BUS RESERVATION SYSTEM

CS23333 – Object Oriented Programming Using JAVA Project Report

Submitted by

HARIPRASANTH P - 231001055

HEMANTH C - 231001062

Of

BACHELOR OF TECHNOLOGY

In

INFORMATION TECHNOLOGY

RAJALAKSHMI
ENGINEERING COLLEGE
An AUTONOMOUS Institution
Affiliated to ANNA UNIVERSITY, Chennai

DEPARTMENT OF INFORMATION TECHNOLOGY

RAJALAKSHMI ENGINEERING COLLEGE

NOVEMBER-2024

BONAFIDE CERTIFICATE

Certified that this project titled "Movie Reservation System" is the bonafide work of **HARIPRASANTH P (231001055), HEMANTH C (231001062)** who carried out the project work under my supervision.

SIGNATURE
Dr.P.Valarmathie
HEAD OF THE DEPARTMENT
Department of Information Technology
Rajalakshmi Engineering College

SIGNATURE
Mrs.Usha S
COURSE INCHARGE
Assistant Professor(S.G)
Department of Information Technology
Rajalakshmi Engineering College

This project is submitted for CS23333 – Object Oriented Programming Using JAVA held on _____

INTERNAL EXAMINER **EXTERNAL EXAMINER**

Table of Contents:

LIST OF FIGURES

ACKNOWLEDGEMENT

First, we thank the almighty God for the successful completion of the project. Our sincere thanks to our chairman **Mr.S. Meganathan, B.E., F.I.E** for his sincere endeavour in educating us in his premier institution. We would like to express our deep gratitude to our beloved Chairperson **Dr.Thangam Meganathan,** for her enthusiastic motivation which inspired us a lot in completing this project and Vice-Chairman **Mr. Abhay Shankar Meganathan B.E., M.S.,** for providing us with the requisite infrastructure.

We also express our sincere gratitude to our college principal, **Dr.S.N.Murugesan M.E., PhD.,** for his kind support and facilities to complete our work on time. We extend heartfelt gratitude to **Dr.P.Valarmathie, Professor and Head of the Department of Information Technology** for her guidance and encouragement throughout the work. We are very glad to thank our course faculty **Mrs. Usha S, Professor** of our department for their encouragement and support towards the successful completion of this project. We extend our thanks to our parents, friends, all faculty members, and supporting staff for their direct and indirect involvement in the successful completion of the project for their encouragement and support.

HARIPRASANTH P
HEMANTH C

1.BUS RESERVATION SYSTEM

1.1 Abstact:

The Bus Reservation System is an efficient, user-friendly software application designed to streamline bus ticket reservations. Developed using Java in the Eclipse IDE, the system integrates MySQL for database management and uses a combination of Bootstrap, HTML, CSS, and JavaScript to create an intuitive and responsive front-end interface. The primary goal of the system is to simplify the process of booking bus tickets, checking schedules, and managing reservations, ensuring an enhanced experience for both passengers and administrators.

Passengers can easily browse available buses, select their preferred schedules, and make seat reservations. The system allows them to view detailed information such as bus timings, availability, and ticket pricing. The real-time reservation feature enables users to receive immediate confirmation, providing a seamless booking experience.

On the administrative side, the system offers powerful management tools. Administrators can efficiently manage bus schedules, update availability, and monitor reservations. The system also includes a section for handling customer queries, reducing the need for manual intervention. This feature improves operational efficiency by minimizing administrative workload and ensures that the scheduling process runs smoothly.

The Bus Reservation System is designed to be scalable, secure, and easy to use. By automating the reservation and scheduling processes, it not only reduces the chances of human error but also optimizes the management of bus services. Ultimately, this system aims to enhance user experience while improving operational efficiency for bus service providers.

This abstract is structured to provide an overview of the system's objectives, technologies used, and its impact on improving bus reservation operations.

2.Introduction:

The Bus Reservation System is an innovative software solution aimed at simplifying and automating the process of bus ticket booking, scheduling, and management. As transportation services grow and the demand for more efficient travel solutions increases, the need for an automated system that manages reservations, schedules, and customer service becomes essential. This system is designed to address these challenges by offering an easy-to-use interface for both passengers and administrators.

The system is built using Java with the Eclipse IDE, leveraging MySQL for database management to store information about buses, schedules, bookings, and users. The front-end interface is created using HTML, CSS, Bootstrap, and JavaScript, ensuring a responsive and visually appealing design that provides a seamless experience across devices.

For passengers, the system offers a simple, user-friendly platform where they can browse available buses, view schedules, and make reservations. The system allows passengers to check the availability of buses on specific dates and book seats with just a few clicks, ensuring an efficient and quick process. The online platform eliminates the need for manual intervention, reducing the time spent on booking tickets and providing immediate confirmation of reservations.

From the administrative perspective, the system provides tools to manage bus schedules, monitor bookings, and respond to customer queries. Administrators can easily update bus details, adjust seat availability, and track the status of reservations. This reduces the workload of customer service teams and minimizes human errors, ensuring a smooth and efficient operation.

This system also addresses operational challenges faced by bus service providers by offering real-time updates, automated ticketing, and efficient management of reservations. The integration of an online system enhances the reliability and accessibility of bus services, making it easier for both passengers and administrators to manage travel logistics. Overall, the Bus Reservation System not only improves the customer experience but also helps streamline operations, making it a valuable tool for transportation companies.

1.3 Purpose:

The purpose of the Bus Reservation System is to automate the bus ticket booking process, making it convenient for passengers to reserve seats online, view schedules, and manage bookings. Additionally, the system helps bus operators manage bus schedules, track reservations, and handle customer queries more efficiently. The project aims to eliminate manual booking, reduce human errors, and provide users with real-time data about available buses and routes. By using modern technologies like Java, MySQL, and responsive web design, the system ensures ease of use and accessibility for both passengers and administrators.

