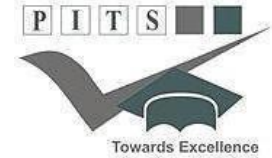




RAILWAY RESERVATION SYSTEM



A MINI PROJECT REPORT

Submitted by

SABARISH.K (821321104030)

ERSHAD AHAMED.U (821321104009)

SAM CHARLES.F (821321104032)

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE & ENGINEERING

PARISUTHAM INSTITUTE OF TECHNOLOGY AND SCIENCE,

THANJAVUR

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2024



ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report “**RAILWAY RESERVATION SYSTEM**” is the Bonafide work of “**ERSHAD AHAMED.U (821321104009), SABARISH.K (821321104030), SAM CHARLES.F (821321104032)**” who carried out the project work under my supervision.

SIGNATURE

Prof. Dr. R. Kavitha, B.E., MTech., Ph.D

HEAD OF THE DEPARTMENT

PROFESSOR

Department of computer Science
and Engineering,

Parisutham Institute of Technology and
Science, Thanjavur.

SIGNATURE

Ms. R. Vinodharasi, M.E.,

SUPERVISOR

ASSISTANT PROFESSOR

Department of computer Science
and Engineering,

Parisutham Institute of Technology
and Science, Thanjavur.

Submitted to the viva voce held on.....

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

At the outset, we present our humble prayer to **The Almighty** for giving us strength, courage, determination not only in pursuance of this task but also in every walk of life.

We express our sincere and affectionate thanks to our beloved Chairman **Mr. S. P. Anthonisamy, M.A., B.L.**, for allowing us to do the mini project work.

With immense pleasure and heartfelt gratitude, we express our sincere thanks to our Dean-Academics, **Prof. Dr. J. Nirmala, B.E., M.Tech., Ph.D** for her great support for mini our project.

We are much obliged and grateful to our Head of the Department and mini project Coordinator, **Prof. Dr. R. Kavitha, B.E., M.Tech., Ph.D** Professor, Department of Computer Science and Engineering, for her continuous support not only in this task but also in all our academic activities.

We would like to show our sincere gratitude to our mini project guide **Mrs. R. Vinodharasi, M.E.**, Assistant Professor, Department of Computer Science and Engineering, for her constant guidance and support all through the mini project work.

Last but not the least, we thank our beloved Parents, Faculty Members and Friends for their timely help and valuable suggestions.

ABSTRACT

The Railway Reservation System is an efficient platform that simplifies the process of booking train tickets, checking seat availability, and accessing essential train details. With a user-friendly interface, passengers can seamlessly search for train routes, select travel dates, and make reservations. The system allows users to choose their preferred seats during booking, provides real-time information on seat availability, and confirms bookings if seats are open. Additionally, passengers can access comprehensive train details such as schedules, stops, and routes. Whether planning a single journey or a return trip, this system enhances the overall passenger experience by streamlining the booking process and ensuring convenience.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	iv
	LIST OF FIGURES	vii
1	INTRODUCTION	1
	1.1 OVERVIEW	1
	1.2 OBJECTIVE	1
2	LITERATURE SURVEY	3
	2.1 ADVANCED TRAIN RESERVATION	3
	2.2 CONFIGURING AN APPLICATION WHICH ALLOWS ONLINE BOOKING	4
	2.3 RAILWAY E-VERIFICATION INFORMATION	5
3	SYSTEM ANALYSIS	6
	3.1 PROBLEM STATEMENT	6
	3.2 EXISTING SYSTEM	6
	3.2.1 DISADVANTAGES	6
	3.3 PROPOSED SYSTEM	7
	3.3.1 ADVANTAGES	7
4	SYSTEM SPECIFICATION	8
	4.1 HARDWARE REQUIREMENTS	8
	4.2 SOFTWARE REQUIREMENTS	8
	4.3 SOFTWARE DESCRIPTION	8
5	SYSTEM DESIGN	11
	5.1 ARCHITECTURE DIAGRAM	11
	5.2 DATA FLOW DIAGRAM	12
	5.3 UML DIAGRAM	13

6	SYSTEM IMPLEMENTATION	19
	6.1 MODULES	19
	6.2 MODULES DESCRIPTION	19
7	TESTING	22
	7.1 AUTOMATION TOOL	22
8	CONCLUSIONS	24
9	FUTURE ENHANCEMENT	25
10	APPENDICES	26
	10.1 SOURCE CODE	26
	10.2 SNAPSHOTS	35
11	REFERENCES	41

LIST OF FIGURES

FIGURE NO.	DIAGRAM	PAGE NO.
5.1	System Architecture	11
5.2	Data Flow Diagram	12
5.3	Usecase Diagram	13
5.4	Class Diagram	14
5.5	Sequence Diagram	15
5.6	Collaboration Diagram	16
5.7	Statechart Diagram	17
5.8	Activity Diagram	18
7.1	Input Validation Testing	23
7.2	Functionality Testing	23
10.1	Register New User	35
10.2	User Home	35
10.3	View Profile	36
10.4	Change Password	36
10.5	Search Train	37
10.6	Fare Enquiry	37
10.7	Fare Result	38
10.8	Seat Availability	38
10.9	Ticket Booking	39
10.10	Add Train	39
10.11	Cancel Train	40

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

The Railway Reservation System is a digital platform designed to streamline the process of booking train tickets, managing reservations, and facilitating a seamless travel experience for passengers. It serves as an efficient alternative to traditional ticket booking methods, offering convenience, accessibility, and real-time information to users. the Railway Reservation System offers a user-friendly, efficient, and reliable platform for booking train tickets and managing travel-related services, catering to the needs of modern-day travelers.

1.2 OBJECTIVES

The primary objective of this project is to enhance the existing Railway Reservation System to provide passengers with an improved and user-friendly experience. The focus will be on addressing the challenges related to seat selection during reservation and integrating a seamless food ordering system to streamline the entire travel process. Provide a foundation for the design and architecture of the enhanced Railway Reservation System, guiding developers in creating a solution that aligns with the specified requirements.

- **Ease of Booking:** The system should allow passengers to book tickets easily, either through online portals, mobile applications, or at railway station counters.
- **Availability Information:** Passengers should be able to check the availability of seats or berths on various trains for their desired routes and dates.
- **Seat/berth selection:** Passengers should have the option to select their preferred seats or berths based on availability and personal preferences.

- **Cancellation and Refund:** The system should facilitate the cancellation of tickets and provide refunds as per the cancellation policy, ensuring a hassle-free process for passengers.
- **Efficient Allocation:** The system should allocate seats or berths efficiently to maximize occupancy while ensuring passenger comfort and safety.
- **Integration with Other Systems:** Integration with other railway systems such as ticketing, scheduling, and payment systems to ensure seamless operations and data exchange.

CHAPTER 2

LITERATURE SURVEY

2.1

PAPER TITLE : Advanced Train Reservation
AUTHOR : Ms.B.T. Tharani Sri Sakthi
PUBLISHED ON : 2014

Firstly Railways can provide adequate and necessary basic facilities to the passengers like pure drinking water hygienic food, good sanitation , proper seating arrangement, electronic scrolling indicator inside the trains, electronic charging facility and proper enquiry facilities at all stations and in trains to enhance the convenience of the passengers thereby attracting more number of passengers. The majority of the passengers are having strong unfavorable opinion about the Railways regarding theft and robbery. Even in the ladies compartment, female passengers are of the opine that they are in secured and facing lot of problems. This arrangement will definitely boost up the image of the Railways in the minds of the female Firstly Railways can provided equate and necessary basic facilities to the passengers like pure drinking water hygienic food ,good sanitation, proper seating arrangement, electronic scrolling indicator inside the trains, electronic charging facility and proper enquiry facilities at all stations and in trains to enhance the convenience of the passengers thereby attracting more number of passengers. The majority of the passengers are having strong unfavorable opinion about the Railways regarding theft and robbery.

2.2

PAPER TITLE : Configuring an application which allows
online booking
AUTHOR : Romania
PUBLISHED ON : 2020

The necessity of such web application starts from the idea that, at the moment, in Romania there is no common information system, which allows the reservation and online purchase of travel tickets for rail and road transport. Moreover, at present, in Romania there is not even a common system for issuing travel tickets, which will connect all the rail passenger transport operators. However, in the near future, creation of this system becomes mandatory. Amendment of Law no. 202/20161, by transposing Directive (EU) 2016/23702, provides for the introduction of a common system of information on rail passenger transport and an integrated ticketing system, direct tickets and reservations. Participation in these systems is part of the public service obligation regarding rail passenger transport. Servations is manage by Romanian Railway Reform Authority, a public legal entity independent of any railway operator or of an association of all railway operators, which operates passenger rail services. The construction of these two systems, the one proposed by the theme of this paper, and the one that will be introduced to meet the requirements that have arisen as a result of the transposition of Directive (EU) 2016/2370, may represent the starting point for the creation of a common system for multimodal transport (road, rail, naval and air transport), integrated ticketing, direct tickets and reservations. Online booking and purchasing a travel ticket offers a lot of benefit not only for passengers, but also for passenger transport companies. For example, those who want to travel using several modes of transport will save time, being able to get their transport ticket from anywhere, using their mobile phone.

2.3

PAPER TITLE : Railway E-Verification Information
AUTHORS : Sourodeep Chatterjee, Soham Das.
PUBLISHED ON : 2020

The Indian Railway System is the World's largest one. The idea behind this paper is To improve the current system that exists in the Indian Railways. This paper includes various changes in many aspects of the system that exists currently. In the present, people riding the train do not have any control over the place of the seat that they will travel in. This paper provides dynamic seat allocation, a person can select where he/she wants to sit in a coach and his/her seat is allocated accordingly. Another aspect of the current system is to address the illegal transaction that the TTE does by assigning the seat to people without any proper going to the government. In our system, there will be a real-time charting system where the TTE will be provided with the passenger list of only the people boarding from the current station. This will be done by matching the PIN code of the current station and the Live PIN code fetched by GPS. The data for that full journey will be downloaded into the local the world is moving towards digitalization but the Indian Railway System still runs on the traditional pen and paper system. The TTE gets the chart of the passengers for every intermediate station on the very first station from where the train has started. Another issue with the existing system is that the seat layout is not provided to the passenger, they cannot book the seat as per their wish. Some aged people might face difficulty with upper berth which is randomly allocated to them. The problem with the system is that TTE abuses his power and illegally allocates the seat to passenger for which the government gets no revenue.

