**FACULTY OF SCIENCE, ENGINEERING AND COMPUTING**

**School of *Computer Science & Mathematics***

**BSc DEGREE**

**IN**

**Computer Science (Software Engineering)**

**PROJECT INTERIM REPORT**

Name: Fathima Ranna

ID Number: E187002

Project Title: Mobitix Digital Bus Ticket Booking App

Project Type: Build

Date:

Supervisor: Ms. Maleesha Ekanayake

KU London Logo

Did you discuss and agree the viability of your project idea with your supervisor? Yes or No

Did you submit a draft of your proposal to your supervisor? Yes or No

Did you receive feedback from your supervisor on any submitted draft? Yes or No

**Abstract**

The current bus ticket reservation systems in Sri Lanka have to deal with issues related to lack of real-time tracking, poor refund management, overcrowded buses, and time-consuming manual ticketing processes, which ultimately lead to dissatisfaction among passengers. Therefore, Mobitix, a Digital Bus Ticket Reservation mobile app, will be designed to resolve these issues. As such, Mobitix will transform public transportation ticketing by offering passengers and administrators an easy and efficient way to manage bus ticketing procedures. It provides the advantage of purchasing and reserving tickets, managing user profiles, dynamically choosing seats, and making secure online payments. It also maintains real-time tracking of the bus using GPS and Google Maps to keep passengers updated with the latest travel information. Moreover, the system provides trip history tracking and refund management. Mobitix will also provide an administrator panel that can be used to manage bus schedules, reservations, and performance analytics. Major features offered include generating tickets and real-time integration with bus schedules, improving the user experience and operational efficiency. Accordingly, Mobitix aims to simplify the bus reservation process of Sri Lanka’s public transportation system in this manner.

To achieve these goals, Mobitix leverages modern technologies such as Flutter for cross-platform mobile development, PHP for backend services, and MySQL for database management. The integration of Google Maps API ensures accurate real-time bus tracking, while local payment gateways like Genie, Koko, and eZcash facilitate secure and convenient transactions. By addressing the inefficiencies of traditional ticketing systems, Mobitix not only enhances the passenger experience but also provides bus operators with tools to optimize scheduling, reduce operational costs, and improve overall service quality. This digital transformation aligns with global trends toward sustainable and efficient public transportation, making Mobitix a forward-thinking solution for Sri Lanka’s evolving transport needs.

Contents

[1. Introduction & Literature Review 1](#_Toc191588406)

[1.1 Introduction 1](#_Toc191588407)

[1.2 Background and Motivation 1](#_Toc191588408)

[1.3 Problem in brief 3](#_Toc191588409)

[1.4 Aim & Objectives 3](#_Toc191588410)

[1.4.1 Aim 3](#_Toc191588411)

[1.4.2 Objectives. 3](#_Toc191588412)

[1.5 Scope 4](#_Toc191588413)

[1.6 Deliverables 12](#_Toc191588414)

[1.7 Literature Review 13](#_Toc191588415)

[2. Analysis 18](#_Toc191588416)

[3. Design 30](#_Toc191588417)

[3.1 Design Techniques 30](#_Toc191588418)

[3.2 System Overview 41](#_Toc191588419)

[References 59](#_Toc191588420)

[Appendices 60](#_Toc191588421)

**List of Figures**

[Figure 1 Gantt Chart 9](#_Toc191588422)

[Figure 2 Use case diagram 21](#_Toc191588423)

[Figure 3 User Registration Flow Chart 30](#_Toc191588424)

[Figure 4 Ticket Booking Flow Chart 31](#_Toc191588425)

[Figure 5 Ticket Cancellation Flow Chart 32](#_Toc191588426)

[Figure 6 Real-time Bus Tracking Flow Chart 33](#_Toc191588427)

[Figure 7 Admin Login Flow Chart 34](#_Toc191588428)

[Figure 8 Admin Managing bus schedules flow chart 35](#_Toc191588429)

[Figure 9 User Activity Diagram 36](#_Toc191588430)

[Figure 10 Admin Activity Diagram 37](#_Toc191588431)

[Figure 11 Sequence Diagram 38](#_Toc191588432)

[Figure 12 System Architecture Diagram 41](#_Toc191588433)

[Figure 13 Entity Relationship Diagram 44](#_Toc191588434)

[Figure 14 Relational Schema 46](#_Toc191588435)

[Figure 15 Class Diagram 47](#_Toc191588436)

[Figure 16 Login page wireframe 51](#_Toc191588437)

[Figure 17 Home Screen Wireframe 52](#_Toc191588438)

[Figure 18 Search Screen Wireframe 53](#_Toc191588439)

[Figure 19 Seat Selection Screen wireframe 1 54](#_Toc191588440)

[Figure 20 Seat Details screen 55](#_Toc191588441)

[Figure 21 Payment Screen 56](#_Toc191588442)

[Figure 22 Ticket Confirmation wireframe 57](#_Toc191588443)

[Figure 23 Settings Screen 58](#_Toc191588444)

List of Tables

[Table 1 SWOT Analysis 7](#_Toc191588445)

[Table 2 PEST Analysis 8](#_Toc191588446)

[Table 3 Risk Management 11](#_Toc191588447)

[Table 4 Methodology Phases 17](#_Toc191588448)

[Table 5 User Table 48](#_Toc191588449)

[Table 6 Ticket Table 48](#_Toc191588450)

[Table 7 Refund Table 49](#_Toc191588451)

[Table 8 Payment Table 49](#_Toc191588452)

[Table 9 Bus Schedule Table 49](#_Toc191588453)

[Table 10 Bus Table 50](#_Toc191588454)

[Table 11 Route Table 50](#_Toc191588455)

[Table 12 GPS table 50](#_Toc191588456)

**Glossary of Terms**

* GPS – Global Positioning System
* API – Application Program Interface

# Introduction & Literature Review

## 1.1 Introduction

Modern transportation is rapidly changing to digital modes, where technology will be the key to efficiency and ease. Mobitix, a mobile-based bus ticket reservation system which is designed especially for Sri Lanka, is an important solution in this area. The goal of this app is to help commuters in avoiding overcrowded buses, the lack of seats, and the inability to purchase or refund tickets on a real-time basis. Customers will be able to search for buses, choose seats, and pay online with Mobitix’s digital platform, which will save precious time and provide a better travel experience.

The need for a solution like Mobitix is evident in Sri Lanka’s busy public transport sector, where nearly 50% of the people depend on buses for daily commutes and intercity travel, according to. (The Department of Census and Statistics Sri Lanka, 2022). Traditional ticketing systems are mostly in a manual and paper-based format, which are considerably prone to inefficiencies and errors. Therefore, Mobitix wishes to fill this gap in service with features such as dynamic seat selection, secure online payment options, and digital tickets to meet the needs of both urban and rural commuters. The system integrates modern technology to make it accessible to anyone with a smart device, and it’s user-friendliness would cater to a growing audience’s demands.

## 1.2 Background and Motivation

Public transportation is part of the economic development process in any country, and Sri Lanka is no exception. The major issues concerning the bus transport sector in Sri Lanka are overcrowding, unmanaged arrival times, and inefficiency in manual ticketing systems. Many passengers are forced to waste much time in buses, especially during peak hours. Moreover, the absence of automation in ticketing develops human errors, and not having real-time information for passengers leads to problems in managing travel schedules.

Before the emergence of modern technologies, reservation procedures in buses were carried out manually, where several problems were encountered, such as wastage of time and frauds related to this. (Jayasuriya, 2021) discusses a few of them where human errors occurred in most cases with a manual system and was unable to meet the growing demand. The introduction of an automated system, such as that involving prepayment mechanisms, has been proposed as part of the solution to these issues. But the challenge remains in finding a solution that is both accessible and effective, especially in a country like Sri Lanka, where not all passengers are comfortable with using smart devices.

The importance of developing a completed automated bus ticket reservation and tracking app that includes real-time tracking, online payment systems, and user-friendly interfaces is emphasized by recent research, such as those conducted by (Vimukthi, 2023), Vimukthi discusses about how real-time tracking and GPS integration would make it possible to monitor the whereabouts of the buses and make travel decisions. This strategy may shorten wait times, improve user satisfaction greatly, and improve bus timetable administration.

While many web-based bus reservation systems have been proposed or are in use, these often fail to provide a complete solution. (Haleem, 2016) discusses the use of RFID and microcontroller-based systems for bus tracking, but web-based systems often lack essential features such as refund management, real-time updates and dynamic ticketing options. For example, when a passenger needs to cancel or modify a ticket, web-based systems frequently fail to offer a proper refund or adjustment process, leaving users frustrated. Moreover, many web systems provide incomplete or outdated information, particularly when it comes to real-time bus tracking. These drawbacks lead to dissatisfaction among the users and further hinder the potential of the system in serving passengers efficiently.

Considering the above-mentioned challenges, the mobile application was decided upon instead of the web-based system. The accessibility of a mobile app is higher compared to a web-based system, especially in a country like Sri Lanka, where smartphones are more in use among people than desktop computers. They also offer other advantages, like GPS-based location tracking, and offline functionality, which is quite necessary for a real-time and user- friendly experience. Also, the mobile application makes it easy to integrate the payment system, dynamic selection of seats, and on-screen instant generation of tickets, which are important features that would be difficult to implement in a traditional web-based platform.

This project will solve key issues in the present bus reservation system by developing a mobile application using Flutter at the frontend, PHP and sqflite at the backend, and GPS technology for real-time tracking. The application will have features like ticket booking, seat selection. Refund, trip history, and real-time tracking of the bus, thus maintaining all the travel details of the user directly from their smartphone.

## 1.3 Problem in brief

The main problem addressed by this project is the inefficiency of current bus ticket reservation systems in Sri Lanka, like refund management, seat selection, and real-time tracking. These issues lead to overcrowded busses, inefficient manual ticketing processes, insufficient management of travel schedules, and ultimately, passenger dissatisfaction.

Existing web-based platforms often fail to address these needs, such as offering complete, user-friendly platforms for online bookings, cancellations, and refunds. Additionally, many commuters in Sri Lanka rely on mobile devices rather than desktop systems, making accessibility a considerable challenge for web-based systems.

Therefore, there is a pressing need for a user-friendly solution to improve ticket booking, travel convenience and the overall efficiency of the public transportation system.

## 1.4 Aim & Objectives

### 1.4.1 Aim

To improve the efficiency of Sri Lanka's bus ticket reservation and management, addressing the issues of overcrowding, manual errors , and lack of real-time updates, for a better public transportation experience among passengers

### 1.4.2 Objectives.

* Conduct a critical review of the risks and inefficiencies in existing bus reservation systems
* Study the existing technologies and methods to address these challenges in detail
* Design and develop a mobile application that integrates real-time tracking, ticket booking, refund management, and smooth payment integration.
* Perform thorough testing to measure the performance and user satisfaction of the developed
* system by analyzing feedback.
* Prepare final documentation, such as technical and user documentation, to support the
* implementation and usability of the system.

## 1.5 Scope

The scope of Mobitix project is carefully defined to make sure that it addresses key challenges in Sri Lanka’s bus ticketing system while remaining feasible within the constraints of time, budget and resources. By focusing on the key functionalities and strengths like real-time tracking and secure payments, this app aims to deliver a reliable and user-friendly mobile application for public transportation. As such the scope can be further elaborated as follows.

**Features for Customers**

* Ticket Booking - Users can search for buses, select seats dynamically, and book tickets online.
* Real-Time Bus Tracking - Integration with Google Maps API to provide real-time bus location and estimated arrival times.
* Secure Online Payments - Support for local payment gateways like Genie, Koko, and eZcash for easy transactions.
* Refund Management - Users can request refunds for canceled tickets, with a transparent refund process.
* Trip History - Users can view their past trips and booking details.
* User Profiles - Users can create and manage profiles, including personal information and payment methods.

**Administrator Panel**

* Bus Schedule Management - Admins can add, update, or delete bus schedules.
* Reservation Management - Admins can view and manage ticket reservations.
* Performance Analytics - Admins can access data on bus performance, booking trends, and user feedback.

**Technical Implementation:**

* Frontend Development: The app will be developed using Flutter for cross-platform compatibility (Android and iOS).
* Backend Development: The backend will use PHP for RESTful API development and MySQL for database management.
* Integration with Third-Party Services: Integration with Google Maps API for real-time tracking and local payment gateways for secure transactions.

**Justification for the Scope**

The included features directly address the key issues in Sri Lanka’s bus ticketing system, such as overcrowding, lack of real-time updates, and inefficient refund management. These features are essential to achieving the project’s aim of improving public transportation efficiency and user satisfaction. The scope is designed to be achievable within the project timeline and available resources. Advanced features such as multi-language support and integration with other transport modes are excluded to ensure timely delivery of the core functionalities.

The scope aligns with the expectations of key stakeholders, including passengers, bus operators, and administrators. Features like real-time tracking and secure payments are prioritized based on stakeholder feedback and market research.

**Key Assumptions**

* It is assumed that smartphone penetration in Sri Lanka will continue to grow, making the app accessible to a larger user base.
* Local payment gateways like Genie, Koko, and eZcash will remain stable and widely used.
* The app will comply with data protection laws like GDPR and local regulations in Sri Lanka regarding user data privacy and refund policies.
* Third-party services like Google Maps API and payment gateways will function reliably without significant downtime or API changes.

