Final Report

Sri-Lanka’s Railway Reservation System

Open University of Sri Lanka

Bachelor of Software Engineering

Cluster: 22.3
Mentor: Mrs. Ahalikai Suthaharan

Team Members:

321425097 – M.L. Saiyaf – s92065097

721428968 – M.R.R. Sarose – s92068968

421435450 – W.D. Ishani – s92075450

721446082– M.H.A. Rahman– s92086082

Contents

1.Introduction.

The Sri Lankan Railway system plays a vital role in the country's transportation infrastructure, facilitating the daily commute for hundreds of thousands of people. However, one of the major challenges faced by the railway is the issue of long queues at ticket counters, leading to significant time wastage, increased stress, and frustration among commuters.

The current system lacks an adequate number of ticket counters to handle the crowds during peak times, exacerbating the problem. Furthermore, the manual process of issuing tickets and handling change further slows down transactions, contributing to the inconvenience experienced by passengers.

To address these challenges, an automated system is proposed as a solution. However, it is crucial to consider the specific context of Sri Lanka when implementing automation in the railway system. While there is an existing system that allows booking via phone calls and a website provided by the railway department, it can be further enhanced to save customers' time and money, providing a more efficient and user-friendly experience.

The revamped system will involve the development of a web application that utilizes a basic Client-Server Architecture. This web application will offer various improvements to streamline the ticketing process and enhance the overall commuting experience for passengers. Some of these improvements include Flexible Payment Options, Comprehensive Reservation Information, User-Friendly Interface

By implementing these improvements, the Sri Lankan Railway aims to streamline the ticketing processes, reduce waiting times, and enhance the overall commuting experience for passengers. The automated system will alleviate the stress and frustration associated with long queues, enabling passengers to save time and have a more convenient journey. Additionally, by offering flexible payment options and comprehensive reservation information, the system will cater to the diverse needs of passengers, making their interactions with the railway more efficient and satisfying.

2.Problem Statement & Project Objectives.

The existing railway system in Sri Lanka lacks important features such as user account management, self-cancellation capabilities, and the provision of shortest route details from the passengers' current locations. Additionally, it fails to offer access to the passengers' previous booking history and information regarding festive season offers and discounts.

The key objectives of this project proposal are as follows:

a. Enhancing automated ticketing system: The project aims to Enhance the automated ticketing system that replaces the existing automated process and improves the automated system.

b. Enhance payment options: The proposed system will offer flexible payment options to passengers, including online payment gateways, and electronic ticketing.

c. Improve reservation information: The system will provide real-time updates on seat availability, train schedules, and fare details, enabling passengers to plan their journeys effectively and reduce uncertainty.

d. Develop a user-friendly interface: The ticketing system will have an intuitive and easy-to-navigate interface, accommodating users with varying levels of technological proficiency for seamless access and usage.

e. Multilingual Support: Recognizing the diverse linguistic landscape in Sri Lanka, the ticketing system can provide multilingual support, allowing passengers to choose their preferred language for accessing the interface. This will enhance accessibility and ensure a user-friendly experience for passengers who may not be proficient in the country's official languages.

f. Customer Feedback and Ratings: The system can include a feedback and rating mechanism, allowing passengers to provide their input on the service quality and overall experience. This will enable the Sri Lankan Railway to gather valuable insights, identify areas for improvement, and continuously enhance the ticketing system based on customer feedback.

3. Introduction to similar type of System.

The Sri Lankan railway system has taken a significant step towards modernizing its services by developing an online reservation system accessible through mobile phones and personal computers. This system was introduced with the aim of providing convenience and ease of access for passengers. Initially, the system garnered attention and saw a considerable number of users due to its core functionalities such as

- Search train schedules.
- Can book tickets online.
- Can view recently used source and destination.

However, despite the initial success, the usage of the system began to decline after a few months of its release to the public. This decline can be attributed to the system's limitations and its failure to deliver certain crucial features such as

