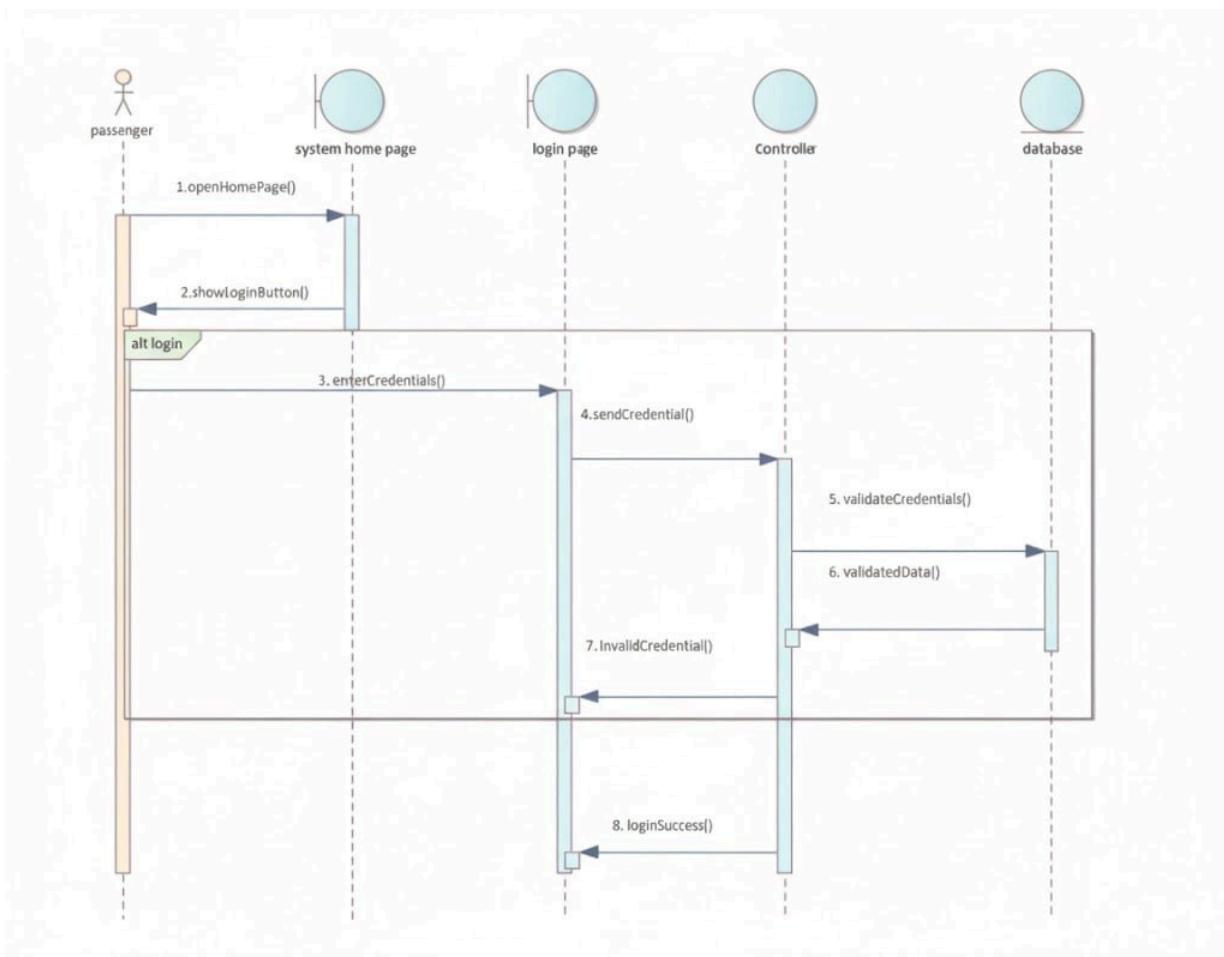


Figure 5.3 Sequence diagram of login

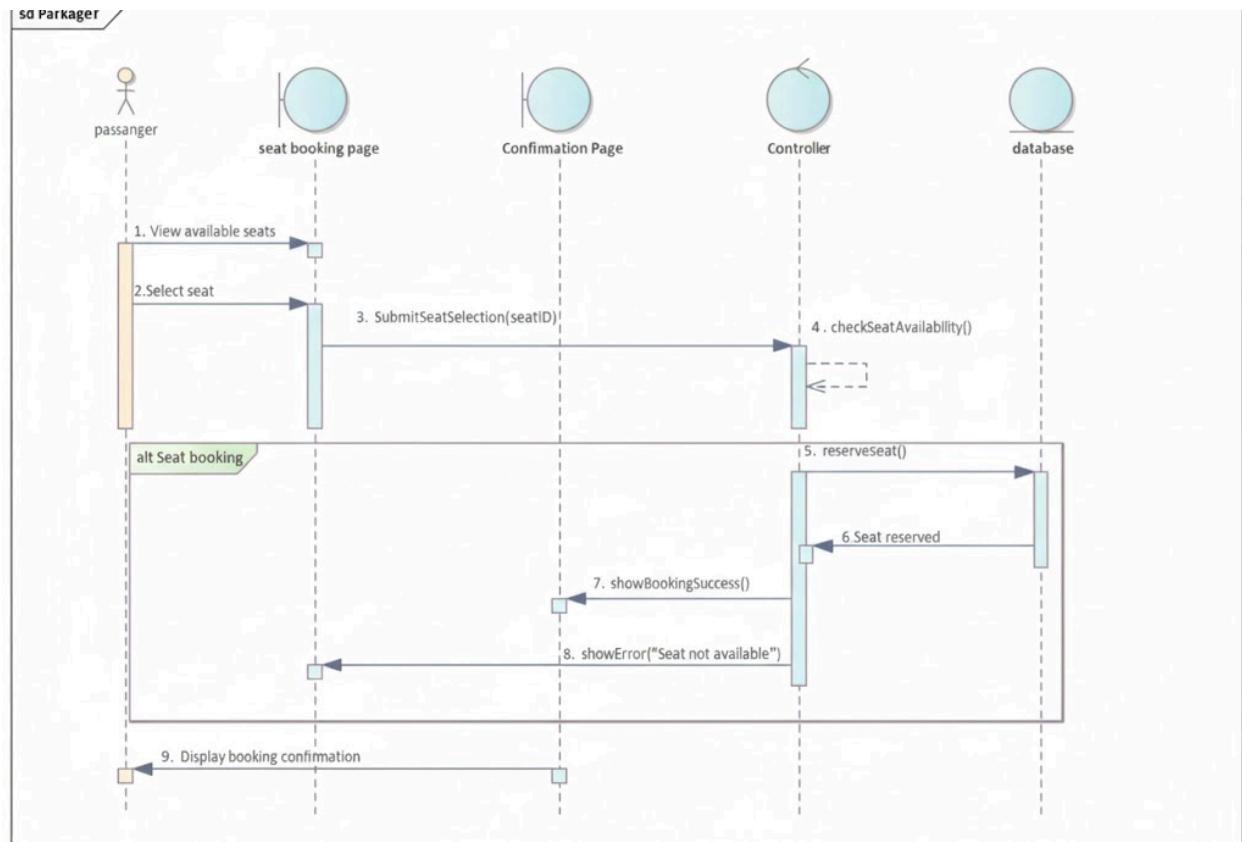


Figure 5.4 sequence diagram of seat booking

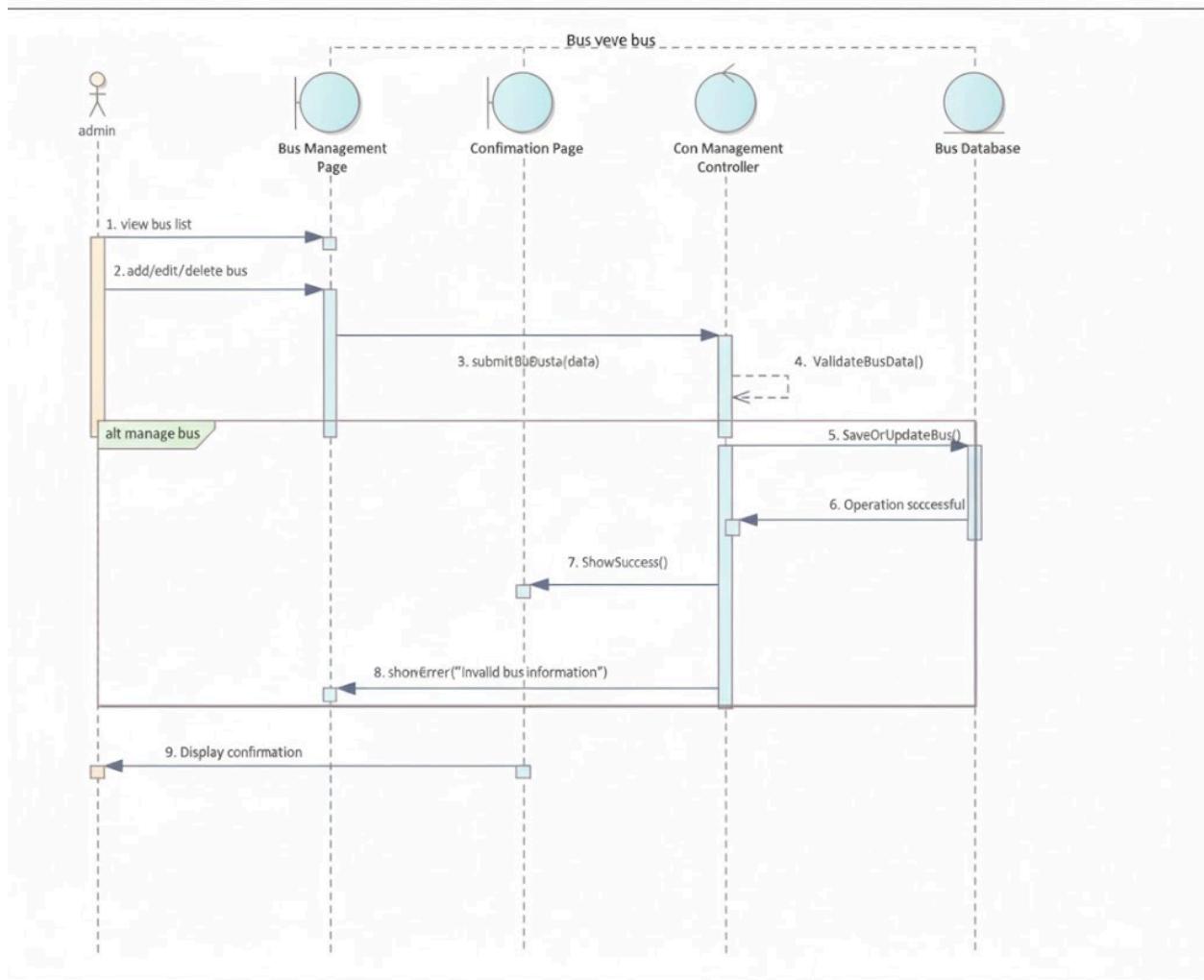


Figure 5.5 sequence diagram of Manage bus