CHAPTER 3

SYSTEM ANALYSIS

3.1 PROBLEM STATEMENT

The existing railway reservation system faces various challenges and limitations, necessitating the development of an advanced and efficient Railway Reservation System. The current system is plagued by issues such as manual ticket booking, long queues, data inconsistencies, and limited accessibility. The current system has allotted our seats automatically so we recover this problem and book exact seats our wishes like a movie tickets. additional facility like current ticket booking system and order foods while traveling. Limited online reservation will be recovered.

3.2 EXISTING SYSTEM

The existing railway reservation system typically encompasses several components and functionalities to facilitate the booking and management of train tickets. Here's an overview of the components commonly found in an existing railway reservation system Online Booking Portal, Mobile Applications, Seat Selection, Cancellation and Refund, PNR Status Enquiry etc.

3.2.1 DISADVANTAGES

- Limited Accessibility
- Limited Booking Channels
- Manual Data Entry Errors
- Complex Refund Process
- Customer Service Challenges

3.3 PROPOSED SYSTEM

A proposed system for a railway reservation system aims to address the limitations and challenges of the existing system while enhancing the overall booking experience for passengers. Here's an outline of key features and improvements in the proposed system Improved Stability and Reliability, Enhanced User Interface, Dynamic Seat Allocation etc.

3.3.1 ADVANTAGES

- Convenience
- Real-Time Information
- Multiple Payment Options
- Automated Confirmation and E-Tickets
- Cancellation and Refund

CHAPTER 4

SYSTEM SPECIFICATION

4.1 HARDWARE REQUIREMENTS

- RAM : 4GB
- Processor : intel CORE i5

4.2 SOFTWARE REQUIREMENTS

- Operating System : Windows 11
- Frontend : HTML
- Backend : Java
- Framework : Eclipse EE
- Database : MySQL

4.3 SOFTWARE DESCRIPTION

4.3.1 OPERATING SYSTEM: WINDOWS 11

Windows 11 is the latest version of Microsoft's operating system, succeeding Windows 10. It brings several new features, improvements, and a refreshed user interface. Windows 11 introduces a centered Start Menu, with app icons aligned in a grid and easy access to recent files and recommended apps. The taskbar is now centered by default and features new animations and effects for a more modern look. Enhanced window management with Snap Layouts and Snap Groups, allowing users to easily organize and arrange multiple windows on the screen. The Microsoft Store has been redesigned with a focus on better curation, performance, and compatibility. It now supports a wider range of apps, including Android apps through the Amazon Appstore.

4.3.2 FRONTEND: HTML

HTML, which stands for Hypertext Markup Language, is the standard language used to create and design web pages. It serves as the backbone of web development, providing the structure and organization for content on the World Wide Web. HTML enables the creation of documents containing hyperlinks, which allow users to navigate between different web pages and resources. These hyperlinks are typically displayed as clickable text or images. HTML consists of markup tags that define the structure and layout of web content.

4.3.3 BACKEND: JAVA

Java is a widely-used programming language and computing platform first released by Sun Microsystems in 1995. Developed with a focus on portability, security, and performance, Java has become one of the most popular programming languages in the world, powering a vast array of applications and systems across various domains, including web development, mobile apps, enterprise software, scientific computing, and more. Java is designed to be platform-independent, meaning that Java programs can run on any device with a Java Virtual Machine (JVM), regardless of the underlying hardware and operating system.

4.3.4 FRAMEWORK: ECLIPSE EE

Eclipse EE (Enterprise Edition), also known as Eclipse IDE for Enterprise Java Developers, is a powerful integrated development environment (IDE) specifically tailored for Java EE (Enterprise Edition) and web application development. It is built on the Eclipse platform and provides comprehensive tools and features to streamline the development, testing, and deployment of enterprise-grade Java applications.

4.3.5 DATABASE: MYSQL

MySQL is an open-source relational database management system (RDBMS) that is widely used for managing structured data. Developed by MySQL AB (now owned by Oracle Corporation), MySQL is known for its reliability, scalability, and performance. It is commonly used in web applications, content management systems (CMS), e-commerce platforms, and various other types of software where data storage and retrieval are essential.

CHAPTER 5

SYSTEM DESIGN

5.1 SYSTEM ARCHITECTURE

System Architecture involves the high level structure of software system abstraction, by using decomposition and composition, with architectural style and quality attributes. A Software Architecture design must conform to the major functionality and performance requirements of the system, as well as satisfy the non-functional requirements such as reliability, scalability, portability, and availability.

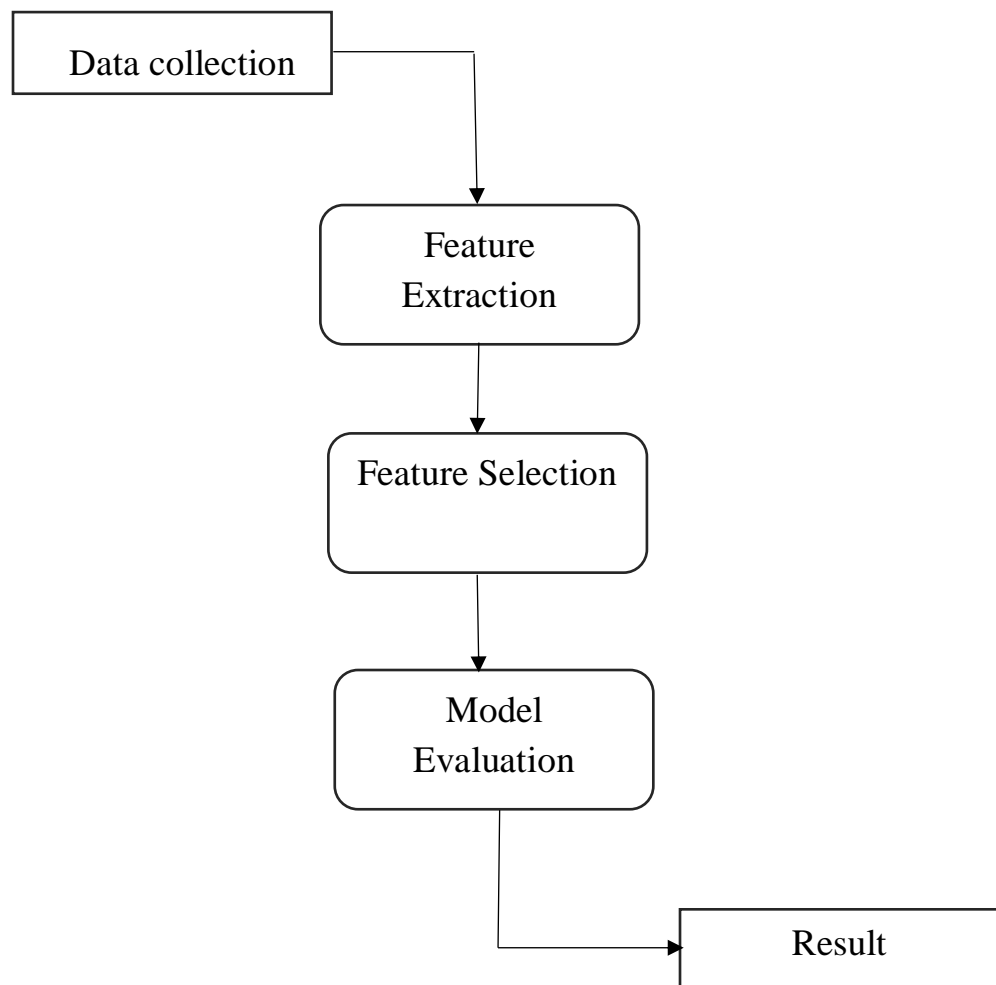


Fig. 5.1 - System Architecture

5.2 DATA FLOW DIAGRAM

A Data Flow Diagram maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination.

Data Flowcharts can range from simple, even hand- drawn process overviews, to in- depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one. Like all the best diagrams and charts, a DFD can often visually “say” things that would be hard to explain in words, and they work for both technical and nontechnical audiences.

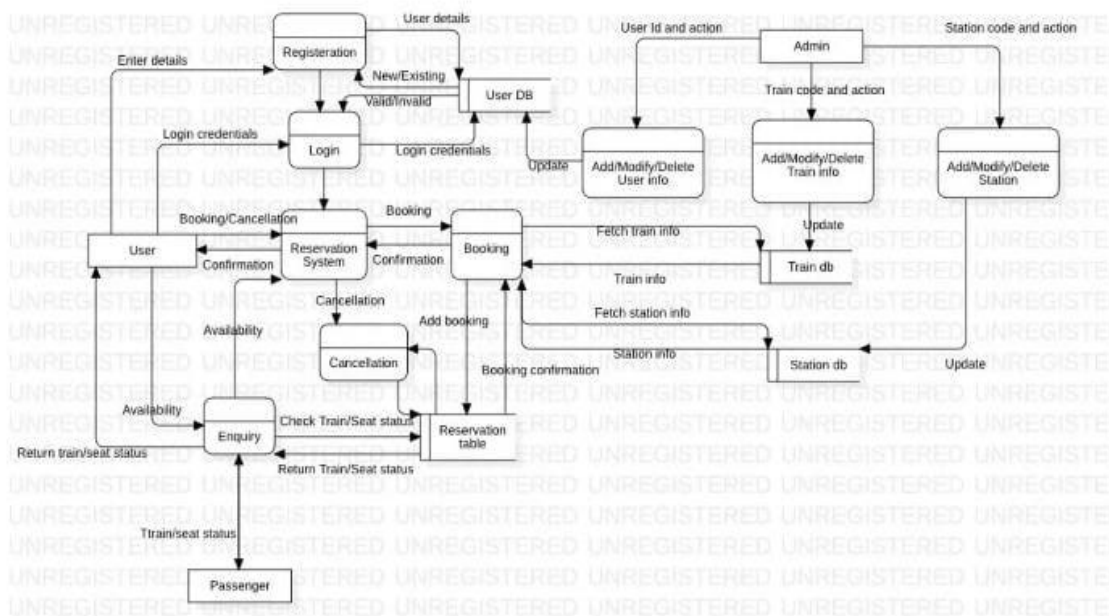


Fig. 5.2 - Data Flow Diagram

5.3 UML Diagram

5.3.1 USECASE DIAGRAM

A Use Case is a methodology used in system analysis to identify, clarify, and organize system requirements. In this context, the term "system" refers to something being developed or operated, such as a mail-order product sales and service Web site. Use Case Diagrams are employed in UML (Unified Modeling Language), a standard notation for the modeling of real- world objects and systems.