**Excluded from Scope**

1. Integration with Other Transport Modes:

The app will focus solely on bus ticketing and will not include other modes of transport like trains or flights.

1. Offline Functionality:

While some features may work offline (e.g., viewing booked tickets), core functionalities like real-time tracking and online payments require an internet connection.

1. Custom Hardware Integration:

The app will not involve the development or integration of custom hardware (e.g., RFID scanners or ticketing kiosks).

1. Advanced Data Analytics:

While basic performance analytics will be provided, advanced data analytics or machine learning-based predictions are out of scope for this project.

1. Multi-Language Support:
2. The app will initially support English, with potential expansion to Tamil  and Sinhala in future updates.

**Constraints**

* The project must be completed within the academic timeline, limiting the scope to essential features.
* The project has limited funding, so expensive third-party services or custom hardware integrations are excluded
* The development team consists of a single student, which limits the complexity and scale of the project.

**Validation with Stakeholders**

* The scope has been discussed and agreed upon with the project head-supervisor, Mr. Vikum, ensuring alignment with academic and stakeholder expectations.
* Feedback from potential users (eg: commuters and bus operators) has been considered to prioritize features such as real-time tracking and secure payments.

**Flexibility for Adjustments**

* A structured change management process will be in place to handle any necessary adjustments to the scope. Changes will be documented and approved by the supervisor before implementation.
* The Agile methodology will allow for iterative developments. Therefore, changes can be made based on feedback and testing results.

1.5.1 SWOT ANALYSIS

Table SWOT Analysis

|  |  |
| --- | --- |
| * STRENGTHS | * WEAKNESSES |
| * User-Friendly interface (Flutter, dynamic seat selection, real-time tracking) * Real-Time Tracking (Google Maps API for accurate bus locations) * Secure Payments (Local gateways like Genie, Koko, eZcash). * Cross-platform (Works on Android & iOS) * Admin Panel (Efficient bus schedule and reservation management). | * Dependency on Third-party APIs (Google Maps, payment gateways). * Limited Offline Functionality (Internet required for key features). * Complex Integration (Challenges with local bus schedules and payment systems). * High development complexity (Dynamic seat selection, refund management) |
| * OPPORTUNITIES | * THREATS |
| * Growing Smartphone usage (Large user base in Sri Lanka) * Government/ private sector support (Collaboration for public transport efficiency) * Expansion to other transportation modes (Trains, intercity buses) * Data Analytics (Improve services using user and bus performance data). * Sustainability (Reduce traffic congestion, promote eco-friendly transport) | * Competition (Existing bus ticketing systems) * Resistance to change (Rural users may prefer traditional methods). * Cybersecurity Risks (Sensitive user data handing). * Regulatory challenges (Compliance with GDPR and local laws) * GPS Accuracy issues (Poor signal or environmental factors) |

1.5.2 PEST ANALYSIS

Table PEST Analysis

|  |  |
| --- | --- |
| Political Factors | * Government Support : Potential collaboration with Sri Lankan government to improve public transport efficiency. * Need to adhere to data protection laws (eg: GDPR) and local regulations. * Government initiatives to modernize public transport could benefit the app. |
| Economic Factors | * Growing Smartphone penetration : Increasing smartphones usage in Sri Lanka provides a larger user base * Local Payment Gateways : integration with local payment systems (eg: Genie, Koko, eZcash) support economic accessibility * Cost of Development : Budget constraints may limit advanced features or scalability. |
| Social Factors | * User adoption : Resistance to digital systems in rural areas may slow adoption. * Accessibility : The app must cater to diverse users, including those with disabilities. * Sustainability : Promotes eco-friendly public transport, aligning with societal trends toward sustainability. |
| Technological Factors | * Real-time tracking : Integration with Google Maps API for accurate bus tracking. * Cross-platform Development : Flutter ensures compatibility with both Android and iOS * Cybersecurity risks : Handling sensitive user data requires strong encryption and security measures. |

1.5.3 Project Timeline

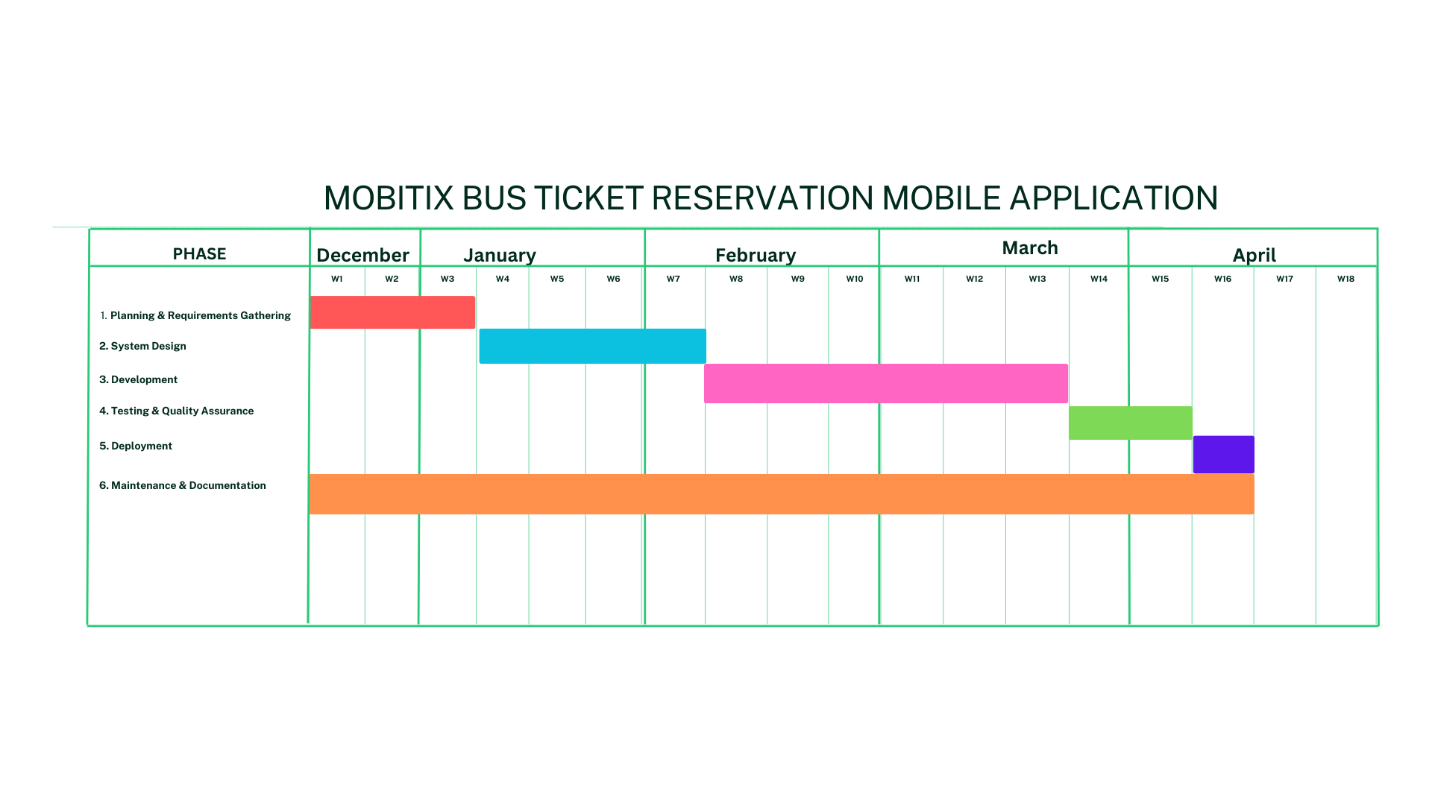


Figure Gantt Chart

1.5.4 Key Stakeholders

The success of the Mobitix Digital Bus Ticket Booking App depends on the involvement and collaboration of several key stakeholders. These stakeholders play important roles in the development, implementation, and adoption of the app. The key stakeholders can be denoted as follows;

*1. Passengers (End Users)*

Passengers are the primary users of the app. They rely on the app to book tickets, track buses in real-time and manage their travel schedules. Passengers seek a convenient, reliable, and user-friendly platform that simplifies the bus ticketing process and enhances their overall travel experience. Their feedback and adoption rates are important for the app’s success and continuous improvement.

*2. Bus operators and Administrators.*

Bus operators and administrators manage bus schedules, reservations, and performance analytics through the app’s admin panel. They require an efficient system to conduct operations, reduce manual errors, and improve bus utilization. Their cooperation is essential for integrating the app with existing bus systems and ensuring accurate real-time data.

*3. Government and Regulatory Bodies*

Government agencies and regulatory bodies oversee public transportation systems to ensure compliance with local laws and regulations. They aim to modernize public transport, reduce traffic congestion, and promote sustainable travel options. Their support can help the app’s adoption and integration into the broader public transport infrastructure.

*4. Payment Gateway Providers*

Providers like Genie, Koko, eZcash provide secure online payments within the app. They seek to expand their user base and provide seamless integration with the app. Their reliability and performance directly affect the app’s payment functionality and user trust.

*5. The Developer.*

The development team (the author, in this case) is responsible for designing, building, and maintaining the app. The developer should aim to deliver a high-quality, functional app within the project timeline and budget. The developer’s expertise and efficiency will determine the app’s technical performance and ability to meet user needs.

*6. Supervisors and Academic Advisors*

Supervisors and academic advisors provide guidance, feedback, and oversight throughout the project. They ensure the project aligns with academic standards and achieves its objectives. Their input helps refine the app’s design and functionality, ensuring it meets both user and academic requirements.

1.5.5 Risk Management

Table Risk Management

|  |  |
| --- | --- |
| Risk | Mitigation method |
| Delay in development due to unforeseen issues | * Extend sprint duration or reduce non-critical features to prioritize delivery * Conduct regular risk assessments during sprint planning to identify potential delays early. * Include buffer time in the project schedule to accommodate unexpected delays. |
| Payment gateway Integration challenges | * Switch to a reliable alternative gateway or provide a manual payment option * Test payment gateway integrations early in the development phase to identify and resolve issues. * Integrate multiple payment gateways as backups to ensure uninterrupted service. |
| GPS/ Google Maps API integration issues | * Use alternative mapping services or temporarily disable GPS feature * Continuously monitor API performance and availability. |
| Performance issues during testing | * Improve code and conduct performance testing on multiple devices * Optimize code and database queries for better performance. |
| Data Security or privacy concerns | * Implement encryption, secure storage. And follow data protection best practices * Conduct regular security audits and vulnerability assessments. * Anonymize user data used for testing to protect privacy. |
| Inaccurate GPS real-time tracking | * Use alternative APIs or use data requests to reduce errors * Implement data validation to ensure GPS data accuracy. * Allow users to report inaccurate tracking for quick resolution. |
| User dissatisfaction with UI/UX design | * Conduct feedback sessions to refine the user interface * Conduct usability testing with real users during the design phase. |
| Unexpected bugs in seat selection and reservation | * Increase QA focus on code improvements * Conduct regular code reviews to catch bugs early. |
| Insufficient time for deployment | * Plan app store submission in earlier sprints and address compliance requirements early * Create a checklist to ensure all app store requirements are met. |
| Integration issues with the Bus Schedule data | Use dummy datasets for testing purposes |

## 1.6 Deliverables

Upon the completion of the Mobitix Digital Bus Ticket Booking App project, the following deliverables will be provided.

**Technical deliverables**

* System Architecture – Detailed documentation explaining the app’s structure, including frontend (Flutter), backend (PHP), and database (MySQL) components
* Database Schema – a well-defined schema for storing user profiles, ticket bookings, and trip history.
* API integration Guides – Documentation for integrating Google Maps API and local payment gateways (Genie, Koko, eZcash).
* User Manuals – Guides for both passengers and administrators to operate the app effectively.

**Functional Deliverables**

* Fully Functional Mobile App – A cross-platform app (Android and iOS) with features like ticket booking, real-time bus tracking, secure payments, refund management, and trip history.
* Admin Panel – a backend system for managing bus schedules, reservations, and performance analytics.
* Testing reports – comprehensive test cases, bug fixes, and user acceptance testing (UAT) results.