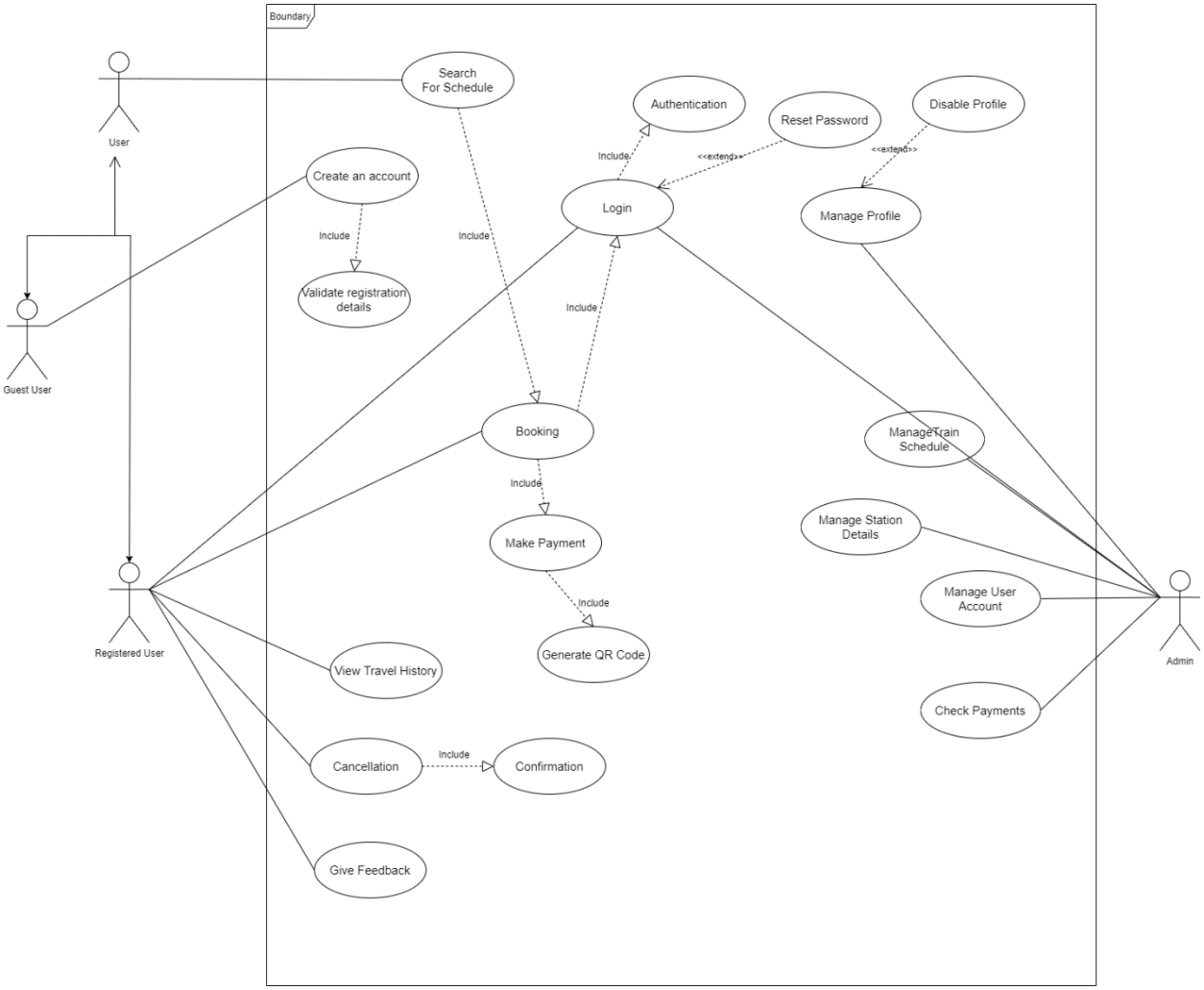
- Cannot cancel train tickets online.
- Payments can only Done via railway stations.
- Cannot search route between two places.
- Cannot search nearby places.

Furthermore, to address these issues It is necessary to revamp the system that incorporates the missing features and resolves the existing issues.

4.Project Solution.

The proposed web-application aims to address the limitations of the current railway system by providing a user-friendly platform for online booking and self-cancellation. It offers features such as account management, transaction history, and updates on festive offers/discounts. Users can easily access train schedules, make reservations, and receive notifications on reservation status and cancellations. The system emphasizes the need for a registered account to facilitate seamless payments and refund processes, ensuring a comprehensive solution for user requirements.

High – Level Use case Diagram



5. Technology Used.

SwiftTail is a powerful website made using a bunch of modern tools and tech. It mixes together the parts you see (like buttons and colors, Functions) with the hidden stuff that makes it all work smoothly. We used different computer languages and special tools to create a website that looks great and does cool things when you use it.

Front-end Technologies:

1. HTML & CSS:

HTML (Hypertext Markup Language): Used for structuring the web pages, defining the content layout, and organizing elements.
CSS (Cascading Style Sheets): Employed for styling and designing the visual presentation of HTML elements, ensuring a consistent and appealing user interface.

