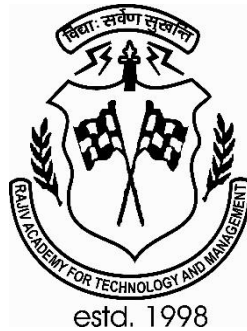


RAJEEV ACADEMY FOR TECHNOLOGY & MANAGEMENT, Mathura



MINI-PROJECT REPORT

On

RAILWAY RESERVATION MANAGEMENT SYSTEM

Submitted for the partial fulfillment towards the award
of the degree in

MASTER OF COMPUTER APPLICATION

Of

Dr. A.P.J. Abdul Kalam Technical University,

Uttar Pradesh, Lucknow

Submitted By

TRAPTI KASHYAP

Roll Number: 2301730140034

Session: 2024-25

Under the Guidance of

SAIYAM VARSHNEY

Department of MCA

PREFACE

The **Railway Reservation Management System** project is an endeavor to bridge the gap between traditional railway booking systems and modern digital advancements. This project serves as a practical demonstration of implementing theoretical knowledge into a real-world application, thereby enhancing the understanding of system development, database management, and user-interface design.

The purpose of this project is to develop a comprehensive and efficient system that simplifies and automates the railway reservation process. With an increasing demand for seamless and accessible services in the transportation sector, this system provides a user-friendly platform for customers to search for train availability, book tickets, and manage their reservations effortlessly.

This report is prepared as part of the summer training program, aimed at fostering a deeper understanding of software development processes and familiarizing students with the dynamic requirements of the industry. The project emphasizes critical aspects such as:

- Creating a **centralized database** to store and manage real-time train schedules, ticket availability, and user data securely.
- Developing an **intuitive web interface** that caters to the needs of both passengers and railway administrators.
- Providing enhanced accessibility and transparency in ticket booking processes, reducing errors, and improving user satisfaction.

The Railway Reservation Management System is more than just a software solution; it represents an integration of technology and convenience tailored to meet the evolving needs of the transportation sector. This preface is a testament to the dedication and collaborative efforts that went into designing, developing, and documenting this project to achieve its goals effectively.

CERTIFICATE OF GUIDANCE

To whom it may Concern

Department Of Computer Science

This is certified that the project entitles “RAILWAY RESERVATION MANAGEMENT SYSTEM” has been developed under the guidance and supervision during session 2024-25 by following student of M.C.A III Semester.

TRAPTI KASHYAP

It is further certified that these students have submitted the present volume as a project report to the department of computer science. This is for partial fulfillment of award of degree of Master of Computer Application from Dr. APJ Abdul Kalam Technical University, Lucknow.

Mr. Saiyam Varshney
(Project Guide)

STUDENT'S DECLARATION

I, **TRAPTI KASHYAP**, hereby declare that this work entitled “**RAILWAY RESERVATION MANAGEMENT SYSTEM**” is the result of summer training under taken. The findings and conclusions expressed in this report are genuine, authentic and are for academic purpose. Any resemblance to earlier research work is purely coincidental.

TRAPTI KASHYAP
(MCA III Sem)
Roll No. 2301730140034

REPORT OF SUPERVISOR CERTIFICATION

I, **Mr. Saiyam Varshney**, hereby certify that this research work entitled “**RAILWAY RESERVATION MANAGEMENT SYSTEM**” is the result of research under taken by **TRAPTI KASHYAP (2301730140034)** the findings and conclusions expressed in this report are genuine, authentic and are for academic purpose only. Any resemblance to earlier work is purely coincidental.

Signature

**Saiyam Varshney
(MCA)**

ABSTRACT

The Railway Reservation Management System is a modern software application aimed at revolutionizing the traditional methods of railway ticket booking, reservation, and management. The primary objective of this project is to provide an efficient, user-friendly, and secure online platform for customers and railway administrators. By transitioning from manual systems to an advanced digital platform, the RRMS significantly enhances the overall experience of railway ticket management.

Railway reservation management system is an online ticket booking website, which is capable of booking ticket and search the train availability.

The system features a centralized database that securely stores all essential data, including train schedules, ticket availability, booking details, and user profiles. This centralized approach ensures real-time synchronization and accuracy across all operations. The RRMS provides an intuitive website interface where users can easily search for train availability, view ticket prices, and make reservations online. The system displays real-time updates on ticket availability, helping users make informed decisions efficiently.

Registered users benefit from personalized access, allowing them to view their booking history and reservation details through a unique booking ID. The system is designed to accommodate multiple user categories, including individual passengers and railway administrators, each with tailored functionalities to meet their specific needs. For administrators, the system simplifies the management of train schedules, ticket inventory, and user data.

In summary, the Railway Reservation Management System is an indispensable tool for modernizing railway operations. It reduces the dependency on manual processes, minimizes errors, and enhances customer satisfaction by offering a seamless and reliable ticket booking experience. By integrating advanced digital technologies and ensuring robust data management, the RRMS represents a significant leap toward the digitization of railway services.

ACKNOWLEDGEMENT

At the onset I must bow down in reverence to the almighty that blessed us with the understanding & prevalence that is needed in this kind of project report.

With great pleasure I express my heartiest thanks to **Mr. Durgesh Nandan Pathak** (Head – MCA Deptt.). I would like to extend my sincere thanks to **Mr. Saiyam Varshney** [Designation– Deptt. of Computer Science Studies], without whose unrelated support and guidance, this project would just not have been possible. I am very thankful for his invaluable guidance, support, and affable & friendly nature. He/She guided me at each and every stage of project.

I am equally indebted to my friends who always inspired and motivated me to do something better through out this project.

At last I would like to extend my sincere thanks to all the respondents to whom I visited for giving their support and valuable information, which helps me in completing my project work.

TRAPTI KASHYAP

MCA III Sem

Roll No. 2301730140034

TABLE OF CONTENTS

PREFACE.....	i
CERTIFICATE OF GUIDANCE.....	ii
STUDENT'S DECLARATION.....	iii
REPORT OF SUPERVISOR CERTIFICATON.....	iv
ABSTRACT.....	v
ACKNOWLEDGEMENT.....	vi
1. INTRODUCTION.....	2
1.1 INTRODUCTION OF THE PROJECT.....	2
1.2 PROBLEM STATEMENT.....	2
1.3 OBJECTIVE OF THE PROJECT.....	3
1.4 FEATURES OF THE PROJECT.....	4
1.5 SCOPE OF THE PROJECT.....	5
2. FEASIBILITY STUDY.....	7
2.1 WHAT IS FEASIBILITY STUDY.....	7
2.2 THREE PHASES OF FEASIBILITY STUDY.....	7
2.3 STEPS INVOLVED IN FEASIBILITY ANALYSIS.....	8
3. SOFTWARE REQUIREMENT SPECIFICATION.....	12
3.1 INTRODUCTION TO SRS.....	12
3.2 SOFTWARE REQUIREMENTS.....	12
3.3 HARDWARE REQUIREMENTS.....	13
4. SYSTEM DESIN.....	15
4.1 PROCESS FLOW DIAGRAM.....	15
4.2 ER DIAGRAM.....	16
4.3 DATA FLOW DIAGRAM.....	18
5. TOOLS AND TECHNOLOGY.....	22
5.1 DEVELOPMENT TOOLS.....	22
5.2 TECHNOLOGY USED.....	23
5.3 WEB-SERVER.....	24

6. SCREENSHOTS.....	26
7. CODING.....	32
config.php.....	32
index.php.....	33
register.php.....	37
login.php.....	38
find_trains.php.....	39
reservation.php.....	45
payment.php.....	53
history.php.....	59
about.hhtml.....	65
8. TESTING.....	70
8.1 WHAT IS TESTING.....	70
8.2 TYPES OF TESTING.....	70
8.3 TESTCASES OF THE PROJECT.....	72
9.CONCLUSION.....	75
REFERENCES	76

TABLE OF FIGURES

Figure 1 FLOWCHART	15
Figure 2 COMPONENTS OF AN ER DIAGRAM	16
Figure 3 ER DIAGRAM	17
Figure 4 SYMBOLS FOR DFD.....	18
Figure 5 LEVEL 0 DFD	19
Figure 6 LEVEL 1 DFD.....	19
Figure 7 LEVEL 2 DFD.....	20
Figure 8 HOME PAGE.....	26
Figure 9 SIGNUP PAGE.....	26
Figure 10 LOGIN PAGE.....	27
Figure 11 SELECT TRAIN PAGE.....	27
Figure 12 RESERVATION PAGE.....	28
Figure 13 PAYMENT PAGE.....	28
Figure 14 BOOKING HISTORY PAGE	29
Figure 15 ABOUT PAGE.....	29
Figure 16 DATABASE 1.....	30
Figure 17 DATABSE 2.....	30

CHAPTER-1

1. INTRODUCTION

1.1 INTRODUCTION OF THE PROJECT

The Railway Reservation Management System is a web-based application designed to transform the traditional railway ticketing and reservation process into an efficient, automated, and user-friendly system. This project aims to simplify the experience of booking train tickets for passengers while providing railway administrators with tools to manage schedules, reservations, and ticket availability seamlessly.

The system addresses the limitations of manual ticketing methods, such as errors, delays, and inefficiencies, by offering a centralized platform that integrates various functionalities. Users can search for trains, check seat availability, view ticket prices, and make reservations from the convenience of their homes or offices. Additionally, registered users can track their booking details and history, ensuring a personalized and transparent experience.

For railway administrators, the system provides capabilities to manage train schedules, monitor reservations, and maintain the integrity of user data efficiently. The centralized database ensures that all information, such as train schedules, ticket inventories, and user details, is stored securely and updated in real time.

The Railway Reservation Management System is a step forward in digitizing transportation services, making them accessible and reliable for users while supporting the operational needs of the railway industry. By leveraging advanced technology, this system aims to enhance customer satisfaction and streamline administrative processes, contributing to a more modern and effective railway service.

1.2 PROBLEM STATEMENT

The traditional railway ticketing and reservation system faces several challenges that hinder its efficiency and effectiveness. Manual processes are time-consuming, prone to errors, and lack real-time synchronization, often resulting in discrepancies in ticket availability and reservation status. Passengers frequently experience

difficulties in accessing updated train schedules, checking seat availability, and booking tickets without visiting physical counters. This not only causes inconvenience but also creates long queues and delays.

For railway administrators, managing train schedules, ticket inventories, and user data manually is a cumbersome task that can lead to inefficiencies and mismanagement. Additionally, there is a lack of transparency and accountability in manual systems, which affects the overall user experience and trust in the service. The need for a modern, automated solution is evident to address these issues and provide a streamlined process for passengers and administrators alike.

1.3 OBJECTIVE OF THE PROJECT

The objective of the Railway Reservation Management System is to develop a user-friendly and efficient digital platform that automates the process of booking and managing railway tickets. The system aims to provide passengers with real-time access to train schedules, ticket availability, and fare details, enabling them to make reservations conveniently from anywhere. Additionally, it seeks to streamline administrative tasks for railway personnel by offering tools to manage train schedules, monitor ticket inventories, and maintain user data securely.

This project is designed to enhance the overall experience of railway services by ensuring accuracy, reducing manual errors, and promoting transparency in operations. By leveraging modern technologies, the system aspires to improve accessibility for passengers while simplifying and optimizing the reservation and ticketing process for administrators, ultimately contributing to the modernization of railway operations.

1.4 FEATURES OF THE PROJECT

The Railway Reservation Management System offers an extensive range of features to enhance user convenience and operational efficiency. These features include:

(a) User Authentication

- **Login:** Registered users can securely log in using their email and password to access their accounts.
- **Signup:** New users can easily create accounts by providing required details and setting up their login credentials.