- Represent the goals of systems and users
- Specify the context a system should be viewed in
- Specify User requirements

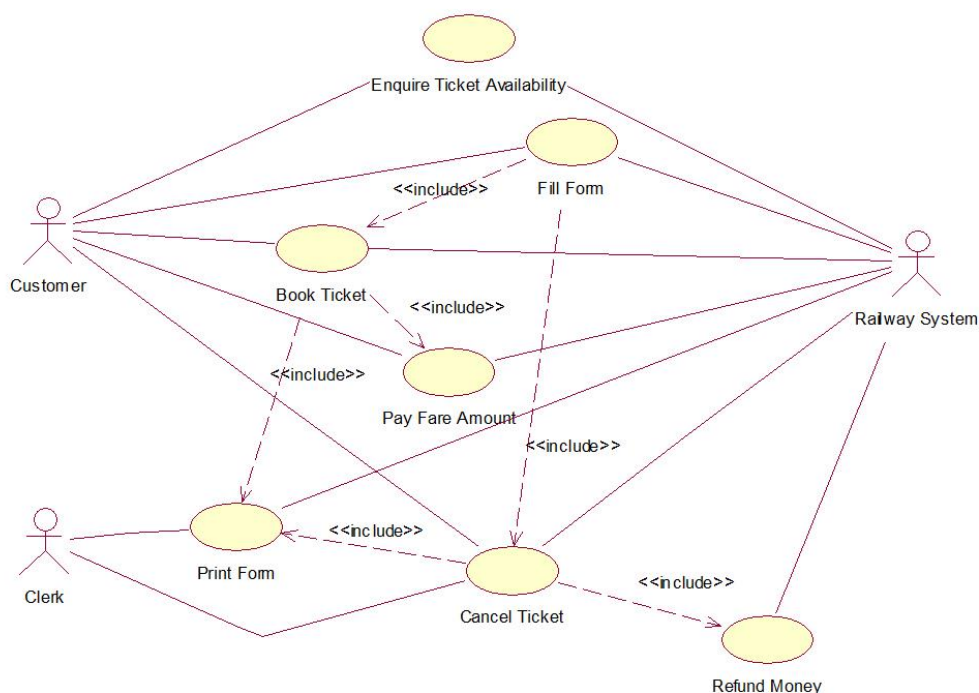


Fig. 5.3 - Usecase Diagram

5.3.2 CLASS DIAGRAM

A Class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.

- Shows Static Structure of classifiers in a system.
- Diagram provides a basic notation for other Structure Diagrams prescribed by UML.
- Helpful for developers and other team members too.
- Business Analysts can use Class Diagrams to model systems from a business perspective.

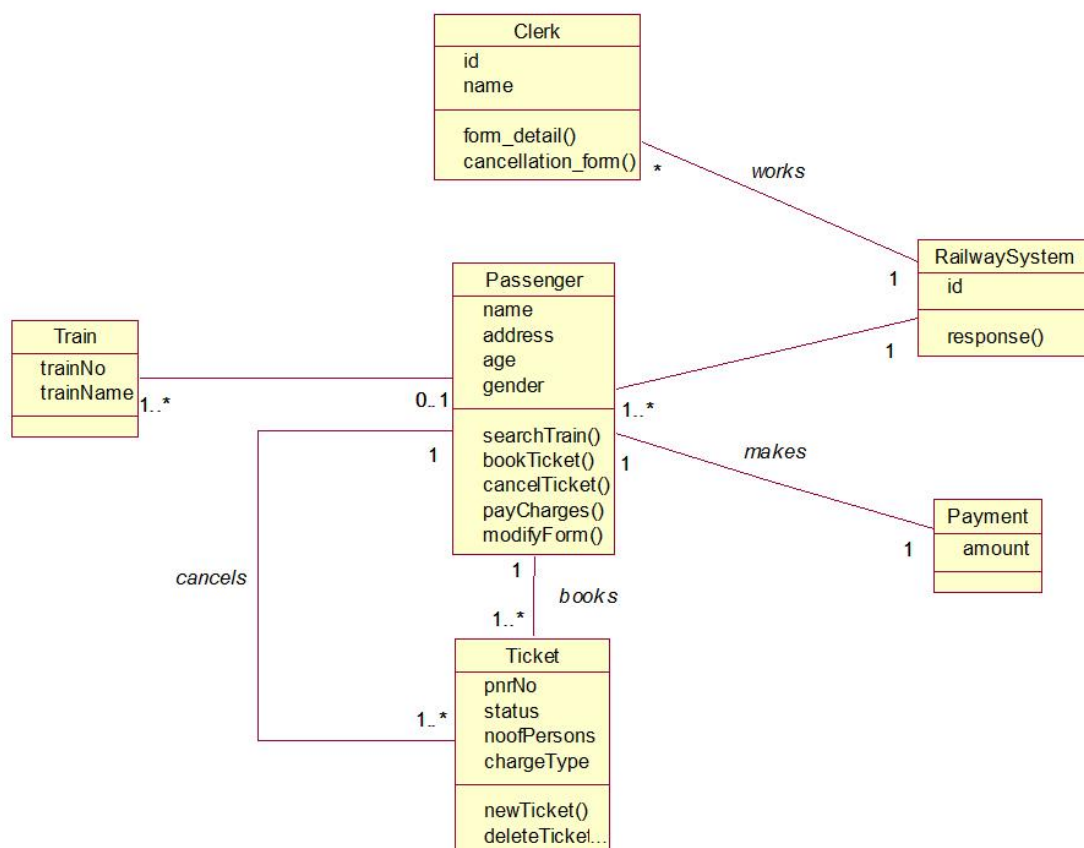


Fig. 5.4 - Class Diagram

5.3.3 SEQUENCE DIAGRAM

UML Sequence diagrams are interaction diagram that detail how operations are carried out. They capture the interaction between objects in the context of collaboration. Sequence diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent.

- Model high-level interaction between active objects in a system.
- Model the interaction between object instances within collaboration that realize use case.

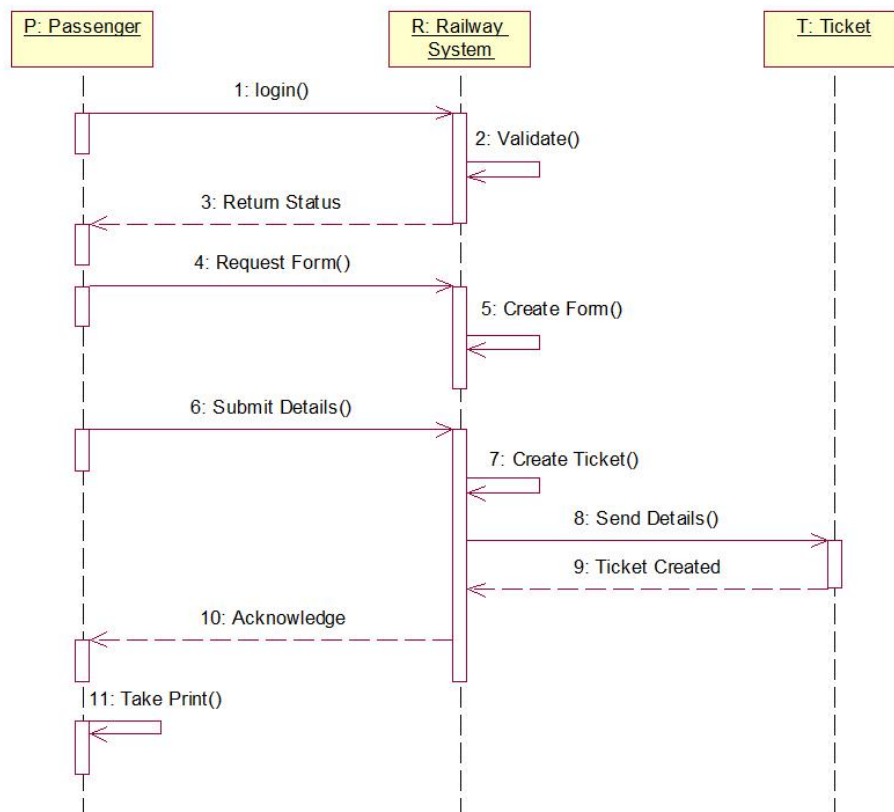


Fig. 5.5 - Sequence Diagram

5.3.4 COLLABORATION DIAGRAM

A Collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software object in the (Unified Modeling Language) UML. These diagram can be used to portray the dynamic behavior of a particular use case and define the role of each object.

