

PROFESSIONAL TRAINING REPORT

at

Sathyabama Institute of Science and Technology

(Deemed to be University)

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Information Technology

By

SOUMYADEEP DAS
REG. NO. : 40120086



DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING

SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A” by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600 119, TAMIL NADU

APRIL - 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF INFORMATION TECHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bona fide work of **Soumyadeep Das**
(Reg. No.: 40120086) who carried out the project entitled "**TICKET RESERVATION
SYSTEM**" under my supervision from January 2023 to April 2023.

E. Murali
Internal Guide

Mr. E. Murali B.E., M.Tech.

R. Subhashini
Head of the Department

Dr. R. SUBHASHINI M.E., Ph.D.,

Submitted for Viva voce Examination held on

29/04/2023

Internal Examiner
Internal Examiner

External Examiner
External Examiner

DECLARATION

I, **Soumyadeep Das** hereby declare that the Project Report entitled **TICKET RESERVATION SYSTEM** done by me under the guidance of **Mr. E. Murali** is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Information Technology.

DATE : 29/04/2023

PLACE : CHENNAI, TAMIL NADU

Soumyadeep Das
SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D, Dean**, School of Computing **Dr. R. Subhashini** Head of the Department of Information Technology for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Mr. E. Murali** for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Information Technology** who were helpful in many ways for the completion of the project.

ABSTRACT

The project objective is to book train tickets in online. The train reservation system is an Internet based application that can be accessed throughout the net and can be accessed by anyone who has a net connection. This application will reserve the tickets. This online ticket reservation system provides a website for the Indian Railway where any user of internet can access it. User is required to login to the system and needs a credit card/UPI for booking the tickets. The website provides complete information regarding currently running trains between capital stations in India and gives the information of the available seats and accommodation type. Ticket reservations are done using credit card and can be cancelled if needed. Our online tickets reservation system is one of the best opportunities for those who cannot afford enough time to get their tickets reserved standing in long queues. People can book tickets online at any time of day or night. Our reservation system also provides option to cancel the tickets which are reserved previously and our portal even gives discount offers for the customers to avail.

CONTENTS

CHAPTER	TITLE	PAGE NO.
	ABSTRACT	I
	LIST OF FIGURES	VII-VIII
	LIST OF TABLES	IX
	LIST OF ABBREVIATIONS	X
1	INTRODUCTION	1
1.1	INTRODUCTION ABOUT WEB-APPLICATION	1
1.2	WORKING OF A WEB-APPLICATION	1-2
1.3	FLOW OF THE WEB-APPLICATION	3
1.4	ADVANTAGES OF WEB-APPLICATION	4
1.5	DISADVANTAGES OF WEB APPLICATION	5
1.6	INTRODUCTION TO TICKET RESERVATION SYSTEM	6
1.7	FEATURES OF TICKET RESERVATION SYSTEM	7
1.8	ADVANTAGES OF TICKET RESERVATION SYSTEM	8

1.9	DISADVANTAGES OF TICKET RESERVATION SYSTEM	9
1.10	OBJECTIVE OF TICKET RESERVATION SYSTEM	10
1.11	INSPIRATION	11
2	AIM AND SCOPE OF THE PRESENT INVESTIGATION	12
2.1	AIM	12
2.2	SCOPE	13
2.3	SYSTEM REQUIREMENTS	14
2.3.1	HARDWARE REQUIREMENTS	15
2.3.2	SOFTWARE REQUIREMENTS	15
2.4	TECHNOLOGIES USED	16-25
2.4.1	HTML	16-17
2.4.2	CSS	18-19
2.4.3	JAVASCRIPT	20-21
2.4.4	PHP	21-22
2.4.5	BOOTSTRAP	22-24

2.4.6	XAMPP	24-25
2.5	ESTIMATED DATASET INFORMATION	26
2.6	LOGIN CREDENTIALS FOR ADMIN	26
3	METHODOLOGY	27
3.1	SYSTEM ARCHITECTURAL WORKFLOW	27
3.1.1	ER DIAGRAM	27-30
3.2	DIAGRAMS IN TICKET RESERVATION SYSTEM	30
3.3	IDEATION MAP WORKFLOW	31-37
3.3.1	USE CASE DIAGRAM	31
3.3.2	CLASS DIAGRAM	32
3.3.3	SEQUENCE DIAGRAM	33
3.3.4	ACTIVITY DIAGRAM	34
3.3.5	COLLABORATION DIAGRAM OR COMMUNICATION DIAGRAM	35
3.3.6	STATECHART DIAGRAM	36
3.3.7	COMPONENT DIAGRAM	37

4	RESULTS AND DISCUSSION	38
4.1	RESULT	38-39
4.2	PROJECT OUTPUT	40-45
4.2.1	HOME PAGE OF TICKET RESERVATION SYSTEM	40
4.2.2	ABOUT PAGE OF TICKET RESERVATION SYSTEM	40
4.2.3	INSPIRATION PAGE OF TICKET RESERVATION SYSTEM	41
4.2.4	SERVICES PAGE OF TICKET RESERVATION SYSTEM	41
4.2.5	CONTACT PAGE OF TICKET RESERVATION SYSTEM	42
4.2.6	ITINERARY PAGE OF TICKET RESERVATION SYSTEM	42
4.2.7	ACCOMODATION PAGE OF TICKET RESERVATION SYSTEM	43
4.2.8	ACCOMODATION PAGE OF TICKET RESERVATION SYSTEM (BOOKED SUCCESSFULLY)	43
4.2.9	PAYMENT INFO PAGE OF TICKET RESERVATION SYSTEM	44
4.2.10	LOGIN PAGE OF TICKET RESERVATION SYSTEM	44
4.2.11	RESERVATION PAGE OF TICKET RESERVATION SYSTEM	45