(b) Train Search and Information

- **Find Trains:** Passengers can search for trains by specifying departure and arrival stations along with their travel dates.
- **Train Details:** The system provides comprehensive train information, including schedules, routes, seat availability, and class options.

(c) Reservation and Booking

- **Seat Selection:** Users can choose their preferred seats and select from various seat classes like 1AC, 2AC, 3AC, Sleeper, and General.
- **Multiple Bookings:** The system supports group travel by allowing multiple passenger bookings in a single transaction.

(d) Payment Gateway

- **Payment Processing:** Integration with a secure payment gateway enables smooth online transactions using credit or debit cards.
- **Card Verification:** Users can safely enter their card details for processing payments, including card number, name, expiry date, and CVV.

(e) Booking Confirmation and PNR Generation

- **Confirmation:** Passengers receive a booking confirmation and a unique Passenger Name Record (PNR) upon successful payment.
- **PNR Tracking:** Users can track their booking status or modify their reservations using the PNR.

(f) Booking History

- **Booking Records:** The system maintains a detailed history of all past and upcoming bookings for users to access anytime.

(g) User Profile Management

- **Profile Updates:** Users can manage and update their contact information, preferences, and password through the system.

(h) Admin Panel

- **User Management:** Administrators can manage user accounts, approve or reject registrations, and handle account suspensions.
- **Booking Oversight:** Admins can oversee booking records, address issues, and manage reservation or cancellation processes.

1.5 SCOPE OF THE PROJECT

Railway passengers frequently need to know about their ticket reservation status, ticket availability on a particular train or for a place, train arrival or departure details, special trains etc. Customer information centers at the railway stations are unable to serve such queries at peak periods.

The number of the reservation counters available to the passengers and customers are very less. On most of the reservation systems there are long queues, so it takes a long time for any individual to book the ticket. As now there are no call centers facilities available to solve the queries of the passengers.

The online railway ticket reservation system aims to develop a web application which aims at providing trains details, trains availability, as well as the facility to book ticket in online for customers. So, we thought of developing a web-based application which would provide the users all these facilities from his terminal only as well as help them in booking their tickets.

The Application was to be divided into two parts namely the user part, and the administrator part. And each of these has their corresponding features. We decided to give the name of the website railway reservation management system.

CHAPTER-2

2. FEASIBILITY STUDY

2.1 WHAT IS FEASIBILITY STUDY

A feasibility study is a comprehensive analysis and evaluation of the potential success of a proposed project. It is conducted before the project is officially launched to determine whether the project is viable, practical, and economically and technically feasible. The purpose of a feasibility study is to provide decision-makers with the necessary information to make informed choices about whether to proceed with the project or not.

A feasibility study—sometimes called a feasibility analysis or feasibility report—is a way to evaluate whether a project plan could be successful. A feasibility study evaluates the practicality of your project plan in order to judge whether you can move forward with the project.

2.2 THREE PHASES OF FEASIBILITY STUDY

(a) Technical Feasibility

The Railway Reservation Management System is technically feasible given the availability of advanced web development technologies and secure payment gateway integration. The project utilizes existing resources such as programming languages (e.g., HTML, CSS, PHP, and JavaScript), a centralized database system, and hosting platforms capable of supporting high-traffic web applications. Additionally, expertise in software development and database management ensures that the necessary technical requirements can be met within the allocated budget and timeline.

(b) Financial Feasibility

The project demonstrates strong financial feasibility, with the potential to reduce long-term operational costs associated with manual ticketing systems. Initial investments include development, infrastructure setup, and training costs, while operating expenses involve server maintenance and periodic updates. Revenue generation can be facilitated through streamlined operations, improved customer satisfaction, and potential additional services. Financial metrics such as break-even

analysis and return on investment (ROI) indicate that the project is economically viable in the long run.

(c) Operational Feasibility

The system is designed to integrate seamlessly into the existing railway operations without disrupting daily activities. With an intuitive interface, minimal training is required for users and administrators. The system's centralized nature ensures consistency and efficiency in managing ticket reservations and schedules. Furthermore, the platform supports scalability, allowing it to adapt to future increases in user demand or additional features.

The feasibility analysis concludes that the Railway Reservation Management System is a practical, financially sound, and operationally viable solution for modernizing railway ticketing and management processes.

2.3 STEPS INVOLVED IN FEASIBILITY ANALYSIS

A feasibility study typically involves several key steps to comprehensively evaluate the viability of a proposed project. The specific steps may vary based on the nature of the project and the organization's preferences, but here is a general outline of the common steps involved in a feasibility study:

(a) Project Definition and Scope:

- Clearly define the objectives and goals of the project.
- Define the scope of the project, including its deliverables, functionalities, and features.

(b) Preliminary Analysis:

- Conduct an initial assessment of the project's potential benefits, risks, and challenges.
- Identify the stakeholders involved and their interests.

(c) Market Analysis:

- Analyze the target market to understand its size, demographics, trends, and demand for the product or service.
- Identify potential customers and their preferences.

- Evaluate the competitive landscape and assess competitors' strengths and weaknesses.

(d) Technical Analysis:

- Assess the technical requirements of the project.
- Evaluate the project's technical feasibility and potential challenges.

(e) Financial Analysis:

- Estimate the initial investment costs, including equipment, resources, and development expenses.
- Forecast operating costs, including personnel, maintenance, and other ongoing expenses.
- Project potential revenue based on market demand and pricing strategies.
- Calculate financial metrics such as Net Present Value (NPV), Internal Rate of Return (IRR), Payback Period, and Return on Investment (ROI).

(f) Operational Analysis:

- Assess how the project will integrate into existing operations and workflows.
- Evaluate staffing requirements, training needs, and organizational changes.
- Identify any operational bottlenecks or challenges that could arise.

(g) Legal and Regulatory Analysis:

- Identify relevant laws, regulations, and permits that the project needs to comply with.
- Assess potential legal obstacles or risks.

(h) Environmental and Social Analysis:

- Evaluate the project's potential environmental impacts and sustainability considerations.
- Consider social and ethical implications of the project on the community and stakeholders.

(i) Resource Analysis:

- Identify and evaluate the availability of necessary resources, including human resources, raw materials, and suppliers.
- Determine the feasibility of securing and managing these resources.

(j) Risk Assessment:

- Identify potential risks and uncertainties that could impact the project's success.
- Analyze the potential impact of these risks and develop strategies to mitigate them.

(k) Conclusion and Recommendation:

- Summarize the findings from each analysis.
- Make a recommendation on whether to proceed with the project, modify it, or abandon it.
- Provide a rationale for the recommendation based on the feasibility study's results.

(l) Final Report:

- Compile all the findings, analyses, and recommendations into a comprehensive feasibility study report.
- Present the report to decision-makers and stakeholders for review and approval.

CHAPTER-3

3. SOFTWARE REQUIREMENT SPECIFICATION

3.1 INTRODUCTION TO SRS

A Software Requirements Specification (SRS) is a comprehensive document that outlines the functional and non-functional requirements of a software system, serving as a blueprint for development. It details the purpose, scope, target audience, and features of the software, ensuring all stakeholders—developers, testers, clients, and project managers—share a common understanding of the project. The SRS typically includes user requirements, system requirements, interface descriptions, constraints, and performance metrics. By providing a clear and structured framework, the SRS minimizes ambiguities, reduces development risks, and serves as a reference throughout the software lifecycle.

3.2 SOFTWARE REQUIREMENTS

Software Requirement for Server:

Web Server	-	Apache (XAMPP)
Back End	-	PHP
Front End	-	HTML, CSS
Operating System	-	Windows 10 and above
IDE	-	Visual Studio
Database	-	MySQL

Software Requirement for Client's:

Connection	-	Having well-establish internet connection
Operating System	-	Windows 10 and above

Web Browser - Chrome, Edge, etc.

3.3 HARDWARE REQUIREMENTS

Hardware Requirement for Client's:

Monitor/Display - 13" LCD monitor

Keyboard - Qwerty

Processor - Dual-Core 2Ghz or higher

Hard Disk - 32GB

Minimum Free Space - 500 MB free disk space

Minimum RAM - 2GB RAM

Hardware Requirement for Server:

Monitor/Display - 13" LCD monitor

Keyboard - Qwerty

Processor - Dual-Core 2Ghz or higher

Hard Disk - 32GB

Minimum Free Space - 500 MB free disk space

Minimum RAM - 2GB RAM

CHAPTER-4

4. SYSTEM DESIN

4.1 PROCESS FLOW DIAGRAM

Process Flow Diagram and Flowchart is a diagram which uses geometric symbols and arrows to define relationships. It is a diagrammatic representation of the algorithm. The Process Flow Diagram of our application is shown below.

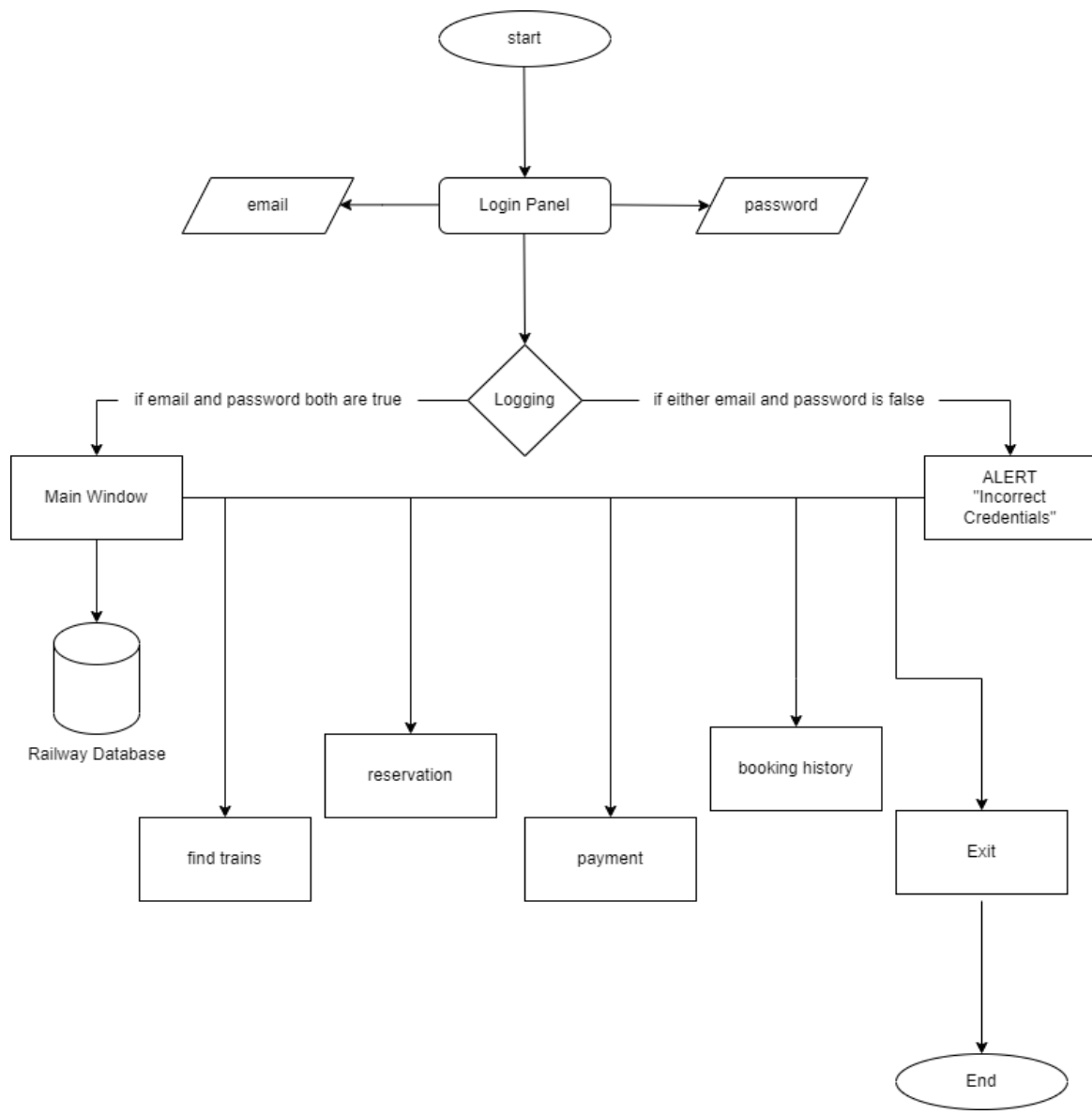


Figure 1 FLOW CHART

4.2 ER DIAGRAM

An ER (Entity-Relationship) diagram is a visual representation of the entities in a system and the relationships between them. It's commonly used in database design to map out the structure of the data and how different entities interact.