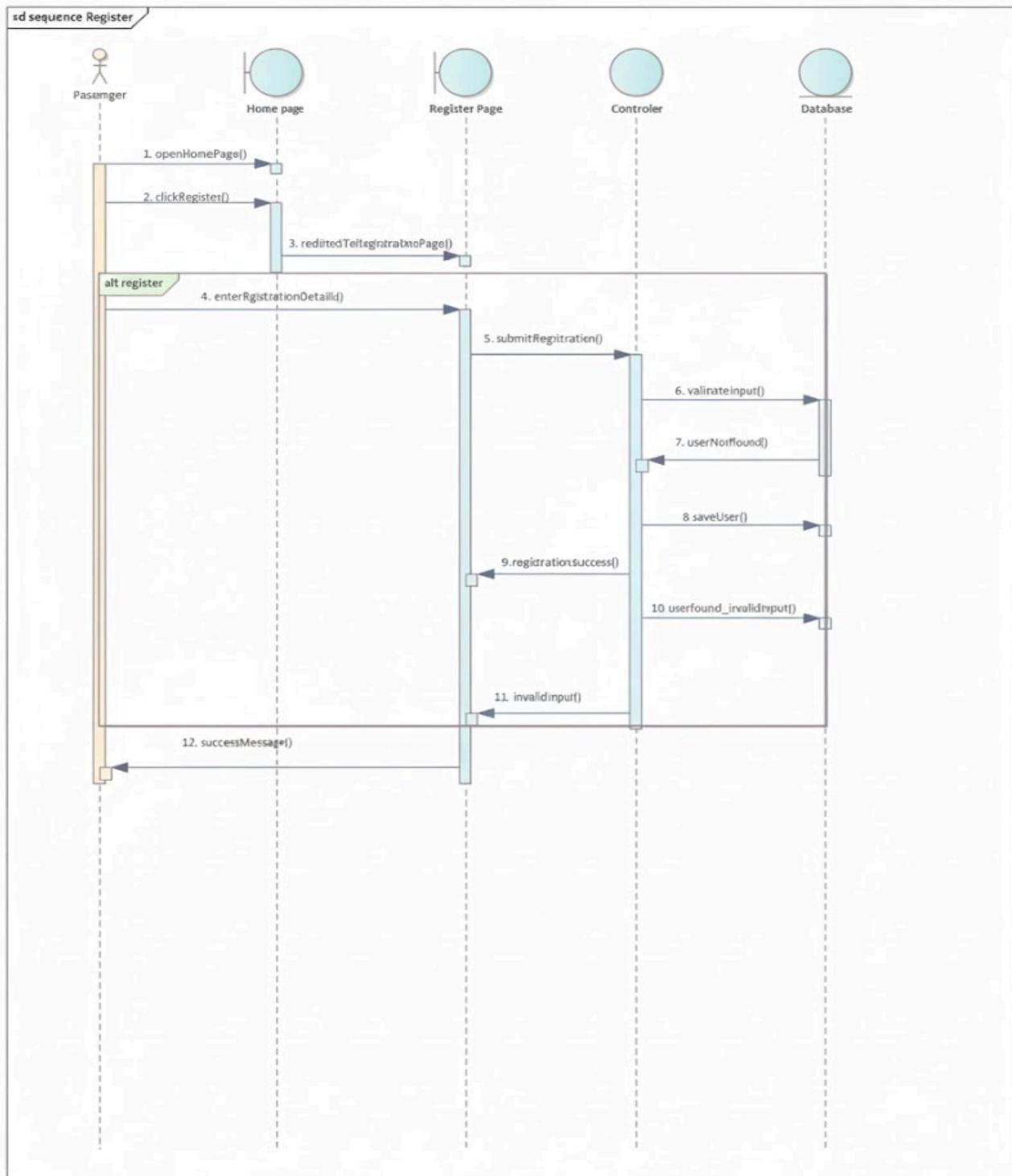


Figure 5.6 sequence diagram of register

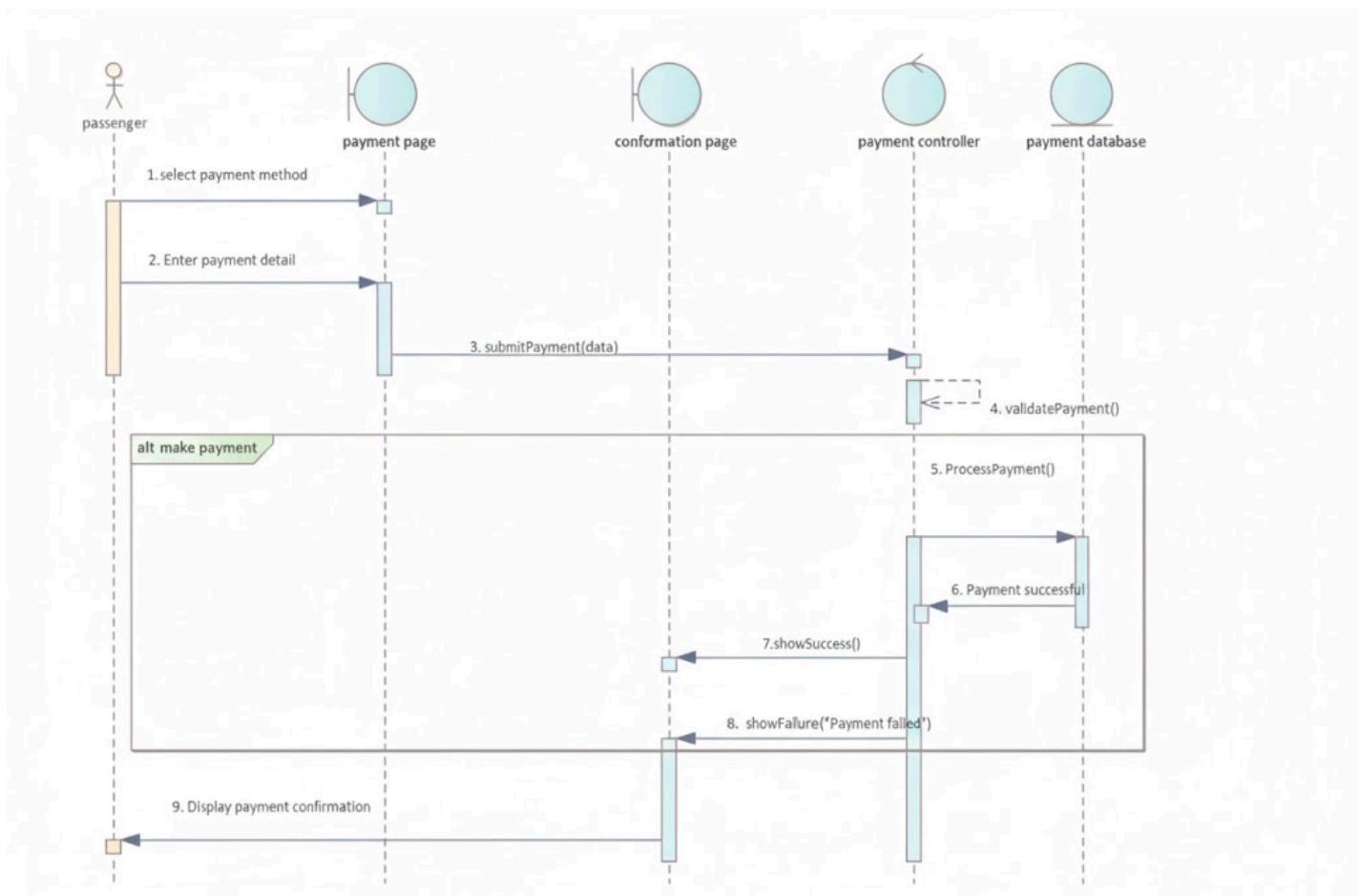


Figure 5.7 sequence diagram of make payment

5.3 Activity Diagrams

Activity diagrams describe the workflow and decision-making processes within the Smart Bus Ticketing System. They model the flow of control from one activity to another, including parallel actions and conditional paths.

These diagrams are used to visualize complex business processes such as booking, concession approval, and ticket verification. Activity diagrams ensure that all possible scenarios and system responses are accounted for and aligned with the requirements specified in the SRS.

Here are some of the main activities of the system represented by activity diagrams.

