Software Requirements Specification

for

Railway Management System

Version 1.0

Prepared by

Prashast Saxena 220911536

Akash Gupta 220911400

Arnav Aradhya 220911612

Manipal Institute of Technology

30th March 2024

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

This document outlines the specifications and requirements for the development of a Railway Reservation System. With the growing population and the relatively slow growth of infrastructure, there is a need for a robust system to manage the increasing demand for railway reservations. The system will consist of two levels: Administrator and User, each with specific rights and functionalities.

1.1 Purpose

The purpose of this document is to define the specifications and functionalities of the Railway Reservation System. It aims to provide a clear understanding of the system's capabilities for both Administrators and Users. The system will facilitate the efficient management of train schedules, bookings, passenger information, ticket reservations, cancellations, and refunds.

1.2 Document Conventions

- Administrator: Refers to the system administrator with elevated privileges to manage the database, train information, and bookings.
- User: Refers to individuals using the system to search for trains, make reservations, and manage their bookings.
- PNR: Stands for Passenger Name Record, a unique identifier for each booked ticket.
- WAMP: Stands for Windows, Apache, MySQL, and PHP, the technologies used for developing the system.
- PHP: Stands for Hypertext Preprocessor, the scripting language used for server-side programming.
- MySQL: The relational database management system used for storing and retrieving data.

1.3 Intended Audience and Reading Suggestions

This document is intended for:

Developers tasked with designing and implementing the Railway Reservation System.
Project Managers oversee the development and deployment of the system.
Stakeholders involved in the railway industry or interested in the functionality of the reservation system.
Readers are encouraged to review the entire document to gain a comprehensive understanding of the system's requirements, functionalities, and limitations.

1.4 Product Scope

The Railway Reservation System will provide the following functionalities:

- Administrator Rights:

- ➢ Add trains to the database with information such as routes, classes, fares, availability, and departure dates.
- ➢ Store and retrieve booking information.
- ➢ Manage passenger records and schedule their journeys.
- ➢ Maintain records of passengers traveling on different trains to various destinations.

- User Rights:

- ➢ View all trains available in the system.
- ➢ Book tickets based on boarding and destination points.
- ➢ Receive confirmed tickets with essential details such as PNR, passenger names, ages, and travel dates.
- ➢ Apply for concessions based on age criteria.
- ➢ Cancel booked tickets and claim refund amounts.

1.5 References:

The development of the Railway Reservation System will involve the following software and technologies:

- WAMP Server v2.5: The software stack consisting of Windows, Apache, MySQL, and PHP for local development and testing.
- PHP: The server-side scripting language used for implementing dynamic web pages and functionalities. https://www.w3schools.com/php/
- MySQL: The relational database management system for storing and managing the system's data. https://hevodata.com/learn/xampp-mysql/
- HTML/CSS/JS: https://www.w3schools.com/html/
- Er-diagram: https://boardmix.com/app/editor/cYMg831htsu4bLmaPn9kIw.

2. Overall Description

The Railway Management System (RMS) is a web-based application designed to support the management of train schedules, bookings, and passenger information. It operates with a 2-level system, distinguishing between Administrators and Users. The system aims to streamline the process of ticket reservations, ensuring efficient handling of the growing demand for railway services.

2.1 Product Perspective

The RMS operates within the context of a larger railway infrastructure, providing a user-friendly interface for both Administrators and Users. It interfaces with a WAMP server (v2.5), utilizing PHP for interactions and MySQL for database management. This system is designed to enhance the efficiency and accessibility of railway reservations, integrating seamlessly with existing railway operations.

2.2 Product Functions

- Administrator Rights

- Addition of trains to the database, including information such as train details, routes, class options, fares, availability, and departure dates.
- Storage and retrieval of booking information.
- Management of passenger records for optimal scheduling.
- Maintenance of comprehensive passenger travel history across trains and destinations.

- User Rights

- View available trains in the database.
- Book tickets based on boarding and destination points.
- Receive confirmed tickets with essential details (e.g., PNR, passenger names, ages, travel dates).
- Options for concessions based on age criteria.
- Ability to cancel tickets and claim refund amounts.