4.2.12	DISCOUNT ON RESERVATION PAGE OF TICKET RESERVATION SYSTEM	45
4.2.13	TRANSACTION PAGE OF TICKET RESERVATION SYSTEM	46
4.3	PERFORMANCE ANALYSIS	47-48
5	CONCLUSION AND FUTURE WORK	49
5.1	CONCLUSION	49
5.2	FUTURE WORK	49-50
5.3	WORKING ENVIRONMENT	51
5.4	APPENDIX : SOURCE CODE	52-54

LIST OF FIGURES

FIGURE NUMBER	FIGURE NAME	PAGE NUMBER
1.1	STRUCTURE OF A WEB APPLICATION	3
2.1	ESTIMATED DATASET INFORMATION	26
3.1	ER DIAGRAM	28
3.2	USE CASE DIAGRAM	31
3.3	CLASS DIAGRAM	32
3.4	SEQUENCE DIAGRAM	33
3.5	ACTIVITY DIAGRAM	34
3.6	COLLABORATION DIAGRAM OR COMMUNICATION DIAGRAM	35
3.7	STATECHART DIAGRAM	36
3.8	COMPONENT DIAGRAM	37
4.1	HOME PAGE OF TICKET RESERVATION SYSTEM	40
4.2	ABOUT PAGE OF TICKET RESERVATION SYSTEM	40
4.3	INSPIRATION PAGE OF TICKET RESERVATION SYSTEM	41

4.4	SERVICES PAGE OF TICKET RESERVATION SYSTEM	41
4.5	CONTACT PAGE OF TICKET RESERVATION SYSTEM	42
4.6	ITINERARY PAGE OF TICKET RESERVATION SYSTEM	42
4.7	ACCOMODATION PAGE OF TICKET RESERVATION SYSTEM	43
4.8	ACCOMODATION PAGE OF TICKET RESERVATION SYSTEM (BOOKED SUCCESSFULLY)	43
4.9	PAYMENT INFO PAGE OF TICKET RESERVATION SYSTEM	44
4.10	LOGIN PAGE OF TICKET RESERVATION SYSTEM	44
4.11	RESERVATION PAGE OF TICKET RESERVATION SYSTEM	45
4.12	DISCOUNT ON RESERVATION PAGE OF TICKET RESERVATION SYSTEM	45
4.13	TRANSACTION PAGE OF TICKET RESERVATION SYSTEM	46

LIST OF TABLES

TABLE NUMBER	TABLE NAME	PAGE NUMBER
2.1	LOGIN CREDENTIALS FOR ADMIN	26
3.1	DIAGRAMS OF THE PROJECT	30

LIST OF ABBREVIATIONS

ABBREVIATION NUMBER	ABBREVIATION	EXPANSION
01	APPL.	APPLICATION
02	PROG.	PROGRAMMING
03	RES.	RESERVATION
04	DIAG.	DIAGRAM

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION ABOUT WEB-APPLICATION

A web-application is an application program that is usually stored on a remote server, and users can access it through the use of Software known as web-browser. It is a type of computer program that usually runs with the help of a web browser and also uses many web technologies to perform various tasks on the internet.

A web appl. can be developed for several uses, which can be used by anyone like it can be used as an individual or as a whole organization for several reasons.

In general, a web application can contain online shops (or we can also say them e-commerce shops),ticket reservation system, webmail's, calculators, social media platforms, etc. There is also some kind of web application that usually requires a special kind of web browser to access them. We cannot access those kinds of web applications by using regular web- browsers. However, most of the web applications available on the internet can be accessed using a standard web browser.

1.2 WORKING OF A WEB-APPLICATION

In general, web-application does not require downloading them because, as we already discussed, the web application is a computer program that usually resides on the remote server. Any user can access it by using one of the standard web browsers such as Google Chrome, Safari, Microsoft Edge, etc., and most of them are available free for everyone.

A web application are generally coded using the languages supported by almost every web-browsers such as HTML, JavaScript because these are the languages that rely on the web browsers to render the program executable. Some of the web applications are entirely static due to which they not required any processing on the server at all while, on the other hand, some web applications are dynamic and require server-side processing.

To operate a web- application, we usually required a web server (or we can say some space on the web-server for our programs/application's code) to manage the clients' upcoming requests and required an application server. The application server performs the task that requested by the clients, which also may need a database to store the information sometimes. Application server technologies range from ASP.NET, ASP, and ColdFusion to PHP and JSP.

A standard web application usually has short development cycles and can be easily developed with a small team of developers. As we all know, most of the currently available web applications on the internet are written using the prog. languages such as the HTML (or HyperText Markup Language), CSS(or Cascading Style Sheets), and Javascript that are used in creating front-end interface (Client-side programming).

To create the web applications script, server-side programming is done by using programming languages such as Java, **Python**, PHP, and Ruby, etc. **Python** and Java are the languages that are usually used for server-side programming.