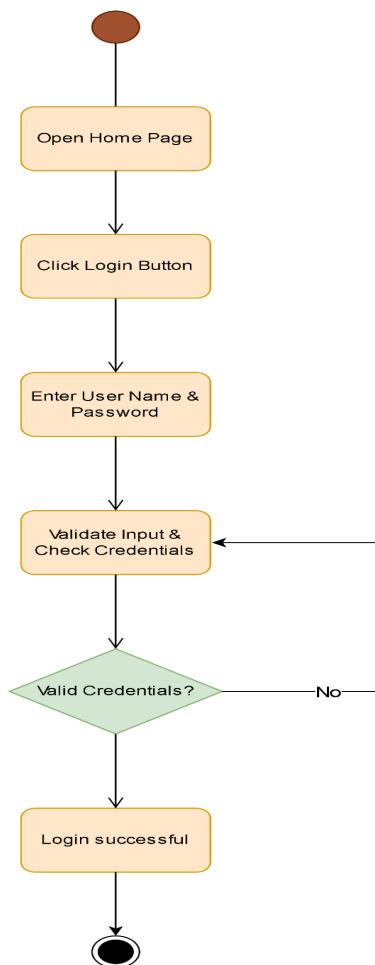


Figure 5.8 Activity diagram of login

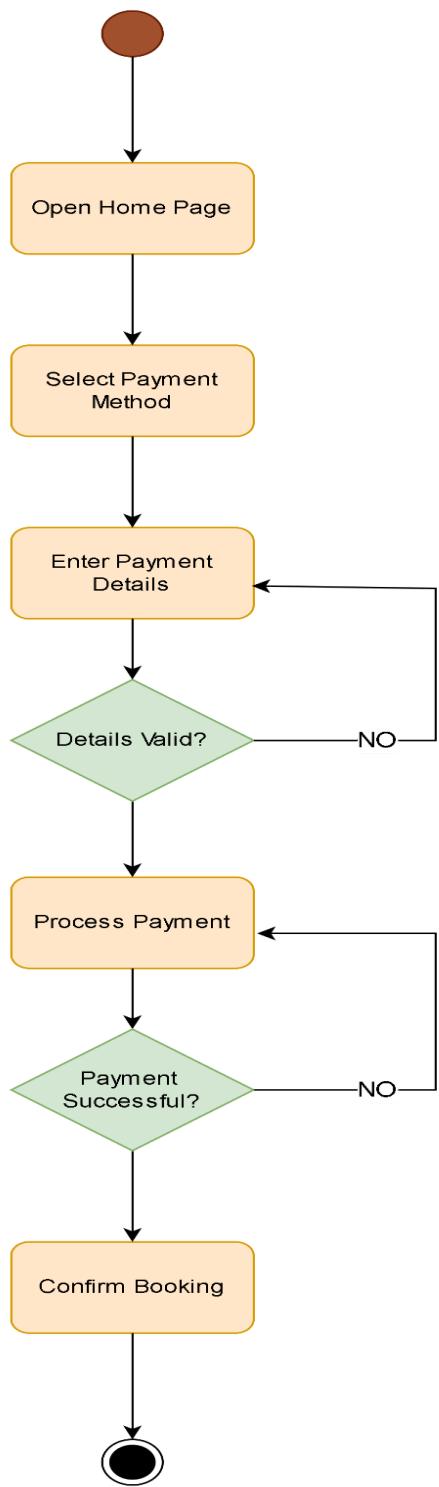


Figure 5.9 Activity diagram of make payment

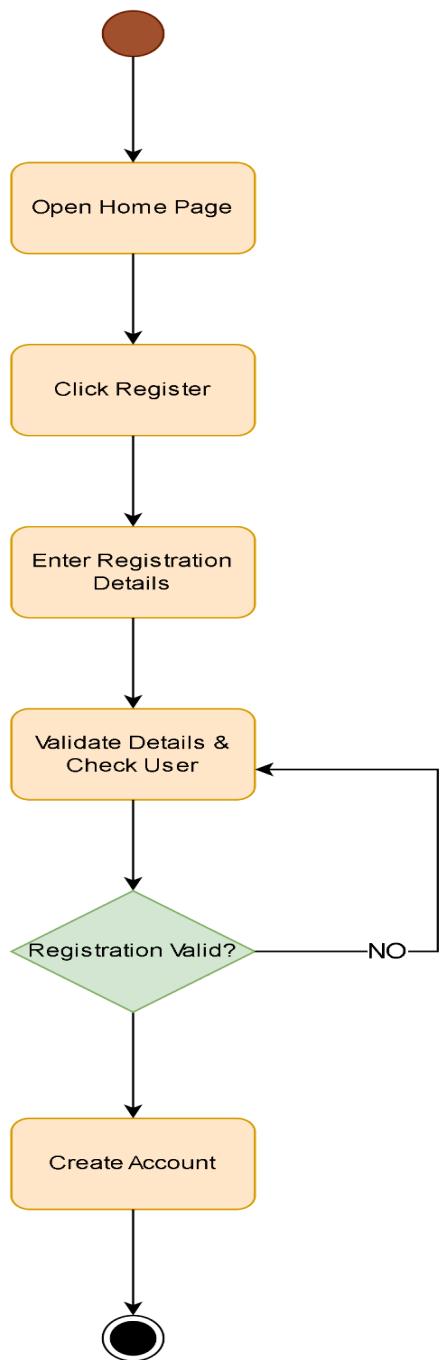


Figure 5.10 Activity diagram of register

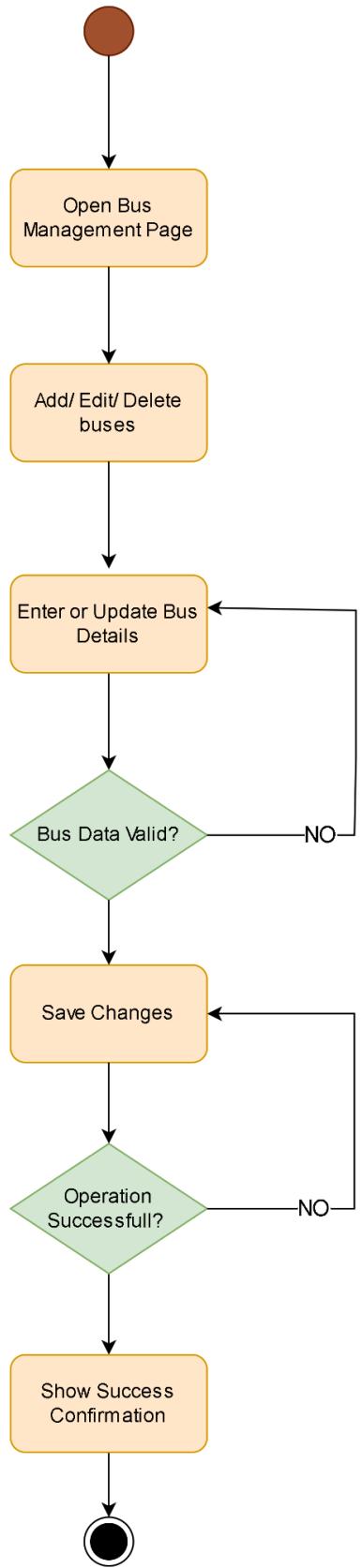


Figure 5.11 Activity diagram of manage bus

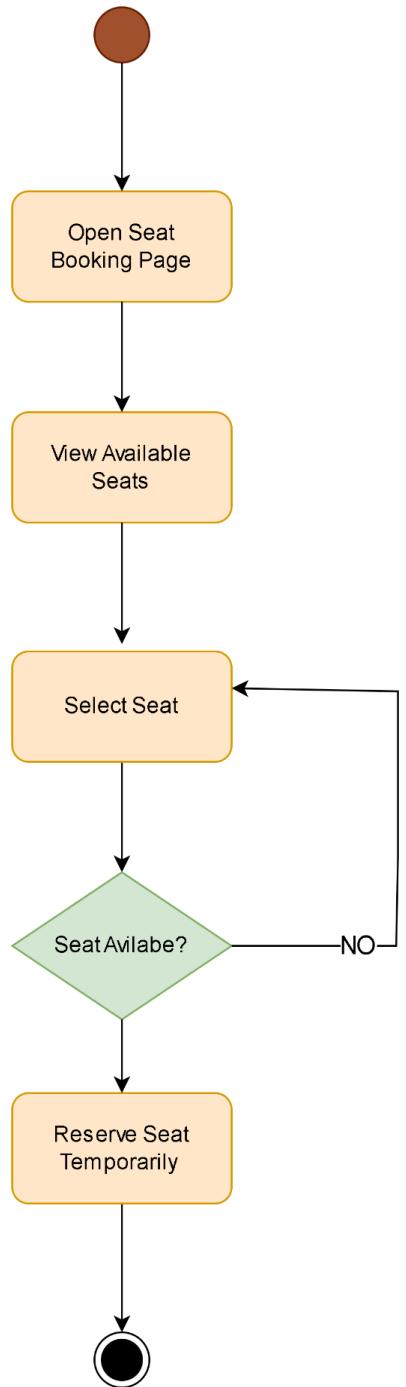


Figure 5.12 Activity diagram of seat booking

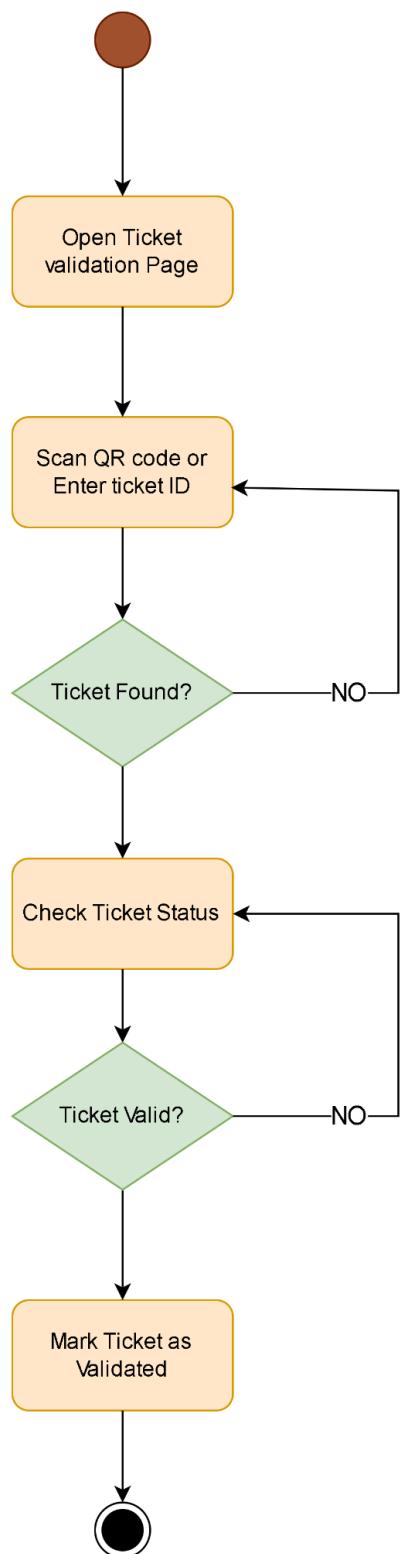


Figure 5.13 Activity diagram of validate ticket

6 Conclusion

The Software Requirements Specification (SRS) for the **Smart Bus Ticketing System (SBTS)** provides a comprehensive roadmap for transitioning from a manual, error-prone ticketing process to a digitized, efficient, and user-centric platform. By addressing the critical challenges identified in the requirements gathering phase—such as long queues, lack of real-time seat information, and insecure payment methods—this system establishes a robust framework for modernizing intercity travel in Ethiopia.

The integration of local payment gateways like **Telebirr** and **CBE Birr**, combined with automated **QR-code ticket validation**, ensures that the system is not only technologically advanced but also culturally and economically relevant to its users. Furthermore, the detailed analysis models, including Use Case, Sequence, and Activity diagrams, ensure that the system's logic is sound and fully traceable to stakeholder needs.

In conclusion, the SBTS is designed to improve operational efficiency for bus operators while significantly enhancing the travel experience for passengers. With a focus on scalability, security, and usability, this document serves as the definitive foundation for the subsequent design, implementation, and testing phases of the software development lifecycle.