- They are very useful for visualizing the relationship between object collaborating to perform particular task

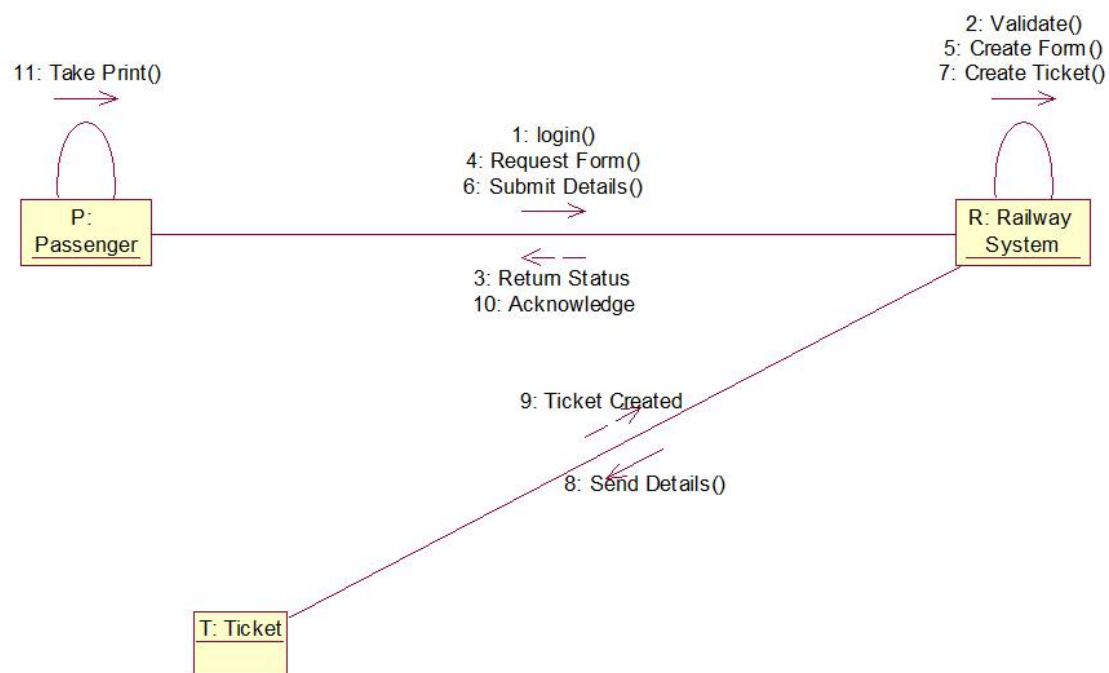


Fig. 5.6 - Collaboration Diagram

UML diagrams are used by software developers, analysts, and designers to communicate and understand the system's architecture, design, and functionality. They serve as a common language for stakeholders to discuss and analyze system requirements, make design decisions, and capture important system details. Unified Modeling Language (UML) is a standardized visual modeling language used to design, visualize, and document software systems.

5.3.5 STATECHART DIAGRAM

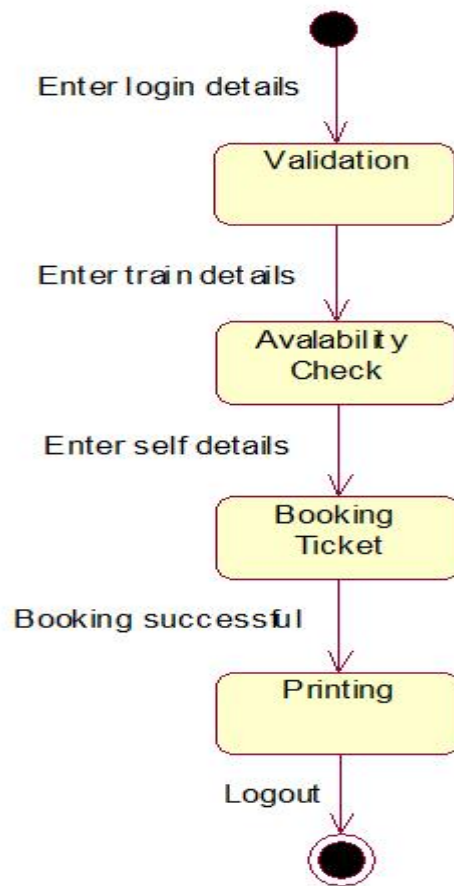


Fig. 5.7 - Statechart Diagram

5.3.6 Activity Diagram

Activity Diagram is another important behavioral diagram in UML diagram to describe dynamic aspects of the system. Activity Diagram is essentially an advanced version of flow chart that modeling the flow from one activity to another activity.

- The flow of activity can be controlled using various control elements in the UML flow diagram
- Activity diagram are used to model processes and workflows. The essence of a useful activity diagram is focused on communicating a specific aspect

of a system's dynamic behaviors. Activity diagram capture the dynamic elements of a system.

- Activity diagram is similar to a flowchart that visualizes flow from one activity to another activity. Activity diagram is identical to the flowchart, but it is not a flowchart.
- In simple words, an activity diagram is used to activity diagram that describe the flow of execution between multiple activities.

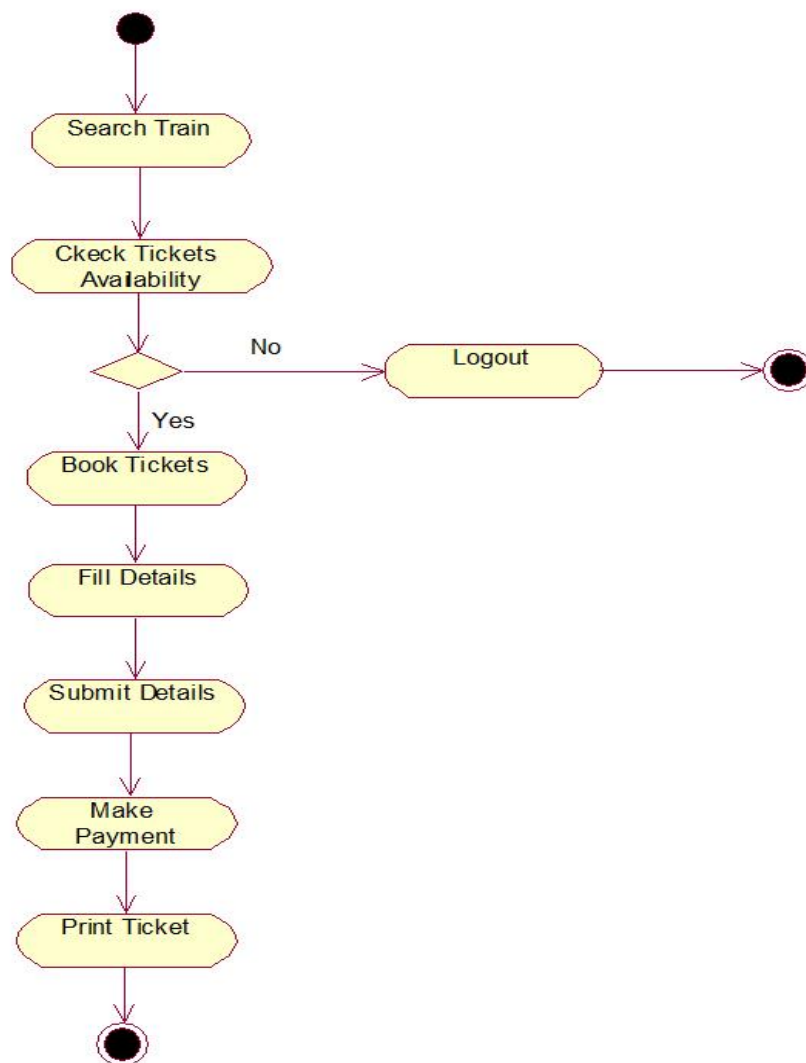


Fig. 5.8 - Activity Diagram

CHAPTER 6

SYSTEM IMPLEMENTATION

6.1 MODULES

Creating a railway reservation system involves several modules to manage different aspects of the reservation process, ticketing, and overall system functioning.

- User Management Module
- Train Management Module
- Reservation Module
- Ticket Management Module

6.2 MODULES DESCRIPTION

6.2.1 COURSE MANAGEMENT

Course management systems are platforms used by educational institutions to facilitate the administration and delivery of courses and educational programs. These systems typically provide a wide range of features to support instructors, students, and administrators throughout the learning process. Instructors can create new courses, set up course details (title, description, objectives, prerequisites, etc.), and configure course settings (enrollment options, grading policies, etc.). Administrators can manage user accounts (students, instructors, administrators), including user registration, authentication, and permissions. Instructors can enroll students in courses and manage course rosters. Instructors can upload course materials such as lecture slides, documents, videos, and other multimedia content. Content organization features allow instructors to structure course materials into modules, units, or lessons. Discussion forums, chat rooms, and messaging systems enable communication between instructors and students, as well as among students.

6.2.2 TRAIN MANAGEMENT MODULE

The Train Management Module is a core component of a railway reservation system responsible for managing all aspects related to trains, including their schedules, routes, seat availability, fares, and other relevant information. Allows administrators to add new trains to the system with details such as train name, number, type (e.g., express, local), and operational schedule. Provides functionality to modify existing train information as needed, including updating schedules, routes, and other attributes. Enables administrators to create and manage train schedules, including departure and arrival times for each station along the route. Supports the addition of special schedules for holidays, events, or maintenance periods. Facilitates the definition and management of train routes, specifying the sequence of stations the train will stop at from origin to destination. Allows for the addition or removal of stations from existing routes and the adjustment of route lengths and configurations. Manages seat availability for different classes (e.g., first class, second class, sleeper, etc.) on each train.

6.2.3 RESERVATION MODULE

The Reservation Module is a fundamental component of a railway reservation system, facilitating the booking process for passengers and managing seat reservations on trains. Allows passengers to search for trains based on criteria such as origin, destination, date of travel, preferred class, and number of passengers. Retrieves real-time information on train schedules, routes, and seat availability from the Train Management Module. Provides passengers with interactive seat maps or seat selection options, allowing them to choose specific seats or berths based on their preferences and requirements. Guides passengers through the booking process, collecting essential details such as passenger information, contact details, and payment information. Integrates with payment gateways to facilitate secure online transactions for booking confirmation and payment processing. Generates electronic tickets (e-tickets) or reservation

confirmations for passengers upon successful booking. Manages booked tickets within the system, maintaining a record of all confirmed reservations and associated passenger information.

6.2.4 TICKET MANAGEMENT MODULE

The Ticket Management Module is a vital part of a railway reservation system, responsible for managing all aspects related to booked tickets, including ticket issuance, modifications, cancellations, and refunds. Generates electronic tickets (e-tickets) or reservation confirmations for passengers upon successful booking through the Reservation Module. Allows passengers to make changes to their booked tickets, such as updating passenger information, changing travel dates, or modifying seat preferences. Facilitates the cancellation of booked tickets by passengers who no longer require them due to changes in travel plans or other reasons. Calculates refunds for canceled tickets based on predefined refund rules, taking into account factors such as cancellation timing, ticket type, and applicable deductions. Sends confirmation notifications to passengers upon successful cancellation of their tickets, providing details of the cancellation and any applicable refunds. Offers options for passengers to reschedule their travel plans by changing the date, time, or route of their booked tickets. Manages waitlisted bookings by automatically processing seat allocations as seats become available due to cancellations or additional inventory. Integrates seamlessly with payment gateways to process refund transactions and update payment statuses related to canceled tickets. Generates reports and analytics on ticket sales, cancellations, refunds, and other ticket-related metrics to facilitate decision-making and performance analysis.