2. Bootstrap:

Front-end Framework: Utilized Bootstrap to streamline front-end development by providing a collection of pre-designed components, grid system, and CSS classes for responsive and modern UI elements.

3. jQuery:

JavaScript Library: Integrated jQuery to simplify client-side scripting and enhance user interactions by providing an easy-to-use interface for DOM manipulation and event handling.

4. Font Awesome:

Icon Toolkit: Incorporated Font Awesome to add scalable vector icons, enhancing the visual aesthetics and usability of the web application by using a wide range of icons.

Back-end Technologies:

1. PHP:

Server-side Scripting Language: Used PHP for server-side logic, handling data processing, interacting with the database, and generating dynamic content for the front-end.

2. MySQL:

Relational Database: Employed MySQL as the database management system to store, manage, and retrieve structured data efficiently for the application.

3. PHP Libraries:

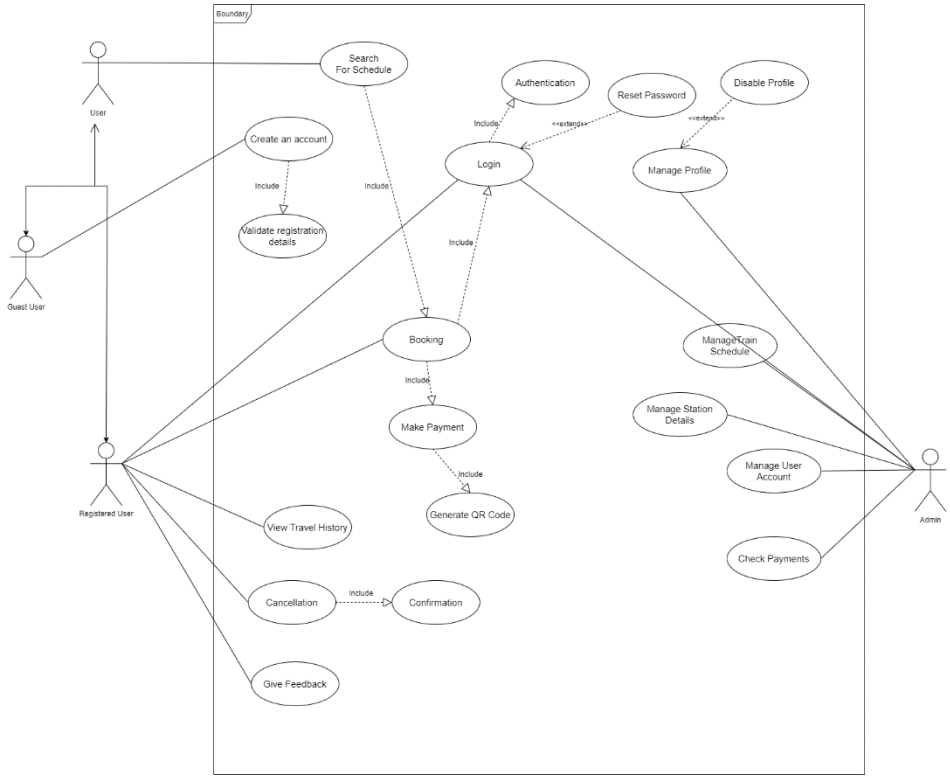
Additional Functionality: Leveraged various PHP libraries to extend the functionality of the back-end, such as handling forms, authentication, data validation, and other tasks to streamline development.