2.3 User Classes and Characteristics

- Administrators

- Authorized personnel responsible for system management.
- Add trains and maintain database integrity.
- Access booking and passenger information for administrative purposes.
- Cannot delete or modify existing data.

- Users

- General public utilizing the system for ticket reservations.
- View available trains and book tickets.
- Receive confirmed tickets with relevant details.
- Can cancel tickets and claim refunds.
- Limited to one booking per journey.
- Cancellation affects the entire PNR.

2.4 Operating Environment

The RMS operates within a web-based environment, accessible through any device with internet connectivity. Users interact with the system through a web browser, while administrators have additional backend access for data management. The system is hosted on a WAMP server (v2.5), utilizing PHP for frontend interactions and MySQL for database management.

2.5 Design and Implementation Constraints

- Hardware limitations: Memory and processing limitations for the database and programs. The data handled is relatively small compared to real world scenarios.
- Other limitations: Lack of software that tracks the exact location of the parcel.

2.6 User Documentation

Comprehensive user documentation will be provided for both Administrators and Users. This documentation will include:

System overview and functionality.
Step-by-step guides for train addition, ticket booking, cancellation, and refund processes.
Troubleshooting tips and frequently asked questions.

2.7 Assumptions and Dependencies

- Assumptions

- ➢ Users have basic internet literacy and can navigate web-based interfaces.
- ➢ Availability of consistent internet connectivity for accessing the system.
- ➢ Users and administrators have valid accounts to access the system.
- ➢ The WAMP server (v2.5) and associated software (PHP, MySQL) are installed and configured correctly.
- ➢ Users provide accurate and valid information during the booking process.

- Dependencies

- ➢ Reliability of the WAMP server (v2.5) and associated software for system functionality.
- ➢ Continued support and maintenance of the web-based application.
- ➢ Timely updates to train schedules and availability in the database.
- ➢ Adherence to railway regulations and policies regarding ticket reservations and cancellations.

3. External Interface Requirements

3.1 User Interfaces

The user interfaces for the Railway Reservation System will be designed to be intuitive and user-friendly for both administrators and regular users.

- Administrator Interface:

- Login Page: Allows the administrator to enter their credentials to access the system.
- Dashboard: Displays summary information such as total trains, bookings, and recent activities.
- Add Train Form: Form to input train details including route, number of classes, fares, availability, and departure dates.
- View Bookings: Interface to view all bookings made, including passenger details and journey information.

- User Interface:

- Home Page: Provides options to search for trains, book tickets, view bookings, and cancel tickets.
- Train Search: Allows users to search for trains based on boarding point, destination, and date.
- Booking Form: Form to input passenger details, select class, and make payment for booking.
- Booking Confirmation: Displays confirmed ticket details with PNR, passenger names, age, travel date, and fare.
- Cancellation Form: Allows users to cancel their bookings and claim refunds.

3.2 Hardware Interfaces

The Railway Reservation System will require standard hardware components to operate smoothly. The hardware requirements include:

Server: Hosting the WAMP server v2.5 for the backend operations and database management.
Client Devices: Any device with a web browser, such as laptops, desktops, tablets, or smartphones, for users and administrators to access the system.
Internet Connection: Reliable internet connectivity is necessary for users to interact with the system.

3.3 Software Interfaces

The system will interact with various software components to perform its functions efficiently. The software interfaces include:

WAMP Server v2.5: Providing the server environment for hosting the system.
PHP: Interfacing layer for dynamic web content generation and backend logic.
MySQL Database: Database management system for storing and retrieving train, booking, and user information.
HTML/CSS/JavaScript: Front-end technologies for designing user interfaces and client-side interactions.

3.4 Communications Interfaces

The Railway Reservation System will rely on communication interfaces to facilitate data exchange and transactions.

HTTP/HTTPS Protocols: Used for communication between the client's web browser and the system's server, ensuring secure data transfer.
Email Notifications: Sending confirmation emails for successful bookings, cancellation confirmations, and other important updates to users.
Payment Gateway: Integration with a secure payment gateway for processing online payments for ticket bookings.
APIs (if applicable): Integration with external services such as SMS gateways for sending booking details via text messages.