## 1.7 Literature Review

The evolution of bus ticket reservation systems from manual to digital platforms has significantly transformed public transportation globally, particularly in developing regions like Sri Lanka. Traditional manual ticketing systems have long been plagued by inefficiencies, including long queues, human errors, and limited accessibility. (Jayasuriya, 2021) highlights that manual processes in Sri Lanka led to overcrowding, poor refund management, and passenger dissatisfaction, underscoring the need for automation. Similarly, (Oloyede, 2014) note that offline systems in Nigeria restricted customers’ ability to compare bus operators or routes, while operators struggled with seat inventory management. Early web-based systems introduced functionalities like online seat selection, payment integration, and report generation, but they often lacked real-time updates and dynamic features such as refund management (Vimukthi, 2023). Mobile applications, such as the proposed Mobitix app, have emerged as a superior alternative due to their accessibility, GPS integration, and offline capabilities. For instance, GPS-based tracking systems in Malaysia, like Katsana and Debezt, enabled real-time bus location updates and automated reporting, significantly improving operational efficiency (Jakubauskas, March 2014)

Secure user authentication is a critical component of digital ticketing systems, ensuring the protection of sensitive data. Systems like Mobitix employ multi-factor authentication (MFA) and role-based access control (RBAC) to prevent unauthorized access. Registration typically requires email or phone verification, which ensures accountability and reduces fraud (Jeewanthi Fernando, September 2016) Dynamic seat selection is another essential feature, allowing users to view real-time seat availability through interactive layouts. For example, BusSeat.lk in Sri Lanka uses color-coded seat maps to indicate booked, available, and reserved seats, reducing overbooking and enhancing transparency (Techkitez, 2015). Payment integration is also crucial, with local payment gateways like Genie, Koko, and eZcash being prioritized in regions like Sri Lanka to align with user preferences and ensure trust. Secure encryption protocols, such as SSL/TLS, protect transaction data, and failed payments trigger instant notifications, allowing users to retry or cancel bookings (Dr. Swapna Ubale, 2024)

Real-time tracking and notifications are vital for improving the user experience. Integration with Google Maps API enables live bus tracking, estimated arrival times, and route optimization, as seen in the Mobitix project (Vimukthi, 2023). Notifications via Firebase Cloud Messaging (FCM) alert users about delays, departures, and refund statuses, further enhancing the system's efficiency (Ubale et al., 2024). In Iraq, mobile bus ticketing systems (MBTS) reduced passenger wait times by 40% through real-time updates, demonstrating the potential impact of such features Refund management is another critical aspect, with automated refund policies streamlining dispute resolution. Admins review requests via dashboards, and approved refunds are processed through integrated gateways. However, inconsistent internet connectivity in rural areas remains a barrier, as highlighted by Jayasuriya (2021).

The technological frameworks and tools used in digital ticketing systems are crucial for their success. Cross-platform frameworks like Flutter dominate mobile app development due to their compatibility with Android and iOS, making them an ideal choice for Mobitix. Flutter’s prebuilt widgets and hot-reload features accelerate UI development, ensuring responsive interfaces (Ubale et al., 2024). PHP and MySQL are widely adopted for backend development due to their scalability and cost-effectiveness. RESTful APIs facilitate communication between frontend and backend modules, while MySQL manages structured data like user profiles and ticket histories (Techkitez, 2015). Third-party integrations, such as Google Maps API for GPS tracking and local payment gateways like Genie, ensure localized transaction support and enhance operational efficiency (Vimukthi, 2023). Firebase handles notifications and user analytics, further improving the system's functionality (Ubale et al., 2024).

Despite the advantages, digital ticketing systems face several challenges. Dependency on third-party APIs, such as Google Maps, poses risks like service interruptions. Mitigation strategies include backup mapping services and manual location updates (Fernando, et al, 2016.). Cybersecurity risks, including data breaches and fraud, are mitigated through encryption, GDPR compliance, and regular security audits. Anonymizing test data further protects user privacy (Vimukthi, 2023). Resistance to digital adoption, particularly in rural areas, is another challenge. Mobitix addresses this through user education campaigns and incentives like discounted first (Fernando, et al, 2016.).Infrastructure limitations, such as poor internet connectivity in rural areas, restrict real-time features. Offline functionalities, such as cached ticket viewing, partially address this gap (Jayasuriya, 2021).

Case studies from various regions demonstrate the scalability and impact of digital ticketing systems. BusSeat.lk, launched in 2015, digitized Sri Lanka’s bus network, serving over 100,000 users. Key features include route maps, seat selection, and multi-operator integration. User feedback highlighted improved efficiency and reduced overcrowding (Techkitez, 2015). In Malaysia, Katsana’s GPS tracking and Debezt’s real-time alerts reduced passenger wait times by 30%, providing operators with performance analytics to optimize schedules (Jakubauskas, 2010). The eBus Services System (EBS), developed using Agile methodology, unified web and mobile platforms for ticket booking. Technologies like Bootstrap and AJAX ensured responsiveness, while PHP-MySQL handled backend operations (Ubale et al., 2024).

Looking ahead, the future of digital ticketing systems lies in expansion to multi-modal transport, AI-driven analytics, and sustainability initiatives. Integrating trains and flights into platforms like Mobitix could create a unified travel ecosystem, enhancing user convenience (Vimukthi, 2023). Machine learning can predict peak travel times, optimize routes, and personalize user experiences, further improving operational efficiency (Ubale et al., 2024). Additionally, digital ticketing reduces paper waste, aligning with global sustainability goals and promoting eco-friendly public transportation (Mezghani)

In conclusion, digital bus ticketing systems address critical inefficiencies in manual processes, offering real-time tracking, secure payments, and enhanced user experiences. Case studies from Sri Lanka, Malaysia, and Nigeria demonstrate their scalability and impact. However, challenges like API dependency and rural adoption require ongoing innovation. Future systems must prioritize AI-driven analytics, multi-modal integration, and robust offline functionalities to achieve universal accessibility. The Mobitix project, with its focus on real-time tracking, secure payments, and user-friendly design, is well-positioned to contribute to this evolution, addressing the specific needs of Sri Lanka’s public transportation system while setting a precedent for future developments in the region

1.8 Methodology

The development of the Mobitix Digital Bus Ticket Booking App follows a structured and iterative approach to ensure the project meets its objectives efficiently.

**1. Development Approach**

The project adopts the Agile Methodology, which is well-suited for software development due to its iterative and flexible nature. Agile allows for continuous feedback from stakeholders, so that quick adjustments and improvements can be done easily throughout the development process. The project is divided into sprints, each focusing on specific features such as ticket booking, real-time tracking, and payment integration. This approach makes sure that the app evolves incrementally, with each sprint delivering a functional version of the system.

**2. Technology Stack**

The technology stack has been carefully selected to align with the project’s objectives and to make the app more user friendly and scalable.

* Frontend – Flutter is sued for cross-platform development, so that the app can be run smoothly on both Android and iOS. Flutter’s rich set of prebuilt widgets and hot reload feature accelerate development and ensure a modern, responsive user interface.
* Backend – PHP is used to build a RESTful API for handling user authentication, ticket reservations, and trip data. MySQL is chosen as the database management system for its reliability and ability to store structured data efficiently.
* Real-time tracking – Google Maps API is integrating for real-time bus tracking, providing accurate location updates and route planning.
* Payment Integration – Local payment gateways like Genie, Koko, and eZcash are supported to ensure secure and convenient transactions for users in Sri Lanka.

**3. System Architecture**

The system follows a client-server architecture, where the mobile app interacts with the backend server via RESTful APIs. Key components include,

* Frontend – the mobile app built with Flutter, providing the user interface for ticket booking, seat selection, and real-time tracking.
* Backend – a PHP-based server handling business logic, user authentication, and data storage in MySQL.
* Third-Party Integrations – Google Maps API for real-time tracking and local payment gateways for secure transactions.

**4. Data Collection.**

To gather user requirements and system data, surveys, stakeholder meetings and system logs were used to ensure that the app is tailored to meet the needs of both passengers and administrators. Surveys were conducted with potential users to understand their needs and pain points with current bus ticketing systems. Stakeholder meetings were conducted for regular discussions with bus operators and administrators and Academic supervisors to gather insights into operational requirements. System logs are used to gather data from live bus schedules and GPS tracking systems are collected to ensure accurate real-time updates.

**5. Implementation**

The implementation process is divided into the following phases.

Table Methodology Phases

|  |  |
| --- | --- |
| **Phases** | **Activities** |
| Planning & Requirements Gathering | * Define project scope * Gather user requirements * Create a sprint backlog |
| System Design | * Develop wireframes * System architecture diagrams * Database schemas |
| Development | * Implement features such as ticket booking, real-time tracking, and payment integration in iterative sprints |
| Testing | * Conduct unit, integration, and user acceptance testing to ensure the app works as expected |
| Deployment | * Deploy the app to the Google Play Store and Apple App Store |
| Maintenance | * Monitor App performance, gather user feedback, and release updates to improve functionality |

**6. Testing methods**

* To ensure the app’s reliability and performance, the following testing methods are employed:
* Unit Testing : Individuals components (eg: ticket booking, payment processing) are tested for functionality
* Integration Testing : Ensures that different modules (eg: Frontend, backend, APIs ) work together seamlessly.
* Security Testing : Verifies that user data is encrypted and secure, with measures like multi-factor authentication (MFA) in place
* User Acceptance Testing : Conducted with a sample group of users to validate the app’s usability and functionality.

**7. Security & Compliance**

Security is a top priority for Mobitix. The following the measures are implemented.

* Data encryption – Sensitive user data, such as payment details and personal information, is encryption during storage and transmission
* Authentication – Multi-factor authentication is used to prevent unauthorized access.
* Compliance – The app adheres to data protection regulations such as GDPR and local laws in Sri Lanka, ensuring user privacy and data security

**8. Limitations.**

The project acknowledges certain limitations, including

* Time Constraints – the project must be completed within the academic timeline, limiting the scope of features.
* Budget Constraints – Limited funding restricts the use of expensive third-party tools or services.
* Hardware Dependencies – The app’s performance may be affected by the quality of users devices and internet connectivity. Especially in rural areas

# 2. Analysis

**Problem Definition**

The core problem addressed by this project is the inefficiency of Sri Lanka’s current bus ticket reservation systems, characterized by the lack of real-time tracking, poor refund management, and overcrowded buses due to manual processes. These issues lead to passenger dissatisfaction, unreliable schedules, and operational challenges for bus operators, hindering the modernization of public transportation.

2.1 Analysis Techniques

2.1.1 SWOT ANALYSIS

The Mobitix Digital Bus Ticket Booking App project represents a great opportunity to improve Sri Lanka’s public transportation system, but it also faces several challenges that needs to be addressed. As such, the Strengths, Weaknesses, Opportunities, and Threats (SWOT) of the project can be analyzed as follows.

**Strengths**

Mobitix has several internal strengths that positions it as a practical solution for modernizing bus ticketing in Sri Lanka. The app’s user-friendly interface, developed using flutter, will provide a smooth experience for passengers, with features like dynamic seat selection and real-time bus tracking improving convenience. The integration of Google Maps API provides accurate real-time tracking, keeping passengers informed about bus locations and estimated arrival times. Moreover, the app supports secure online payments through local gateways like genie, Koko and eZcash, which are widely used in Sri Lanka, making the app more accessible and trustworthy. The cross-platform compatibility of the app will further broaden it’s reach, while the inclusion of an admin panel will allow for efficient management of bus schedules, reservations, and performance analytics, improving operational efficiency for bus operators.

**Weaknesses**

Despite its strengths, Mobitix also has internal weaknesses. The application’s high reliance on third-party APIs such as Google Maps and Payment Gateways risks interruption of services or integration issues. Additionally, the app’s limited offline functionality could undermine usability in areas with poor internet coverage, an issue in rural Sri Lanka. The advanced integration with local bus schedules and payment systems can also be technically demanding. Especially if data formats or APIs are inadequately documented. Moreover, the high level of development complexity for functionalities like dynamic seat selection and refund management can lead to delays or increased expense in development process.

**Opportunities**

The outside world also provides numerous opportunities for Mobitix. Growing smartphone penetration in Sri Lanka provides an enormous and increasing user base, particularly among commuters in urban and rural areas. Collaboration with the government or the private sector may further proper the adoption and use of the app in the public transport system. expansion potential into other modes of transport, such as trains or inter-city buses, which would increase the utility and market size of the app. Leverage of data analytics to analyze user behavior and bus performance would open opportunities for improved service and optimized scheduling. Moreover, the application also fosters global trends toward sustainability as it promotes environmentally responsible public transportation and reduces congestion jams, which would be enhanced by green stakeholders.

**Threats**

However, the project is not exempt from external challenges. Competition from existing bus ticketing systems both offline and online will hinder Mobitix’s penetration into the market. Resistance to change among users, particularly in rural areas where traditional ticketing is more common, can hinder adoption. Management of sensitive user data, such as personal information and payments, exposes the app to cybersecurity risks, which can damage user trust if poorly managed. Regulatory compliance in the form of GDPR and domestic data protection regulations adds an added layer of sophistication. Finally, GPS accuracy issues, caused by poor signal intensity or environmental conditions, can lead to user disappointment if real-time tracking is unpredictable.

Accordingly, Mobitix has great potential to address Sri Lanka’s inefficient bus ticketing system because it’s easy to use, real-time, and provides secure payment. However, the project must overcome challenges such as third-party API reliance, limited offline usage, and resistance to change. By utilizing opportunities like rising smartphone penetration, government subsidies, and data analytics, and countering threats like cyber attacks and regulatory penalties, Mobitix can establish a stable and efficient public transport system in Sri Lanka.