1.3 FLOW OF THE WEB-APPLICATION

Structure of a typical web-application :

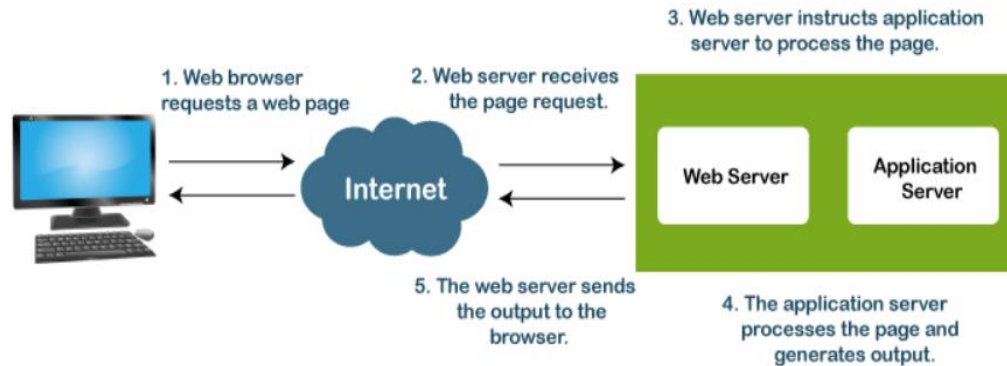


Fig 1.1 : Structure of a Web-Application

In general, a user sends a request to the web-server using web browsers such as Google Chrome, Microsoft Edge, Firefox, etc over the internet. Then, the request is forwarded to the appropriate web application server by the web-server.

Web application server performs the requested operations/ tasks like processing the database, querying the databases; produces the result of the requested data. The obtained result is sent to the web-server by the web application server along with the requested data/inf. or processed data. The web server responds to the user with the requested or processed data/information and provides the result to the user's screen.

1.4 ADVANTAGES OF WEB-APPLICATION

Any typical web application can run or accessible on any operating system such as the Windows, Mac, Linux as long as the browser is compatible. A web application is usually not required to install in the hard drive of the computer system, thus it eliminates all the issues related to the space limitation.

All the users are able to access the same version of the web application, which eliminates all compatibility issues. It also reduces software piracy in subscription-based web applications, for example, SAAS (or Software as a service).

They also reduce the expense for end-users, business owners because the maintenance needed by the business is significantly less. Web applications are flexible. A user can work from any geographical location as long as he has a working internet connection.

It just takes a moment to create a new user by providing a username, password, and URL, and it's all. After the availability of the cloud, storage space is now virtually unlimited as long as you can afford it.

A web application can be programmed to run on a wide variety of operating systems, unlike native applications that can run on a particular platform. Any standard web application is developed with some basic programming languages like HTML, CSS that are compatible and well known among the IT professionals.

1.5 DISADVANTAGES OF WEB-APPLICATION

Internet connection is necessary to access any web application, and without an internet connection, anyone can't use any of the web applications. It is very typical to get an internet connection in our modern cities, still rural area internet connectivity not so well. Several people in business believe that their data on the cloud environment is not that secure and likes to stick with old methods; they even don't want to use new methods.

As we all know that many users like to use different web browsers according to their needs and choices. So while creating a web application, you must remember that your application must support several web browsers, including new and old versions of browsers. Speed-related issues are also affecting the web application's performance because there are several factors on which the performance of a web application depends, and these all factors affect the performance of the web application in their own way.

If a user's web application faces any kind of issues, or if he does not have a good quality corporate website, his web application will not be going to run correctly, smoothly. A user must have to spend enough money to maintain the good condition of his web application, provide an update whenever an issue occurs, and make an attractive user interface, which is not so cheap at all.

A web application must be programmed/ coded in such a way that it will be run regardless of the device's operating system. It may face some issues while running on Windows, Android, or several other operating systems if it is not responsive.

1.6 INTRODUCTION TO TICKET RESERVATION SYSTEM

A ticket reservation system is a software application that allows users to reserve or book tickets for events, transportation, or other activities. The system can be used by customers to browse available options, select their preferred seats or modes of transportation, and make a payment for their reservation.

The ticket reservation system typically consists of a database that stores information about events, available seats, and pricing. It may also include a user interface for customers to search for events, view seat availability and pricing, and complete their booking. The system may also include a payment gateway that allows customers to securely pay for their reservations.

Ticket reservation systems are used in various industries, including:

Transportation - such as airlines, trains, buses, and ferries.

Entertainment - such as movies, concerts, and sports events.

Tourism - such as hotels, tours, and attractions.

Education - such as school admissions and exam registrations.

Hence, a ticket reservation system simplifies the process of booking tickets for customers and streamlines the operations of businesses that rely on ticket sales.

1.7 FEATURES OF TICKET RESERVATION SYSTEM

The features of a ticket reservation system can vary depending on the specific industry or use case, but some common features include:

Availability calendar: A calendar that shows the dates and times of events, transportation schedules, or other activities that can be reserved.

Seat selection: The ability to choose specific seats or areas within a venue, such as a movie theatre, sports arena or ticket reservation system.

Pricing options: The ability to offer different pricing tiers, discounts, or promotional codes to customers.

Payment processing: The ability to securely process payments from customers using various payment methods, such as credit cards, PayPal, or other digital payment systems.

Reservation confirmation: A system that sends confirmation emails or SMS messages to customers once their reservation is complete.

User accounts: The ability for customers to create and manage their own accounts, which may include their reservation history, payment details, and personal information.

Integration with other systems: The ability to integrate with other systems, such as inventory management or customer relationship management (CRM) software, to streamline business processes and improve efficiency.

Mobile responsiveness: The ability for the system to be accessed and used on mobile devices, such as smartphones and tablets.

Customer support: The ability for customers to contact support through various channels, such as email, chat, or phone, to get assistance with their reservations or other issues.

1.8 ADVANTAGES OF TICKET RESERVATION SYSTEM

There are several advantages of using a ticket reservation system, including:

Convenience: Ticket reservation systems allow customers to browse and book tickets from the comfort of their own homes, at any time of the day or night. This eliminates the need to visit a physical ticket office or wait in long queues to purchase tickets.