CHAPTER 7

TESTING

7.1 AUTOMATION TOOL

Selenium is a powerful and widely-used testing tool primarily used for automating web applications. It supports multiple programming languages like Java, Python, C#, and Ruby, making it versatile for developers and testers. Selenium allows for the creation of automated tests that mimic user interactions with web browsers, such as clicking buttons, filling out forms, and navigating through web pages. Its robust API and integration with testing frameworks like TestNG and JUnit enable efficient test scripting, execution, and reporting. Selenium WebDriver, a key component of Selenium, provides a rich set of methods and commands for interacting with web elements, validating functionality, and performing assertions. Additionally, Selenium Grid facilitates parallel test execution across multiple browsers and platforms, enhancing testing scalability and coverage. Overall, Selenium's flexibility, compatibility, and extensive features make it a popular choice for web application testing, helping teams achieve faster feedback cycles, improve test coverage, and maintain software quality.

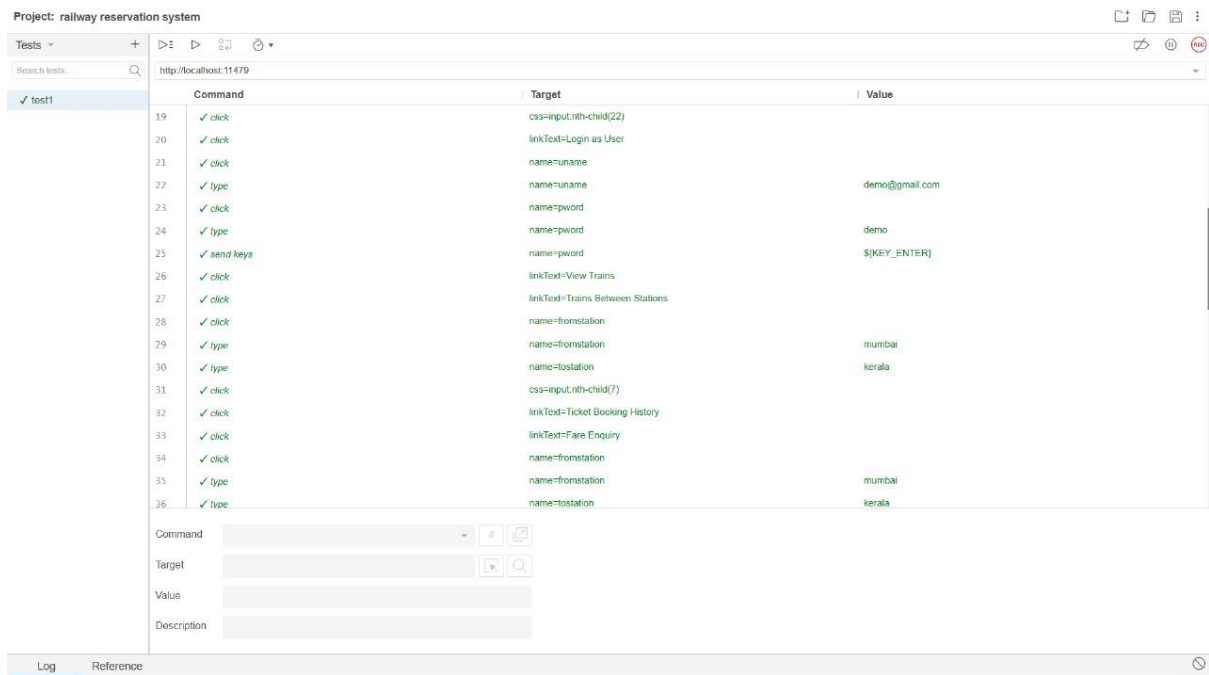


Fig. 7.1 - Input Validation Testing

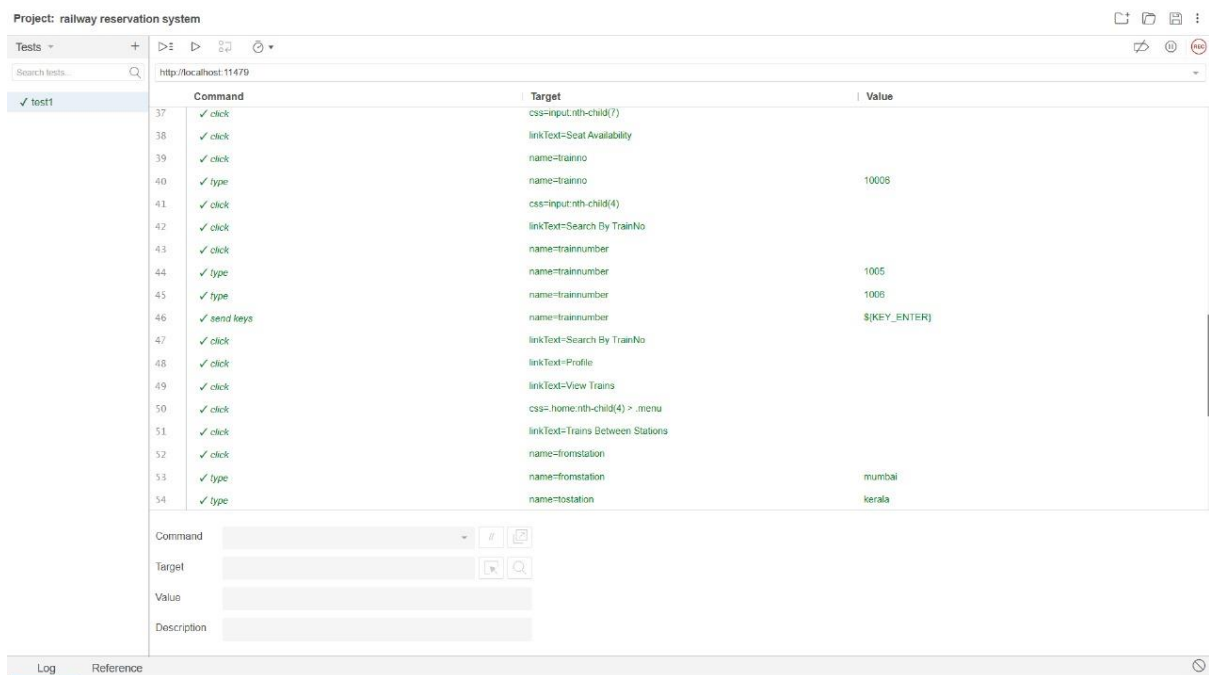


Fig. 7.2 - Functionality Testing

CHAPTER 8

CONCLUSIONS

The development and implementation of the railway reservation system have been a significant endeavor aimed at enhancing the booking experience for passengers while streamlining operations for railway authorities. Through meticulous planning, collaboration, and innovation, the project has achieved several key milestones and delivered tangible benefits to stakeholders. The railway reservation system offers a user-friendly interface that allows passengers to search, book, and manage their journeys seamlessly. Enhanced features such as seat selection, payment processing, and booking confirmation have contributed to an improved booking experience. The system has streamlined ticket booking processes, seat reservation management, and payment transactions, leading to increased efficiency and productivity for railway authorities. Automation of repetitive tasks and integration with external systems have optimized operational workflows. Integrating the reservation system with external systems such as train scheduling, payment gateways, and notification services posed challenges due to disparate technologies, data formats, and communication protocols. However, effective collaboration and problem-solving strategies helped overcome these challenges. Ensuring optimal performance under varying load conditions required extensive testing, analysis, and optimization of system components. Identifying and resolving performance bottlenecks required dedicated efforts and iterative improvements.

CHAPTER 9

FUTURE ENHANCEMENT

Future enhancements for the railway reservation system project can help to further improve functionality, usability, and efficiency while addressing emerging needs and technological advancements. Develop a mobile application for iOS and Android platforms to provide passengers with greater flexibility and convenience in booking and managing their journeys. Implement a recommendation engine that analyzes passenger preferences, booking history, and travel patterns to offer personalized travel suggestions and promotions. Integrate dynamic pricing algorithms and revenue management techniques to optimize ticket pricing based on demand, seat availability, and market trends. Enhance accessibility features of the reservation system to cater to passengers with disabilities and special needs. Implement advanced analytics and business intelligence capabilities to gain insights into passenger behavior, booking trends, and revenue performance.