4. WampServer:

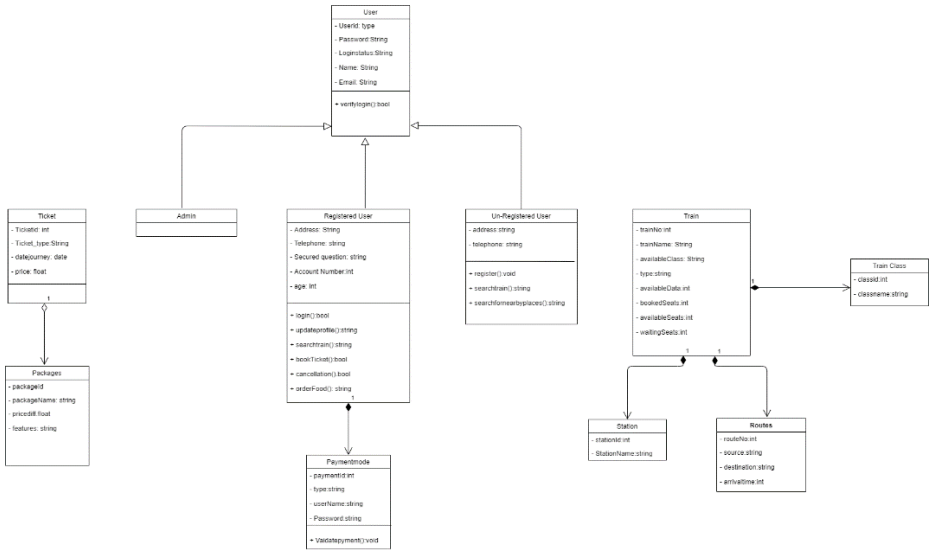
Local Server Environment: Used WampServer to create a local development environment, incorporating Apache, MySQL, PHP, and other tools necessary for hosting and testing the application locally.

6. Design Document.

Use Case Diagram.

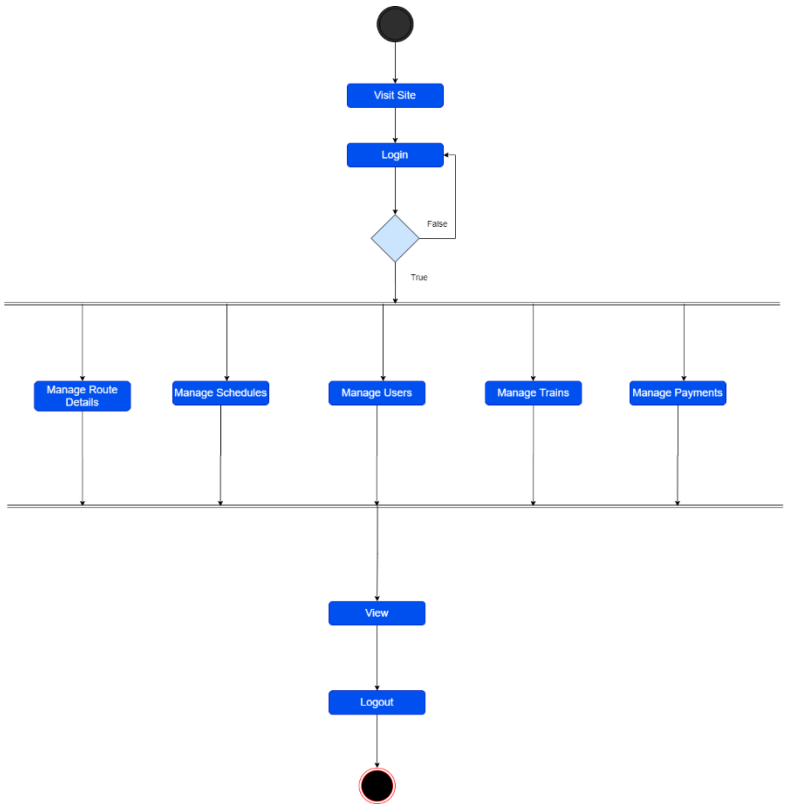


Class Diagram.

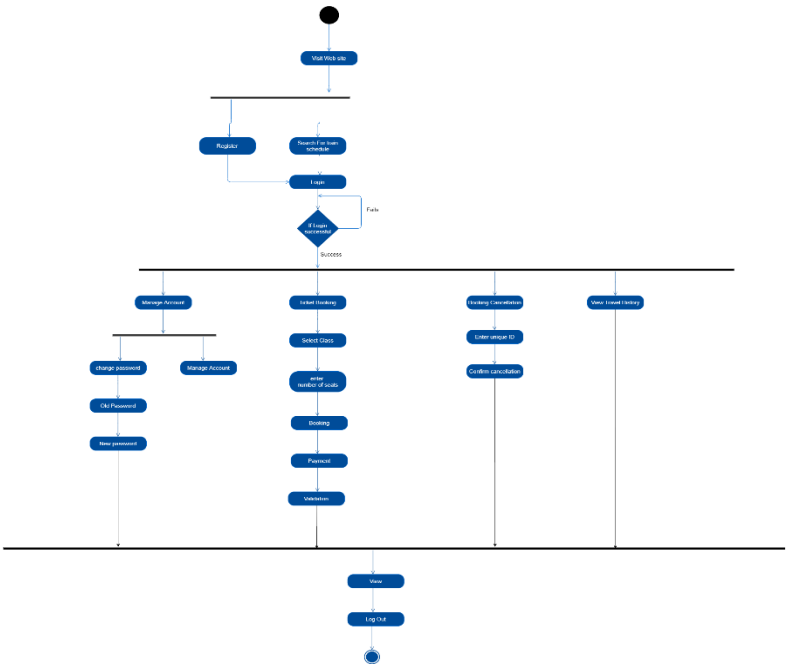


Activity Diagram.