4. System Features

4.1 Train Management

- System Feature 1: Add new trains to the database with relevant information.
  The administrator should be able to input details such as train name, route, number of classes, fares for each class, availability of tickets, departure date, and other necessary information.

- System Feature 2: View existing trains in the database.
  Users should be able to see the list of available trains along with their respective details like route, classes, and fares.

4.2 Booking Tickets

- System Feature 3: Book a ticket based on boarding and destination points.
  Users should be able to select a train, specify boarding point, destination point, class, number of passengers, and make a booking.

- System Feature 4: Generate confirmed tickets with essential details.
  Upon successful booking, users should receive a confirmed ticket with details such as PNR (Passenger Name Record), passenger names, ages, travel date, class, etc.

- System Feature 5: Apply for concessions based on age criteria.
  Users falling under specific age categories should have the option to apply for fare concessions if applicable.

4.3 Ticket Cancellation

- System Feature 6: Cancel a booked ticket and claim refund.
  Users should be able to cancel their booked tickets, which results in the cancellation of the entire PNR, and receive a refund based on the cancellation rules.

4.4 Passenger Management

- System Feature 7: Maintain records of all passengers and bookings.
  The system should keep track of passengers' details, their bookings, travel history, and assigned PNRs.

- System Feature 8: Schedule passenger journeys accordingly.
  The system should allocate passengers to appropriate trains, classes, and seats based on their bookings and availability.

4.5 Constraints and Limitations

- System Feature 9: Enforce constraints on train management.
  Only the addition of trains is allowed; deletion or modification is not permitted.

- System Feature 10: Enforce constraints on user bookings.
  Each user is allowed only one booking per journey.
  Cancellation of a ticket results in the cancellation of the entire PNR.
  All passengers on a PNR must travel in the same class.

4.6 User Interface

- System Feature 11: User-friendly interface for both administrators and users.
  Design an intuitive interface for administrators to add trains and view bookings.
  Provide a user-friendly booking interface for users to search, book, and manage their tickets.

- System Feature 12: Error handling and validation.
  Implement proper error messages and validation checks to ensure data integrity and user input accuracy.

4.7 Reporting and Record-keeping

- System Feature 13: Store and retrieve booking information.
  Enable the system to store and retrieve details of all bookings done till date for reference and reporting purposes.

- System Feature 14: Generate reports on passenger travel.
  Provide options for administrators to generate reports on passengers traveling in different trains, on various dates, and reaching different destinations.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

- Response Time:

  The system should respond to user actions (like booking, canceling, viewing trains) within 3 seconds on average.
  Database queries should be executed within 2 seconds to ensure smooth user experience.

- Scalability:

  The system should handle a load of up to 1000 simultaneous users without significant performance degradation.
  It should be capable of managing data for at least 10,000 bookings per day.

- Reliability:

  The system should have an uptime of at least 99% to ensure availability during peak booking hours.
  Database backups should be scheduled daily to prevent data loss.

5.2 Safety Requirements

- Data Integrity:

  All data entered by users or administrators should be validated to prevent any corruption of the database.
  Regular data integrity checks should be performed to ensure accuracy.

- System Stability:

  The system should handle errors gracefully, displaying user-friendly messages and preventing crashes.
  Proper error logging mechanisms should be in place for easy troubleshooting.

5.3 Security Requirements

- Authentication:

  User login credentials should be securely stored using encryption.
  Administrators should have unique, strong passwords and multi-factor authentication.

- Data Privacy:

Personal passenger information (like names, ages, PNR details) should be stored securely and accessed only on a need-to-know basis.
Encryption should be used for sensitive data transmission, such as during booking and cancellation.

- Protection Against Attacks:

  The system should have measures against SQL injection attacks and other common web vulnerabilities.
  Regular security audits should be conducted to identify and patch any potential security loopholes.

5.4 Software Quality Attributes

- Maintainability:

  Code should be well-documented and follow standard coding practices for easy maintenance.
  Regular code reviews should be conducted to ensure quality and consistency.