1.4 Scope of the Project:

The Bus Reservation Ticket Booking System is designed to address the challenges of manual ticketing by offering a comprehensive and automated solution. Built using Java Spring Boot and MySQL, the system ensures efficient management of schedules, reservations, and user data. Key features include real-time seat availability checks, secure payment options, and robust data validation. The system caters to passengers and administrators, supporting functions like ticket booking, schedule updates, and data analytics for business insights. Emphasizing user-friendliness and security, the project has a broad scope for enhancing bus reservation services.

1.5 Software Requirement Specification:

Introduction

The Bus Reservation Ticket Booking System is designed to manage all aspects of bus ticket reservations, including bus schedules, seat allocations, and passenger details. It serves as an automated alternative to traditional manual reservation processes, enhancing efficiency in transportation management.

Document Purpose

This SRS document outlines the software requirements for the Bus Reservation Ticket Booking System, including design decisions, architectural design, and detailed implementation requirements. It provides a comprehensive overview of the system's functionality and technical specifications for developers and stakeholders.

Product Scope

The Bus Reservation Ticket Booking System is developed for widespread use in bus transport services, aiming to replace outdated paper-based systems. It streamlines the reservation process, enabling passengers to book tickets online while allowing administrators to manage schedules, reservations, and data effectively. The system's versatility supports flexible scheduling and ticketing mechanisms, enhancing the overall user experience.

Definitions, Acronyms, and Abbreviations

- BRTBS: Bus Reservation Ticket Booking System
- SRS: Software Requirements Specification

References and Acknowledgment

[1] JAVA SPRING BOOT

[2] MYSQL

Overall Description

The Bus Reservation Ticket Booking System provides authorized users with seamless access to bus ticket records, streamlining the reservation process for bus operators and passengers worldwide. It simplifies operations in various transportation settings by providing real-time booking and seat availability updates.

Product Perspective

The system uses a client/server architecture and is compatible with major operating systems like Windows and Linux. The backend employs MySQL for database management, while the frontend leverages HTML and CSS for a responsive and user-friendly interface. The application is developed using Java Spring Boot and JDBC, ensuring robust database connectivity and secure transactions.

Product Functionality

The Bus Reservation Ticket Booking System supports the following core functionalities:

- Admin Registration: Allows new administrators to register and access the system.
- Admin Login: Provides secure login for authorized administrators.
- Add Bus Schedule: Facilitates the addition of new bus schedules, including route details and timings.
- View Schedule: Enables users to view and update existing schedules.
- Delete Schedule: Allows administrators to remove outdated schedules.
- Seat Reservation: Supports seat selection and reservation for passengers.
- Update Reservation: Enables modifications to existing reservations.
- Cancel Reservation: Provides the option to cancel bookings.

User and Characteristics

- Qualification: Basic educational qualifications (e.g., matriculation) and familiarity with English.
- Experience: General knowledge of online booking processes is beneficial.
- Technical Experience: Basic computer literacy is sufficient for interacting with the system.

Operating Environment

- Hardware Requirements:
  - Processor: i3 or above
  - RAM: 4GB
  - Storage: 500GB
  - Operating System: Windows 8/10/11
- Software Requirements:
  - Database: MySQL
  - Backend Technology: Java Spring Boot, JDBC
  - Frontend: HTML, CSS
  - IDE: Eclipse

Constraints

- Access to the system is restricted to registered administrators.
- Deletion of schedules or reservations is irreversible and must be handled cautiously.
- Internet connectivity is required for real-time operations.

Assumptions and Dependencies

- Administrators will securely share login credentials with authorized personnel only.
- The system depends on MySQL database connectivity and Java runtime environment.

Specific Requirements

User Interface

The Bus Reservation Ticket Booking System provides intuitive, menu-driven interfaces for:

- Admin Registration: Adding new administrators.
- Admin Login: Logging in securely to manage schedules and reservations.
- Add Schedule: Inputting new bus schedules.
- View Schedule: Viewing and editing existing schedules.
- Delete Schedule: Removing outdated schedules.
- Seat Reservation: Allowing passengers to select and book seats.
- Update Reservation: Modifying existing reservations.
- Cancel Reservation: Canceling specific reservations.

Hardware Interface

- Minimum screen resolution: 1024 x 768 pixels.
- Compatible with any version of Windows or Linux.

Functional Requirements

1. Login Module (LM):
   - Admins log in securely with a username and password.
   - Passwords are masked to ensure confidentiality.
   - Successful login requires validation through the database.
2. Bus Schedule Management Module (BSMM):
   - Admins can add, view, update, or delete bus schedules.
   - Real-time synchronization with the database ensures accurate data updates.
3. Reservation Module (RM):
   - Passengers can view seat availability, select seats, and book tickets.
   - Admins can manage reservations, including updates and cancellations.
4. Database Management Module (DMM):
   - Facilitates secure and efficient data storage for schedules, reservations, and user information.
   - Ensures integrity and consistency of stored data.

Non-functional Requirements

1. Performance:
   a. System should respond within 2 seconds for seat selection and booking confirmation.
2. Reliability:
   a. In the event of a failure, operations must resume within 5 minutes to ensure minimal disruption.
3. Availability:
   a. The system must be available 24/7 to support continuous reservations.
4. Security:
   a. User data, including payment information, must be encrypted.
   b. Unauthorized access to admin functionalities must be prevented.
5. Maintainability:
   a. Clear documentation and well-structured code must allow for future system enhancements and debugging.