2.1.2 Use Case Diagram

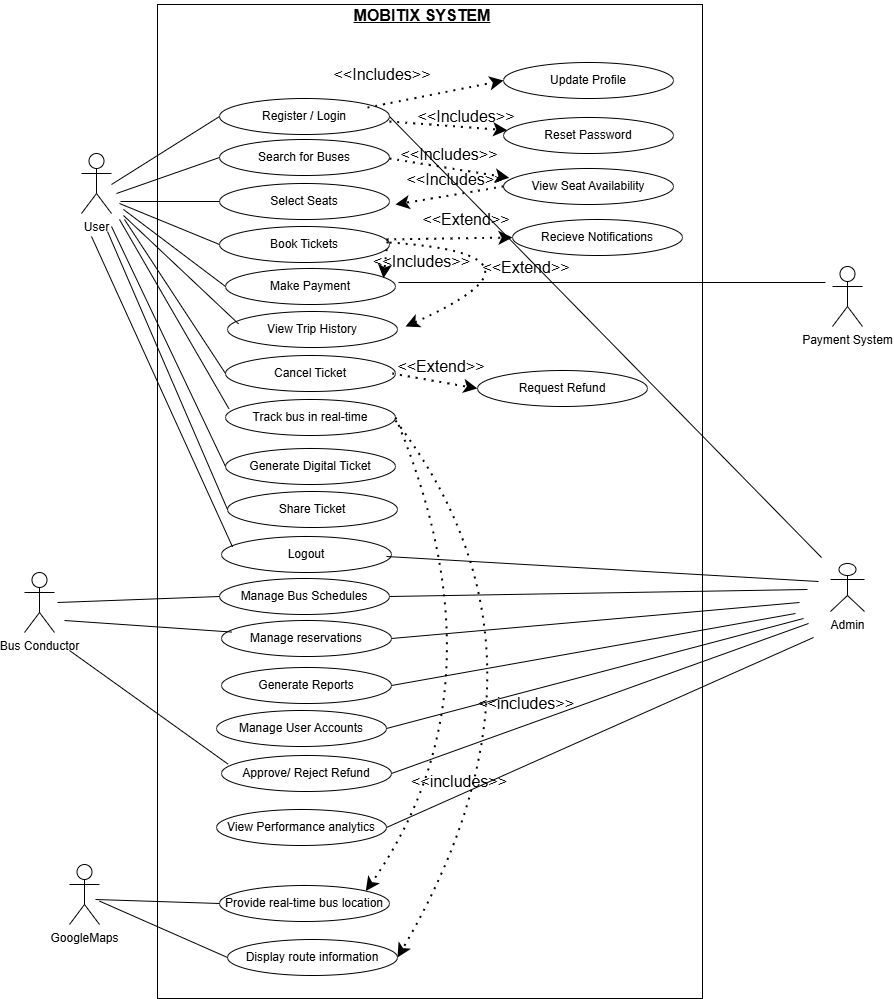


Figure 2 Use case diagram

The use case diagram for the Mobitix Digital Bus Ticket Booking App illustrates the interactions between different users (actors) and the system's functionalities. The primary actors include Users (Passengers), Admin, Bus Conductors, Payment System, and Google Maps.

User Interactions

Passengers (Users) interact with the system by performing essential tasks such as registering/logging in, searching for buses, selecting seats, booking tickets, making payments, viewing trip history, and canceling tickets. These actions are core to the ticket booking experience. Some use cases, such as viewing seat availability and receiving notifications, are included within the main booking process to enhance the user experience. Additional features like tracking the bus in real-time, generating digital tickets, and sharing tickets offer improved convenience.

The Admin has extended control over the system, managing bus schedules, reservations, and user accounts. They can also generate reports, approve/reject refunds, and view performance analytics to oversee operations effectively. The Bus Conductor is responsible for tasks such as managing bus schedules and reservations, ensuring smooth operations.

The Payment System is linked to processing payments and handling refund requests, ensuring financial transactions are integrated seamlessly. Additionally, Google Maps provides real-time bus tracking and displays route information, allowing users to stay updated on their bus’s status.

Some actions, such as resetting passwords, updating profiles, and viewing seat availability, are included as additional functionalities that support the primary use cases. Others, like receiving notifications and requesting refunds, extend from main use cases, indicating they are optional but crucial enhancements.

2.1.3 User Stories

Passenger User Stories

**1. Ticket Booking & Reservation**

**User Story:**  
*"As a passenger, I want to search for available buses by date, time, and route so that I can find a suitable option for my trip."*

**Acceptance Criteria:**

* The user can input travel date, time, and destination to search for buses.
* The system displays a list of available buses with details (departure time, duration, seat availability).
* Users can filter/search buses by operator, price, or seat availability.

**2. Dynamic Seat Selection**

**User Story:**  
*"As a passenger, I want to select my preferred seat from a bus seating layout so that I can choose a comfortable seat for my journey."*

**Acceptance Criteria:**

* The system displays a real-time seating layout for the selected bus.
* Users can view available, booked, and reserved seats.
* Once a seat is selected, it is temporarily locked until payment is completed.

**3. Online Payment Processing**

**User Story:**  
*"As a passenger, I want to make an online payment for my ticket so that I can confirm my booking instantly."*

**Acceptance Criteria:**

* Users can choose from multiple payment options (Genie, Koko, eZ Cash).
* The system securely processes payments and confirms the transaction.
* A payment receipt and e-ticket are generated upon successful payment.
* If payment fails, the system provides an appropriate error message.

**4. Real-time Bus Tracking**

**User Story:**  
*"As a passenger, I want to track my bus in real-time so that I can estimate its arrival time and plan accordingly."*

**Acceptance Criteria:**

* Users can access a live map showing the real-time location of the booked bus.
* The system provides estimated arrival times at the boarding point.
* Notifications are sent in case of delays or changes in the schedule.

**5. Trip History & Refund Requests**

**User Story:**  
*"As a passenger, I want to view my past and upcoming trips so that I can manage my bookings and request refunds if necessary."*

**Acceptance Criteria:**

* Users can view a list of past and upcoming trips.
* The system allows users to request a refund for eligible bookings.
* Refund status updates are sent via notifications.

Admin User Stories

**6. Bus Schedule Management**

**User Story:**  
*"As an administrator, I want to add, edit, and remove bus schedules so that I can keep the system up to date with current routes and timings."*

**Acceptance Criteria:**

* Admins can add new bus schedules with details (route, departure time, stops).
* Admins can update schedules in case of changes.
* Deleted schedules no longer appear in passenger searches.

**7. Reservation & Ticketing Management**

**User Story:**  
*"As an administrator, I want to view and manage ticket reservations so that I can handle customer inquiries and refunds efficiently."*

**Acceptance Criteria:**

* Admins can search and filter reservations by user, date, or bus route.
* Admins can manually cancel or modify bookings when necessary.
* Refund requests are reviewed and approved based on policy rules.

Security and Compliance user stories

**8. Secure Login & Authentication**

**User Story:**  
*"As a user, I want to log in securely so that my personal and payment information is protected."*

**Acceptance Criteria:**

* Users can log in using email/phone number and password.
* Multi-factor authentication (MFA) is available for enhanced security.
* The system enforces secure password policies.

**9. Data Privacy & User Access Control**

**User Story:**  
*"As a user, I want to control who can access my personal and booking information so that I can ensure my privacy and security."*

**Acceptance Criteria:**

* Users can modify account privacy settings.
* The system enforces permissions for user data access

2.1.4 Requirements Engineering

**Functional Requirements**

**User Requirements (Passengers)**

***User registration and Authentication***

 Users must be able to sign up using their email or phone number.

 The system must enforce email/phone number verification before account activation.

 Secure login/logout must be implemented using hashed passwords and multi-factor authentication (MFA).

 Users should be able to reset their password via email or OTP-based authentication.

 The system should allow users to update profile details (name, contact number, payment preferences).

***Ticket Booking and Reservation***

 Users must be able to search for available buses by entering the departure and destination locations along with the date and time.

 The system should provide a list of available buses with details such as route, seat availability, price, and departure time.

 Users should be able to select specific seats dynamically via an interactive seat map.

 The system must allow users to complete their ticket purchases via integrated local payment gateways (Genie, Koko, eZ Cash).

 Upon successful payment, users should receive an electronic ticket (e-ticket) in the app and via email.

 The system must support QR code-based tickets for verification during boarding.

 Users should be able to share e-tickets with others via email or messaging apps.

***Real-Time Bus Tracking***

 The system should integrate with Google Maps API to track buses in real time.

 Users must be able to view the live location of their booked bus and estimated time of arrival (ETA).

 Notifications should be sent for key updates, such as bus departure, delays, and arrival alerts.

 Users should have the ability to set alerts for their stop to receive notifications before arrival.

***Trip Management***

 Users must be able to view their past trips, including date, time, bus details, and fares paid.

 The system should allow users to cancel their booking before departure.

 Refund requests must be processed based on predefined policies (e.g., full refund if canceled within 24 hours, partial refund after that).

 Users should be able to provide feedback and rate their travel experience.

**Payment Integration**

* Users must be able to securely make online payments using supported local payment gateways (Genie, Koko, eZ Cash).

**Administrator Requirements**

***Bus Management***

 Administrators must be able to add, edit, and remove buses and schedules.

 The system should allow assigning drivers and conductors to specific buses.

 Bus route details must be customizable, including stops, departure times, and fares.

***Bus Reservation and Ticket Management***

 Administrators should have access to view all active and past bookings.

 The system must provide an interface for approving or rejecting refund requests.

 Administrators should be able to modify bus schedules in case of route changes or cancellations.

***Performance Analytics***

 The system should generate automated reports on ticket sales, peak travel times, and revenue trends.

 Analytics should track user behavior, including frequently booked routes and customer preferences.

 The dashboard should highlight key performance indicators (KPIs) such as average booking time, customer ratings, and refund rates.

***Non-Functional Requirements***

***Performance Requirements***

 The system must handle at least 1000+ concurrent users without degradation in performance.

 Ticket booking transactions must be processed within 3 seconds on average.

 The real-time bus tracking feature should update location data at least every 10 seconds.

***Security Requirements***

 All user data must be protected in compliance with GDPR and CCPA regulations.

 Payment transactions must be encrypted.

 User authentication must support multi-factor authentication (MFA).

 Role-based access control (RBAC) must be enforced for administrators and passengers.

***Usability and Accessibility***

 The user interface must be intuitive and easy to navigate, with clear CTAs (Call-to-Actions).

 The app should support Sinhala, Tamil, and English languages for broader accessibility.

 The application must be fully responsive and compatible with both Android and iOS devices.

 The app must support screen reader compatibility for visually impaired users.

***Constraints and Assumptions***

 Internet access is required for ticket booking, seat selection, and bus tracking.

 The payment gateways (Genie, Koko, eZ Cash) must provide API documentation for integration.

 Bus operators must update real-time GPS tracking information for accurate live location updates.

 Refund policies must be clearly defined and agreed upon by both users and administrators.

 The mobile app should be optimized for low-bandwidth connections to accommodate users with limited internet access.

2.2 Analysis Outcomes

The analysis of the **Mobitix Digital Bus Ticket Booking App** reveals several key outcomes, providing valuable insights into the project’s feasibility, challenges, and strategic opportunities. These outcomes are derived from the **SWOT analysis, user needs,** and **requirements engineering,** highlighting critical aspects that will shape the development and implementation of the system.

***Key Outcomes***

1. Addressing the inefficiencies in the current system

* The analysis confirms that manual ticketing processes, lack of real-time tracking, and inefficient refund management are major pain points in Sri Lanka’s public transportation system.
* Mobitix aims to improve bus ticket reservations by offering digital booking, dynamic seat selection, and automated refund management, reducing passenger inconvenience.

2. Clear Definition of a System Capabilities

* Functional requirements have been established to include secure user authentication, online ticket purchases, e-ticket generation, and real-time bus tracking.
* The integration of Google Maps API for live tracking and local payment gateways (Genie, Koko, eZ Cash) ensures a user-friendly and reliable experience for passengers.

3. Identification of Non-Functional Requirements

 The system must meet critical performance, security, scalability, and usability requirements.

* Key performance goals include handling 1000+ concurrent users and ensuring that ticket booking transactions are processed within 3 seconds.
* Security requirements emphasize GDPR compliance, data encryption, and multi-factor authentication to safeguard sensitive user information.

4. Technological Feasibility and Potential Challenges

* The choice of technologies, including Flutter (for cross-platform development), PHP (backend), and Google Maps API (for real-time tracking),confirms the technical feasibility of the project.
* However, challenges such as integration issues with third-party APIs (Google Maps, local payment gateways) and ensuring seamless real-time tracking accuracy need to be addressed.

5. Market Readiness and Adoption Challenges

* The growing smartphone penetration in Sri Lanka presents a significant opportunity for app adoption.
* However, resistance to change, especially among rural commuters accustomed to traditional ticketing methods, may slow initial adoption. A user education campaign and incentives (e.g., discounts on first bookings) could mitigate this challenge.

6. Competitive and Regulatory Considerations

* Competition from existing ticketing systems (offline and online) may impact market penetration. Strategic partnerships with bus operators and government transport agencies will be crucial for widespread adoption.
* Compliance with data protection regulations (GDPR, local data privacy laws) must be strictly maintained to avoid legal risks and ensure user trust.