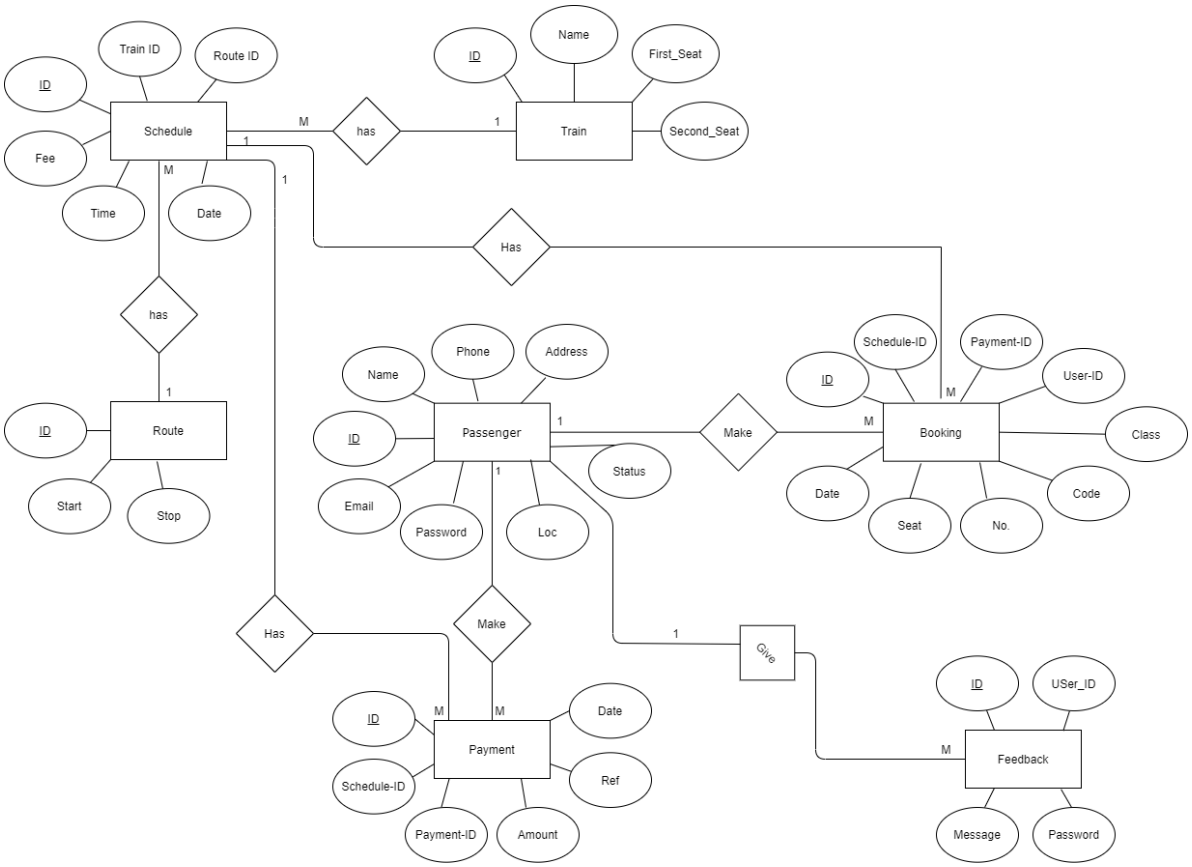
Admin



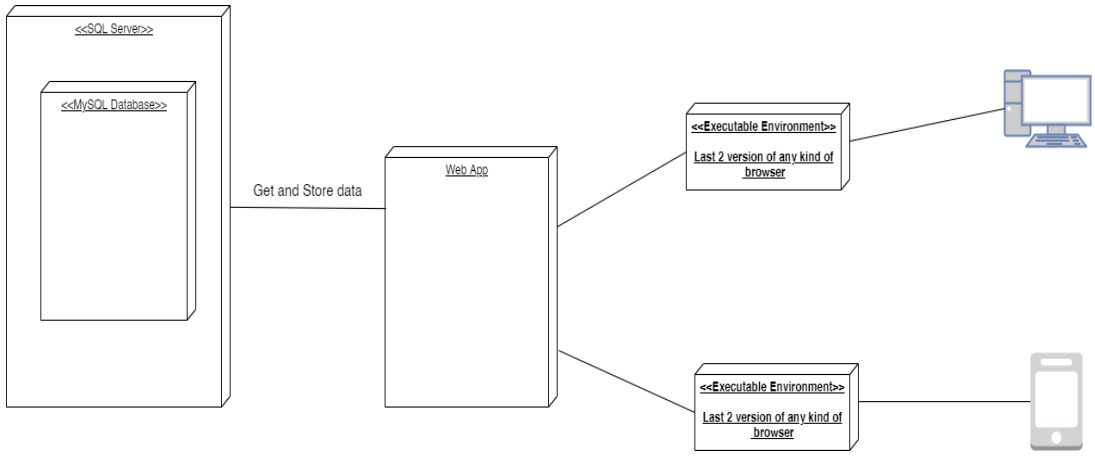
Passenger.



ER Diagram.



Overall Architecture-Block Diagram.



7. Test Case Document & Test Result.

This document presents the comprehensive Test Case Report for the SwiftTail project. The objective of this report is to outline the detailed test scenarios, procedures, and outcomes conducted during the testing phase of the SwiftTail web application.

Click here to view the Test Case Report:
https://docs.google.com/spreadsheets/d/1tu7dEuTq9fi-3KDqHathCUueFkq9XAx /edit?usp=drive_link&ouid=101682450833822827777&rtpof=true&sd=true

8. Project limitations & Conclusion.

Project Limitations:

a. Language Support: Implementing multilingual support could be challenging due to the diverse linguistic landscape of Sri Lanka, requiring ongoing updates and translations.

b. Infrastructure Challenges: The success of the automated system heavily relies on stable internet connectivity, which might be a limitation in certain remote or underdeveloped areas of Sri Lanka.

c. Technological Accessibility: While efforts are made to create a user-friendly interface, there might still be challenges for passengers with limited technological proficiency in utilizing the system efficiently.

d. Dependency on Payment Gateways: Relying on external payment gateways may pose challenges in terms of transaction failures or technical issues that are beyond the system's control.

e. System Adoption and User Behavior: Encouraging passengers to transition from traditional ticketing methods to the automated system

might face resistance, requiring effective marketing and outreach strategies.

Conclusion:

In conclusion, the proposed automated ticketing system for the Sri Lankan Railway represents a significant advancement towards enhancing passenger experience and streamlining ticketing processes. However, it is crucial to acknowledge the identified limitations and challenges that could potentially impact the system's widespread adoption and effectiveness. Despite these limitations, the system aims to significantly reduce queues, offer convenience through flexible payment options, provide real-time information, and improve overall user experience.