- Usability:

  The user interface should be intuitive and easy to navigate, even for users unfamiliar with the system.
  Proper error messages and guidance should be provided to assist users during the booking and cancellation process.

- Portability:

  The system should be platform-independent, capable of running on different web browsers without issues.
  It should support multiple devices, including desktops, laptops, and mobile phones.

5.5 Business Rules

- Booking Limits:

  Users are allowed to make only one booking per journey to prevent hoarding tickets.
  Each booking should be for the same class for all passengers on the PNR.

- Concession Criteria:

  Users eligible for age-based concessions should provide valid proof during ticket booking.
  Concessions should be applied automatically based on the age criteria specified.

- Cancellation Policy:

Cancellation of a ticket should result in a refund according to predefined rules and timelines. Cancellations should be allowed up to a certain period before the departure date.

- Train Information Updates:

  Administrators should update train schedules promptly to reflect changes in departure dates, availability, or routes.
  Users should have access to the most recent and accurate information when booking tickets.

- Data Retention:

  Booking history and passenger records should be maintained for a specified period for auditing and reference purposes.
  Archived data should be stored securely and made accessible only to authorized personnel.

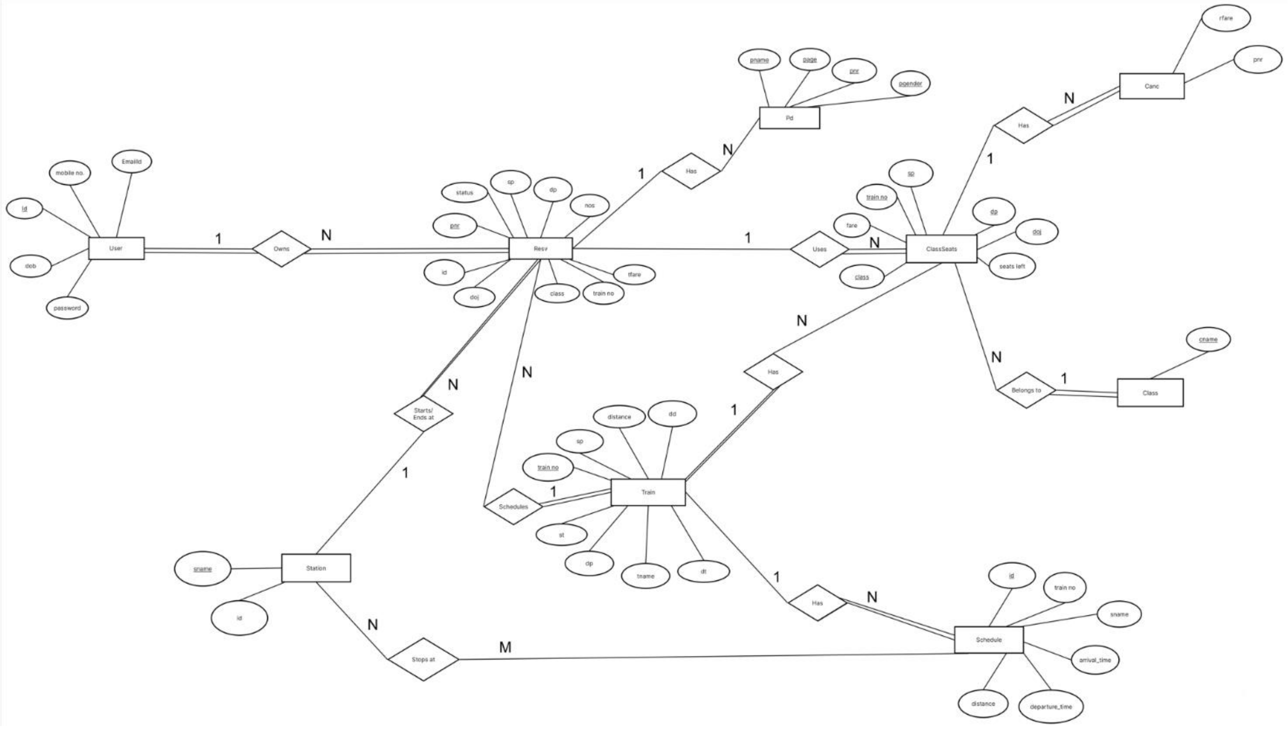
6. Other Requirements: TBD

Appendix A: Glossary

- Administrator: Refers to the system administrator with elevated privileges to manage the database, train information, and bookings.
- User: Refers to individuals using the system to search for trains, make reservations, and manage their bookings.
- PNR: Passenger Name Record, a unique identifier for each booked ticket.
- WAMP: Acronym for Windows, Apache, MySQL, and PHP, the technologies used for developing the system.
- PHP: Hypertext Preprocessor, the scripting language used for server-side programming.
- MySQL: The relational database management system used for storing and retrieving data.
- Railway Reservation System: The web-based application designed to manage train schedules, bookings, and passenger information.
- Train Management: The functionalities related to adding new trains, managing routes, classes, fares, availability, and departure dates.
- Booking Tickets: The process of users selecting trains, specifying boarding and destination points, classes, and making reservations.
- Cancellation: The process where users can cancel their booked tickets and claim refunds based on predefined rules.
- Passenger Management: The system's ability to maintain records of passengers, bookings, travel history, and assign Passenger Name Records (PNRs).
- Dashboard: A summary page displaying relevant information such as total trains, bookings, and recent activities for administrators.