2.System Flow Diagram

2.1 Use Case Diagram

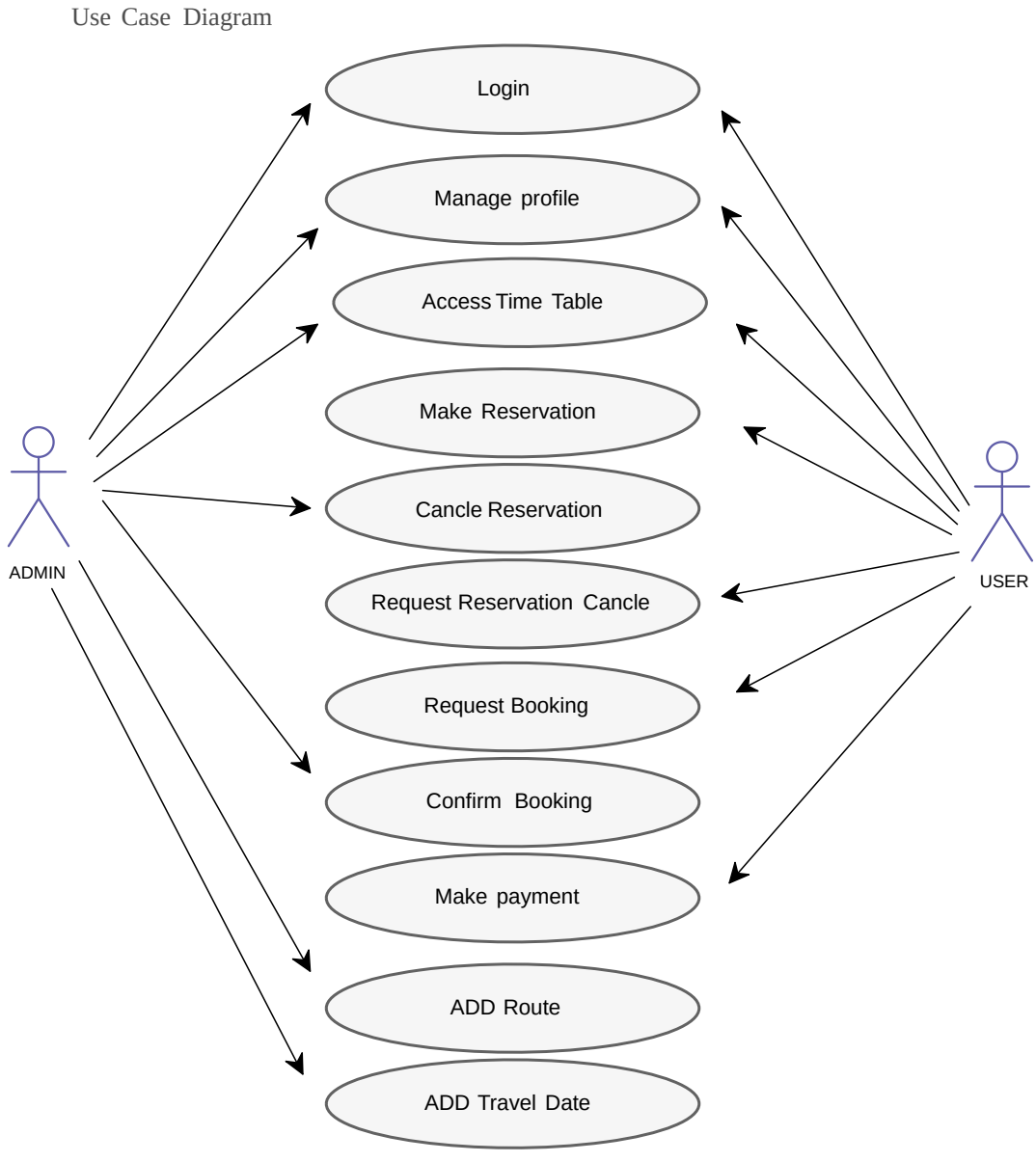

fig:2.1 Use case Diagram

2.2 Entity Relationship Diagram

Here is an Entity-Relationship Diagram (ERD) for the Bus Reservation Ticket Booking System:

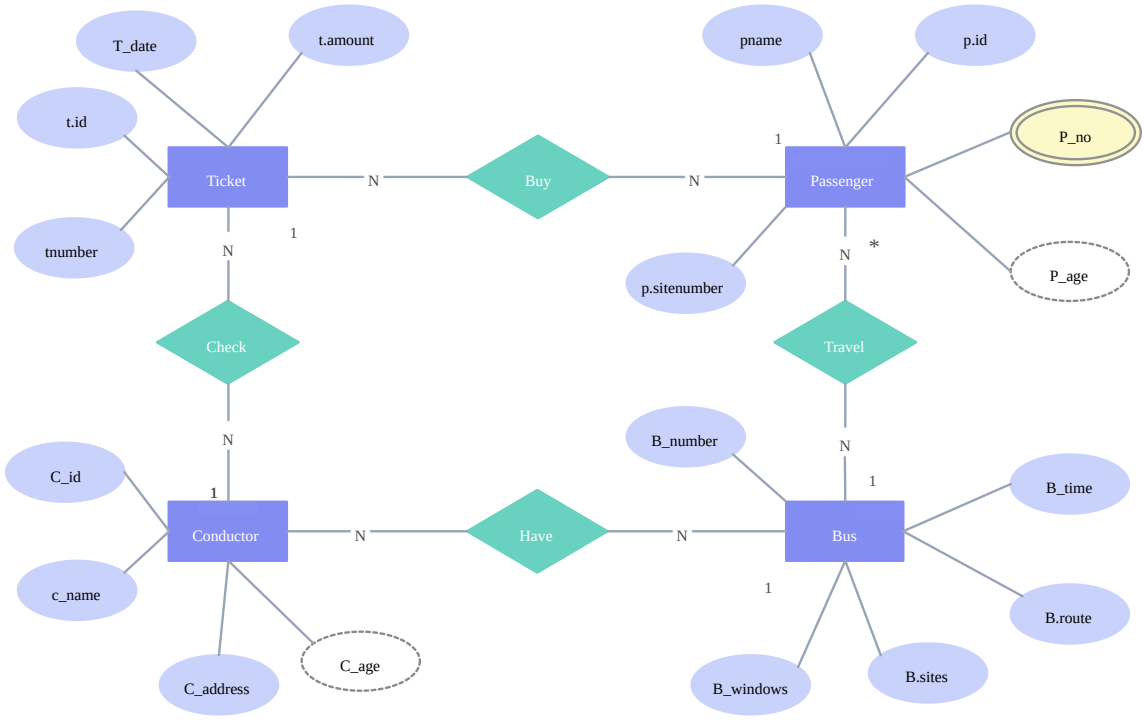

fig:2.2 Entity Relationship Diagram

2.3 Data Flow Diagram

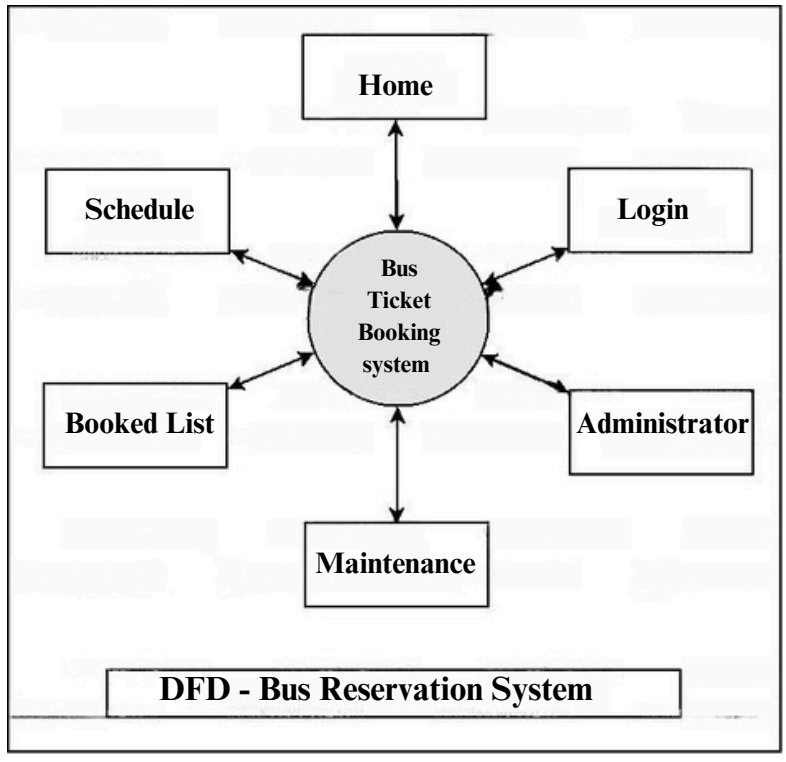

fig:2.3 Data Flow Diagram

3.Module Description

Admin Module

Register:

Admins can register themselves with a unique username and password to gain system access.

Login:

Admins log in using their registered username and password.

After Login:

Add Bus:

- Admins can add details about new buses, including the bus name, type (AC/Non-AC), seat capacity, and registration number.

View Bus:

- Admins can view existing bus details and update information such as bus type or seat capacity.

Delete Bus:

- Admins can remove buses from the system, ensuring outdated or incorrect information is deleted.

Add Schedule:

- Admins can create bus schedules, specifying routes, departure and arrival times, and dates for a particular bus.

View Schedule:

- Admins can review existing schedules and update information such as route details or timing.

Delete Schedule:

- Admins can remove outdated schedules from the system.

Add Reservation*:*

- Admins can create reservations manually by specifying passenger details, bus schedule, and seat number.

Update Reservation:

- Admins can modify reservation details such as the passenger's name, seat allocation, or travel date.

Cancel Reservation:

- Admins can cancel existing reservations, freeing up the allocated seats.

Remove Admin:

- Admins with the necessary permissions can delete admin accounts if needed for security or organizational purposes.

Passenger Module

View Buses:

- Passengers can browse available buses, including details like bus type and seat availability.

Search Schedule:

- Passengers can search for schedules based on routes, dates, or departure times.

Book Ticket:

- Passengers can book tickets by selecting a schedule, choosing their desired seat, and providing payment details.

View Booking History:

- Passengers can view their past reservations, including travel dates, times, and seat numbers.

Cancel Booking:

- Passengers can cancel existing bookings and receive applicable refunds.

Schedule Module

Overview:

Maintains and manages the bus schedules.

Features:

1. Admins can define schedules with routes, departure, and arrival times.
2. Passengers can view schedules based on routes or dates.
3. Ensure no overlapping or conflicting schedules for a single bus.

4.DESIGN:

4.1 Design:

User Page

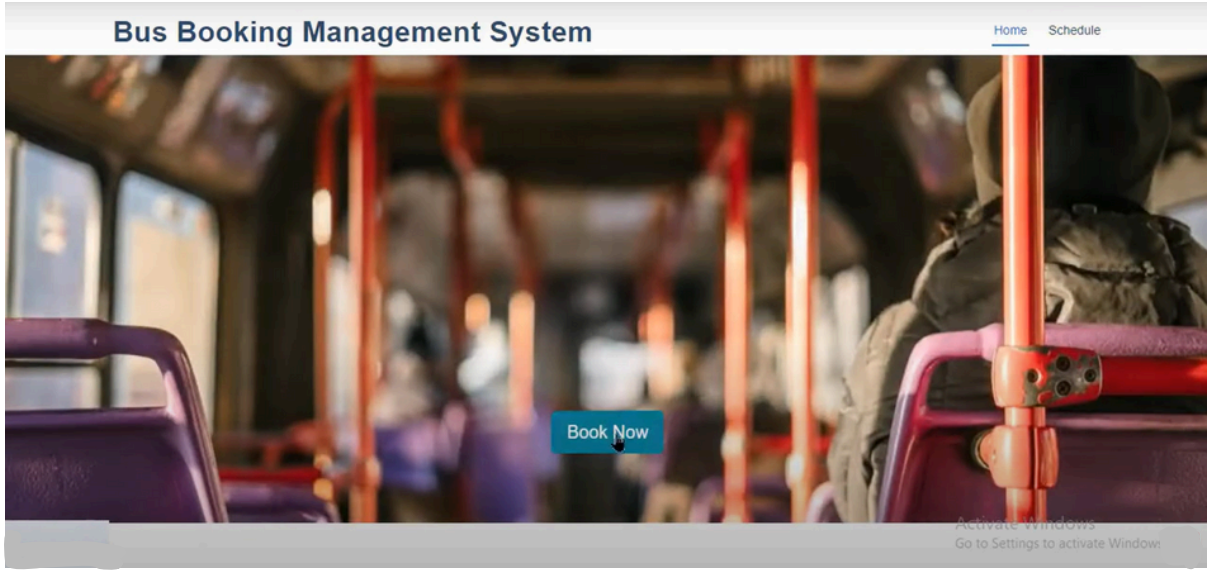

fig:4.1.1 User Page

Booking Page

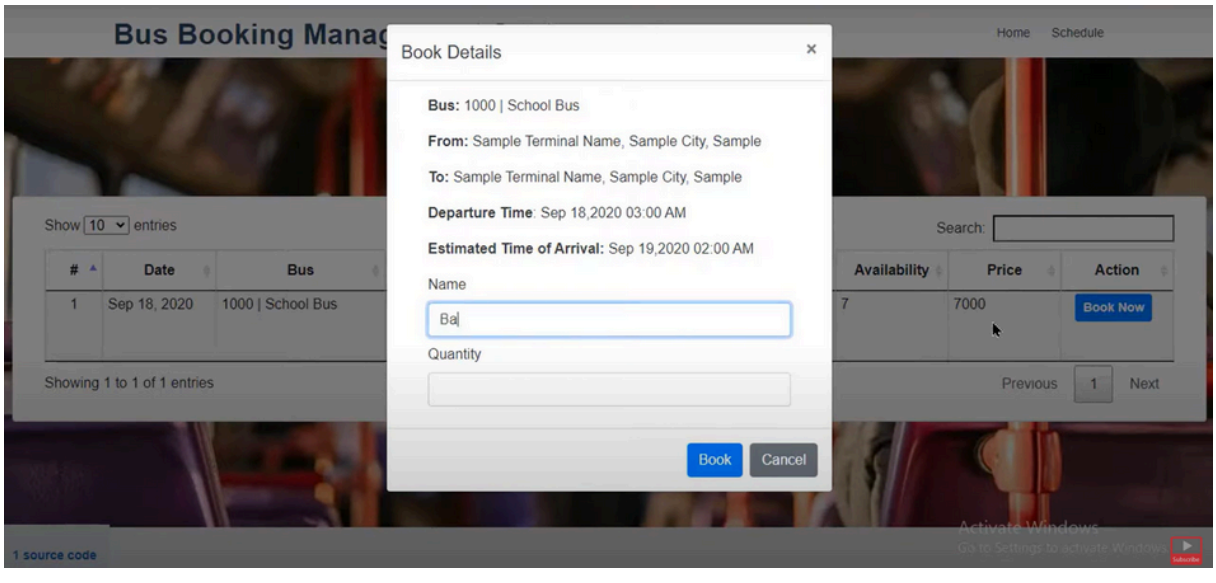

fig:4.1.2 Booking Page

Admin Page:

Login page:

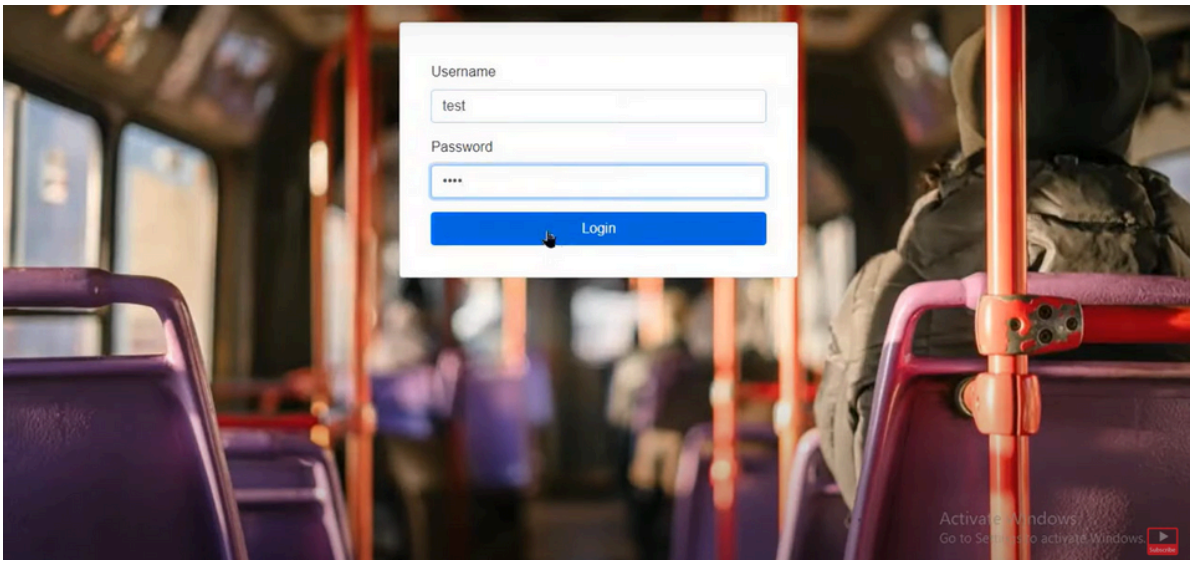

fig:4.1.3 Login Page

Schedule:

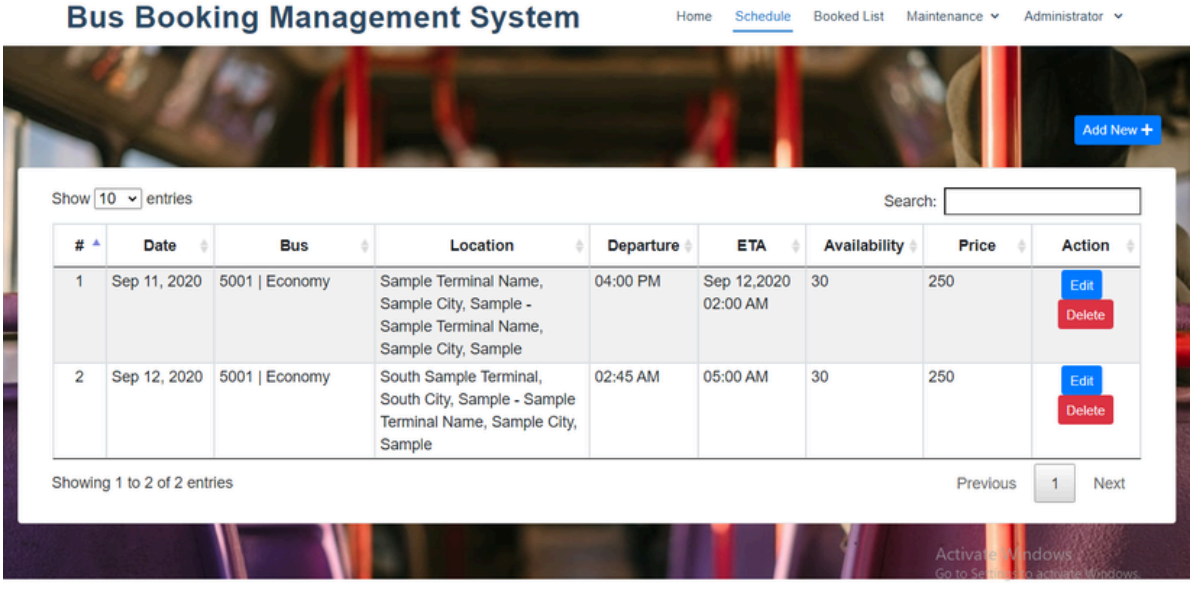

fig:4.1.4 Schedule

Booked List:

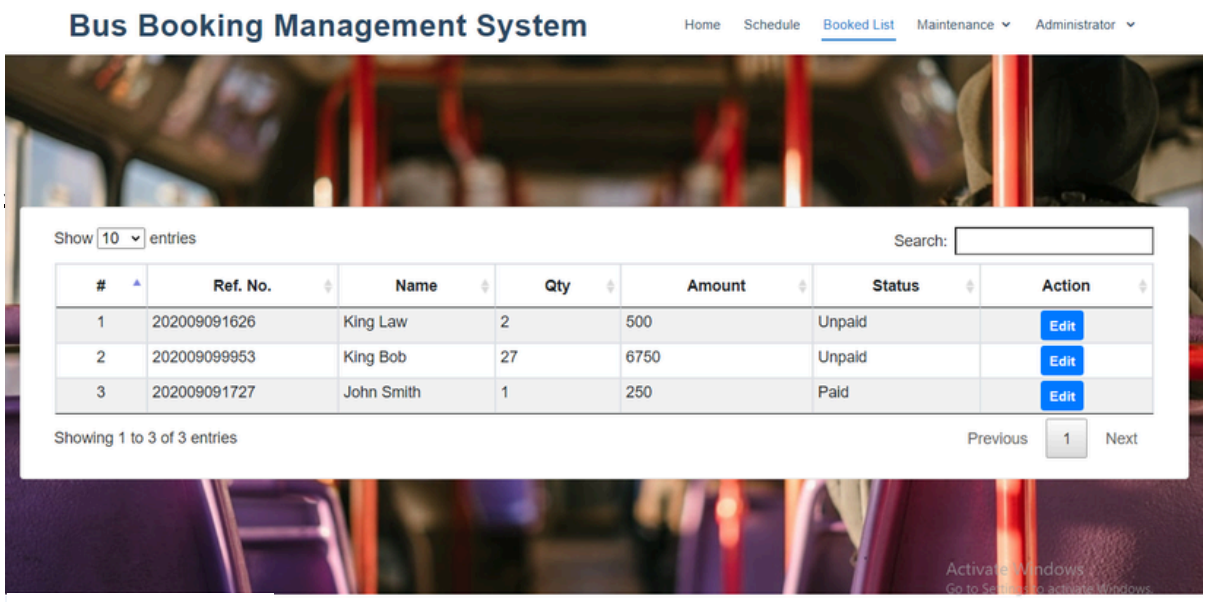

fig:4.1.5 Booked List

Add Schedule:

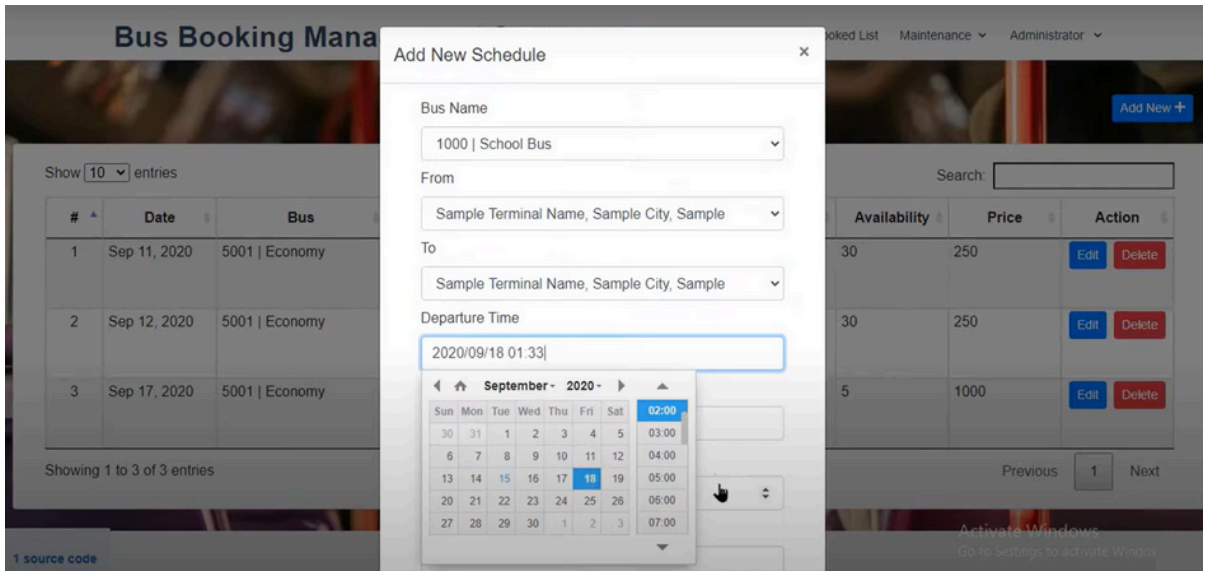

fig:4.1.6 Add Schedule

4.2 Database Design:

The database design for the Bus Reservation Ticket Booking System focuses on efficient data storage and retrieval using MySQL. The database stores essential data related to users, buses, schedules, reservations, and payments. At the analysis stage, key data elements such as user information, bus schedules, seat availability, and booking details were identified and organized.

Normalization techniques are applied to minimize data redundancy, ensuring data consistency and optimizing the database for quick, efficient retrieval and updates. Redundant data is eliminated, relationships between entities are defined, and internal data consistency is maintained to avoid inconsistencies during updates.

The system's relational design ensures that data can be accessed easily, with clear connections between tables like Users, Buses, Reservations, and Payments. This allows for efficient management of bus schedules, passenger bookings, and payments while maintaining minimal storage and improving data retrieval speeds. MySQL's flexibility ensures a scalable solution for future system enhancements.

MySQL Tables:

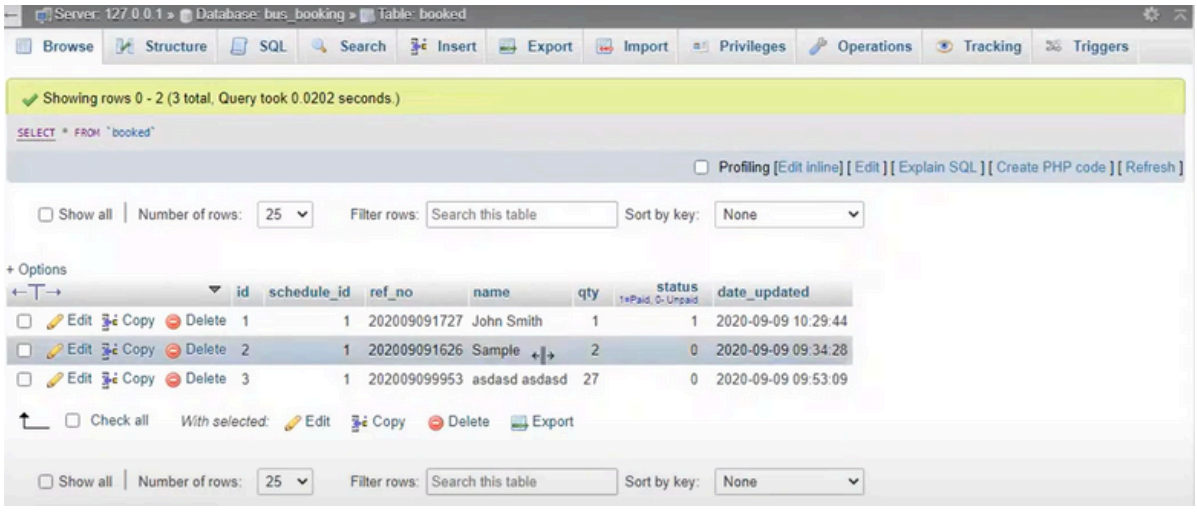

fig:4.2.1 User Table

Location Table

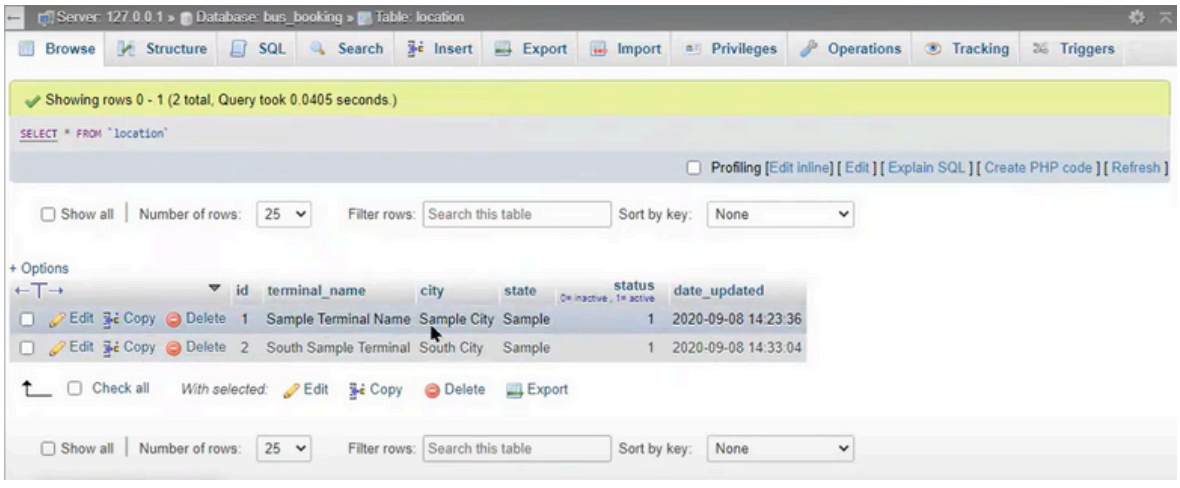

fig:4.2.2 Location Table

Admin Table:

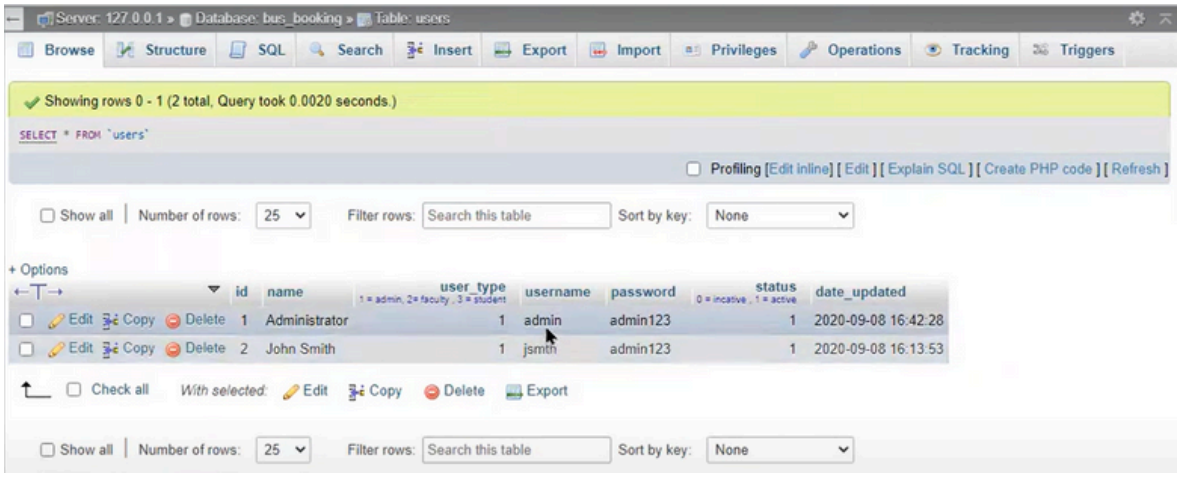

fig:4.2.3 Admin Table

4.3 Implementation (Code):

```
import java.sql.*;

public class BusReservationSystem {

 // Database credentials
 static final String DB_URL = "jdbc:mysql://localhost:3306/bus_reservation";
 static final String USER = "root"; // Replace with your MySQL username
 static final String PASS = "password"; // Replace with your MySQL password

 public static void main(String[] args) {
 Connection conn = null;
 Statement stmt = null;

 try {
 // Step 1: Register JDBC driver
 Class.forName("com.mysql.cj.jdbc.Driver");

 // Step 2: Open a connection
 System.out.println("Connecting to the database...");
 conn = DriverManager.getConnection(DB_URL, USER, PASS);

 // Step 3: Execute SQL queries
 stmt = conn.createStatement();

 // Create a table
 String createTableSQL = "CREATE TABLE IF NOT EXISTS BusSchedule (" +
 "id INT AUTO_INCREMENT PRIMARY KEY, " +
 "bus_name VARCHAR(50), " +
 "route VARCHAR(100), " +
 "departure_time TIME, " +
```

```
"arrival_time TIME" +
")";
stmt.executeUpdate(createTableSQL);
System.out.println("Table 'BusSchedule' created successfully.");

// Insert data
String insertSQL = "INSERT INTO BusSchedule (bus_name, route, departure_time, arrival_time) "
+
"VALUES ('Express Bus', 'City A to City B', '10:00:00', '14:00:00')";
stmt.executeUpdate(insertSQL);
System.out.println("Data inserted successfully.");

// Retrieve data
String selectSQL = "SELECT * FROM BusSchedule";
ResultSet rs = stmt.executeQuery(selectSQL);

// Display retrieved data
System.out.println("Bus Schedule:");
while (rs.next()) {
int id = rs.getInt("id");
String busName = rs.getString("bus_name");
String route = rs.getString("route");
Time departureTime = rs.getTime("departure_time");
Time arrivalTime = rs.getTime("arrival_time");

System.out.printf("ID: %d, Bus: %s, Route: %s, Departure: %s, Arrival: %s%n",
id, busName, route, departureTime, arrivalTime);
}

// Step 4: Clean-up environment
rs.close();
```

```
            stmt.close();
            conn.close();
        } catch (SQLException se) {
            se.printStackTrace();
        } catch (Exception e) {
            e.printStackTrace();
        } finally {
            // Close resources
            try {
                if (stmt != null) stmt.close();
                if (conn != null) conn.close();
            } catch (SQLException se) {
                se.printStackTrace();
            }
        }
        System.out.println("Goodbye!");
    }
}
```

5.CONCLUSION:

The Bus Reservation Ticket Booking System simplifies and automates bus ticket booking and management, providing a user-friendly interface built with Java Spring Boot, HTML, CSS, MySQL, and JDBC. It allows administrators to manage bus schedules, reservations, and customer data efficiently while ensuring data consistency and minimizing redundancy through a robust database design.

This system reduces manual errors, enhances accuracy, and saves time for users and administrators alike. It serves as a scalable and reliable solution, with the potential for future enhancements like online payment integration and mobile compatibility, making it an effective tool for the transportation sector.

6.REFERENCE LINKS:

- https://www.javatpoint.com/java-swing
- https://www.javatpoint.com/sql-tutorial