9. Project Demonstration.

Here's a step-by-step Demonstration to access the SwiftTail web application:

Step 1: Setup Environment

- Ensure you have Apache server or Wamp server installed on your machine.

Step 2: Database Setup

- Locate the database file provided with the project.
- Import this database file into your local database management system (e.g., MySQL) using a tool like phpMyAdmin or the command line.

Step 3: Start the Wamp Server

- Open Wamp server and start the Apache server.

Step 4: Run the Application

- Place the SwiftTail project files into the correct directory in the Wamp server's 'www' folder (usually located at 'C:\wamp64\www' or similar depending on your installation).
- Open a web browser and enter the following URL in the address bar:

http://localhost/SwiftTail (replace 'SwiftTail' with the actual folder name where you placed the project files).

- The application should start loading.

Step 5: Login to the System

- Refer to the 'readme' file provided with the project to find the default username and password.

- Enter the default credentials in the login page to access the SwiftTail system.

This demonstration showcases the primary functionalities available to both administrators and passengers within the SwiftTail project. Administrators have comprehensive control over managing trains, routes, schedules, users, feedback, and payments, while passengers can interact with the system to book trains, manage bookings, and provide feedback.

Admin Panel:

1. **Manage Train Details:**
   - **Insert Train Details:** As an admin, you can add new train details like train number, name, departure, arrival, etc., into the system.
   - **Update Train Details:** Modify existing train information such as schedules, routes, or any other relevant data.
   - **Delete Train:** Remove outdated or unnecessary train records from the system.