Improved customer experience: By using a ticket reservation system, customers can view seat availability, choose their preferred seats, and complete their booking quickly and easily. This can improve their overall experience and make them more likely to return in the future.

Increased revenue: Ticket reservation systems can help businesses to sell more tickets by reaching a wider audience, offering different pricing options, and promoting their events or services to potential customers.

Real-time availability: Ticket reservation systems provide real-time updates on seat availability, which allows customers to make informed decisions about their reservations. This also allows businesses to manage their inventory more efficiently and avoid overbooking.

Reduced operational costs: By automating the ticket reservation process, businesses can reduce the costs associated with manual ticketing systems, such as staffing, printing, and distribution.

Integration with other systems: Ticket reservation systems can be integrated with other systems, such as accounting or marketing software, to streamline business processes and improve efficiency.

Thus, a ticket reservation system can help businesses to improve their customer experience, increase revenue, and reduce operational costs, while providing customers with a convenient and hassle-free way to book their tickets.

1.9 DISADVANTAGES OF TICKET RESERVATION SYSTEM

While ticket reservation systems offer many advantages, there are also some potential disadvantages to consider:

Technical issues: Like any software system, ticket reservation systems can experience technical issues such as downtime, glitches, or system errors. This can cause inconvenience for customers and negatively impact business operations.

Security risks: Ticket reservation systems process sensitive customer information, such as credit card details and personal information. This makes them a target for hackers and cybercriminals, which can result in data breaches and other security risks.

Dependence on technology: Ticket reservation systems require reliable internet connectivity, which can be a challenge in areas with poor internet infrastructure. Additionally, businesses that rely too heavily on technology may become vulnerable to disruptions in service or outages.

Cost: Implementing a ticket reservation system can be expensive, especially for small businesses or organizations with limited budgets. There may be upfront costs for software licensing, hardware, and system integration, as well as ongoing maintenance and support costs.

Lack of personal touch: Online ticket reservation systems may lack the personal touch of traditional ticket sales, which can impact customer loyalty and satisfaction.

Overall, while ticket reservation systems offer many benefits, businesses and organizations should carefully evaluate their needs and consider the potential drawbacks before implementing a system.

1.10 OBJECTIVE OF TICKET RESERVATION SYSTEM

The objectives of a ticket reservation system can vary depending on the specific industry or use case, but some common objectives include:

Improving customer experience: To provide customers with a convenient, easy-to-use system that allows them to browse, select, and purchase tickets quickly and efficiently. This can help to improve customer satisfaction and encourage repeat business.

Maximizing revenue: Another objective of a ticket reservation system is to maximize revenue for the business or organization. This can be achieved by offering different pricing options, promoting ticket sales through targeted marketing campaigns, and leveraging data analytics to identify sales trends and opportunities.

Streamlining operations: By automating the ticket reservation process, businesses and organizations can streamline their operations and reduce the costs associated with manual ticketing systems. This can include reducing staffing requirements, minimizing printing and distribution costs, and improving inventory management.

Enhancing data collection and analysis: Ticket reservation systems can provide businesses with valuable data on customer behavior, sales trends, and other metrics. This data can be used to make informed business decisions, improve marketing strategies, and enhance the overall customer experience.

Hence, the objectives of a ticket reservation system are to improve the customer experience, increase revenue, streamline operations, and enhance data collection and analysis to drive business growth and success.

1.11 INSPIRATION

The inspiration behind the ticket reservation system dates back to the mid-20th century, when airlines began using computerized reservation systems to manage their bookings. These systems allowed customers to make reservations by phone or through travel agents, and provided real-time updates on flight availability and pricing.

Over time, other industries such as the entertainment and sports industries also began adopting similar systems to manage ticket sales. The widespread adoption of the internet and e-commerce in the late 1990s and early 2000s further accelerated the development of online ticket reservation systems, making it easier for customers to purchase tickets from the comfort of their own homes.

Today, ticket reservation systems are used across a wide range of industries, from transportation and hospitality to entertainment and sports. They have become an essential tool for businesses and organizations that rely on ticket sales to drive revenue, and continue to evolve and improve to meet the changing needs of customers and businesses alike.

CHAPTER 2

AIM AND SCOPE OF THE PRESENT INVESTIGATION

2.1 AIM

The aim of a ticket reservation system is to provide a fast, easy, and efficient way for customers to purchase tickets for events or services. By automating the ticketing process, businesses and organizations can improve their operations, reduce costs, and enhance the overall customer experience.

The primary aim of a ticket res. system is to improve the efficiency and accuracy of the ticketing process. This can be achieved by providing customers with an online portal where they can browse available tickets, select their preferred seats or dates, and make payments securely and easily.

In addition to improving the customer experience, the aim of a ticket reservation system is to help businesses and organizations to maximize revenue and profitability. This can be achieved by offering different pricing options, implementing promotional strategies to drive sales, and leveraging data analytics to identify sales trends and opportunities.

Therefore, the aim of a ticket reservation system is to provide a seamless and hassle-free ticketing experience for customers, while also improving the efficiency and profitability of businesses and organizations that rely on ticket sales.

2.2 SCOPE

A ticket reservation system can have a broad scope depending on its purpose and implementation. Here are some examples of areas where ticket reservation systems can be used:

Travel: Airlines, trains, buses, and other transportation companies use ticket reservation systems to manage seat bookings, schedules, and availability.

Entertainment: Ticket reservation systems are widely used for booking tickets for movies, concerts, theater performances, sports events, and other forms of entertainment.