Here's a brief explanation of the components of an ER diagram:

Components of an ER Diagram:

(a) Entities: These are the objects or things in the system that have data stored about them. For example:

- Student
- Course
- Employee

Entities are usually represented by rectangles.

(b) Attributes: These are the data or properties that describe the entities. For example:

- A Student entity might have attributes like Student_ID, Name, Age.
- An attribute is typically represented by an oval.

(c) Relationships: These represent the interactions or associations between entities. For example:

- A Student "enrolls" in a Course.

Relationships are typically represented by diamonds.

Entity Set 	Strong Entity Set	
	Weak Entity Set	
Attributes 	Simple Attribute	
	Composite Attribute	
	Single-valued Attribute	
	Multivalued Attribute	
	Derived Attribute	
	Null Attribute	
Relationship 	Strong Relationship	
	Weak Relationship	

Figure 2 COMPONENTS OF AN ER DIAGRAM

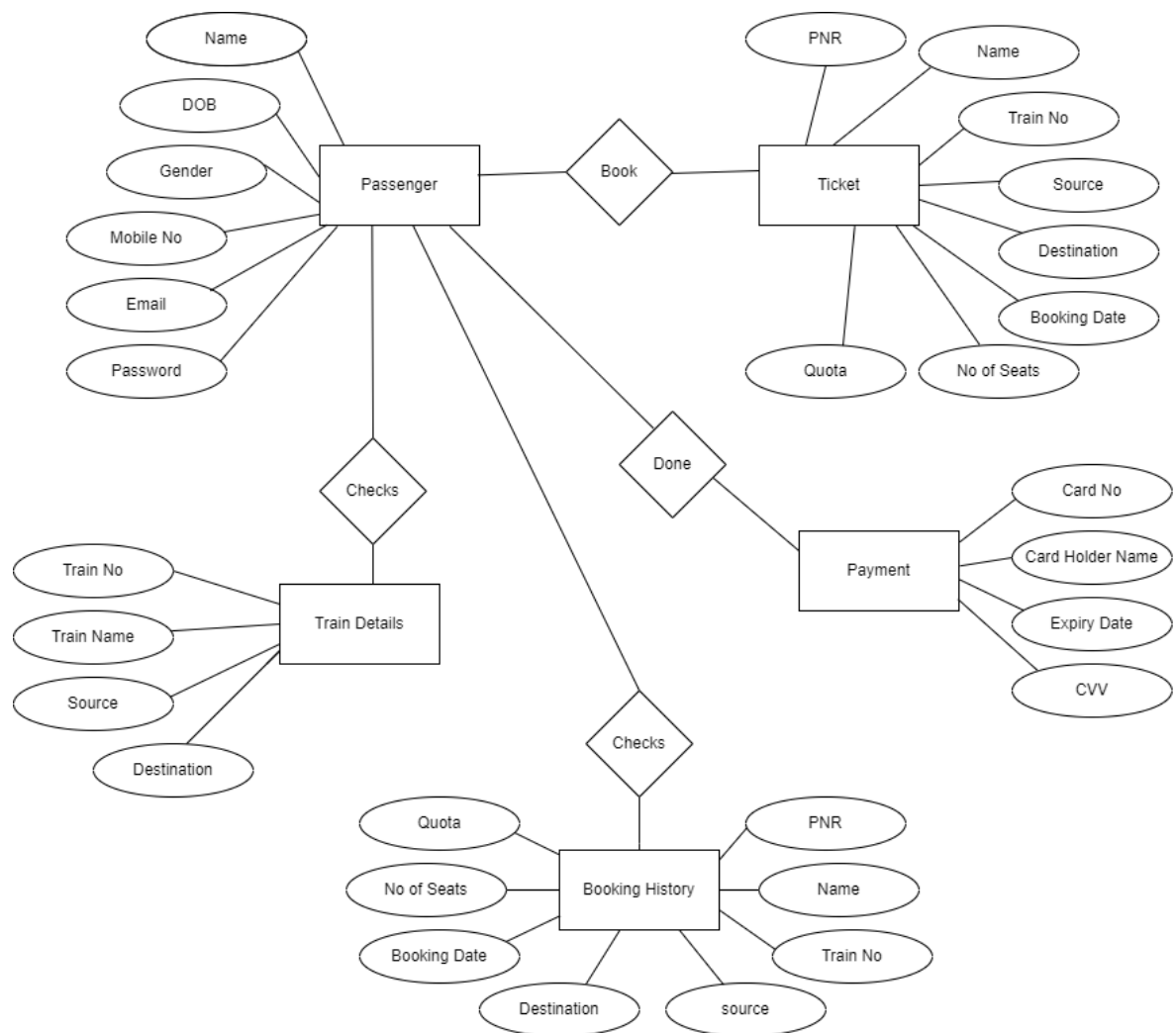



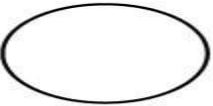


Figure 3 ER DIAGRAM

4.3 DATA FLOW DIAGRAM

A DFD (Data Flow Diagram) is a visual representation that illustrates how data moves within a system, showing how inputs are transformed into outputs through various processes. It is commonly used in software engineering and system analysis to model the flow of data within an application or business process.

Key Components of a DFD:

- (a) Processes:** These represent activities or functions that transform input data into output. Processes are typically represented by circles or rounded rectangles.
- (b) Data Stores:** These are repositories where data is stored for later use. They are usually represented by open-ended rectangles.
- (c) Data Flows:** These represent the movement of data between processes, data stores, and external entities. Data flows are represented by arrows.
- (d) External Entities:** These are sources or destinations of data that are outside the system being modeled. They are typically represented by rectangles or squares.

Symbol	Name	Function
	Data flow	Used to Connect Processes to each other, to sources or Sinks; te arrow head indicates direction of data flow.
	Process	Performs Some transformation of Input data to yield output data.
	Source of Sink (External Entity)	A Source of System inputs or Sink of System outputs.
	Data Store	A repository of data; the arrow heads indicate net inputs and net outputs to store.

Symbols for Data Flow Diagrams

Figure 4 SYMBOLS FOR DFD

Levels of DFD:

(a) Level 0: This is the highest-level DFD, showing the entire system as a single process, with external entities and data flows. The Level 0 DFD of our application is shown below.



Figure 5 LEVEL 0 DFD

(b) Level 1: This level breaks the system into major sub-processes, showing data flows between these processes, data stores, and external entities. The Level 1 DFD of our application is shown below.

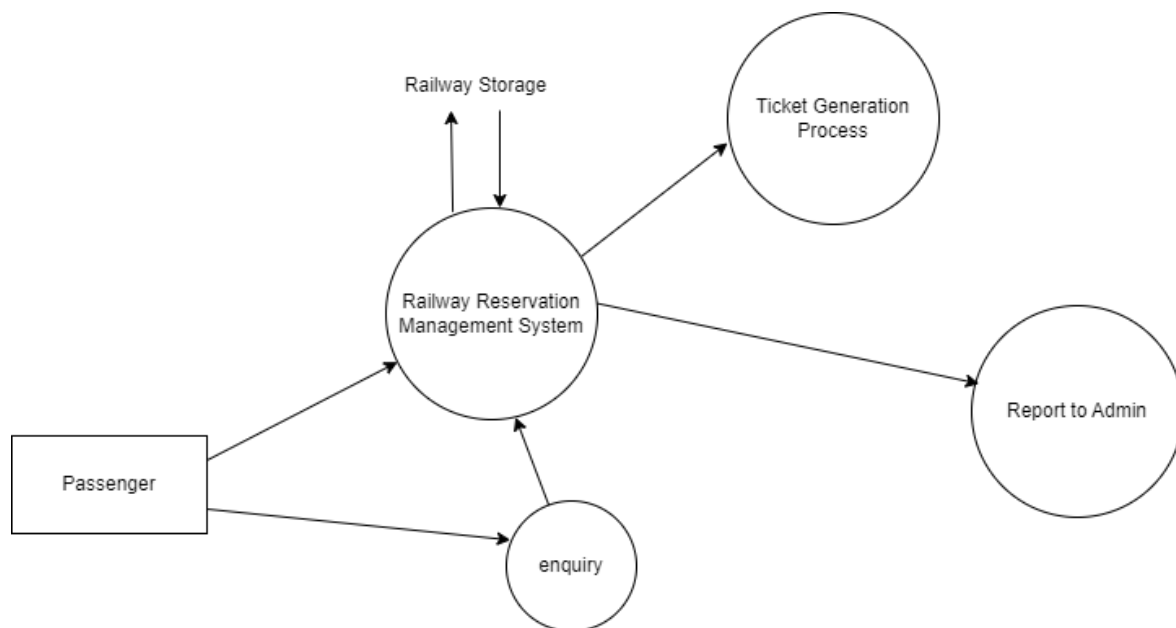


Figure 6 LEVEL 1 DFD

(c) Level 2: These diagrams further decompose each process into more detailed sub-processes to show the finer level of data flow and interactions. The Level 2 DFD of our application is shown below.