7. References

1. IEEE Computer Society

IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications, Version 1998, IEEE Standards Association, 1998.
Source: IEEE Digital Library.

2. IEEE Computer Society

ISO/IEC/IEEE 29148:2018 – Systems and Software Engineering — Life Cycle Processes — Requirements Engineering, Version 2018, ISO/IEC/IEEE, 2018.
Source: ISO Standards Documentation.

3. Pressman, R. S., & Maxim, B. R.

Software Engineering: A Practitioner's Approach, 9th Edition, McGraw-Hill Education, 2020.
Source: Software Engineering textbook.

4. Sommerville, I.

Software Engineering, 10th Edition, Pearson Education Limited, 2016.
Source: Software Engineering reference book.

5. World Wide Web Consortium (W3C)

Web Content Accessibility Guidelines (WCAG) 2.1, Recommendation, June 2018.
Source: W3C Accessibility Standards.

6. PCI Security Standards Council

Payment Card Industry Data Security Standard (PCI DSS) v4.0, Version 4.0, March 2022.
Source: PCI Security Standards Documentation.

7. Telebirr – Ethio Telecom

Telebirr Digital Payment Platform Documentation, Ethio Telecom, Ethiopia.

Source: Telebirr official service documentation.

8. Commercial Bank of Ethiopia (CBE)

CBE Birr Digital Banking System Overview,

Commercial Bank of Ethiopia, Ethiopia.

Source: CBE digital banking service documentation.

9. Oracle Corporation

MySQL 8.0 Reference Manual,

Oracle Corporation, Version 8.0.

Source: MySQL official documentation.

10. The Apache Software Foundation

Apache HTTP Server Documentation,

Version 2.4.

Source: Apache Software Foundation documentation.

11. Mozilla Developer Network (MDN)

Web APIs and JavaScript Documentation,

Mozilla Foundation.

Source: MDN Web Docs.

Appendix

Appendix : Requirement Traceability and Requirement

This appendix provides traceability between stakeholder questionnaire responses and the identified system requirements. It also includes a summary of the questionnaires and responses and the identified system requirements. It also includes a summary of the questionnaires and interviews used during the requirement elicitation phase.

Requirement Traceability Matrix

Questionnaire Item	Requirement ID	Requirement Type	Requirement Description
1. How do you currently purchase bus tickets?	FR-1	Functional	Online booking functionality to replace manual ticketing.
2. What challenges do you face with the manual ticketing process?	FR-2	Functional	Automation to reduce errors and improve efficiency.
3. Would you prefer booking tickets via mobile, web, or both?	NFR-1	NON-Functional	Cross-platform accessibility (web & mobile support).
4. Which payment methods do you trust most (Telebirr, CBE Birr, etc.)?	FR-3	Functional	Integration with local payment gateways.
5. How important is real-time seat availability information to you?	FR-4	Functional	Integration with local payment gateways.
6. What features would make you more likely to use an online ticketing system?	FR-5	Functional	Value-added features (discounts, loyalty, chatbot).
7. How often do you travel intercity by bus?	NFR-2	NON-Functional	Scalability to handle frequent travelers and peak demand.
8. Would you find QR-code	FR-6	Functional	QR-code ticket generation and validation.

based ticket validation useful?			
9. What type of notifications (SMS, email, app alerts) would you prefer for booking updates?	FR-7	Functional	Notification services (SMS, Email, App alerts).
10. What security concerns do you have when making online payments?	NFR-3	NON-Functional	Secure payment processing and data protection.

Questionnaires used to gather requirements

To define the SRS, the team used the following tools:

Bus Operator Interview Summary

- ❖ Automated fare calculation and financial reporting
- ❖ real-time dashboard for route and bus management.
- ❖ QR-based ticket validation to reduce fraud.

Passenger Questionnaire (Sample):

1. How do you currently find bus schedules for intercity travel?
2. What is your biggest frustration with manual ticketing? (Options: Queues, overcharging, lack of seat choice).
3. Would you trust an automated system to handle payments via Telebirr?

Admin/Operator Interview Guide

1. How do you manually track seat occupancy for a specific route?
2. What kind of daily financial reports do you need?
3. How do you currently notify passengers of delays or cancellations?

4. What security measures do you expect in an online ticketing system?

Appendix Usage

This appendix ensures full traceability between stakeholder needs and system requirements. It validates that every requirement is rooted in real user feedback and supports verification during testing, change management, and future system enhancements. The included questionnaires, interview notes, and traceability matrices provide a transparent and auditable record of the requirement elicitation process.