Education: Schools, colleges, and universities use ticket reservation systems to manage registration and enrollment for classes, workshops, seminars, and other educational programs.

Tourism: Tour operators, travel agencies, and hotels use ticket reservation systems to manage bookings for tours, activities, and accommodations.

Healthcare: Hospitals and clinics use ticket reservation systems to schedule appointments for patients and manage their medical records.

Events: Event planners and organizers use ticket reservation systems to manage registration, attendee information, and ticket sales for conferences, trade shows, and other events like sports meet and sports organizations.

Overall, a ticket reservation system can be used in any industry or organization that requires booking or scheduling of services or events.

2.3 SYSTEM REQUIREMENTS

The system requirements for a ticket reservation system depend on the specific needs of the business or organization. However, here are some general requirements to consider:

Server and Database: The system needs a server to host the application, and a database to store and manage the reservation data. The choice of server and database depends on the expected traffic, scalability requirements, and budget.

User Interface: The system should have a user-friendly interface that enables users to search, book, and manage their reservations easily. The interface should be compatible with different devices and web browsers.

Payment Gateway: The system should support different payment methods, such as credit cards, PayPal, or bank transfers, and integrate with a payment gateway for secure transactions.

Availability Management: The system should have a feature to manage the availability of the services, such as seats or rooms, and update the availability in real-time.

Notification System: The system should have a notification system to send confirmation emails, reminders, and alerts to users about their reservations.

Security: The system should have robust security features to protect the user data and prevent unauthorized access or hacking attempts.

Reporting and Analytics: The system should provide reports and analytics to help the business or organization analyse their reservation data, such as occupancy rates, revenue, and customer demographics.

Integration: The system should integrate with other applications, such as booking engines, customer relationship management (CRM) systems, and social media platforms, to streamline operations and improve customer experience.

These are some of the general system requirements to consider when designing a ticket reservation system.

2.3.1 Hardware Requirements

- 1) Server : 4GB RAM or above
- 2) Storage
- 3) Networking
- 4) Load Balancer
- 5) Backup and Recovery
- 6) Security

2.3.2 Software Requirements

- 1) Operating System : Windows, Linux or MacOS
- 2) Web Framework : Angular JS
- 3) Database Management System : MySQL
- 4) Programming Language : HTML, CSS, JS, PHP
- 5) Front-end Framework : AngularJS, Bootstrap
- 6) API Integration
- 7) Testing and Deployment Tools

2.4 TECHNOLOGIES USED

2.4.1 HTML

HTML stands for HyperText Markup Language. It is a markup language used for creating web pages and other types of online content. HTML is used to define the structure and content of a webpage, including text, images, links, and other elements.

HTML documents are made up of HTML tags, which are enclosed in angle brackets and are used to define different elements and attributes of a webpage. For example, the <head> tag is used to define the head section of a webpage, which includes information such as the title of the page, while the <body> tag is used to define the main content of the webpage.

HTML is often used in conjunction with other web technologies such as CSS (Cascading Style Sheets) and JavaScript to create more advanced and interactive web pages. CSS is used for styling and formatting the content of a webpage, while JavaScript is used for creating dynamic and interactive elements such as dropdown menus, pop-up windows, and animations.

HTML is an essential tool for web developers and designers to create websites that are accessible and user-friendly.

FEATURES OF HTML

HTML (HyperText Markup Language) is a markup language used for creating web pages and other types of online content. The following are some of the features of HTML:

Structure: HTML is used to define the structure of a webpage, including headings, paragraphs, lists, and other types of content.

Tags: HTML uses tags to define different elements of a webpage, such as text, images, links, and tables.

Attributes: HTML tags can have attributes, which provide additional information about the element. For example, the `img` tag can have attributes such as `src` (which specifies the image URL) and `alt` (which provides a text description of the image).

Hyperlinks: HTML allows you to create hyperlinks, which are clickable links that take you to another webpage or to a different part of the same webpage.

Forms: HTML provides a way to create forms, which allow users to input data and send it to a server for processing. Forms can include input fields, checkboxes, radio buttons, and other types of elements.

Multimedia: HTML supports the integration of multimedia elements such as images, videos, and audio files.

Accessibility: HTML provides features such as alt text for images, headings for screen readers, and semantic markup to make content more accessible to users with disabilities.

2.4.2 CSS

CSS, or Cascading Style Sheets, is a language used for describing the presentation and layout of web pages. It is used to define the styles of HTML elements, such as their colors, fonts, sizes, positions, and other visual properties. By separating the presentation and content of a web page, CSS enables web designers to create more flexible and responsive designs that can be easily updated and customized.

CSS works by applying rules to HTML elements, which are selected using selectors. A selector is a pattern that matches one or more elements on a web page, and a rule is a set of styles that are applied to those elements.

CSS also allows for the creation of complex layouts and effects using positioning, floating, and flexbox techniques, as well as animations and transitions.

CSS can be included in an HTML document using the `<style>` element or an external stylesheet file, which is linked using the `<link>` element. Modern web development often involves the use of preprocessor languages like Sass or LESS, which allow for more advanced features like variables, mixins, and nesting.

FEATURES OF CSS

CSS, or Cascading Style Sheets, is a powerful language used for designing and styling web pages. Some of the key features of CSS include:

Separation of Content and Presentation: CSS allows the separation of content and presentation, which means that the content of a web page is stored separately from its presentation. This allows for easier maintenance and updates of web pages.

Cascading: CSS has the ability to cascade styles, which means that multiple styles can be applied to a single element and the browser will determine which style takes precedence.