2. **Route Management:**
   - **Insert Route:** Add new routes specifying origins, destinations, stops, and other route-related information.
   - **Update Route:** Modify existing route information to reflect any changes or updates.
   - **Delete Route:** Remove routes that are no longer in use or needed.

3. **Schedule Handling:**
   - **Insert Schedule:** Create schedules for trains, defining departure times, arrival times, and associated details.
   - **Update Schedule:** Modify existing schedules to accommodate changes in timings or other schedule-related information.

- **Delete Schedule:** Remove outdated or redundant schedules from the system.

4. **User Management:**
   - **Manage Users:** Admins can manage user accounts, including creating new accounts, updating user information, or deactivating accounts if necessary.

5. **Feedback and Payments:**
   - **View Feedback and Reply:** Access and review feedback submitted by users, providing responses or support where needed.
   - **View Payment Records:** View all payment-related information for bookings and transactions within the system.

Passenger Interface:

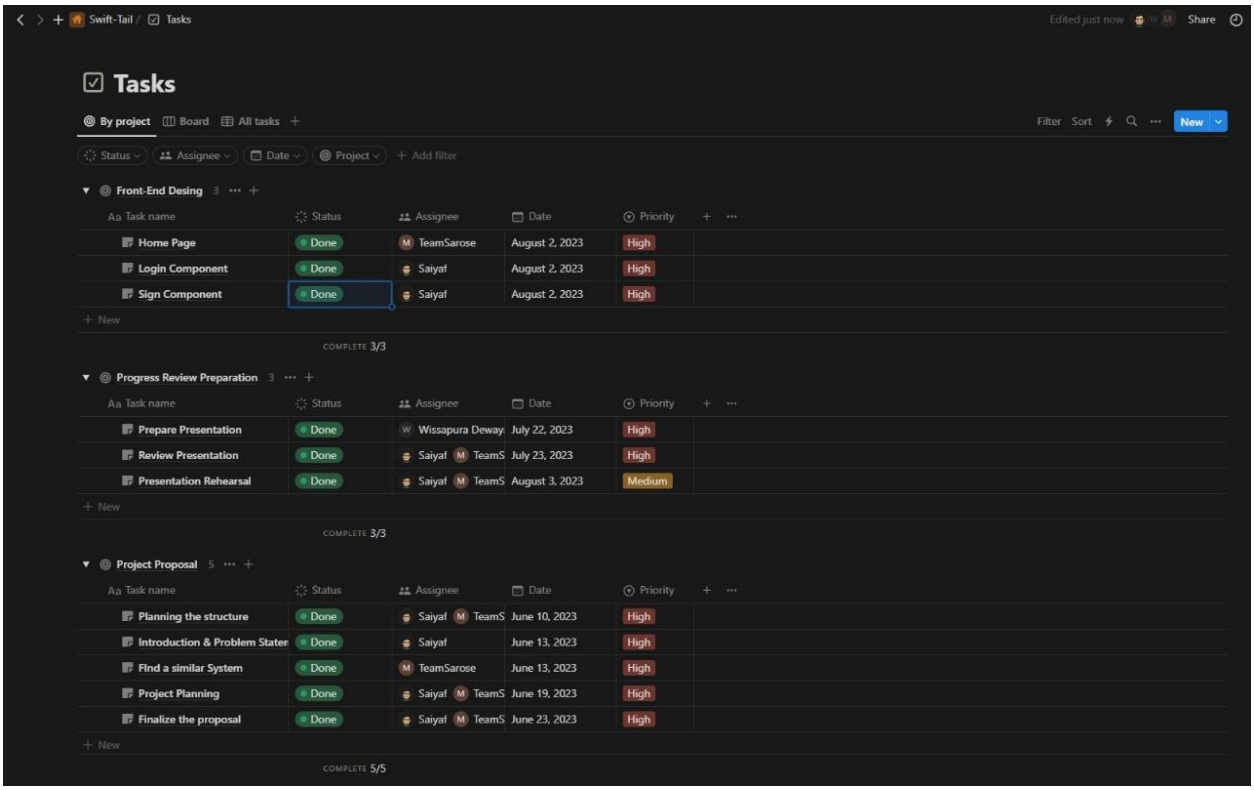
1. **Booking Process:**
   - **Book Train:** Passengers can search for trains, select seats, and make bookings for desired journeys.
   - **Make Payment:** Complete the payment process securely using available payment methods.

2. **View Bookings:**
   - **View Bookings:** Passengers can see their booked train journeys, review details, and manage their bookings.