- Booking Confirmation: The receipt provided to users upon successful booking, containing essential details such as PNR, passenger names, ages, travel dates, and fares.
- Concessions: Special discounts or fares applied based on age criteria for eligible users.
- User Interface: The visual and interactive elements are designed for users to interact with the system, including forms, search options, and booking interfaces.
- Database: A structured collection of data stored electronically, managed by the MySQL relational database management system in this system.
- Web-based Environment: The platform where users access the system through a web browser with internet connectivity.
- Server: The computing device hosting the WAMP server v2.5 for backend operations and database management.
- Client Devices: Devices such as laptops, desktops, tablets, or smartphones used by users and administrators to access the system.
- HTTP/HTTPS Protocols: Protocols used for secure communication between the client's web browser and the system's server.
- Payment Gateway: A secure service integrated into the system for processing online payments for ticket bookings.
- APIs: Application Programming Interfaces, used for integrating external services such as SMS gateways for sending booking details via text messages.

- Error Handling: The system's capability to manage errors gracefully, displaying user-friendly messages and preventing crashes.
- Data Integrity: Ensuring the accuracy and consistency of data entered into the system, preventing corruption of the database.
- Encryption: The process of converting data into a secure code to prevent unauthorized access, used for storing sensitive information and secure data transmission.
- SQL Injection: A common web vulnerability where malicious SQL code is inserted into input fields to manipulate the database.
- Code Maintenance: The process of keeping the system's code well-documented, following standard practices for easy updates and modifications.
- Platform Independence: The system's capability to run on different web browsers and devices without compatibility issues.
- Booking Limits: Rules specifying the number of bookings allowed per user per journey to prevent ticket hoarding.
- Concession Criteria: Requirements and conditions for applying age-based fare concessions to eligible users.
- Cancellation Policy: Rules and guidelines governing ticket cancellations, refunds, and timelines for users.
- Data Retention: The practice of storing and maintaining booking history and passenger records for auditing and reference purposes.
- Archived Data: Old or historical data stored securely and accessible only to authorized personnel for analysis or reference.

Appendix B: ER-Diagram


rfare
pnr
Canc
N
Has
1
pname
page
pnr
pgender
Pd
N
Has
1
mobile no.
EmailId
Id
User
dob
password
1
Owns
N
status
sp
dp
nos
pnr
Resv
id
doj
class
train no
tfare
1
Uses
N
sp
train no
fare
ClassSeats
class
dp
doj
seats left
N
Has
1
N
Belongs to
1
Class
cname
N
Starts/
Ends at
1
N
distance
dd
sp
train no
Train
1
Schedules
st
dp
tname
dt
sname
Station
id
1
Has
N
id
train no
sname
Schedule
arrival_time
distance
departure_time
N
Stops at
M