Selectors: CSS selectors are used to select and target specific HTML elements for styling. Selectors can be based on element type, class, ID, attributes, and more.

Box Model: CSS uses a box model to describe the layout and dimensions of HTML elements. The box model consists of content, padding, border, and margin.

Responsive Design: CSS allows for the creation of responsive web designs, which means that web pages can adapt to different screen sizes and devices. This is achieved through the use of media queries, flexible layouts, and fluid images.

Layouts: CSS provides several layout techniques for positioning and arranging HTML elements, including floats, flexbox, and grid.

Typography: CSS provides a wide range of typography features, including font families, sizes, weights, line heights, and more.

Color and Effects: CSS allows for the customization of colors and the creation of effects such as shadows, gradients, and animations.

Compatibility: CSS is compatible with all modern browsers and is constantly evolving to include new features and improvements.

2.4.3 JAVASCRIPT

JavaScript is a programming language that is primarily used for creating interactive and dynamic web pages. It is often abbreviated as JS and is one of the most popular programming languages in the world. JavaScript allows web developers to create dynamic effects, manipulate the content of web pages, and respond to user interactions.

FEATURES OF JAVASCRIPT

Some of the key features of JavaScript are:

Object-oriented Programming: JavaScript is an object-oriented programming language, which means that it uses objects to represent data and functionality.

Client-side Scripting: JavaScript is mainly used for client-side scripting, which means that it runs in the user's web browser and can manipulate the HTML and CSS of a web page in real-time.

Event-driven Programming: JavaScript is event-driven, which means that it can execute code in response to events such as button clicks, page loads, and form submissions.

Asynchronous Programming: JavaScript supports asynchronous programming, which allows web developers to create code that can run in the background without blocking other code from running.

Cross-platform Compatibility: JavaScript is compatible with all major web browsers and can be used on any platform that supports web development.

Libraries and Frameworks: JavaScript has a vast ecosystem of libraries and frameworks, which can help web developers to create complex and sophisticated web applications more easily.

Expressiveness: JavaScript is a highly expressive language, meaning that it has a syntax that is easy to read and understand. This makes it a popular choice for beginners and experienced programmers alike.

JavaScript is a versatile and powerful language that is essential for modern web development. It provides the tools that web developers need to create interactive, dynamic, and engaging web pages.

2.4.4 PHP

PHP is a server-side scripting language used for creating dynamic web pages and web applications. It is one of the most popular server-side scripting languages in the world, and it is used by many popular websites and applications such as Facebook, Wikipedia, and WordPress.

FEATURES OF PHP

Here are some of the key features of PHP:

Server-Side Scripting: PHP is a server-side scripting language, which means that it runs on the server before the web page is sent to the user's web browser. This allows PHP to generate dynamic content, interact with databases, and perform other server-side tasks.

Cross-Platform Compatibility: PHP is compatible with all major operating systems, including Windows, Linux, and macOS.

Database Connectivity: PHP has built-in support for connecting to many popular databases, including MySQL, PostgreSQL, and Oracle.

Object-Oriented Programming: PHP supports object-oriented programming, which allows developers to write modular, reusable code.

Extensibility: PHP is highly extensible, which means that developers can add new features and functionality through the use of extensions, plugins, and libraries.

Interactivity: PHP provides many tools for creating interactive web applications, including forms, cookies, and sessions.

Security: PHP includes many built-in security features, such as input validation, encryption, and authentication.

PHP is a versatile and powerful language that is essential for building modern web applications. It provides the tools that web developers need to create dynamic, interactive, and secure web pages and applications.

2.4.5 BOOTSTRAP

Bootstrap is a popular open-source front-end framework used for building responsive and mobile-first websites and web applications. It was developed by Twitter and is now maintained by a large community of developers.

Bootstrap provides a set of CSS and JavaScript components, including buttons, forms, typography, navigation, and more, that can be easily customized and used in web development projects. The framework also includes a grid system that helps developers create responsive layouts for different screen sizes.

One of the key benefits of using Bootstrap is that it simplifies web development and speeds up the process by providing pre-built components that can be easily integrated into a project. It also helps ensure consistency and responsiveness across different devices and screen sizes.

Bootstrap has become very popular among developers and is widely used in web development projects of all sizes. It is compatible with all major browsers and can be used with popular web development tools like Visual Studio Code, Sublime Text, and Atom.

FEATURES OF BOOTSTRAP

Bootstrap has several features that make it a popular choice for building responsive and mobile-first websites and web applications. Here are some of its key features:

Responsive Design: Bootstrap provides a responsive grid system that makes it easy to create layouts that adapt to different screen sizes and devices.

Customizable Themes: Bootstrap comes with a variety of built-in themes and templates that can be easily customized to match the design of your website or application.

Cross-Browser Compatibility: Bootstrap is compatible with all major browsers, including Chrome, Firefox, Safari, and Internet Explorer.

Extensive Component Library: Bootstrap provides a large collection of pre-built UI components, such as forms, buttons, alerts, modals, and more, that can be easily integrated into your project.

JavaScript Plugins: Bootstrap includes a variety of JavaScript plugins, such as carousels, tooltips, and popovers, that can add interactivity and functionality to your website or application.

Accessibility: Bootstrap is designed with accessibility in mind and provides several features, such as keyboard navigation and ARIA labels, to make your website or application accessible to all users.

Community Support: Bootstrap has a large and active community of developers who contribute to its development and provide support through forums, documentation, and tutorials.

Bootstrap provides a comprehensive set of tools and features that can help streamline the web development process and create responsive, accessible, and visually appealing websites and applications.