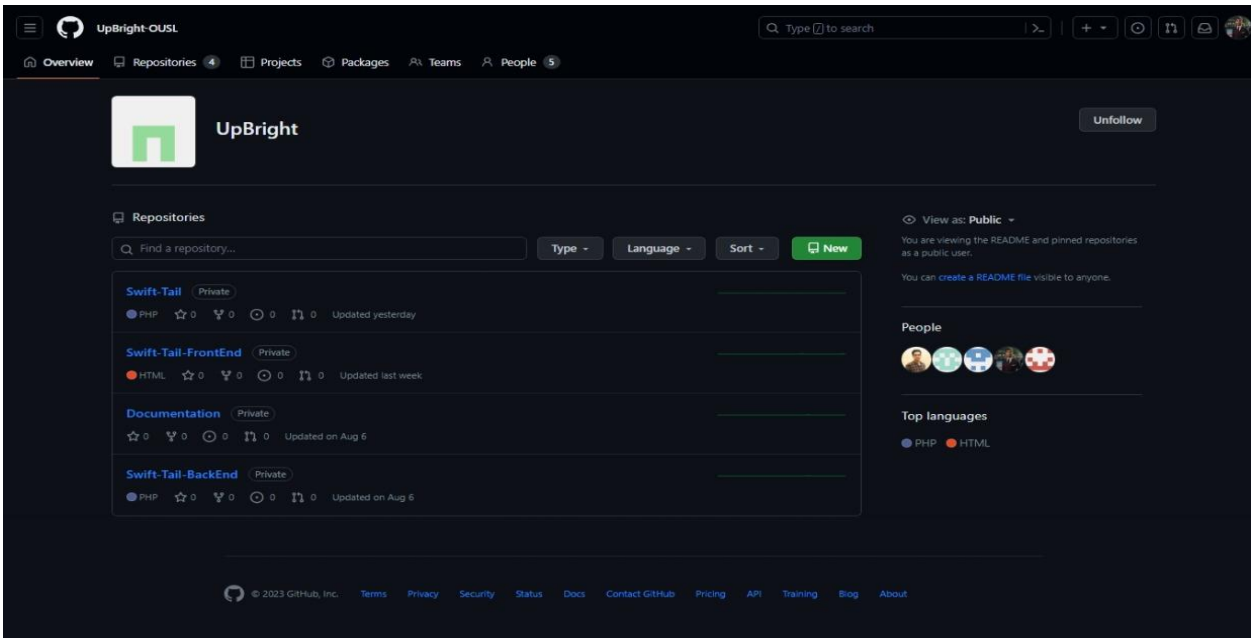
3. **User Account:**
   - **Signup:** New users can sign up for an account to access booking and other features.
   - **Send Feedback:** Passengers can submit feedback or inquiries regarding their experiences or the system.

10. Appendix.

Notion Project Governance Tool used to Manage the project.



GitHub



10.1 CMMI Meating Minutes

Meeting Information			
Meeting Date/Time	On 10/09/2023 At 10.00 AM		
Participants	**Present** Sarose Saiyaf Rahman Ishani		**Absent**
Estimated Time	160mins	**Actual Time**	140mins
Call/Location Information	Gathered on google meet		
Supported Documents	Notion		

Agenda:
- Progress of the development according to the roadmap.
- Discussion about the barriers faced by the members.

Notes/Clarifications:
Everyone should complete the assigned tasks and they should be presented at the next meeting.

Meeting Minutes:
- Discussed how the development went so far.
- Discussed the challenges and how to face them.
- Assigning tasks.

Notes/Clarifications:
Everyone should complete the assigned tasks and they should be presented at the next meeting.

Action item	Person Responsible
Designing Home page.	**Rahman**
Designing Authentication modal.	**Saiyaf**
DB design.	**Sarose**
Implementation of User Registration handler.	**Sarose**
Implementation of User Login handler.	**Ishani**
Build progress review presentation.	**Saiyaf**

10.2 SRS Report

This document outlines the Software Requirements Specification (SRS) for the SwiftTail project. It details the essential requirements, functionalities, and specifications crucial for developing the SwiftTail web application.

Click here to view the SRS Report:
https://drive.google.com/file/d/1vhAX29pXWwNz0CK4503M68amEbc7GT55/view?usp=drive_link

10.3 Final Report Approval

Ms. Ahalikai Suthaharan has reviewed this document and hereby agrees that the contents herein are accurate. Any changes to this document must be communicated in writing and signed off by both parties.

Approved By

Ms. Ahalikai Suthaharan