CHAPTER 10

APPENDICES

10.1 APPENDIX 1

10.1.1 SOURCE CODE

FRONT-END: USERHOME

```
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>E-Trains</title>
<link rel="stylesheet" href="UserHome_Css.css">
<style>
</style>
</head>
<body>
<header>
<h1 class="hd">National Ticket Booking Spot</h1>
<div class="home">
<p class="menu">
<a href="userhome">Home</a>
</p>
</div>
<div class="home">
<p class="menu">
<a href="userviewtrainfwd">View Trains</a>
</p>
</div>
```

```
<div class="home">
<p class="menu">
<a href="trainbwstnfwd">Trains Between Stations</a>
</p>
</div>
<div class="home">
<p class="menu">
<a href="bookingdetails">Ticket Booking History</a>
</p>
</div>
<div class="home">
<p class="menu">
<a href="fareenqfwd">Fare Enquiry</a>
</p>
</div>
<div class="home">
<p class="menu">
<a href="useravailfwd">Seat Availability</a>
</p>
</div>
<div class="home">
<p class="menu">
<a href="usersearchtrain">Search By TrainNo</a>
</p>
</div>
<div class="home">
<p class="menu">
<a href="userprofile">Profile</a>
</p>
```

```

</div>
<div class="home">
<p class="menu">
<a href="userlogout">Logout</a>
</p>
</div>
</header>
</body>
</html>

```

BACK END: USERHOME

```

package com.shashi.servlets;
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import com.shashi.constant.UserRole;
import com.shashi.utility.TrainUtil;
@SuppressWarnings("serial")
@WebServlet("/userhome")
public class UserHome extends HttpServlet {
protected void doGet(HttpServletRequest req, HttpServletResponse res) throws
IOException, ServletException {
res.setContentType("text/html");
PrintWriter pw = res.getWriter();
TrainUtil.validateUserAuthorization(req, UserRole.CUSTOMER);

```

```

RequestDispatcher rd = req.getRequestDispatcher("UserHome.html");
rd.include(req, res);

pw.println("<div class='tab'>" + "<p1 class='menu'>" + " Hello "
TrainUtil.getCurrentUserName(req)
+ " ! Welcome to our new NITRTC Website" + "</p1>" + "    </div>");
pw.println("<div class='main'><p1 class='menu'>User Home</p1></div>");
pw.println("<div class='tab'>Hello " + TrainUtil.getCurrentUserName(req)
+ " ! Good to See You here.<br/> Here you can Check up the train "
+ "details, train schedule, fare Enquiry and many more information.
<br/>Just go to the Side Menu Links and "
+ "Explore the Advantages.<br/><br/>Thanks For Being Connected with us!" +
"</div>");
}
}

```

FRONT-END: USERLOGIN

```

<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>LOGIN</title>
<link rel="stylesheet" href="UserHome_Css.css">
</head>
<body>
<header>
<h1 class="hd">National Ticket Booking Spot</h1>
<div class="main">
<span class="menu">
<a href="UserLogin.html">Login as User</a>
</span>

```

```

<span class="menu">
<a href="UserRegister.html">New User Register</a>
</span>
<span class="menu">
<a href="AdminLogin.html">Login as Admin</a>
</span>
</div>
</header>
<div class="tab green">User Login</div>
<form action="userlogin" class="tab brown" method="post">
<br /> UserName: <input type="text" name="uname"
placeholder="Enter Your EmailId"> <br /> <br /> Password:
<input type="password" name="pword"> <br /> <br />
<input type="submit" value=" LOGIN "><br />
</form><br />
</body></html>

```

BACK-END: USERLOGIN

```

package com.shashi.servlets;
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import com.shashi.constant.ResponseCode;
import com.shashi.constant.UserRole;
import com.shashi.utility.TrainUtil;

```

```

@WebServlet("/userlogin")

public class UserLoginServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

protected void doPost(HttpServletRequest req, HttpServletResponse res) throws
IOException, ServletException {

    PrintWriter pw = res.getWriter();
    res.setContentType("text/html");

    String uName = req.getParameter("uname");
    String pWord = req.getParameter("pword");

    String responseMsg = TrainUtil.login(req, res, UserRole.CUSTOMER, uName,
pWord);

    if (ResponseCode.SUCCESS.toString().equalsIgnoreCase(responseMsg)) {
        RequestDispatcher rd = req.getRequestDispatcher("UserHome.html");
        rd.include(req, res);

        pw.println("<div class='main'><p1 class='menu'>Hello " + uName
+ " ! Welcome to our new NITRTC Website</p1></div>");

        pw.println("<div class='tab'>Hello " + uName
+ " ! Good to See You here.<br/> Here you can Check up the train "
+ "details and train schedule,fare Enquiry and many more information.<br/>Just
go to the Side Menu Links and "
+ "Explore the Advantages.<br/><br/>Thanks For Being Connected with us!" +
"</div>");
    } else {

        RequestDispatcher rd = req.getRequestDispatcher("UserLogin.html");
        rd.include(req, res);

        pw.println("<div class='tab'><p1 class='menu'>" + responseMsg +
"</p1></div>");
    }
}
}
}

```

FRONT-END: USERREGISTER

```
<!DOCTYPE html>

<html>

<head>

<meta charset="ISO-8859-1">

<title>LOGIN</title>

<link rel="stylesheet" href="UserHome_Css.css">

</head>

<body>

<header>

<h1 class="hd">National Ticket Booking Spot</h1>

<div class="main">

<span class="menu"> <a href="UserLogin.html">Login as User</a>

</span> <span class="menu">

<a href="UserRegister.html">New UserRegister</a>

</span> <span class="menu">

<a href="AdminLogin.html">Login asAdmin</a>

</span></div></header>

<table class="tab brown hd">

<tr>

<td>New User Registration</td>

</tr></table>

<table class="tab green">

<tr>

<td>

<form action="userreg" method="post">

Email Id : <input type="email" name="mailid" placeholder="Enter your emailId" required><br /> <br />
```

Password : <input type="password" name="pword" pattern="(?!.*\d)(?!.*[a-z])(?!.*[A-Z]).{8,}" title="Must contain at least one number and one uppercase and lowercase letter, and at least 8 or more characters" required>

 FirstName: <input type="text" name="firstname" required>

 Last Name: <input type="text" name="lastname" required>

 Address : <input type="text" name="address" required>

value=" I AGREE FOR ALL TERMS & CONDITIONS ! REGISTER ME "

<style="text-align: center">

</form></td></tr>

</table></body></html>

BACK-END: USERREGISTER

```
package com.shashi.servlets;

import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import com.shashi.beans.TrainException;
import com.shashi.beans.UserBean;
import com.shashi.constant.UserRole;
import com.shashi.service.UserService;
import com.shashi.service.impl.UserServiceImpl;

@SuppressWarnings("serial")
@WebServlet("/userreg")

public class UserRegServlet extends HttpServlet {

    private UserService userService = new
    UserServiceImpl(UserRole.CUSTOMER);
```



```

protected void doPost(HttpServletRequest req, HttpServletResponse res) throws
IOException, ServletException {
    res.setContentType("text/html");
    PrintWriter pw = res.getWriter();
    try {
        UserBean user = new UserBean();
        user.setMailId(req.getParameter("mailid"));
        user.setPWord(req.getParameter("pword"));
        user.setFName(req.getParameter("firstname"));
        user.setLName(req.getParameter("lastname"));
        user.setAddr(req.getParameter("address"));
        user.setPhNo(Long.parseLong(req.getParameter("phoneno")));
        String message = userService.registerUser(user);
        if ("SUCCESS".equalsIgnoreCase(message)) {
            RequestDispatcher rd = req.getRequestDispatcher("UserLogin.html");
            rd.include(req, res);
            pw.println("<div class='tab'><p1 class='menu'>User Registered
            Successfully !</p1></div>");
        } else {
            RequestDispatcher rd = req.getRequestDispatcher("UserRegister.html");
            rd.include(req, res);
            pw.println("<div class='tab'><p1 class='menu'>" + message + "</p1></div>");
        }
    } catch (Exception e) {
        throw new TrainException(422, this.getClass().getName() + "_FAILED",
        e.getMessage());
    }
}
}
}