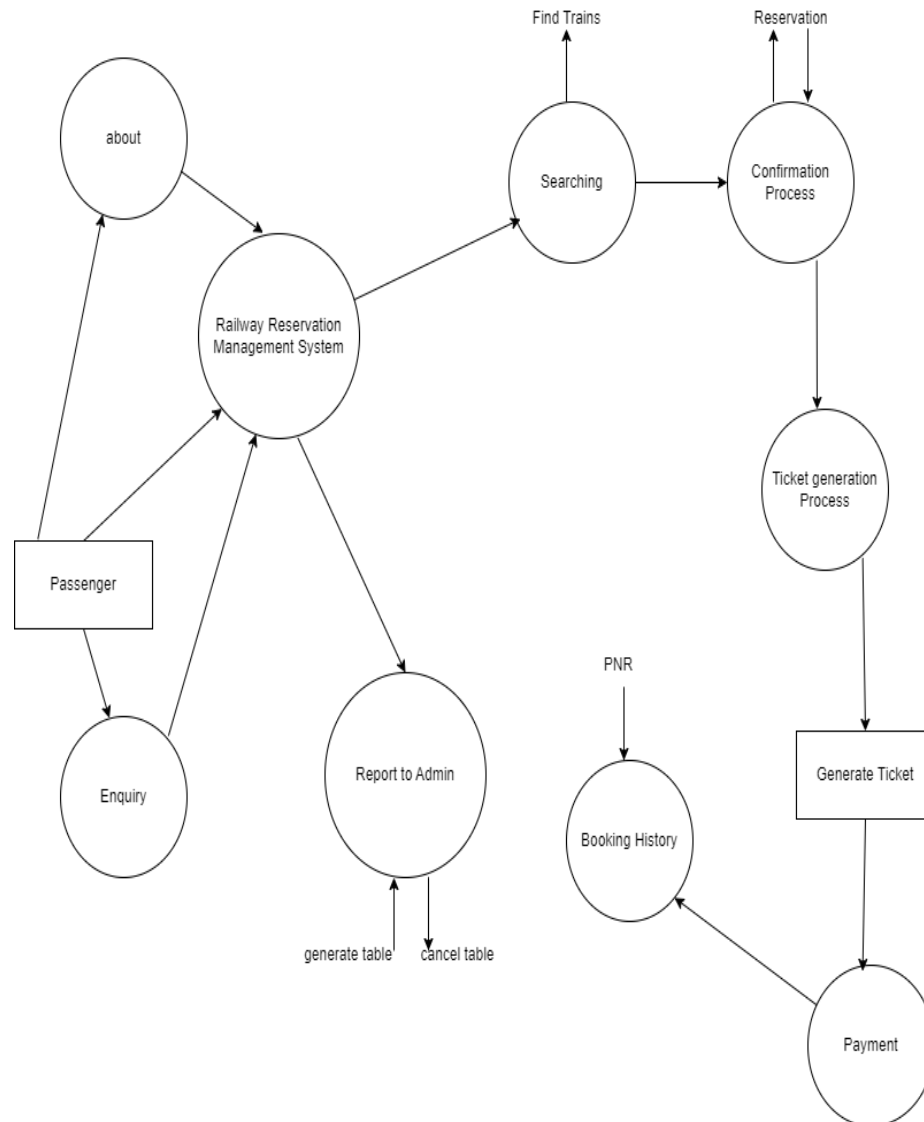


Figure 7 LEVEL 2 DFD

CHAPTER-5

5. TOOLS AND TECHNOLOGY

5.1 DEVELOPMENT TOOLS

VISUAL-STUDIO-CODE:

Visual Studio Code (VS Code) is a popular, lightweight, and versatile code editor developed by Microsoft. It supports a wide range of programming languages and is equipped with features like syntax highlighting, intelligent code completion, debugging, and Git integration, making it a powerful tool for developers. VS Code's extensibility allows users to enhance its functionality by adding extensions for specific frameworks, languages, or tools, enabling customization for various development needs. Its user-friendly interface and cross-platform compatibility (Windows, macOS, and Linux) make it an ideal choice for developers of all levels, from beginners to professionals. Whether you're building websites, writing scripts, or developing software, VS Code provides a seamless and efficient coding experience. VS Code provided integrated terminal to run shell commands without leaving the editor and also Out-of-the-box support for languages like JavaScript, Python, C++, and many more.

XAMPP:

XAMPP is a free, open-source software package that provides an easy-to-install environment for web development. It includes essential tools like Apache (a web server), MySQL (a database management system), PHP (a server-side scripting language), and **Perl**. XAMPP is designed to allow developers to set up a local server environment on their computers for testing and development purposes, without the need for complex configuration. It is especially popular among developers working with PHP-based applications, as it simplifies the process of creating, testing, and debugging websites and web applications locally before deploying them to a live server. XAMPP supports cross-platform use, making it available for Windows, macOS, and Linux.

5.2 TECHNOLOGY USED

PHP:

PHP (Hypertext Preprocessor) is a widely-used, open-source server-side scripting language designed primarily for web development. It is embedded within HTML and is especially suited for creating dynamic and interactive web pages. PHP is known for its simplicity, flexibility, and extensive library of functions, making it a favorite among developers. It efficiently interacts with databases like MySQL, PostgreSQL, and others, enabling the development of robust web applications such as content management systems, e-commerce platforms, and social networking sites. With its broad community support, frequent updates, and compatibility with various operating systems, PHP remains a reliable and cost-effective solution for web development.

MYSQL-SERVER/phpMyAdmin):

MySQL, a powerful and widely-used open-source relational database management system, is often paired with PHP to create dynamic and data-driven web applications. It allows developers to efficiently store, retrieve, and manage large volumes of structured data. SQL (Structured Query Language) is used to interact with MySQL databases, providing commands to query and manipulate data. phpMyAdmin, a user-friendly web-based interface written in PHP, simplifies the management of MySQL databases. It allows developers to perform database tasks such as creating tables, running queries, and managing users without requiring extensive command-line knowledge. Together, MySQL and phpMyAdmin provide a seamless and effective environment for developing, testing, and managing web applications.

HTML:

HTML (HyperText Markup Language) is the standard language for creating and structuring content on the web. It provides the foundation for web pages by using elements and tags to define text, images, links, and other multimedia content. HTML is easy to learn and forms the backbone of web development, often working

alongside CSS for styling and JavaScript for interactivity. It is versatile and widely supported, enabling developers to build websites that are accessible across various devices and browsers. With advancements like HTML5, it supports modern web features such as multimedia playback, graphics rendering, and responsive design.

CSS:

CSS (Cascading Style Sheets) is a stylesheet language used to control the presentation and layout of web pages written in HTML. It allows developers to apply styles such as colors, fonts, spacing, and positioning to elements, creating visually appealing and organized websites. By separating content (HTML) from design (CSS), it enhances the maintainability and scalability of web projects. CSS supports responsive design, enabling web pages to adapt seamlessly to different screen sizes and devices. With features like animations, transitions, and grid layouts, CSS adds dynamic and interactive elements to web pages.

5.3 WEB-SERVER

APACHE:

Apache (or Apache HTTP Server) is a widely-used open-source web server software that serves content over the web. It is responsible for handling HTTP requests from clients (typically web browsers), processing those requests, and returning the appropriate content, such as HTML pages, images, or other files. Apache is highly configurable and can run on various operating systems, including Linux, Windows, and macOS.

As a web server, Apache is one of the most popular choices for hosting websites and web applications. It can be used to serve both static content (like HTML files) and dynamic content (such as PHP or CGI scripts). Apache also supports features like URL rewriting, access control, security configurations, and modules for extending its functionality, such as integrating with PHP or SSL for secure connections.

In summary, Apache is the server that hosts and delivers web content to users, playing a critical role in the infrastructure of most websites.