7. opportunities for future enhancements

* Beyond bus ticketing, the platform has the potential to expand into train and inter-city transport reservations, broadening its market reach.
* Data analytics could be leveraged to analyze user behavior, predict peak travel times, and optimize bus schedules, improving both user experience and operational efficiency for bus operators.
* Integrating AI-based recommendations for route planning and bus schedules could further enhance the app’s value proposition.

8. Strategic planning for Implementation

* The Agile development approach will be used to iteratively refine features, incorporate user feedback, and address emerging challenges.
* Initial deployment will target urban areas with strong internet connectivity and smartphone adoption, followed by expansion to rural regions with necessary adaptations (e.g., offline booking support).

# 3. Design

## 3.1 Design Techniques

3.1.1. Flow Charts

User Registration Flow Chart

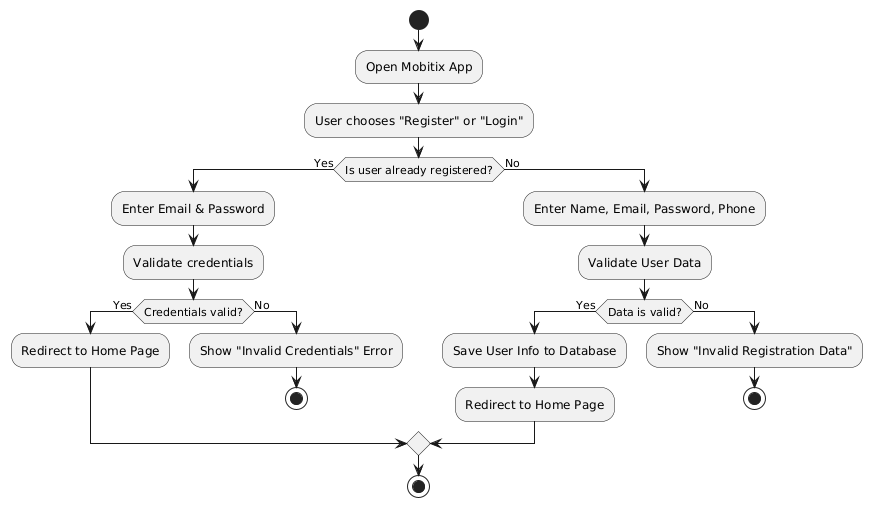


Figure User Registration Flow Chart

Ticket Booking Flow Chart

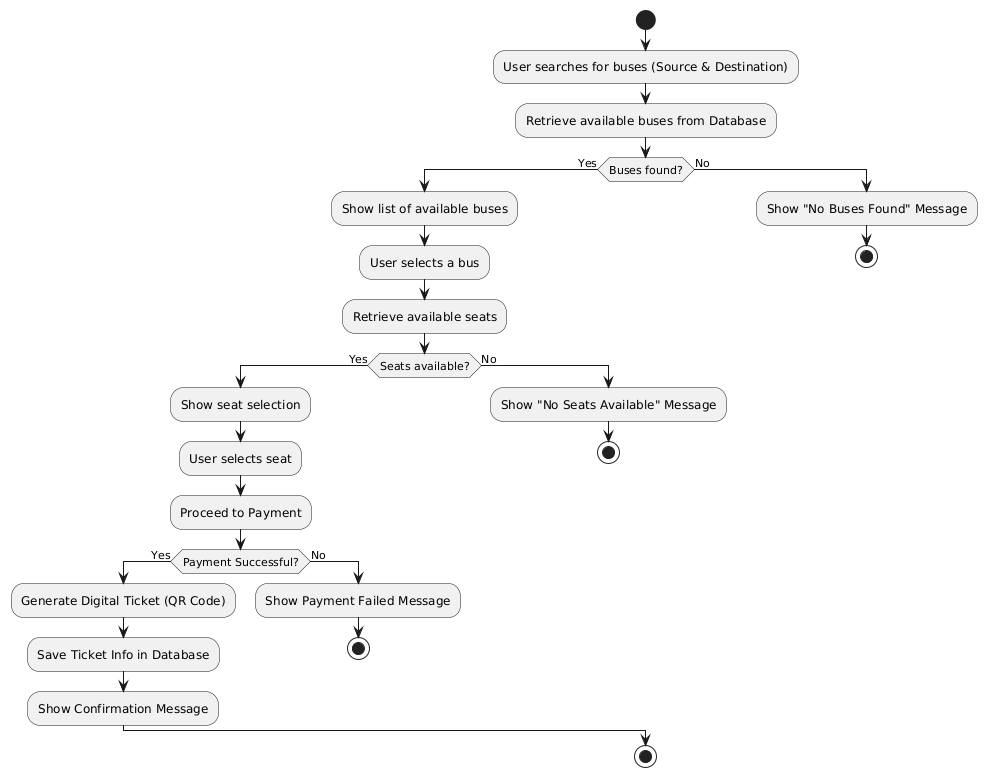


Figure Ticket Booking Flow Chart

Ticket Cancellation Flow Chart

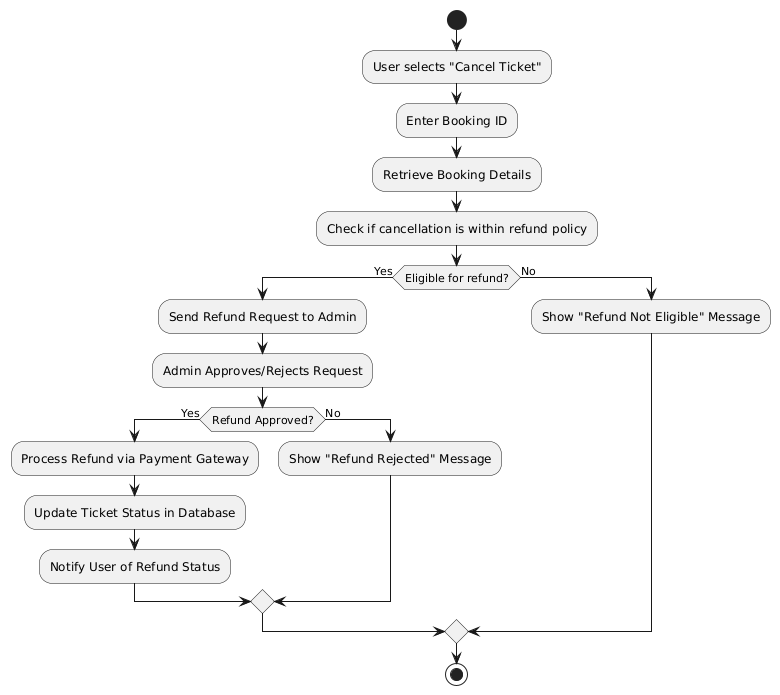


Figure Ticket Cancellation Flow Chart

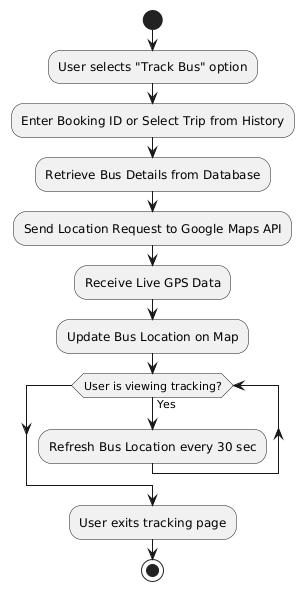


Figure Real-time Bus Tracking Flow Chart

Admin Login Flow Chart

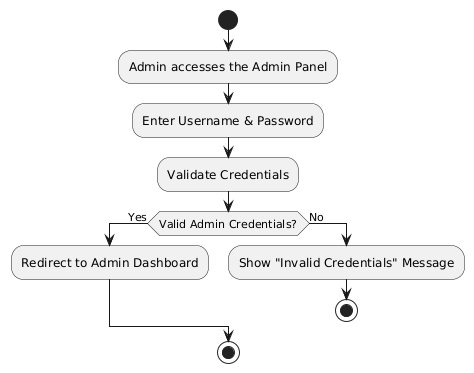


Figure Admin Login Flow Chart

Admin Manage Bus Schedule Flow Chart

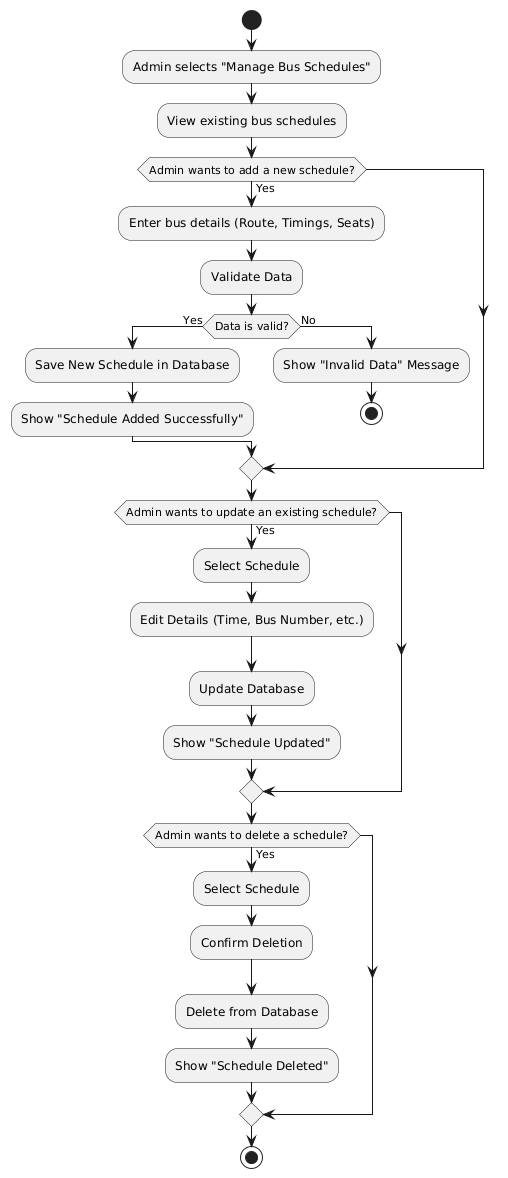


Figure Admin Managing bus schedules flow chart

3.1.2 Activity Diagrams

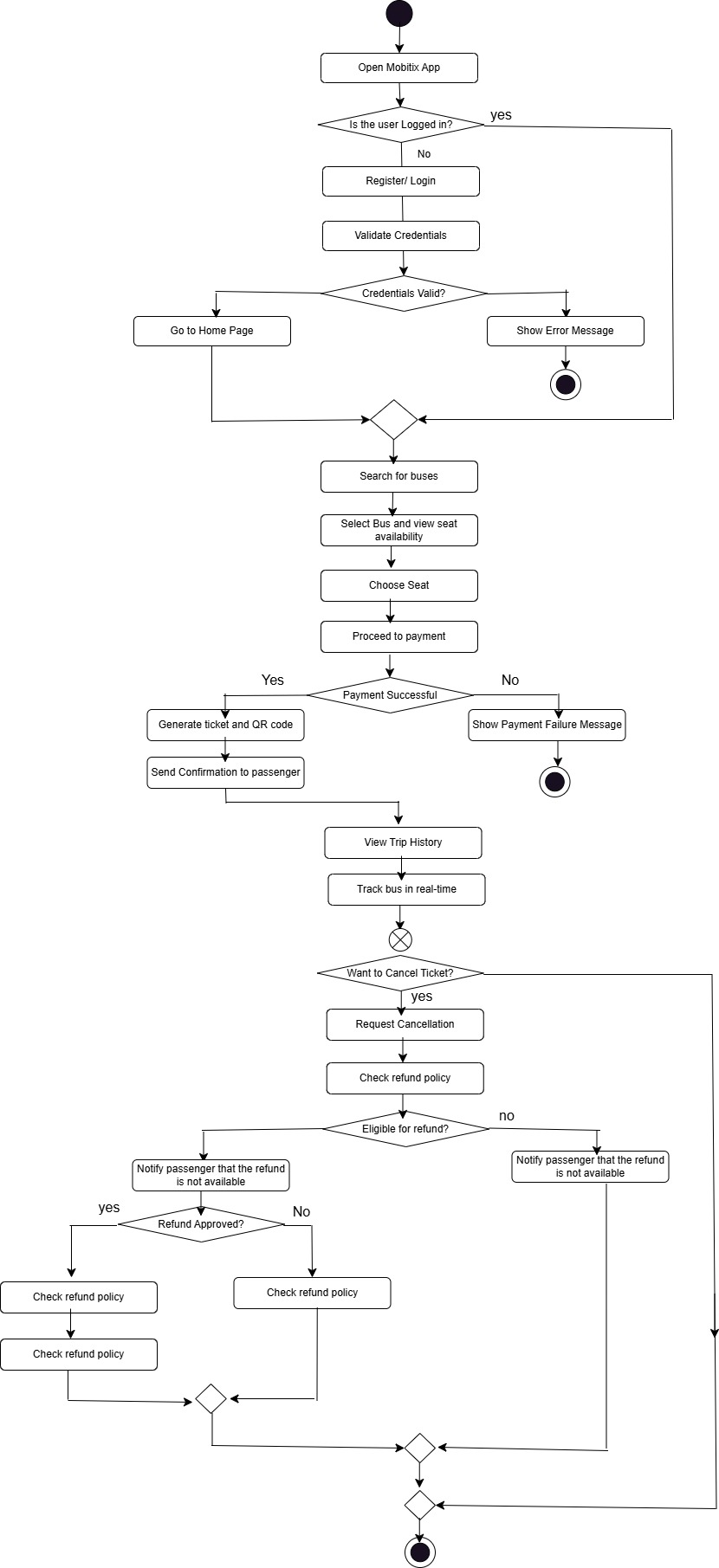


Figure User Activity Diagram

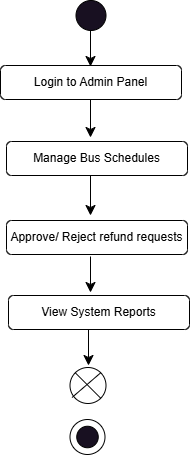


Figure Admin Activity Diagram

3.1.3 Sequence Diagram

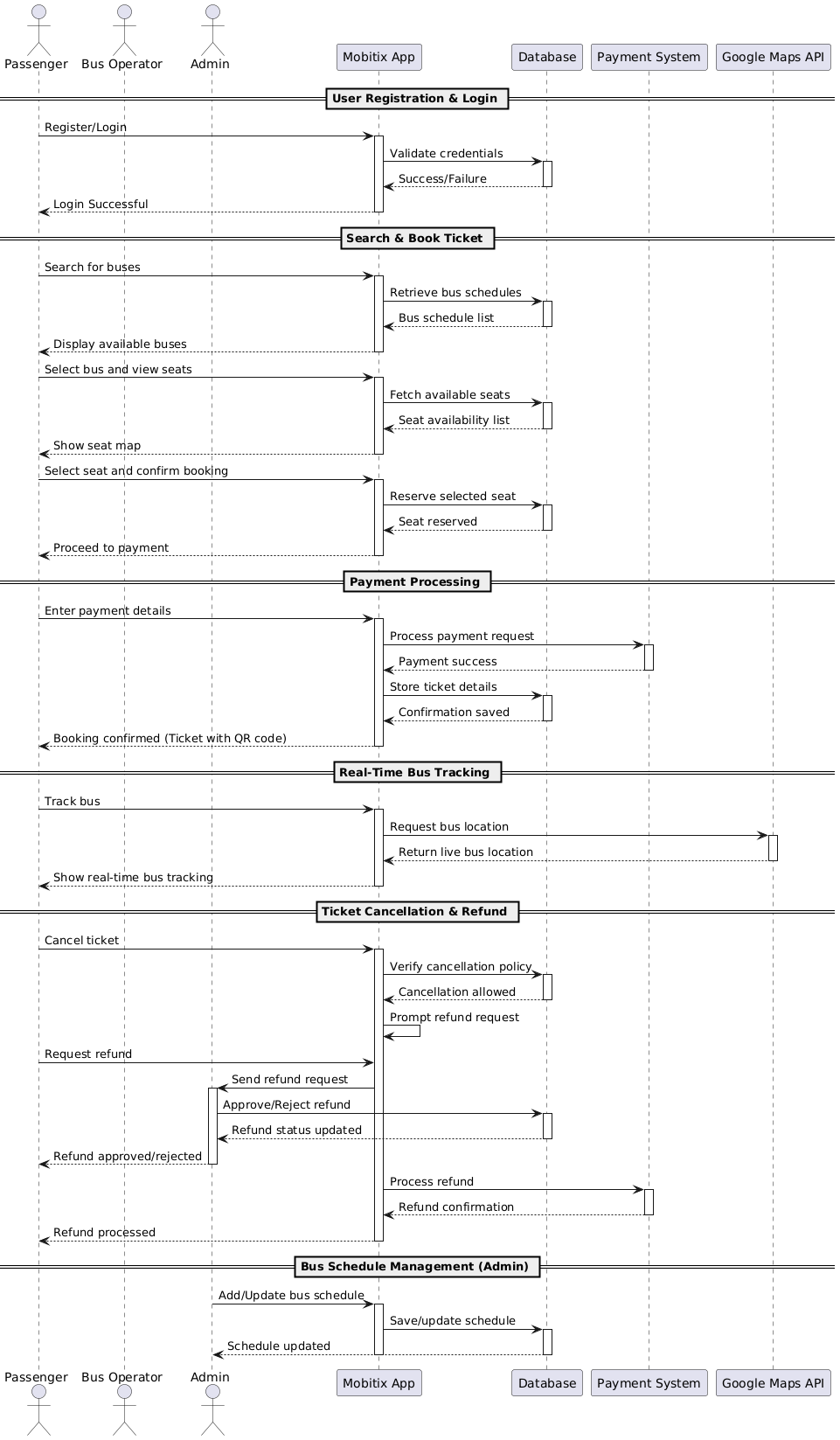


Figure Sequence Diagram

3.1.4 Design Narrative

***Key Components***

User Interface (UI)

* Frontend Technologies: The system's UI will be built using Flutter, a cross-platform framework that enables a seamless, responsive, and visually appealing experience for both Android and iOS users.
* Design Principles: The UI will follow modern design principles, ensuring intuitive navigation, accessibility, and ease of use for passengers and administrators.
* Key Interfaces:
  + Passenger Dashboard: Displays ticket booking options, real-time bus tracking, and trip history.
  + Admin Dashboard: Provides tools for bus schedule management, user analytics, and refund approvals.
  + Booking Page: Features a dynamic seat selection system, bus search filters, and a secure payment gateway.
  + Notifications Panel: Displays alerts for bus departures, cancellations, estimated arrival times, and payment confirmations

Backend Architecture

* Programming Language: The backend will be developed using PHP, chosen for its efficiency in handling RESTful API requests and database operations.
* Database Management: MySQL will be used to store and manage user profiles, ticket reservations, transaction records, and bus schedules. The database will be designed to support high-speed queries and ensure data consistency.
* Data Processing: The system will leverage server-side validation and caching mechanisms to optimize performance and ensure secure data handling.

Integration with External Systems

* Payment Gateway Integration: Mobitix will integrate with local Sri Lankan payment gateways such as Genie, Koko, and eZ Cash via API connections, enabling secure and convenient digital payments.
* Real-time Bus Tracking: The Google Maps API will provide live GPS tracking of buses, allowing users to view real-time location updates and estimated arrival times.
* Notification Services: Firebase Cloud Messaging (FCM) will be used to send push notifications for bus status updates, payment confirmations, and refund alerts.

Agile Methodology

The development of the Mobitix Digital Bus Ticket Booking App will follow the Agile methodology, which emphasizes flexibility, iterative development, and continuous improvement. Agile allows for incremental progress, enabling the project to adapt to changes and feedback throughout the development lifecycle.

* Iterative Development: The project will be broken down into manageable tasks, with each feature developed, tested, and refined before moving on to the next.
* Continuous Testing & Improvement: Regular testing will be conducted to ensure functionality, security, and usability, allowing for early identification and resolution of issues.
* Adaptive Planning: Instead of rigid upfront planning, development priorities can be adjusted based on technical feasibility and evolving requirements.
* Focus on Deliverables: The project will be developed in stages, ensuring that each component (e.g., authentication, ticket booking, payment integration) is fully functional and optimized before deployment.

Major Functionalities

Passenger Features