```

10.2 APPENDIX 2

10.2.1 SNAPSHOTS

10.2.1.1 REGISTER NEW USER

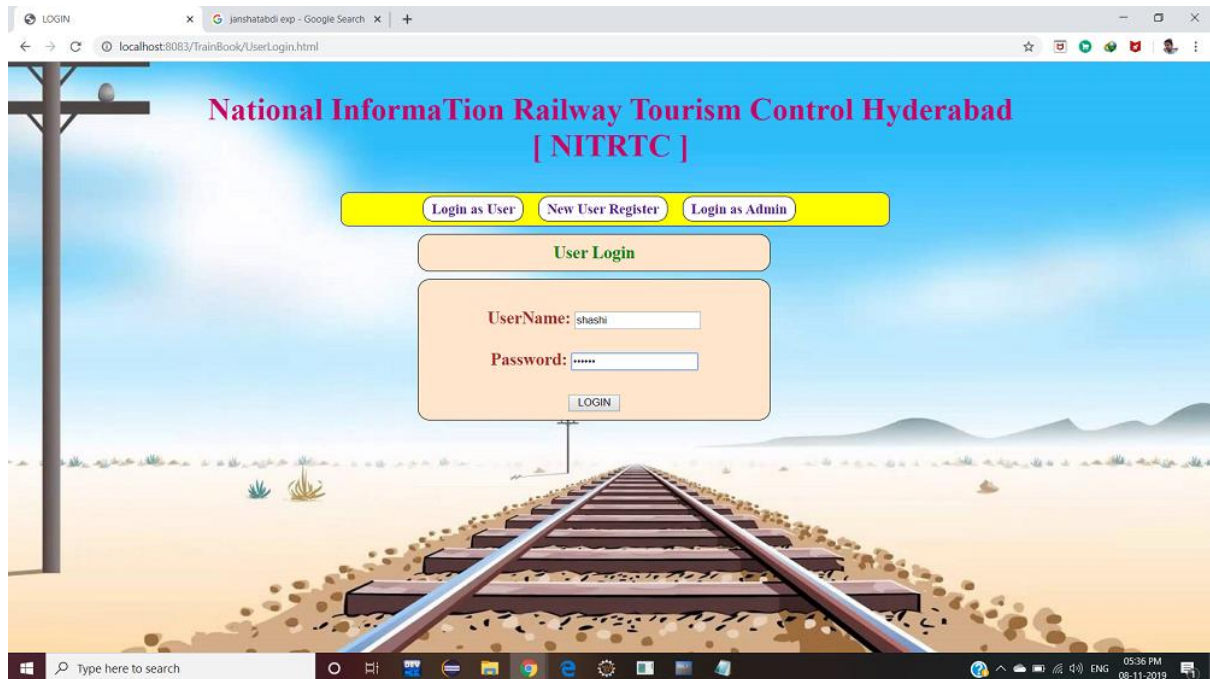


Fig. 10.1 - Register New User

10.2.1.2 USER HOME

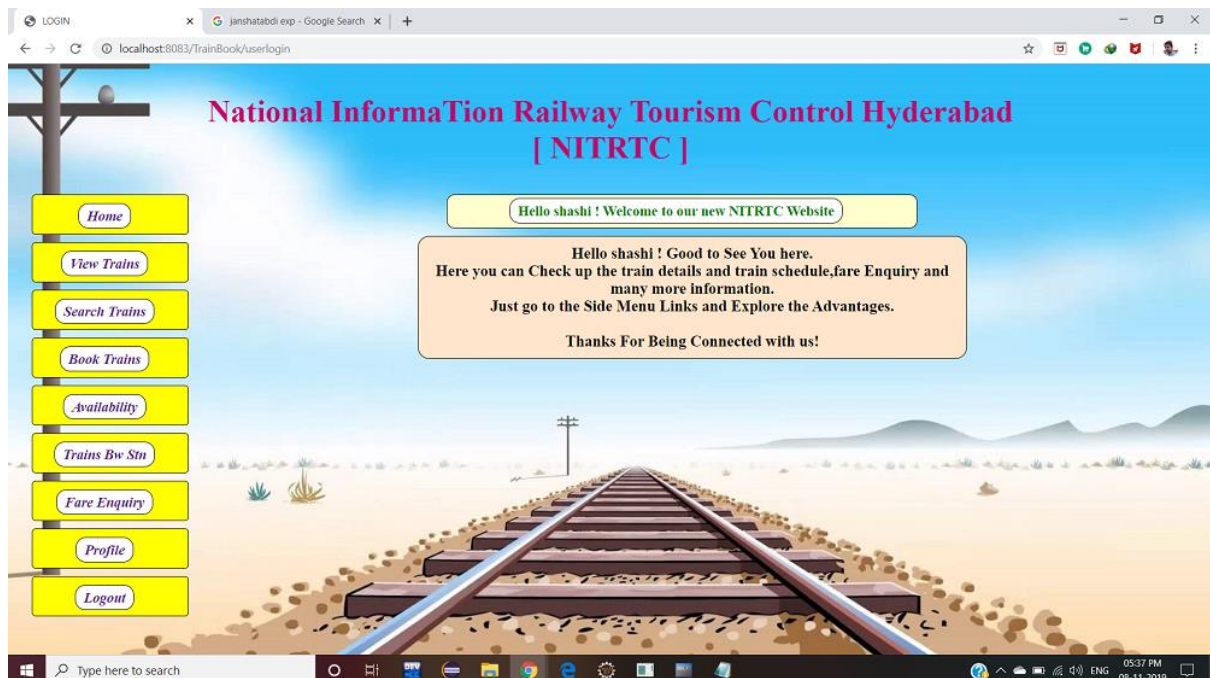
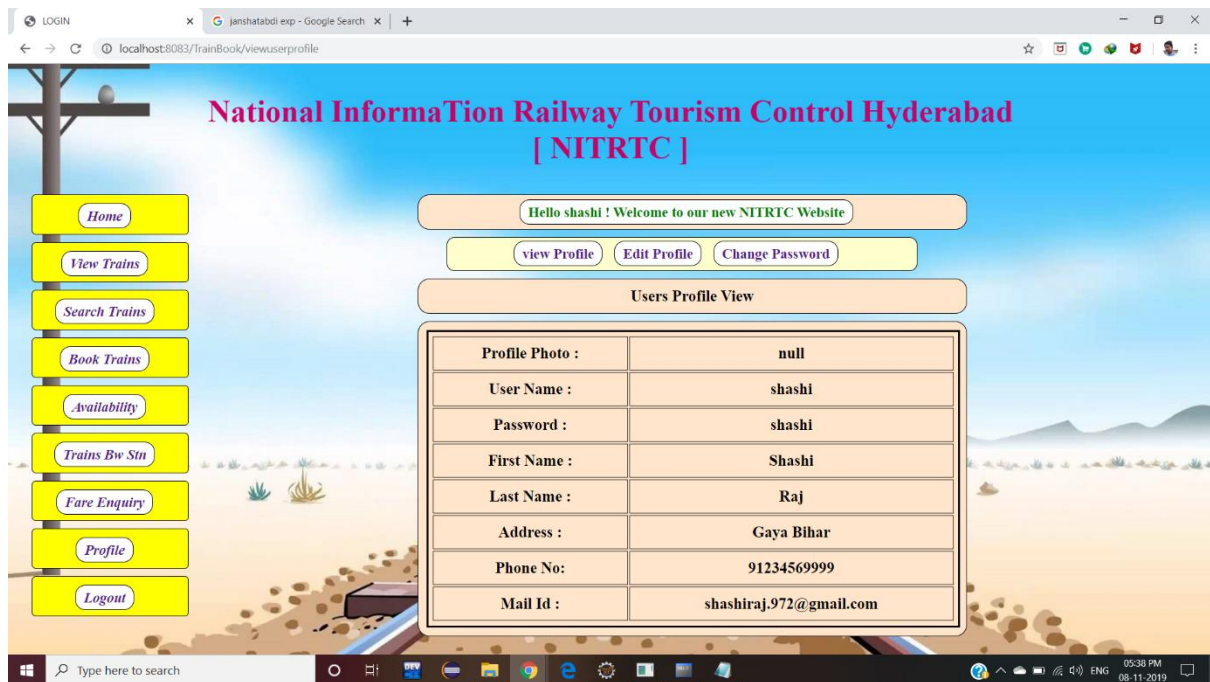


Fig. 10.2 - User Home

10.2.1.3 VIEW PROFILE



National InformaTion Railway Tourism Control Hyderabad [NITRTC]

Hello shashi ! Welcome to our new NITRTC Website

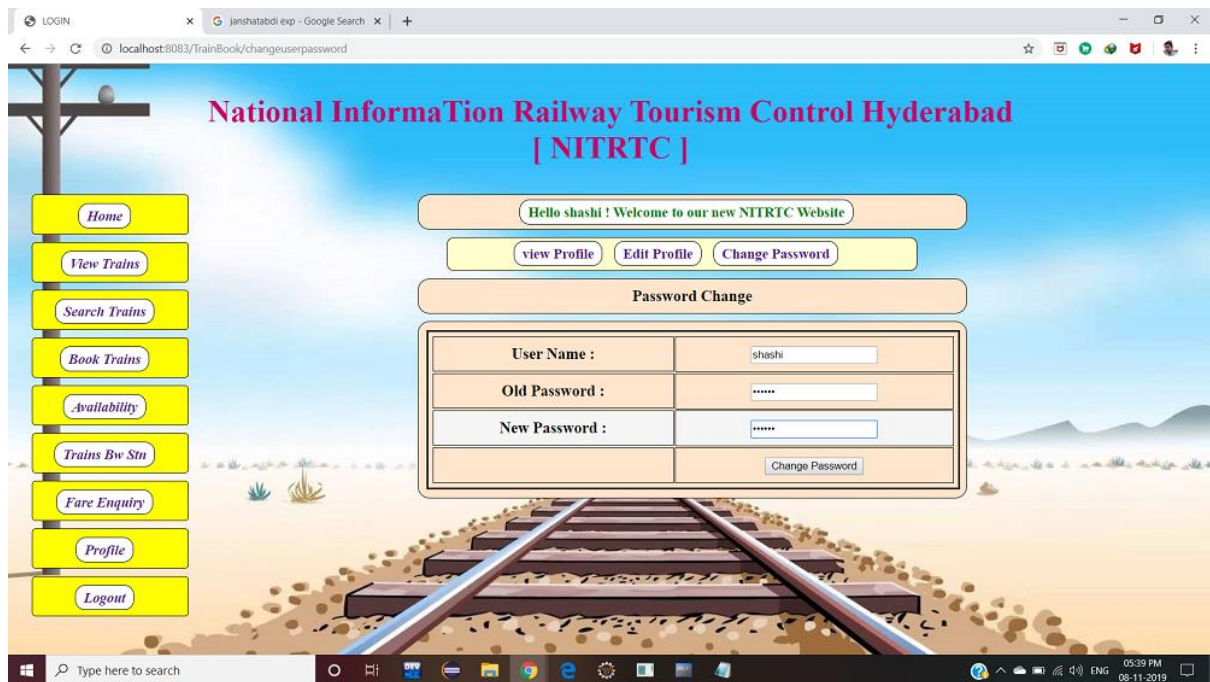
[view Profile](#) [Edit Profile](#) [Change Password](#)

Users Profile View

Profile Photo :	null
User Name :	shashi
Password :	shashi
First Name :	Shashi
Last Name :	Raj
Address :	Gaya Bihar
Phone No:	91234569999
Mail Id :	shashiraj.972@gmail.com

Fig. 10.3 - View Profile

10.2.1.4 CHANGE PASSWORD



National InformaTion Railway Tourism Control Hyderabad [NITRTC]

Hello shashi ! Welcome to our new NITRTC Website

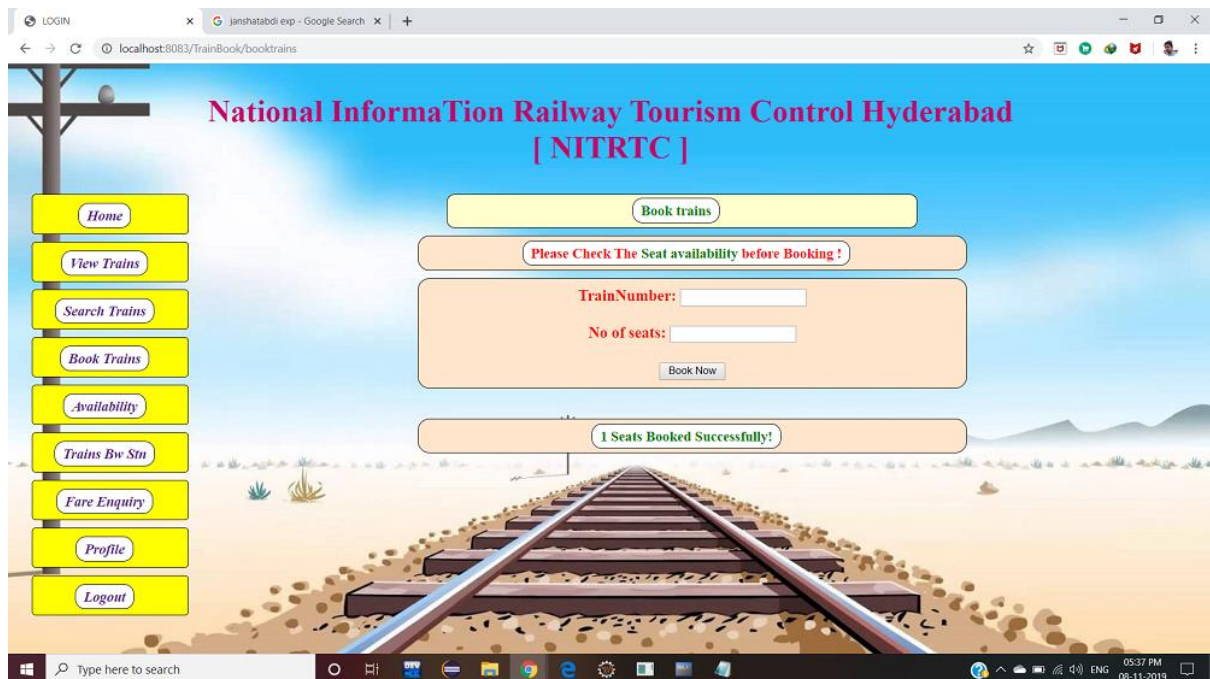
[view Profile](#) [Edit Profile](#) [Change Password](#)