2.4.6 XAMPP

XAMPP is an open-source, cross-platform web server solution that includes Apache, PHP, MySQL, and Perl. It is a popular tool used for local development and testing of web applications.

The name XAMPP is an acronym that stands for:

X (cross-platform)

Apache (the web server)

MySQL (the database management system)

PHP (the programming language)

Perl (another programming language)

XAMPP is available for Windows, Linux, and macOS and can be easily installed as a complete package, making it a convenient and easy-to-use solution for web developers who need to set up a local development environment quickly.

FEATURES OF XAMPP

Some of the features of XAMPP include:

Easy Installation: XAMPP can be installed quickly and easily, requiring minimal configuration.

Cross-Platform Support: XAMPP is compatible with Windows, Linux, and macOS, making it a flexible solution for developers working on different operating systems.

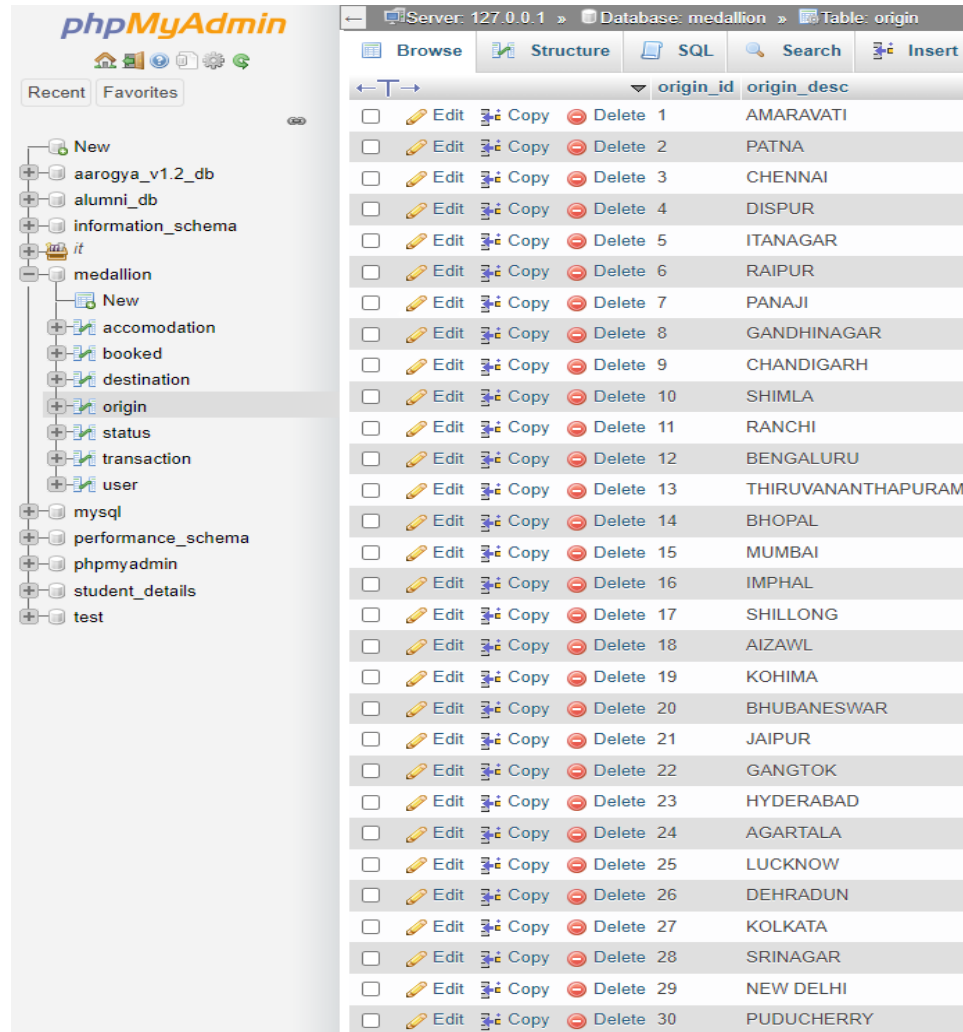
Complete Package: XAMPP includes Apache, MySQL, PHP, and Perl, providing a complete web server solution for local development and testing.

User-Friendly Control Panel: XAMPP comes with a user-friendly control panel that allows developers to start and stop the web server, manage the database, and configure other settings.

Extensions and Add-ons: XAMPP supports a range of extensions and add-ons, such as phpMyAdmin, that can help simplify development and testing tasks.

Security: XAMPP is designed to be secure by default, with features like a firewall and security checks that help protect the web server and the data stored on it.

2.5 ESTIMATED DATASET INFORMATION



origin_id	origin_desc
1	AMARAVATI
2	PATNA
3	CHENNAI
4	DISPUR
5	ITANAGAR
6	RAIPUR
7	PANAJI
8	GANDHINAGAR
9	CHANDIGARH
10	SHIMLA
11	RANCHI
12	BENGALURU
13	THIRUVANANTHAPURAM
14	BHOPAL
15	MUMBAI
16	IMPHAL
17	SHILLONG
18	AIZAWL
19	KOHIMA
20	BHUBANESWAR
21	JAIPUR
22	GANGTOK
23	HYDERABAD
24	AGARTALA
25	LUCKNOW
26	DEHRADUN
27	KOLKATA
28	SRINAGAR
29	NEW DELHI
30	PUDUCHERRY

Fig 2.1 : Estimated Dataset Information

2.6 LOGIN CREDENTIALS FOR ADMIN

Table 2.1 : Login Credentials for Admin

USERNAME	PASSWORD
Admin	Admin

CHAPTER 3

METHODOLOGY

3.1 SYSTEM ARCHITECTURAL WORKFLOW

3.1.1 ER DIAGRAM

An ER (Entity-Relationship) diagram is a graphical representation of entities and their relationships to each other in a database. ER diagrams are commonly used in database design to visualize the structure of a database and to help identify the relationships between different tables.