CHAPTER-6

6. SCREENSHOTS

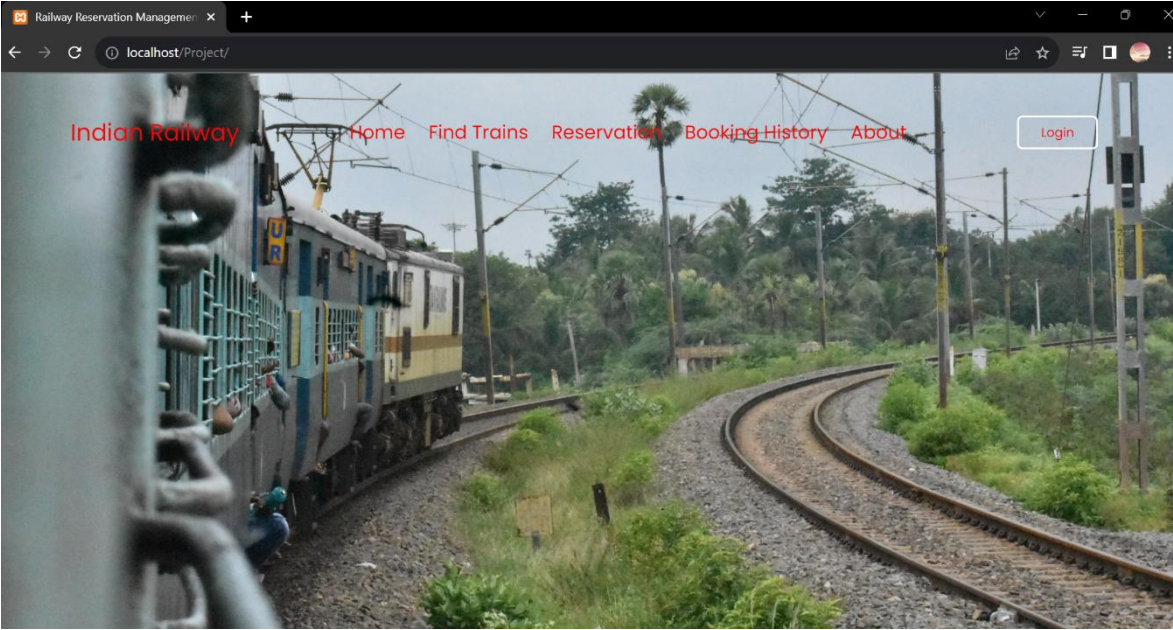


Figure 8 HOME PAGE

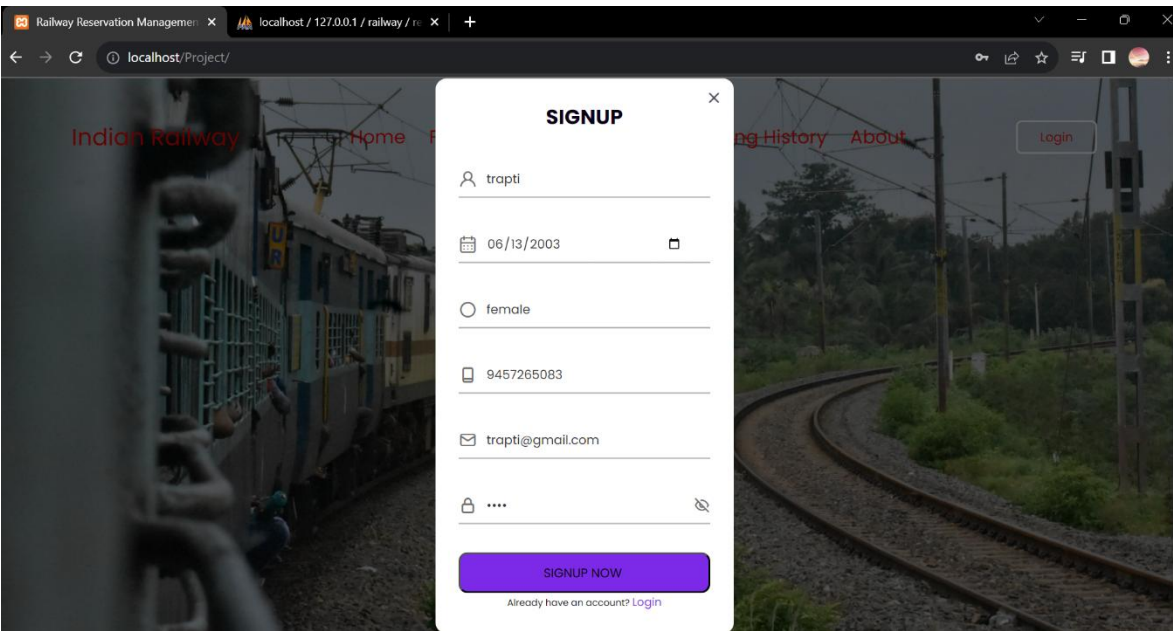


Figure 9 SIGNUP PAGE

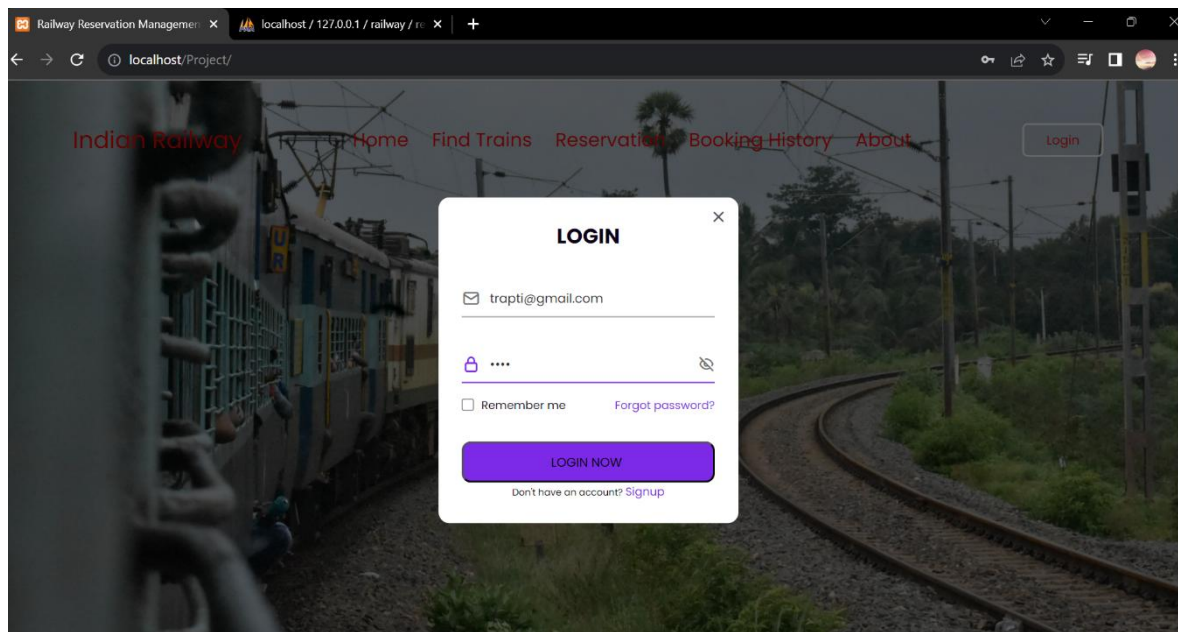


Figure 10 LOGIN PAGE

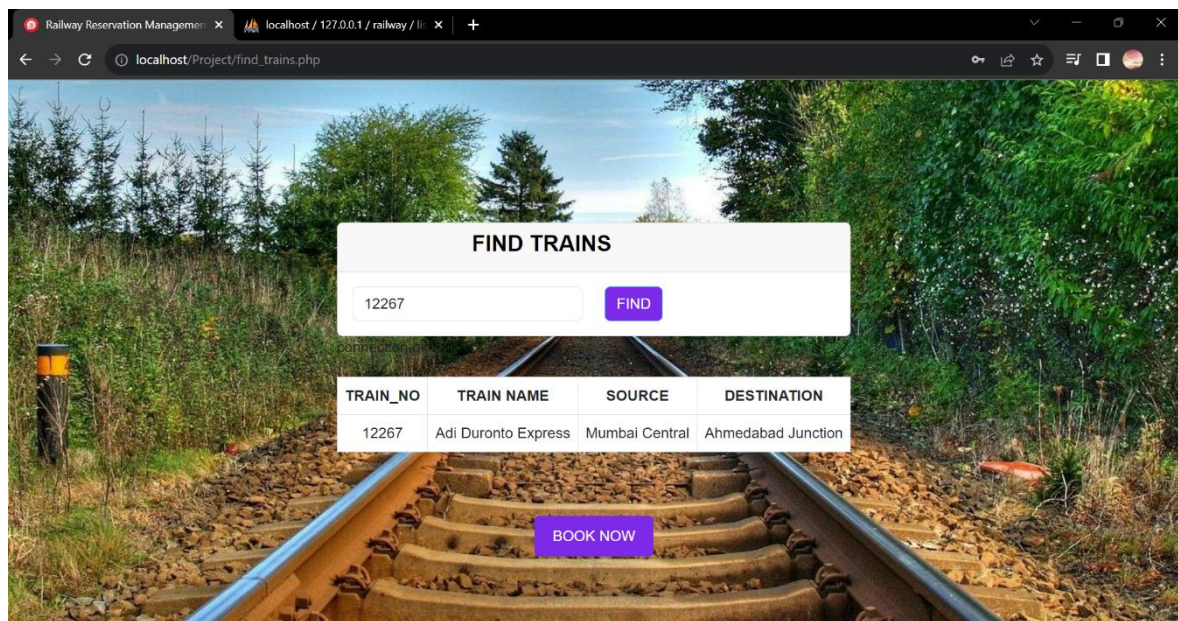


Figure 11 SELECT TRAIN PAGE

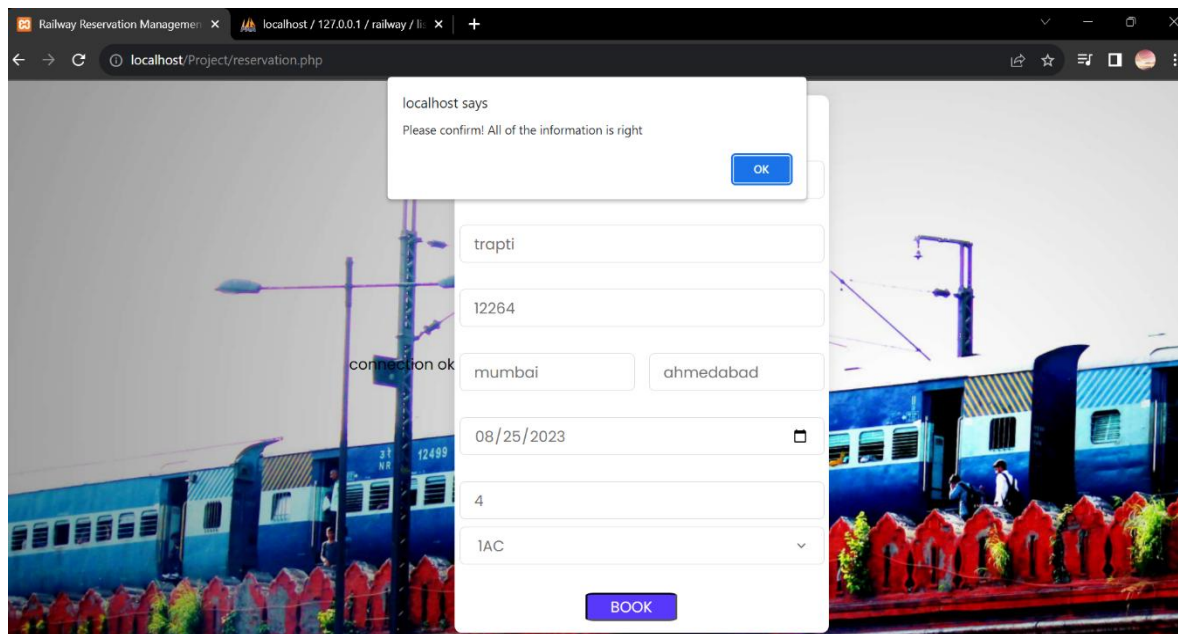


Figure 12 RESERVATION PAGE

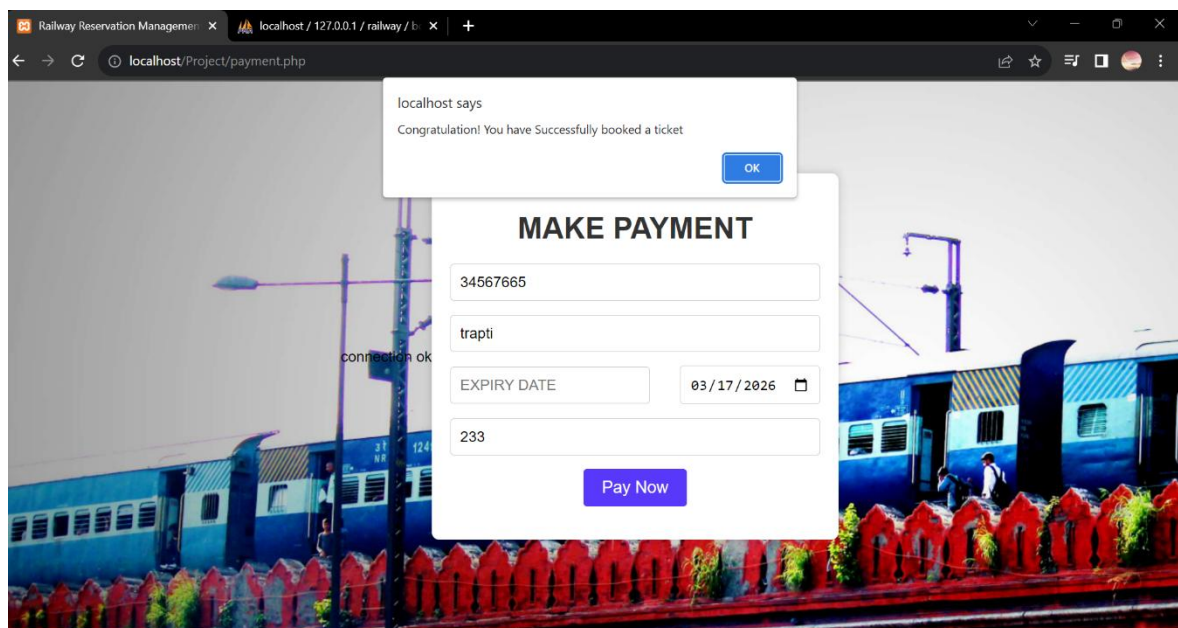


Figure 13 PAYMENT PAGE

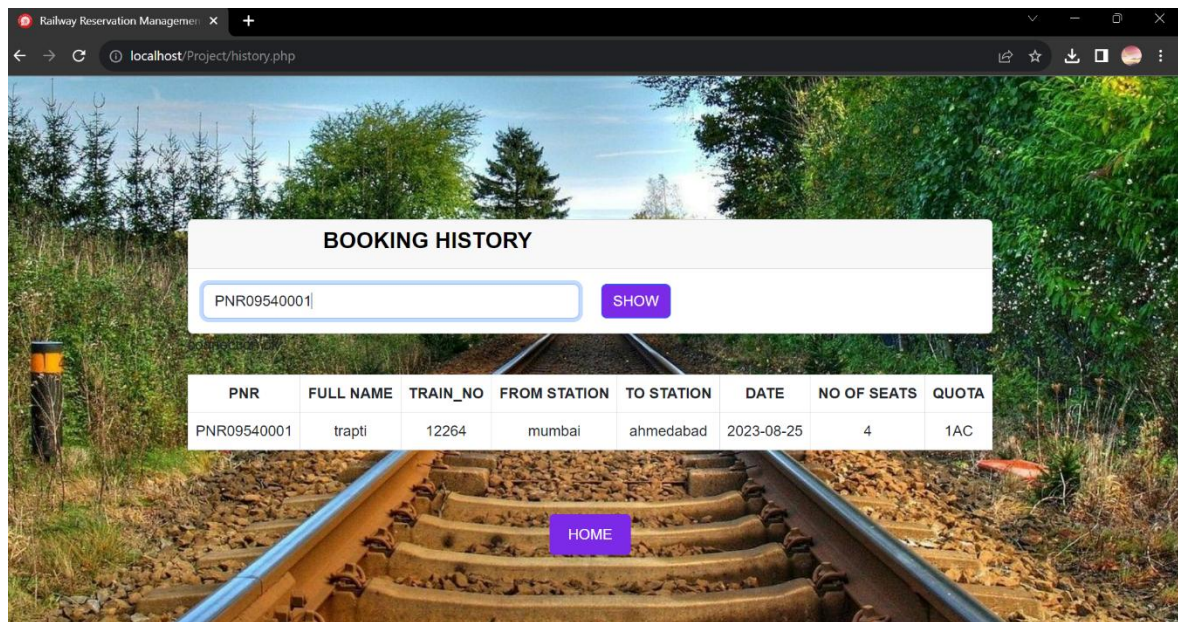


Figure 14 BOOKING HISTORY PAGE

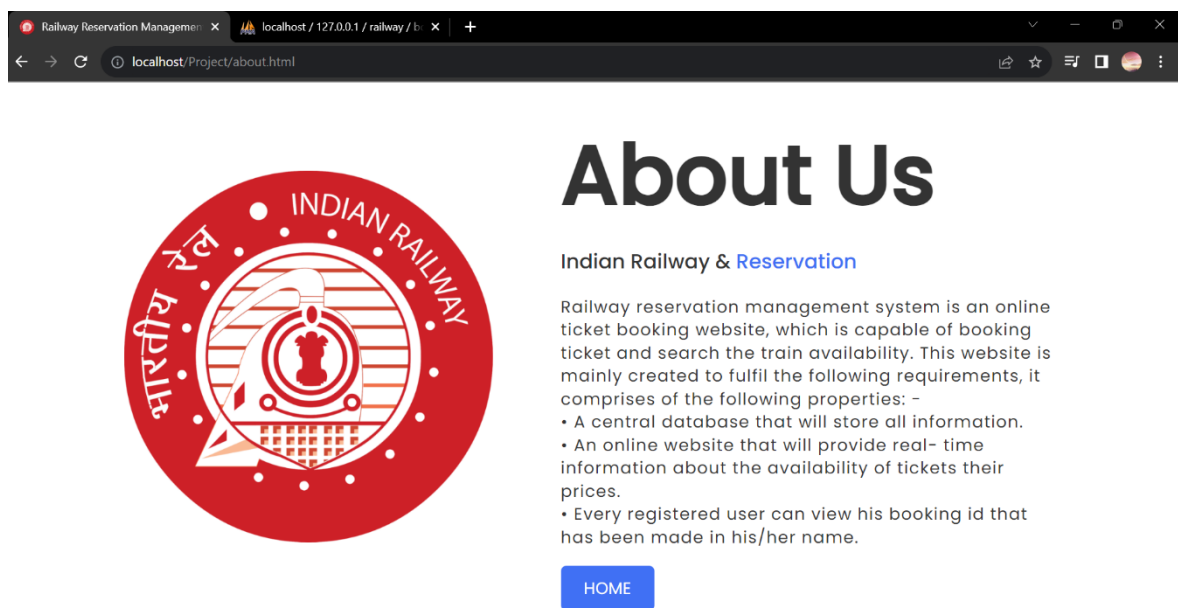


Figure 15 ABOUT PAGE

			SNO	U_NAME	DOB	GENDER	MOBILE_NO	EMAIL	PASS_WORD				
<input type="checkbox"/>		Edit		Copy		Delete	1	charu	2004-03-06	female	9275892132	charu@gmail.com	0606
<input type="checkbox"/>		Edit		Copy		Delete	2	bharti	2003-08-04	female	8123247254	bharti@gmail.com	0303
<input type="checkbox"/>		Edit		Copy		Delete	3	trapti	2003-06-13	female	7825388634	trapti@gmail.com	1313
<input type="checkbox"/>		Edit		Copy		Delete	4	preeti	2023-08-08	female	9812345678	preeti@gmail.com	1234
<input type="checkbox"/>		Edit		Copy		Delete	5	neha	2002-04-24	female	7612345678	neha@gmail.com	2424
<input type="checkbox"/>		Edit		Copy		Delete	6	mahima	2023-08-08	female	6712345678	mahima@gmail.com	0880
<input type="checkbox"/>		Edit		Copy		Delete	7	ravi	2023-08-09	male	7735475287	ravi@gmail.com	4321
<input type="checkbox"/>		Edit		Copy		Delete	8	trapti	2003-06-13	female	9457265083	traptik@gmail.com	1306
<input type="checkbox"/>		Edit		Copy		Delete	9	bhawna	2002-11-18	female	7932524758	bhawna@gmail.com	6767

☐

Check all

With selected:

Edit

Copy

Delete

Export

☐

Show all

Number of rows:

25

Filter rows:

Search this table

Sort by key:

None

Figure 16 DATABASE 1

				TRAIN_NO	TRAIN NAME	SOURCE	DESTINATION
<input type="checkbox"/>				04101	Prayagraj Sangam Kanpur Anwarg	Prayagraj Sangam	Kanpur Anwarganj
<input type="checkbox"/>				05414	Agra Fort Kasganj Special	Agra Fort	Kasganj Junction
<input type="checkbox"/>				07166	Cuttack Hyderabad Special	Cuttack Junction	Hyderabad Deccan
<input type="checkbox"/>				09007	Bct Bikaner Special	Mumbai Central	Bikaner Junction
<input type="checkbox"/>				1111	Bharatpur memu	mathura	bharatpur
<input type="checkbox"/>				12267	Adi Durlonto Express	Mumbai Central	Ahmedabad Junction
<input type="checkbox"/>				12609	Bangalore Express	Chennai Centra	Bangalore Cy Junction
<input type="checkbox"/>				12722	Dakshin Super Fast Express	Hazrat Nizamuddin	Hyderabad Deccan
<input type="checkbox"/>				12723	Andhra Pradesh Express	Hyderabad Decan	New Delhi
<input type="checkbox"/>				12954	Ag Kranti Rajdhani	H Nizamuddinl	Mumbai Centra

☐ Check all With selected: Edit Copy Delete Export

Console

Figure 17 DATABASE 2

CHAPTER-7

7.CODING

config.php:

```
<?php
/*
This file contains database configuration assuming you are running mysql using
user "root" and password ""
*/

$SERVER = "localhost";
$USERNAME = "root";
$PASSWORD = "";
$DBNAME = "railway";
// Try connecting to the Database
$conn = mysqli_connect($SERVER, $USERNAME, $PASSWORD, $DBNAME);
//Check the connection
if($conn)
{
    echo "connection ok";
}
else{
    echo "connection failed".mysqli_connect_error();
}
?>

<?php include("config.php");
?>
```

index.php:

```
<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta http-equiv="X-UA-Compatible" content="IE=edge" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<title>Railway Reservation Management System</title>

<link rel="shortcut icon" href="Logo.png" type="image/x-icon">

<link rel="stylesheet" href="style.css" />

<link rel="stylesheet"
href="https://unicons.iconscout.com/release/v4.0.0/css/line.css" />

</head>

<body>

<!-- Header -->

<header class="header">

<nav class="nav">

<a href="#" class="nav_logo">Indian Railway</a>

<ul class="nav_items">

<li class="nav_item">

<a href="index.php" class="nav_link">Home</a>

<a href="find_trains.php" class="nav_link">Find Trains</a>

<a href="reservation.php" class="nav_link">Reservation</a>

<a href="history.php" class="nav_link">Booking History</a>

<a href="about.html" class="nav_link">About</a>

</li>
```

```
</ul>

<button class="button" id="form-open">Login</button>

</nav>

</header>
```

```
<!-- Home -->

<section class="home">

<div class="form_container">

<i class="uil uil-times form_close"></i>
```

```
<!-- Login From -->

<div class="form login_form">

<form action="login.php" method="POST">

<h2>LOGIN</h2>

<div class="input_box">

<input type="email" name="email" autocomplete="off" placeholder="ENTER
YOUR EMAIL" required />

<i class="uil uil-envelope-alt email"></i>

</div>

<div class="input_box">

<input type="password" name="pwd" autocomplete="off" placeholder="ENTER
YOUR PASSWORD" required />

<i class="uil uil-lock password"></i>

<i class="uil uil-eye-slash pw_hide"></i>