Password Change

User Name :	<input type="text" value="shashi"/>
Old Password :	<input type="password" value=""/>
New Password :	<input type="password" value=""/>
<input type="button" value="Change Password"/>	

Fig. 10.4 - Change Password

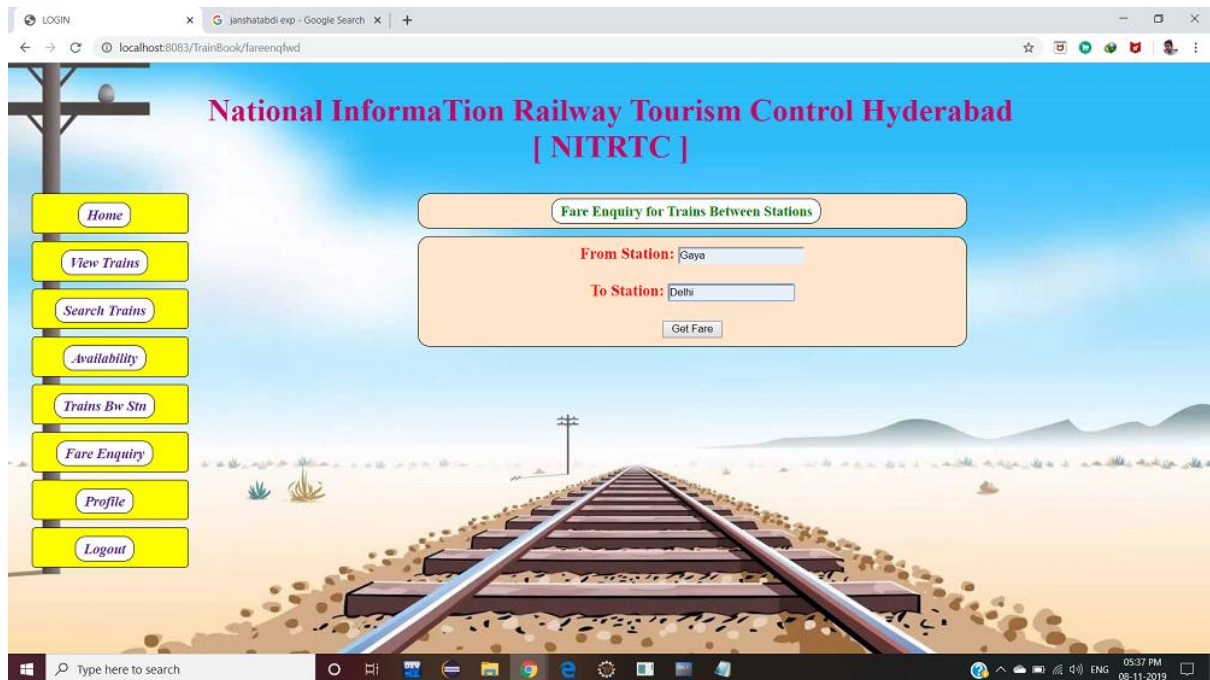
10.2.1.5 SEARCH TRAIN



The screenshot shows a web browser window with the URL `localhost:8083/TrainBook/booktrains`. The page title is "National InformaTion Railway Tourism Control Hyderabad [NITRTC]". On the left, there is a vertical menu with buttons: Home, View Trains, Search Trains, Book Trains, Availability, Trains Bw Stn, Fare Enquiry, Profile, and Logout. The main content area has a "Book trains" button at the top. Below it is a message: "Please Check The Seat availability before Booking!". Then, there are input fields for "TrainNumber:" and "No of seats:", followed by a "Book Now" button. At the bottom of the main area, a green message box says "1 Seats Booked Successfully!". The background features a desert landscape with a long railway track receding into the distance. The Windows taskbar at the bottom shows the time as 05:37 PM on 08-11-2019.

Fig. 10.5 - Search Train

10.2.1.6 FARE ENQUIRY



The screenshot shows a web browser window with the URL `localhost:8083/TrainBook/fareenqfwd`. The page title is "National InformaTion Railway Tourism Control Hyderabad [NITRTC]". The left sidebar menu is identical to the previous screenshot. The main content area has a "Fare Enquiry for Trains Between Stations" button. Below it, there are input fields for "From Station:" (with "Gaya" entered) and "To Station:" (with "Delhi" entered), followed by a "Get Fare" button. The background is the same desert landscape with a railway track. The Windows taskbar at the bottom shows the time as 05:37 PM on 08-11-2019.

Fig. 10.6 - Fare Enquiry

10.2.1.7 FARE RESULT

The screenshot shows the NITRTC website interface. On the left is a vertical menu with buttons: Home, View Trains, Search Trains, Book Trains, Availability, Trains Bw Stn, Fare Enquiry, Profile, and Logout. The main content area has a blue header with the text "National InformaTion Railway Tourism Control Hyderabad [NITRTC]". Below the header, a welcome message says "Hello shashi ! Welcome to our new NITRTC Website". A yellow box displays "Fare for Trains BetWeen Station Gaya and Delhi is RS 840". A table shows the train details:

Train Name	Train No	From Stn	To Stn	Seats	Fare (INR)
Mahabodhi Express	10103	Gaya	Delhi	73	840 RS

The background of the website features a landscape with a railway track receding into the distance under a blue sky.

Fig. 10.7 - Fare Result

10.2.1.8 SEAT AVAILABILITY

The screenshot shows the NITRTC website interface. On the left is a vertical menu with buttons: Home, View Trains, Search Trains, Book Trains, Availability, Trains Bw Stn, Fare Enquiry, Profile, and Logout. The main content area has a blue header with the text "National InformaTion Railway Tourism Control Hyderabad [NITRTC]". Below the header, a welcome message says "Hello shashi ! Welcome to our new NITRTC Website". A yellow box displays "Available Seats are 73 Seats". A table shows the train details:

Train Name :	Mahabodhi Express
Train Number :	10103
From Station :	Gaya
To Station :	Delhi
Available Seats:	73
Fare (INR) :	840 RS

The background of the website features a landscape with a railway track receding into the distance under a blue sky.

Fig. 10.8 - Seat Availability

10.2.1.9 TICKET BOOKING

National InformaTion Railway Tourism Control Hyderabad [NITRTC]

Your Ticket Booking Information

USER ID:	shashi	Train NO:	10102
From Station:	Gaya	To Station :	Mumbai
Journey Date:	08-11-2019	No of Seats:	1

Pay And Book

Fig. 10.9 - Ticket Booking

10.2.1.10 ADD TRAIN

National InformaTion Railway Tourism Control Hyderabad [NITRTC]

Add Trains

Train Number: 10104

Train Name: Garib Rath Express

From Station: Patna

To Station: Mumbai

Available: 26

Fare (INR): 924

ADD TRAIN

Fig. 10.10 - Add Train

10.2.1.11 CANCEL TRAIN



Fig. 10.11 - Cancel Train

CHAPTER 11

REFERENCES

1. Desdemona Isabela Scarisoreanu” Configuring an Application Which Allows Online Booking And Purchase Of Travel Tickets For Railway And Road Transport - Unified Modeling Language” International Conference On Mathematics And Computers In Science And Engineering (Macise) (2020).
2. Godson Michael d'silva, Anoop Kunjumon Scariah, Lukose Roy Pannapara, Jessica John Joseph “Smart Ticketing System For Railways In Smart Cities Using Software As A Service Architecture “International Conference On I-Smac (Iot In Social, Mobile, Analytics And Cloud) (I-Smac) (2017).
3. Hairong Dong, Xuan Liu, Min Zhou, Wei Zheng, Jing Xun, Shigen Gao, Haifeng Song, Yidong Li, Fei-Yue Wang “Integration Of Train Control And Online Rescheduling For High-Speed Railways In Case Of Emergencies“ Computational Social Systems(2022).
4. Nur W. Rahayu, Dhery D. Andika “Assessing the Quality Of Rail Ticket Reservation Systems: Cases From Indonesia “International Conference On Computer And Information Sciences (Iccoins) (2014).
5. Ritesh Shivraj Pokarne “Train Ticket Reservation System” Research Gate (2021).

6. Ryo Takagi “Required Characteristics of The Ticketing Systems For Reservation-Compulsory Commuting Railways” International Conference On Intelligent Rail Transportation (Icirt) (2018).
7. Sara Nawghare, Rachna K. Somkunwar, Zarina Shaikh “Indian Railways Smart Ticketing Validation System With Improved Alert Approach” Sustainable Computing And Smart Systems (Icscss) (2023).
8. Sofiar Yusliansyah “Adoption Factors of Online-Web Railway Ticket Reservation Service (A Case from Indonesia) “International Conference of Information And Communication Technology (Icoict) (2017).
9. Sourodeep Chatterjee, Soham Das, Divisha, Bhaskar Goswami, Pallab Nag, Chittaranjan Pradhan “Railway E-Verification Information and Ticketing System” International Conference on Communication And Signal Processing (2020).
10. Ms. B.T. Tharani Sri Shakti, Mr. Jessen Joseph Leo, Ms. R. Monisha, Dr.S.M Ramesh "Advanced Train Reservation and Passenger Intimation with Safety System" International Conference On Innovation And Challenges In Engineering Sciences (Icices) (2014).