* Seamless Registration & Authentication: Users can sign up using email or phone number, with OTP verification for security.
* Dynamic Seat Selection: An interactive seat map will allow passengers to select their preferred seats.
* Secure Online Ticket Purchase: Users can pay via integrated payment gateways, with instant e-ticket generation and a QR code-based boarding pass.
* Trip Management: Passengers can view trip history, cancel bookings, and request refunds based on predefined policies.

Administrator Features

* Bus Schedule Management: Admins can add, edit, and delete bus schedules, update fares, and assign drivers.
* Booking & Refund Approvals: Admins can view real-time booking statistics and process refund requests efficiently.
* Performance Analytics: Data insights will help operators analyze ticket sales trends, peak travel times, and user preferences.

Key Benefits

* Enhanced Public Transport Efficiency: Real-time tracking, online booking, and digital payments will significantly reduce congestion, waiting times, and manual errors.
* Improved User Experience: A modern, intuitive mobile app will provide greater convenience for passengers compared to traditional ticketing systems.
* Operational Optimization: Bus operators will benefit from automated scheduling, ticket sales reports, and enhanced customer insights, allowing them to refine services.
* Security & Compliance: Adherence to data privacy regulations (GDPR, CCPA) ensures user trust and legal compliance.

## 3.2 System Overview

3.2.1 System Architecture

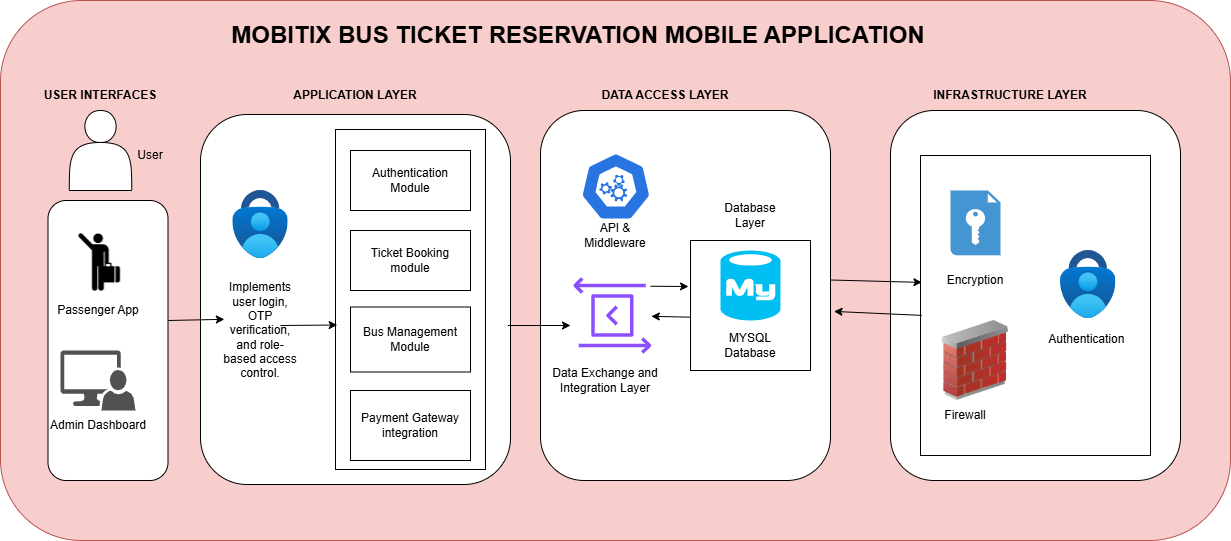


Figure System Architecture Diagram

The System Architecture Diagram for the Mobitix Digital Bus Ticket Booking App visually represents the interaction between various system components of the mobile application. The diagram is structured into multiple layers, each playing a distinct role in the system's functionality.

At the top, the User Interface Layer includes the Passenger App, and the Admin Dashboard. These interfaces enable users & administrators to interact with the system by searching for routes, booking tickets, managing schedules, and monitoring transactions. Arrows from these interfaces point toward the Application Layer, indicating user requests being processed by the backend system.

The Application Layer serves as the core processing unit of the system. It comprises several modules, including Authentication, Ticket Booking, Bus Management, and Payment Gateway Integration. The Authentication Module handles user login and role-based access control, while the Ticket Booking Module manages seat selection, confirmation, and ticket generation. The Bus Management Module ensures that schedules and routes are properly maintained, and the Payment Gateway facilitates secure transactions through external providers. This layer interacts with the Data Access Layer via APIs and middleware, ensuring smooth communication between the frontend and backend.

The Data Access Layer consists of APIs & Middleware and the Data Exchange & Integration Layer, which act as intermediaries between the backend services and the Database Layer. This setup ensures efficient data retrieval and synchronization across different modules. The Database Layer, utilizing SQL/NoSQL databases like MySQL, stores critical system data, including user details, booking history, and bus schedules.

The Security Layer in the infrastructure layer implements encryption, firewalls, and intrusion detection to protect sensitive user data. Bidirectional arrows indicate interactions between the Application Layer and Cloud Infrastructure, ensuring a secure and responsive system.

3.2.2. Data Model

The data model of the **Mobitix Digital Bus Ticket Booking App** is structured to efficiently manage bus ticket bookings, user information, and financial transactions while ensuring data integrity and security. The key components of the model include:

***Entities***

* **User:** Represents passengers, admins, and bus operators. Stores user details such as full name, email, role, phone number, and password.
* **Ticket:** Stores booking details, including schedule ID, user ID, seat number, price, booking time, payment status, and ticket status.
* **Payment:** Manages payment transactions, including payment ID, user ID, ticket ID, payment method, payment status, amount, timestamp, and transaction ID.
* **Refund:** Handles refund requests, linking users and tickets, including approval status and request date.
* **Bus:** Represents buses in the system, storing details such as bus number, operator name, total seats, and operational status.
* **Route:** Defines bus routes, storing departure time, arrival time, date, and available seats.
* **Bus Schedule:** Manages bus schedules by linking buses to schedules and tracking available seats.
* **GPS:** Stores bus tracking information, including timestamp, latitude, and longitude.