</div>

<div class="option_field">

<span class="checkbox">

<input type="checkbox" id="check" />
```

```

<label for="check">Remember me</label>

</span>

<a href="#" class="forgot_pw">Forgot password?</a>

</div>

<input type="submit" class="form_control" name="login" value="LOGIN NOW
"><div class="login_signup">Don't have an account? <a href="#"
id="signup">Signup</a></div>

</form>

</div>

<!-- Signup Form -->

<div class="form signup_form">

<form action="register.php" method="POST">

<h2>SIGNUP</h2>

<div class="input_box">

<input type="text" name="name" autocomplete="off" placeholder="NAME"
required />

<i class="uil uil-user name"></i>

</div>

<div class="input_box">

<input type="date" name="dob" autocomplete="off" placeholder="DOB" required
/>

<i class="uil uil-calendar-alt date"></i>

</div>

<div class="input_box">

<input type="text" name="gender" autocomplete="off" placeholder="GENDER"
required />

<i class="uil uil-circle gender"></i>

```

```

</div>

<div class="input_box">

<input type="tel" name="mbl"  autocomplete="off" placeholder="MOBILE NO"
required />

<i class="uil uil-mobile-android-alt tel"></i>

</div>

<div class="input_box">

<input type="email" name="email"  autocomplete="off" placeholder="ENTER
YOUR EMAIL" required />

<i class="uil uil-envelope-alt email"></i>

</div>

<div class="input_box">

<input type="password" name="pwd"  autocomplete="off" placeholder="CREATE
PASSWORD" required />

<i class="uil uil-lock password"></i>

<i class="uil uil-eye-slash pw_hide"></i>

</div>

<input type="submit" class="form_control" name="insert_value" value="SIGNUP
NOW ">

<div class="login_signup">Already have an account? <a href="#"
id="login">Login</a></div>

</form>

</div>

</div>

</section>

<script src="script.js"></script>

</body>

</html>

```

register.php:

```
<?php
include ('config.php');

if (isset($_POST['insert_value']))
{
$name = $_POST['name'];
$dob = $_POST['dob'];
$gender = $_POST['gender'];
$mbl = $_POST['mbl'];
$email = $_POST['email'];
$pwd = $_POST['pwd'];

$query = "INSERT INTO register
(U_NAME,DOB,GENDER,MOBILE_NO,EMAIL,PASS_WORD )
VALUES ('$name', '$dob', '$gender', '$mbl', '$email', '$pwd')";
$data = mysqli_query($conn,$query);
if($data)
{
header("Location:find_trains.php");
}
else
{
header("Location:index.php");
}
}
?>
```

login.php:

```
<?php
include('config.php');

if (isset($_POST['login']))
{
$email=$_POST['email'];
$pwd=$_POST['pwd'];

$query = "SELECT * FROM register WHERE email = '$email' AND pass_word=
'$pwd' ";
$data = mysqli_query($conn, $query);
if($data)
{
header("Location:find_trains.php");
}
else
{
header("Location:index.php");
}
}
?>
```


find_trains.php:

```
<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Railway Reservation Management System</title>

<link rel="shortcut icon" href="Logo.png" type="image/x-icon">

<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
4bw+/aepP/YC94hEpVNVgiZdgIC5+VKNBQNGCHeKRQN+PtmoHDEXuppvnDJz
Qlu9" crossorigin="anonymous">

<style>

body {

font-family: Arial, sans-serif;

position: relative;

height: 100vh;

width: 100%;

background-image: url("wp.jpg");

background-size: cover;

background-position: center;

justify-content: center;

align-items: center;

margin: 0;

display: flex;

flex-direction: column;

}
```

```
.container {  
margin-top: 20px;  
display: flex;  
justify-content: center;  
align-items: center;  
height: 60vh;  
}  
.card-title {  
color: #000000;  
font-size: 24px;  
font-weight: bold;  
margin-left: 130px;  
}  
.btn-primary {  
background-color: #7d2ae8;  
border-color: #007bff;  
}  
.btn-primary:hover {  
background-color: #0056b3;  
border-color: #0056b3;  
}  
table {  
width: 100%;  
border-collapse: collapse;  
margin-top: 20px;  
}  
th, td {
```

```
padding: 10px;
text-align: center;
}
th {
background-color: #f2f2f2;
font-weight: bold;
}
tr:nth-child(even) {
background-color: #f2f2f2;
}
a {
color: #007bff;
text-decoration: none;
display: block;
margin-top: 20px;
text-align: center;
}
.button {
display: inline-block;
padding: 10px 20px;
background-color: #7d2ae8;
color: white;
text-decoration: none;
border-radius: 5px;
border: none;
cursor: pointer;
margin-top: 20px;
```

```

}
.button:hover {
background-color: #0056b3;
}
</style>
</head>
<body>
<div class="container">
<div class="row">
<div class="col-md-12 mt-4">
<div class="card">
<div class="card-header">
<h4 class="card-title">FIND TRAINS</h4>
</div>
<div class="card-body">
<form action="" method="POST" class="row">
<div class="col-md-6">
<div class="form-group">
<input type="text" name="tn" class="form-control" placeholder="TRAIN NO"
required>
</div>
</div>
<div class="col-md-6">
<button type="submit" name="search_by_tn" class="btn btn-
primary">FIND</button>
</div>
</form>
</div>