***Relationships:***

* A User can book multiple Tickets.
* A Ticket is associated with a Bus Schedule and a User.
* A User can make multiple Payments for Tickets.
* A User can request multiple Refunds, which are linked to a specific Ticket.
* A Bus follows a Route and is assigned to multiple Schedules.
* A Bus Schedule is linked to a Bus and multiple Tickets.
* A Bus has a GPS tracking record.

**Normalization:**

The data model is designed with normalization principles to eliminate redundancy and ensure data consistency. Key attributes are separated into different tables, and foreign key constraints maintain referential integrity.

**Security Considerations:**

* User authentication: Passwords are securely stored.
* Payment security: Transactions are linked to payment gateways, ensuring encrypted and secure processing.
* Access control: Role-based access control (RBAC) restricts functionalities based on user roles.

**Data Types:**

* The model uses optimized data types for efficient storage and retrieval:
* VARCHAR for text attributes such as names and emails.
* INTEGER for IDs and numerical attributes.
* DECIMAL for monetary values.
* DATETIME for timestamps and booking times.
* ENUM for predefined categories such as user roles and payment statuses.

***Entity Relationship Diagram***

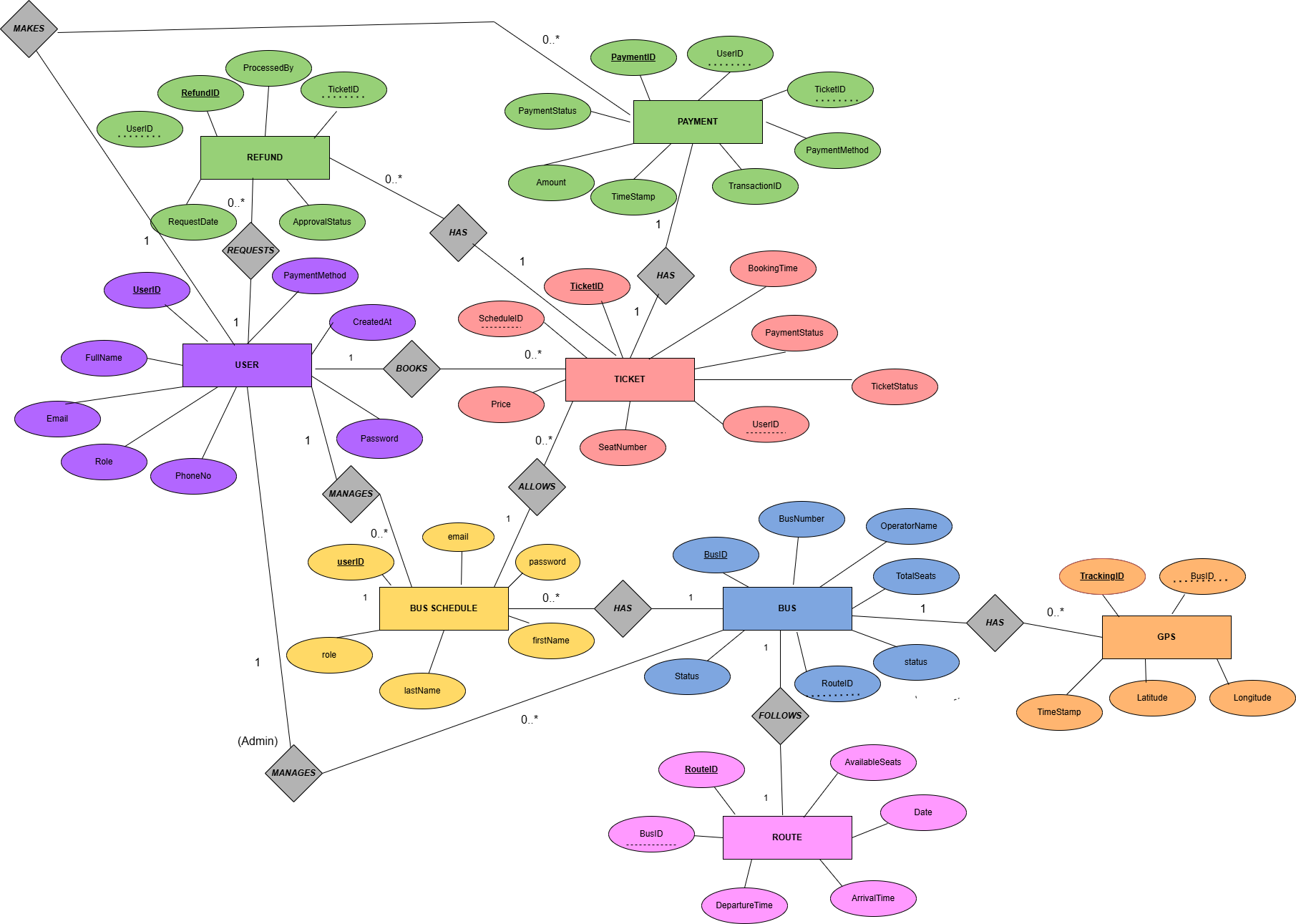
******

Figure Entity Relationship Diagram

This ER diagram represents the structure of the Mobitix bus ticket reservation system, detailing the relationships between key entities such as users, tickets, buses, routes, payments, refunds, and schedules.

The User entity, identified by UserID, includes attributes such as FullName, Email, Role, PhoneNo, Password, and CreatedAt. Users can book tickets, make payments, and request refunds. The system distinguishes between regular users and administrators, with the latter managing bus schedules.

The Ticket entity, identified by TicketID, contains attributes such as ScheduleID, Price, SeatNumber, BookingTime, PaymentStatus, TicketStatus, and UserID. Each ticket is associated with a schedule and a user who books it. A user can book multiple tickets, while each ticket corresponds to a single schedule.

The Payment entity, linked to tickets, includes PaymentID, UserID, TicketID, PaymentMethod, Amount, TimeStamp, and TransactionID. It ensures that every ticket purchase is recorded along with payment details.

Similarly, the Refund entity, identified by RefundID, consists of UserID, TicketID, RequestDate, ApprovalStatus, and ProcessedBy. Users can request refunds, which are either approved or rejected by an administrator.

The Bus Schedule entity, managed by administrators, includes UserID, Email, Password, FirstName, and LastName. It ensures that schedules are controlled by authorized personnel.

The Bus entity, identified by BusID, contains BusNumber, OperatorName, TotalSeats, Status, and RouteID. Each bus follows a specific route and has a status indicating its availability.

The Route entity, identified by RouteID, consists of BusID, AvailableSeats, Date, DepartureTime, and ArrivalTime. It defines the journey details for buses.

Additionally, the GPS entity, linked to buses, includes TrackingID, BusID, TimeStamp, Latitude, and Longitude, allowing real-time tracking of buses.

These entities are connected through relationships such as "BOOKS" between users and tickets, "MAKES" between users and payments/refunds, "HAS" between tickets and payments, "MANAGES" between administrators and bus schedules, "FOLLOWS" between buses and routes, and "HAS" between buses and GPS tracking. These relationships ensure smooth operation of the bus reservation app by managing user interactions, payments, ticketing, and bus tracking efficiently.

**Relational Schema**

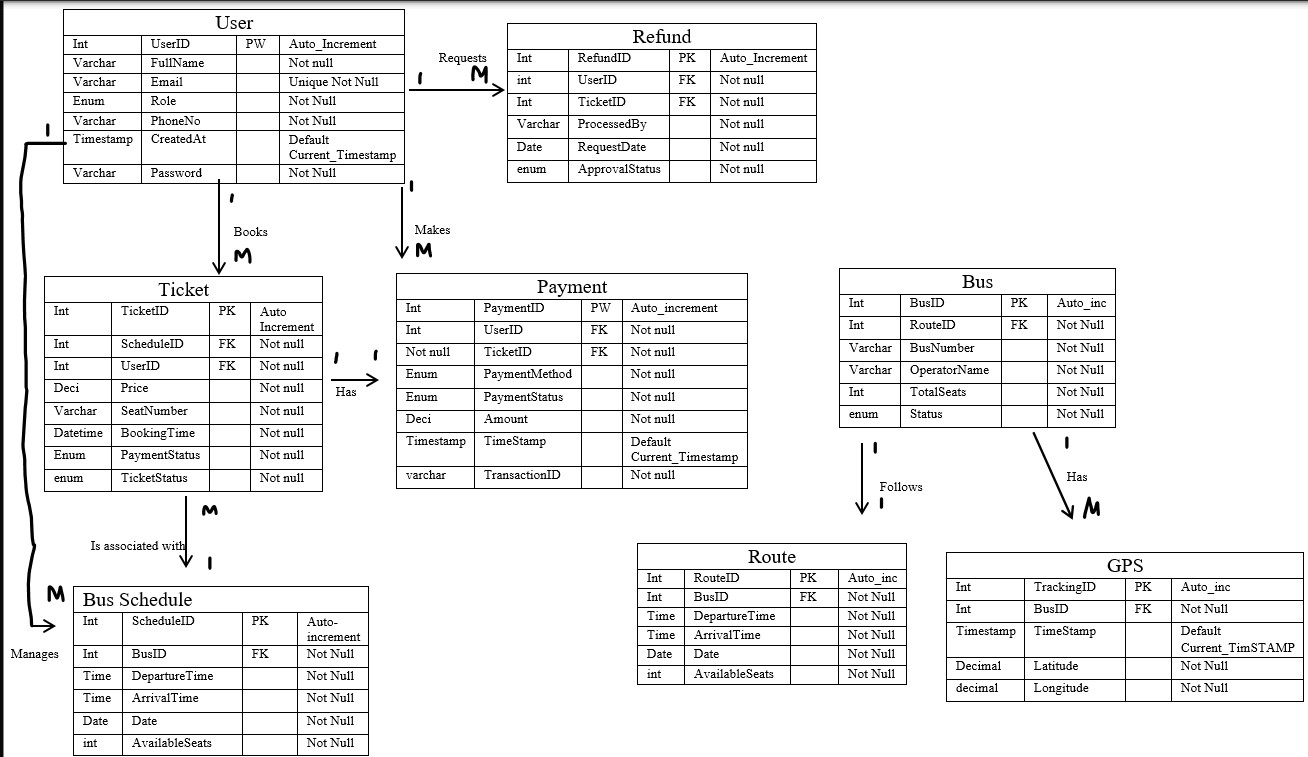


Figure Relational Schema

**Main Entities & Their Roles:**

1. User – People using the system (passengers, admins). They can book tickets, make payments, and request refunds.
2. Ticket – Represents a bus booking, linked to a User and a Schedule.
3. Payment – Stores payment details when users buy tickets.
4. Refund – Allows users to request refunds, which an admin can approve or reject.
5. Bus – Represents a physical bus, linked to a Route and Schedule.
6. Route – Defines the start and end locations, including departure and arrival times.
7. Bus Schedule – Defines when a bus operates on a given route.
8. GPS – Tracks bus locations using latitude and longitude.

**Relationships:**

* A User can book multiple Tickets.
* A Ticket is linked to a Payment.
* A User can request a Refund for a ticket.
* An Admin manages Bus Schedules.
* A Bus follows a Route and has a GPS tracker.

***Class Diagram***

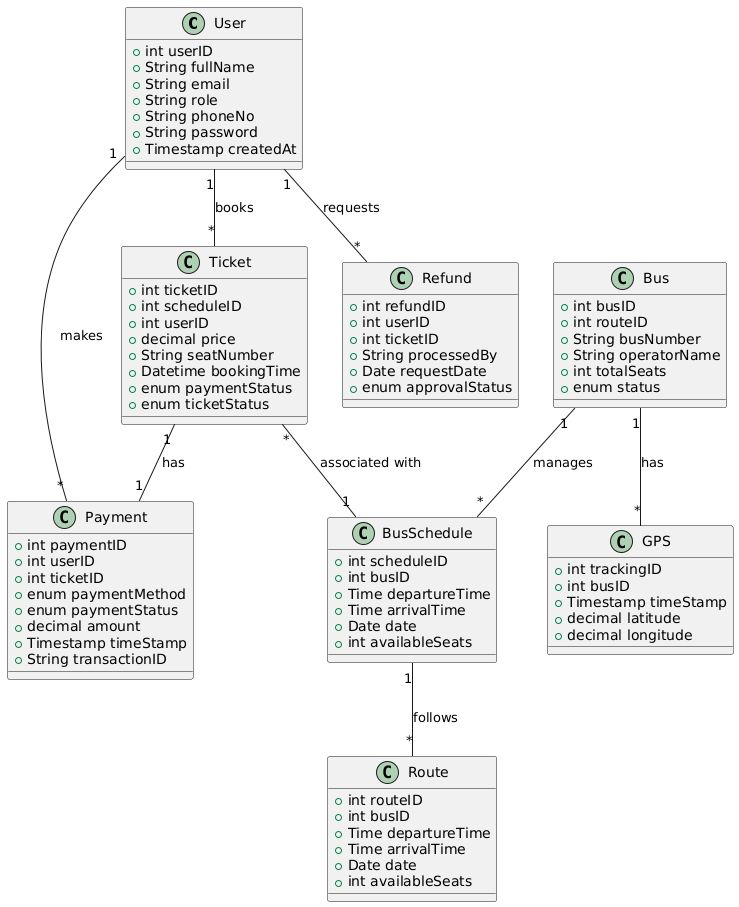


Figure Class Diagram

The UML class diagram for the Mobitix Digital Bus Ticket Booking App represents the core entities and their relationships within the system. The User class, which includes attributes such as FullName, Email, and PhoneNo, interacts with other classes like Ticket, Payment, and Refund. A User can book multiple Tickets, which are associated with a BusSchedule that manages bus timings and available seats. The Bus class is linked to Route, defining travel details like DepartureTime and ArrivalTime. The Payment class ensures secure transactions, while the Refund class handles ticket cancellations and approvals. Additionally, the GPS class tracks buses in real-time. Relationships such as one-to-many (eg: a user can have multiple tickets) and many-to-one (eg: multiple buses following a route) ensure a well-structured and scalable system.