```

```

</div>

<div class="table-responsive">
<table class="table table-bordered">
<thead></thead>

<tr>
<th scope="col">TRAIN_NO</th>
<th scope="col">TRAIN NAME</th>
<th scope="col">SOURCE</th>
<th scope="col">DESTINATION</th>
</tr>
</thead>
<tbody>
<?php
include('config.php');
if(isset($_POST['search_by_tn']))
{
$tn = $_POST['tn'];
$query = "SELECT * FROM list WHERE TRAIN_NO = '$tn'";
$query_run = mysqli_query($conn,$query);
if(mysqli_num_rows($query_run)>0)
{
while($row = mysqli_fetch_array($query_run))
{
?>

<tr>

<td><?php echo $row['TRAIN_NO'];?></td>
<td><?php echo $row['TRAIN NAME'];?></td>

```

```

<td><?php echo $row['SOURCE'];?></td>
<td><?php echo $row['DESTINATION'];?></td>
</tr>
<?php
}
}
else
{
?>
<tr>
<td colspan="6">No Record Found</td>
</tr>
<?php
}
}
?>
</tbody>
</table>
</div>
</div>
</div>
</div>
</div>
</div>
</div>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-

```

```
HwwvtgBNo3bZJJLYd8oVXjrBZt8cqVSpeBNS5n7C8IVInixGAoxmnlMuBnhbgrkm"
crossorigin="anonymous"></script>
```

```
</body>
```

```
<a href="reservation.php" class="button">BOOK NOW</a>
```

```
</html>
```

reservation.php:

```
<?php
```

```
include ('config.php');
```

```
error_reporting(0);
```

```
//PNR
```

```
$read = "SELECT*FROM booking ORDER BY PNR DESC LIMIT 1";
```

```
$result = mysqli_query($conn,$read);
```

```
if($result)
```

```
{
```

```
$fetch = mysqli_fetch_assoc($result);
```

```
$lastpnr = $fetch['PNR'];
```

```
if($lastpnr == null)
```

```
{
```

```
$newpnr = "PNR09540000";
```

```
}
```

```
else
```

```
{
```

```
$newpnr = str_replace("PNR0954", "", $lastpnr);
```

```
$newpnr = str_pad($newpnr+1,4,0,STR_PAD_LEFT);
```

```
$newpnr = "PNR0954" . $newpnr;
```

```
}
```

```

}
else
{
    echo "server down";
}
if (isset($_POST['submit']))
{
    $pnr = $_POST['pnr'];
    $na = $_POST['c_na'];
    $tn = $_POST['tn'];
    $fstn = $_POST['fstn'];
    $tstn = $_POST['tstn'];
    $date = $_POST['date'];
    $nos = $_POST['nos'];
    $qu = $_POST['qu'];

    $query = "INSERT INTO booking
    (PNR,C_NAME,TRAIN_NO,FSTN,TSTN,B_DATE,NO_OF_SEATS,QUOTA)
    VALUES ('$pnr', '$na', '$tn', '$fstn', '$tstn', '$date', '$nos', '$qu')";
    $data = mysqli_query($conn,$query);
    if($data)
    {
        header("Location:payment.php");
    }
    else
    {
        header("Location:index.php");
    }
}

```



```

}
}
?>

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<meta http-equiv="X-UA-Compatible" content="ie=edge" />

<title>Railway Reservation Management System</title>

<link rel="shortcut icon" href="Logo.png" type="image/x-icon">

<link rel="stylesheet" href="" />

<style>

@import
url("https://fonts.googleapis.com/css2?family=Poppins:wght@200;300;400;500;600;700&display=swap");

* {

margin: 0;

padding: 0;

box-sizing: border-box;

font-family: "Poppins", sans-serif;

}

body {

height: 600px;

display: flex;

align-items: center;

justify-content: center;

padding: 20px;

```

```
background-image: url("bgpp.jpg");
}

.container {
position: relative;
max-width: 400px;
height: 570px;
width: 100%;
background: #fff;
padding: 5px;
border-radius: 8px;
box-shadow: 0 0 15px rgba(0, 0, 0, 0.1);
}

h1 {
margin-bottom: 20px;
color: #333;
text-align: center;
}

.container .form {
margin-top: 10px;
}

.form .input-box {
width: 100%;
margin-top: 20px;
}

.input-box label {
color: #333;
}
```

```
.form :where(.input-box input, .select-box) {  
  position: relative;  
  height: 40px;  
  width: 100%;  
  outline: none;  
  font-size: 1rem;  
  color: #707070;  
  margin-top: 8px;  
  border: 1px solid #ddd;  
  border-radius: 6px;  
  padding: 0 15px;  
}  
.input-box input:focus {  
  box-shadow: 0 1px 0 #7d2ae8;  
}  
.input-box input:hover {  
  background: #cfbde7;  
}  
.form .column {  
  display: flex;  
  column-gap: 15px;  
}  
address :where(input, .select-box) {  
  margin-top: 15px;  
}  
.select-box select {  
  height: 100%;
```

```

width: 100%;
outline: none;
border: none;
color: #707070;
font-size: 1rem;
}

.container .form_control {
height: 30px;
width: 25%;
color: #fff;
font-size: 1rem;
font-weight: 400;
margin-top: 30px;
border-radius: 10%;
cursor: pointer;
transition: all 0.2s ease;
background: #7d2ae8;
margin-left: 135px;
}

.container .form_control:hover {
background: rgb(88, 56, 250);
}

@media screen and (max-width: 500px) {
.form .column {
flex-wrap: wrap;
}

.form :where(.gender-option, .gender) {

```

```

row-gap: 15px;
}
}
</style>
</head>
<body>
<section class="container">
<h1>RESERVATION FORM</h1>
<form action="" method="POST" class="form">
<div class="input-box">
<input type="text" id="pnr" value="<?php echo $newpnr; ?>" name="pnr"
readonly/>
</div>
<div class="input-box">
<input type="text" name="c_na" placeholder="ENTER FULL NAME" required />
</div>
<div class="input-box">
<input type="text" name="tn" placeholder="TRAIN_NO" required />
</div>
<div class="column">
<div class="input-box">
<input type="text" name="fstn" placeholder="FROM STATION" required />
</div>
<div class="input-box">
<input type="text" name="tstn" placeholder="TO STATION" required />
</div>
</div>

```

```

<div class="input-box">
<input type="date" name="date" placeholder="DATE" required />
</div>

<div class="input-box">
<input type="value" name="nos" placeholder="NO OF SEATS" required />
</div>

<div class="column">
<div class="select-box" >
<select name="qu">
<option hidden name="qu">QUOTA</option>
<option>1AC</option>
<option>2AC </option>
<option>3AC</option>
<option>SL</option>
<option>GEN</option>
</select>
</div>
</div>

<input type="submit" class="form_control" name="submit" value="BOOK"
onclick="btn()">

</form>

</section>

<script>
function btn()
{
alert("Please confirm! All of the information is right")
btn2()

```

```

}
function btn2()
{
alert("Now! Make the Payment");
}
</script>
</body>
</html>

```

payment.php:

```

<?php
include ('config.php');
if (isset($_POST['make_payment']))
{
$cno = $_POST['cno'];
$chn = $_POST['chn'];
$exp = $_POST['exp'];
$cvv = $_POST['cvv'];
$query = "INSERT INTO payment (CARD_NO,CARD_H_NAME,E_DATE,CVV )
VALUES ('$cno', '$chn', '$exp', '$cvv')";
$data = mysqli_query($conn,$query);
if($data)
{
header("Location:history.php");}
else
{

```

```
echo "Data not inserted";
}
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Railway Reservation Management System</title>
<link rel="shortcut icon" href="Logo.png" type="image/x-icon">
<link rel="stylesheet" href="style3.css">
<style>
body {
font-family: Arial, sans-serif;
background-color: #f5f5f5;
display: flex;
justify-content: center;
align-items: center;
height: 100vh;
margin: 0;
background-image: url("bgpp.jpg");
}
.container {
background-color: #fff;
padding: 20px;
border-radius: 8px;
```



```

box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);
max-width: 400px;
width: 100%;
text-align: center;
}

.column {
display: flex;
justify-content: space-between;
align-items: center; /* Center vertically */
}

.input-group.flex {
flex-basis: calc(50% - 10px);
}

.column .input-group input[type="text"],
.column .input-group input[type="date"] {
width: 100%; /* Make the inputs take full available width */
padding: 10px;
border: 1px solid #ccc;
border-radius: 4px;
font-size: 16px;
outline: none;
transition: border-color 0.3s;
}

h1 {
margin-bottom: 20px;
color: #333;
}

```

```
.input-group {  
margin-bottom: 15px;  
text-align: left;  
}  
  
label {  
display: block;  
margin-bottom: 5px;  
color: #333;  
font-weight: bold;  
}  
  
input[type="text"] {  
width: 100%;  
padding: 10px;  
border: 1px solid #ccc;  
border-radius: 4px;  
font-size: 16px;  
outline: none;  
transition: border-color 0.3s;  
}  
  
input[type="text"]:focus {  
border-color: #7ed321;  
}  
  
input[type="submit"] {  
background-color: #7d2ae8;  
color: #fff;  
border: none;  
padding: 10px 20px;
```

```

border-radius: 4px;
font-size: 18px;
cursor: pointer;
transition: background-color 0.3s;
display: block;
margin: 0 auto;
}
input[type="submit"]:hover {
background-color: rgb(88, 56, 250);
}
</style>
</head>
<body>
<div class="container">
<h1>MAKE PAYMENT</h1>
<form action="payment.php" method="POST">
<div class="input-group">
<input type="text" id="card-number" name="cno" placeholder="CARD NUMBER"
required>
</div>
<div class="input-group">
<input type="text" id="card-holder" name="chn" placeholder="CARD HOLDER
NAME" required>
</div>
<div class="column">
<div class="input-group">
<input type="text" id="expiry" name="exp" placeholder="EXPIRY DATE" readonly>
</div>

```

```
<div class="input-group">

<input type="date" id="expiry" name="exp" placeholder="EXPIRY DATE"
required></div>

</div>

<div class="input-group">

<input type="text" id="cvv" name="cvv" placeholder="CVV" required>

</div>

<div class="input-group">

<input type="submit" name="make_payment" value="Pay Now" onclick
="btn()">

</div>

</form>

</div>

<script>
function btn(){
alert("Please confirm! All of the information is right")
btn2()
}
function btn2(){
alert("Congratulation! You have Successfully booked a ticket");
}
</script>

</body>

</html>
```

history.php:

```
<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Railway Reservation Management System</title>

<link rel="shortcut icon" href="Logo.png" type="image/x-icon">

<link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
4bw+/aepP/YC94hEpVNVgiZdgIC5+VKNBQNGCHeKRQN+PtmoHDEXuppvnDJz
Qlu9" crossorigin="anonymous">

</head>

<style>

body {

font-family: Arial, sans-serif;

position: relative;

height: 100vh;

width: 100%;

background-image: url("wp.jpg");

background-size: cover;

background-position: center;

justify-content: center;

align-items: center;

margin: 0;

display: flex;

flex-direction: column;
```

```
}  
.container {  
margin-top: 20px;  
display: flex;  
justify-content: center;  
align-items: center;  
height: 60vh;  
}  
.card-title {  
color: #000000;  
font-size: 24px;  
font-weight: bold;  
margin-left: 130px;  
}  
.btn-primary {  
background-color: #7d2ae8;  
border-color: #007bff;  
}  
.btn-primary:hover {  
background-color: #0056b3;  
border-color: #0056b3;  
}  
table {  
width: 100%;  
border-collapse: collapse;  
margin-top: 20px;  
}
```

```
th, td {  
padding: 10px;  
text-align: center;  
}  
th {  
background-color: #f2f2f2;  
font-weight: bold;  
}  
tr:nth-child(even) {  
background-color: #f2f2f2;  
}  
a {  
color: #007bff;  
text-decoration: none;  
display: block;  
margin-top: 20px;  
text-align: center;  
}  
.button {  
display: inline-block;  
padding: 10px 20px;  
background-color: #7d2ae8;  
color: white;  
text-decoration: none;  
border-radius: 5px;  
border: none;  
cursor: pointer;
```

```

margin-top: 20px; }

.button:hover {
background-color: #0056b3;
}
</style>
</head>

<body>
<div class="container">
<div class="row">
<div class="col-md-12 mt-4">
<div class="card">
<div class="card-header">
<h4 class="card-title">BOOKING HISTORY</h4>
</div>
<div class="card-body">
<form action="" method="POST" class="row">
<div class="col-md-6">
<div class="form-group">
<input type="text" name="pnr" class="form-control" placeholder="PNR" required>
</div>
</div>
<div class="col-md-6">
<button type="submit" name="search_by_id" class="btn btn-
primary">SHOW</button>
</div>
</form>

```



```

</div>
</div>
<div class="table-responsive">
<table class="table table-bordered">
<thead></thead>
<tr>
<th scope="col">PNR</th>
<th scope="col">FULL NAME</th>
<th scope="col">TRAIN_NO</th>
<th scope="col">FROM STATION</th>
<th scope="col">TO STATION</th>
<th scope="col">DATE</th>
<th scope="col">NO OF SEATS</th>
<th scope="col">QUOTA</th>
</tr>
</thead>
<tbody>
<?php
include('config.php');
if(isset($_POST['search_by_id']))
{
$id = $_POST['pnr'];
$query = "SELECT * FROM booking WHERE PNR = '$id'";
$query_run = mysqli_query($conn,$query);
if(mysqli_num_rows($query_run)>0)
{
while($row = mysqli_fetch_array($query_run))

```

```

{
?>

<tr>
<td><?php echo $row['PNR'];?></td>
<td><?php echo $row['C_NAME'];?></td>
<td><?php echo $row['TRAIN_NO'];?></td>
<td><?php echo $row['FSTN'];?></td>
<td><?php echo $row['TSTN'];?></td>
<td><?php echo $row['B_DATE'];?></td>
<td><?php echo $row['NO_OF_SEATS'];?></td>
<td><?php echo $row['QUOTA'];?></td>
</tr>
<?php
}
}
else
{
?>

<tr>
<td colspan="6">No Record Found</td>
</tr>
<?php
}
}
?>

</tbody>

```

```

</table>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/js/bootstrap.bundle.min.js"
integrity="sha384-
HwwvtgBNo3bZJJLYd8oVXjrBZt8cqVSpeBNS5n7C8IVInixGAoxmnlMuBnhbgrkm"
crossorigin="anonymous"></script>

</body>

<a href="index.php" class="button">HOME</a>

</html>

```

about.html:

```

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8" />

<meta name="viewport" content="width=device-width, initial-scale=1.0" />

<meta http-equiv="X-UA-Compatible" content="ie=edge" />

<title>Railway Reservation Management System</title>

<link rel="shortcut icon" href="Logo.png" type="image/x-icon">

<link rel="stylesheet" href="style1.css" />

<style>

```

```
body {
font-family: Arial, sans-serif;
background-color: #f5f5f5;
display: flex;
justify-content: center;
align-items: center;
height: 100vh;
margin: 0;
background-image: url("bgpp.jpg");
.container {
background-color: #fff;
padding: 20px;
border-radius: 8px;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);
max-width: 400px;
width: 100%;
text-align: center;}
h1 {
margin-bottom: 20px;
color: #333;}
.input-group {
margin-bottom: 15px;
text-align: left;}
label {
display: block;
margin-bottom: 5px;
color: #333;
```

```
font-weight: bold;
}
input[type="text"] {
width: 100%;
padding: 10px;
border: 1px solid #ccc;
border-radius: 4px;
font-size: 16px;
outline: none;
transition: border-color 0.3s;
}
input[type="text"]:focus {
border-color: #7ed321;
}
input[type="submit"] {
background-color: #7d2ae8;
color: #fff;
border: none;
padding: 10px 20px;
border-radius: 4px;
font-size: 18px;
cursor: pointer;
transition: background-color 0.3s;
display: block;
margin: 0 auto;
}
input[type="submit"]:hover {
```

```

background-color: rgb(88, 56, 250);
}
</style>
</html>
<body>
<section class="about-us">
<div class="about">

<div class="text">
<h2>About Us</h2>
<h5>Indian Railway & <span>Reservation</span></h5>
<p>Railway reservation management system is an online ticket booking website,
which is capable of booking ticket and search the train availability. This website is
mainly created to fulfil the following requirements, it comprises of the following
properties: -<br>
• A central database that will store all information.<br>
• An online website that will provide real- time information about the availability of
tickets their prices.<br>
• Every registered user can view his booking id that has been made in his/her
name.<br>
</p>
<div class="data">
<a href="index.php" class="hire">HOME</a>
</div>
</div>
</div>
</div>
</section>
</body>
</html>

```

CHAPTER-8

8. TESTING

8.1 WHAT IS TESTING

Testing is a critical phase in the software development lifecycle that involves evaluating and validating a project to ensure it meets the required specifications and functions correctly. The primary goal of testing is to identify bugs, errors, or discrepancies in the system and verify that the software performs as expected under various conditions. For the Railway Reservation Management System, testing ensures that the application is reliable, efficient, and user-friendly.

8.2 TYPES OF TESTING

(a) Unit Testing: Unit testing is a method of testing individual units or components of a software application. It is typically done by developers and is used to ensure that the individual units of the software are working as intended. Unit tests are usually automated and are designed to test specific parts of the code, such as a particular function or method. Unit testing is done at the lowest level of the software development process, where individual units of code are tested in isolation.

(b) Integration Testing: Integration testing is a method of testing how different units or components of a software application interact with each other. It is used to identify and resolve any issues that may arise when different units of the software are combined. Integration testing is typically done after unit testing and before functional testing, and is used to verify that the different units of the software work together as intended.

(c) Regression Testing: Regression testing is a method of testing that is used to ensure that changes made to the software do not introduce new bugs or cause existing functionality to break. It is typically done after changes have been made to

the code, such as bug fixes or new features, and is used to verify that the software still works as intended.

(d) System Testing: System Testing is carried out on the whole system in the context of either system requirement specifications or functional requirement specifications or in the context of both. The software is tested such that it works fine for the different operating systems.

(e) Performance Testing: It is designed to test the run-time performance of software within the context of an integrated system. It is used to test the speed and effectiveness of the program. It is also called load testing. In it we check, what is the performance of the system in the given load.

(f) Black box Testing: Black Box Testing is a software testing method where the tester focuses on testing the functionality of the application without any knowledge of the internal workings or code structure.

(g) White box Testing: White Box Testing (also known as clear-box testing or structural testing) is a software testing method where the tester has full knowledge of the internal workings of the application, including the code, design, and architecture.

(i) Alpha Testing: Alpha testing is the initial testing phase where the software is tested by the internal development team or QA team. The goal is to identify bugs and issues before releasing the software to external users. It is performed in a controlled environment within the development organization.

(j) Beta Testing: Beta testing is conducted by a select group of external users (beta testers) before the software is released to the public. Beta testers provide feedback on the software's performance, usability, and potential issues. It helps identify any last-minute bugs or improvements.

8.3 TEST CASES OF THE PROJECT

TEST CASE 1: USER REGISTRATION

(a) Scenario: User wants to create a new account.

(b) Steps:

- User clicks on the "Signup" option.
- Enters valid details in the registration form.
- Submits the form.

(c) Expected Result: User account is successfully created, and the user is redirected to the main menu.

TEST CASE 2: USER LOGIN

(a) Scenario: User wants to log into their account.

(b) Steps:

- User clicks on the "Login" option.
- Enters valid credentials.
- Submits the form.

(c) Expected Result: User is successfully logged in and directed to the main menu.

TEST CASE 3: TRAIN SEARCH

(a) Scenario: User wants to search for available trains.

(b) Steps:

- User selects "Find Trains" from the main menu.
- Enters valid source, destination, and travel date.
- Submits the search request.

(c) Expected Result: System displays a list of available trains matching the criteria.

TEST CASE 4: SEAT RESERVATION

(a) Scenario: User wants to reserve seats on a selected train.

(b) Steps:

- User selects a train from the available options.
- Chooses seats and enters valid passenger details.
- Submits the reservation request.

(c) Expected Result: System confirms successful reservation and initiates payment.

TEST CASE 5: BOOKING HISTORY

(a) Scenario: User wants to view their booking history.

(b) Steps:

- User selects "Booking History" from the main menu.

(c) Expected Result: System displays a list of past and upcoming bookings for the user.

TEST CASE 6: LOGOUT

(a) Scenario: User wants to log out of their account.

(b) Steps:

- User selects "Logout" from the main menu.

(c) Expected Result: User session is terminated, and the user is redirected to the login page.

TEST CASE 7: INVALID LOGIN ATTEMPT

(a) Scenario: User tries to log in with incorrect credentials.

(b) Steps:

- User enters invalid username and password.
- Submits the login form.

(c) Expected Result: System displays an error message and prompts the user to re-enter valid credentials.

CHAPTER-9

9. CONCLUSION

In conclusion, the Railway Reservation Management System is an integrated solution that simplifies the entire train reservation process, enhancing user convenience, efficiency, and accessibility. By providing a set of essential modules such as login/signup, train search, reservation, payment handling, and booking history, the system ensures that users can easily and securely manage their travel plans. The modular structure of the system not only enhances usability but also ensures that it is maintainable, scalable, and adaptable for future upgrades or expansions.

The user authentication feature offers secure login and registration processes, safeguarding users' personal information. The train search module helps users quickly find relevant train details based on their travel preferences, while the reservation module ensures a smooth booking experience, including seat selection and payment processing. Additionally, the booking history feature provides users with an organized overview of their previous and upcoming trips, promoting transparency and easy access to booking details.

Overall, the Railway Reservation Management System offers a well-rounded solution that meets the needs of both passengers and administrators, contributing to a streamlined, user-friendly, and efficient railway booking experience.

REFERENCES

For the successful completion of this project, we referred to multiple sources for gathering information and data.

These sources include various related internet sites.

These can be summarized as-

[1] GeeksforGeeks. Available from: <https://www.geeksforgeeks.org/i>

[2] Stackoverflow. Available from <https://stackoverflow.com/i>

[3] PHP Documentation. Available from:
<https://www.php.net/manual/en/function.include.php>