***Database Table Structures***

1. User Table

Table User Table

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| UserID | Int | PRIMARY KEY |
| FullName | Varchar | NOT NULL |
| Email | Varchar | UNIQUE  NOT NULL |
| Role | Enum | NOT NULL |
| PhoneNo | Varchar | NOT NULL |
| CreatedAt | Timestamp | DEFAULT  Current\_Timestamp |
| Password | Varchar | NOT NULL |

2. Ticket Table

Table Ticket Table

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| TicketID | Int | PRIMARY KEY  AUTO INCREMENT |
| ScheduleID | Int | FOREIGN KEY  NOT NULL |
| UserID | Int | FOREIGN KEY  NOT NULL |
| Price | Deci | NOT NULL |
| SeatNumber | Varchar | NOT NULL |
| BookingTime | Datetime | NOT NULL |
| PaymentStatus | Enum | NOT NULL |
| TicketStatus | enum | NOT NULL |

3. Refund Table

Table Refund Table

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| RefundID | Int | PRIMARY KEY  AUTO INCREMENT |
| UserID | int | FOREIGN KEY  NOT NULL |
| TicketID | Int | FOREIGN KEY  NOT NULL |
| ProcessedBy | Varchar | NOT NULL |
| RequestDate | Date | NOT NULL |
| ApprovalStatus | enum | NOT NULL |

4. Payment Table

Table Payment Table

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| PaymentID | Int | PRIMARY KEY  AUTO INCREMENT |
| UserID | Int | FOREIGN KEY  NOT NULL |
| TicketID | Not null | FOREIGN KEY  NOT NULL |
| PaymentMethod | Enum | NOT NULL |
| PaymentStatus | Enum | NOT NULL |
| Amount | Deci | NOT NULL |
| TimeStamp | Timestamp | DEFAULT  Current\_Timestamp |
| TransactionID | varchar | NOT NULL |

5. Bus Schedule Table

Table Bus Schedule Table

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| ScheduleID | Int | PRIMARY KEY  AUTO INCREMENT |
| BusID | Int | NOT NULL |
| DepartureTime | Time | NOT NULL |
| ArrivalTime | Time | NOT NULL |
| Date | Date | NOT NULL |
| AvailableSeats | int | NOT NULL |

6. Bus Table

Table Bus Table

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| BusID | Int | PRIMARY KEY  AUTO INCREMENT |
| RouteID | Int | FOREIGN KEY  NOT NULL |
| BusNumber | Varchar | UNIQUE  NOT NULL |
| OperatorName | Varchar | NOT NULL |
| TotalSeats | Int | NOT NULL |
| Status | enum | NOT NULL |

7. Route Table

Table Route Table

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| RouteID | Int | PRIMARY KEY  AUTO INCREMENT |
| BusID | Int | FOREIGN KEY  NOT NULL |
| DepartureTime | Time | NOT NULL |
| ArrivalTime | Time | NOT NULL |
| Date | Date | NOT NULL |
| AvailableSeats | int | NOT NULL |

8. GPS table

Table GPS table

|  |  |  |
| --- | --- | --- |
| Column Name | Data Type | Constraints |
| TrackingID | Int | PRIMARY KEY  AUTO INCREMENT |
| BusID | Int | FOREIGN KEY  NOT NULL |
| TimeStamp | Timestamp | DEFAULT  Current\_Timestamp |
| Latitude | Decimal | NOT NULL |
| Longitude | decimal | NOT NULL |

3.2.3 Wireframes

1. Login Page

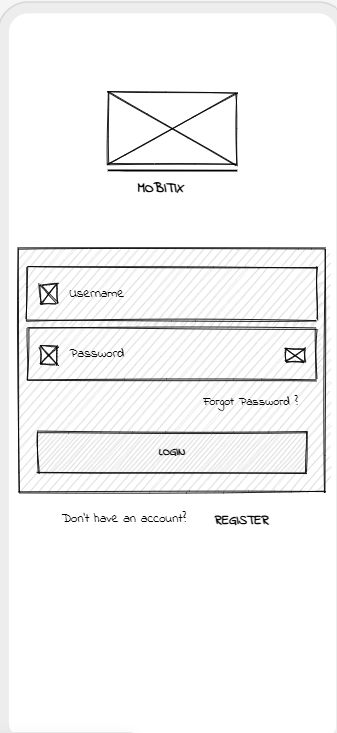


Figure Login page wireframe

2. Home Screen

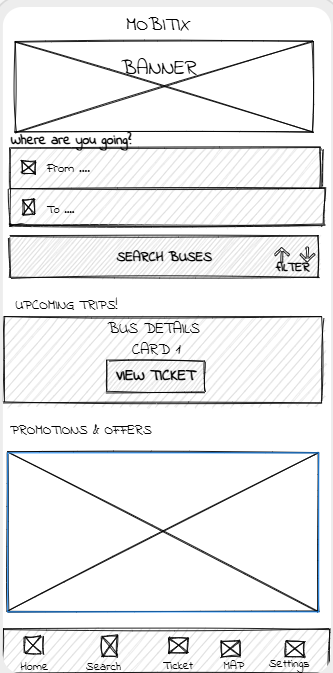


Figure Home Screen Wireframe

3. Search screen wireframe and Navbar

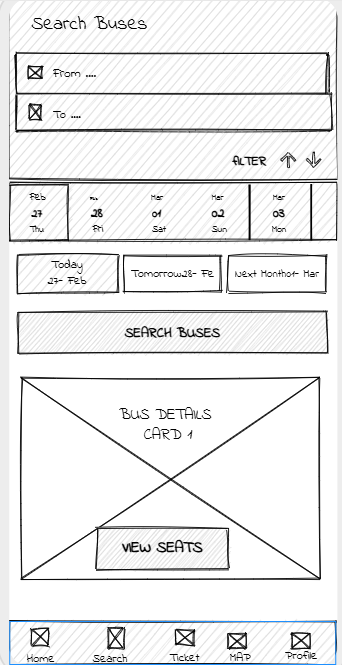


Figure Search Screen Wireframe

4. Seat Selection Screen

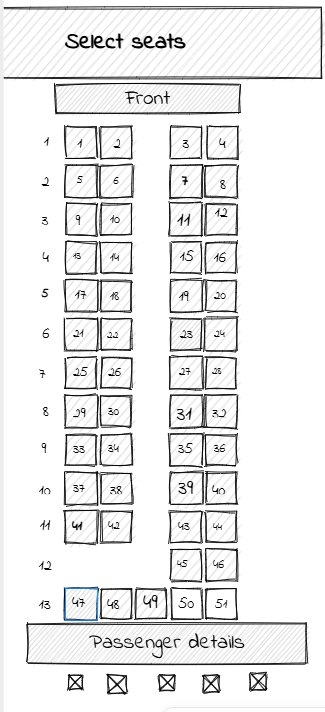


Figure Seat Selection Screen wireframe 1

5. Seat Details

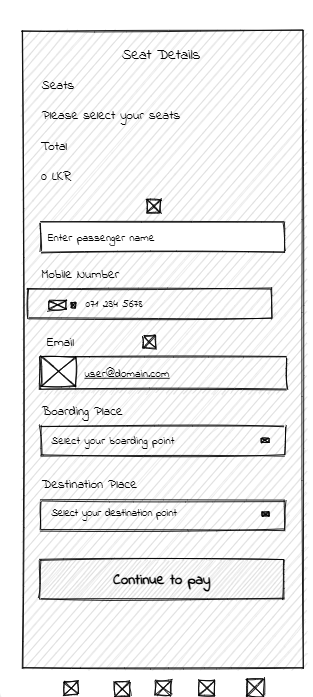


Figure Seat Details screen

6. Payment Screen

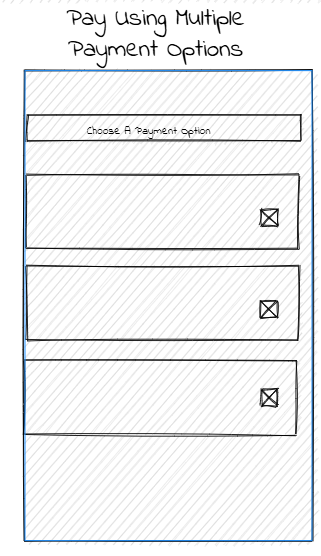


Figure Payment Screen

7. Ticket Confirmation Screen

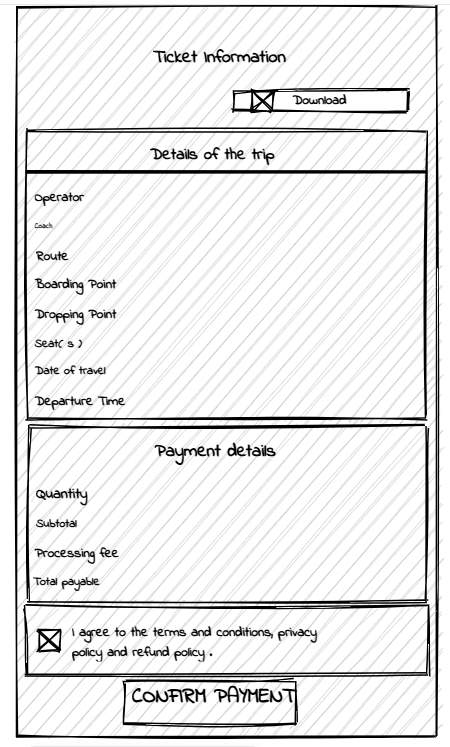


Figure Ticket Confirmation wireframe

8. Settings Screen

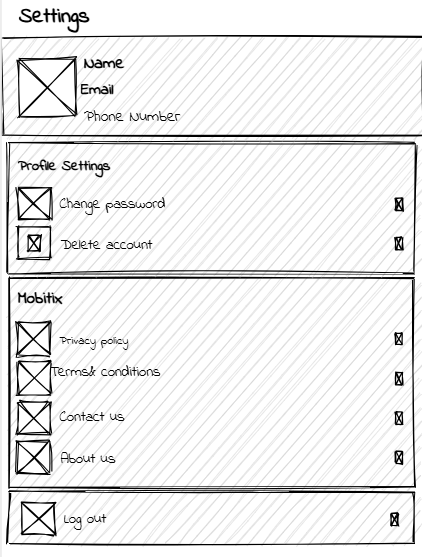


Figure Settings Screen

# References

* Ceipidor, U. e. (2013). Mobile ticketing with NFC management for transport companies, Problems and Solutions. *2013 5th International Workshop on Near Field Communication, NFC 2013*.
* Dr. Swapna Ubale, e. a. (2024). BUS TICKET BOOKING APPLICATIONS ON ANDROID PLATFORM. *International Research Journal of Modernization in Engineering Technology and Science*, 945- 948.
* Jakubauskas, G. (March 2014). Improvement of urban passenger transport ticketing systems by. *Transport*, 37-41.
* Jayasuriya, J. (2021). Smart Ticketing and Seat Reservation System for Sri Lankan Railway. *PDF (ir.kdu.ac.lk). Faculty of Computing, Department of Information Technology, General Sir John Kotelawala Defence University, Southern Campus, Sri Lanka*, 52.
* Jeewanthi Fernando, e. a. (September 2016). ONLINE BUS TICKET RESERVATION SYSTEM TO THE NATIONAL TRANSPORTATION SERVICE IN SRI LANKA. *Proceedings in Computing, 9th International Research Conference-KDU, Sri Lanka*, 96-100.
* Mezghani, M. (n.d.). Study on Electronic Ticketing in Public Transport. . *November, 2014*.
* Oloyede, M. A. (2014). Development of an Online Bus Ticket Reservation System for a Transportation Service in Nigeria . *Computer Engineering and Intelligent Systems* .
* The Department of Census and Statistics Sri Lanka. (2022). *Statistical Pocket Book.* Baththaramulla: Ministry of Finance, Economic Stabilization and National Policies Sri Lanka .
* Vimukthi, D. (2023). Online Bus Ticketing and Tracking System. *Informatics Institute of Technology*.

# Appendices

*You may have several appendixes (Appendix 1, Appendix 2 or Appendix A, Appendix B) to refer to further details related to chapters like: Technology adapted, Analysis and Design, Implementation, evaluation, etc.*