An ER diagram consists of three main components: entities, attributes, and relationships.

Entities: An entity is a real-world object, such as a person, place, or thing, that can be distinguished from other objects. In an ER diagram, entities are represented by rectangles.

Attributes: An attribute is a characteristic or property of an entity. For example, a person entity might have attributes such as name, age, and address. In an ER diagram, attributes are represented by ovals.

Relationships: A relationship is a connection between two or more entities. In an ER diagram, relationships are represented by diamonds.

There are three types of relationships that can be represented in an ER diagram:

One-to-one: In a one-to-one relationship, each record in the first table is related to one and only one record in the second table, and vice versa.

One-to-many: In a one-to-many relationship, each record in the first table can be related to one or more records in the second table, but each record in the second table is related to only one record in the first table.

Many-to-many: In a many-to-many relationship, each record in the first table can be related to one or more records in the second table, and vice versa.

ER diagrams can be created using various tools, such as Microsoft Visio, Lucidchart, or draw.io. These tools allow users to drag and drop shapes onto a canvas, connect them with lines, and add text to define the entities, attributes, and relationships.

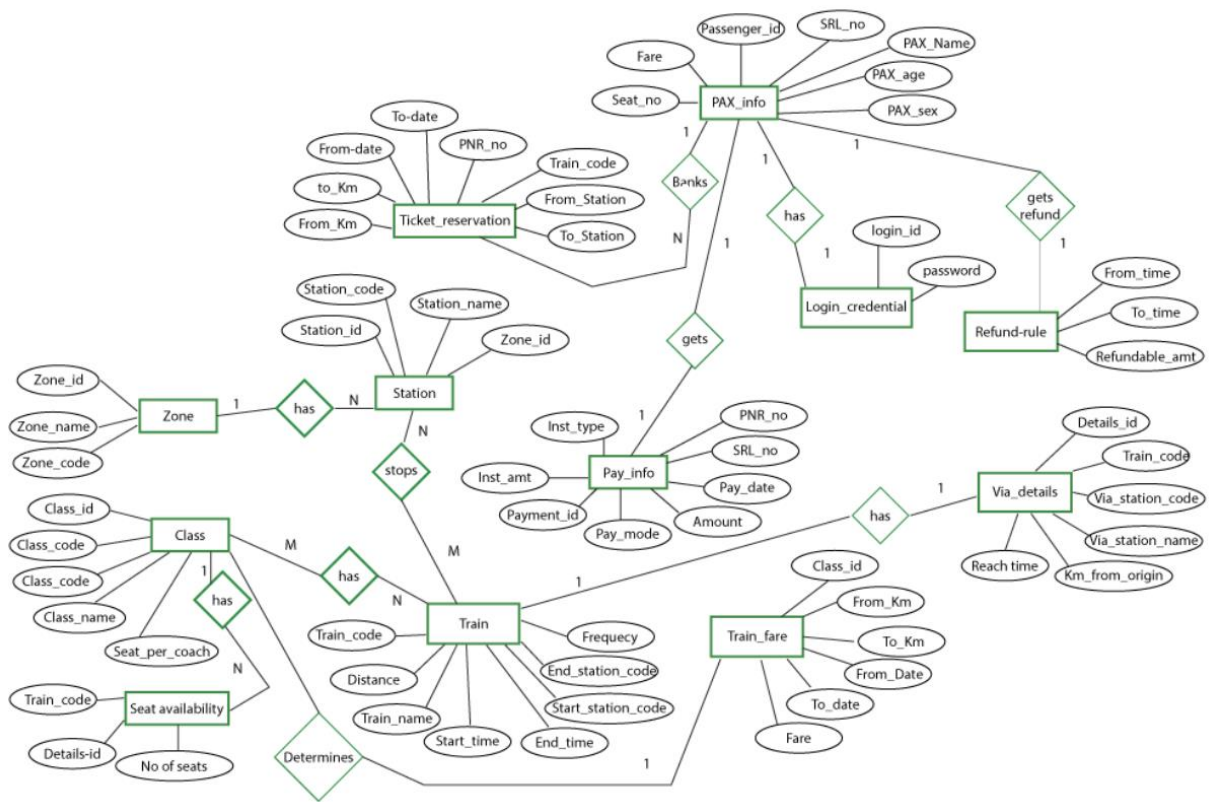


Fig 3.1 : ER Diagram

Entities and their attributes are:

PAX_info: Attributes of PAX_info entity are Passenger_id (primarykey) ,SRL_no ,PAX_name ,PAX_age ,PAX_sex ,fare,seat_no.

2 Login_credentials: Attributes of Login_credentials entity are login_id(PK) , password.

3 Ticket_reservation: Attributes of Ticket_reservation entity are PNR_no(pk),to-date, from-date, to-km, from-km, to-station, from-station, Train_code.

4 Refund_rule: Attributes of refund_rule entity are to-time, from-time, refundable- amt.

5 via_details: Attributes of via_details entity are Details_id(PK), Train_code, via_station_code, km_from_origin, Reach_time.

6 train_fare: Attributes of train_fare entity are to-date, from-date, to-km, from-km, Fare, Class_id.

7 Train: Attributes of Train entity are Train_code(PK), Distance, Train_name, Start_time, End_time, Start_station_code, End_station_code, Frequency.

8 Seat_availability: Attributes of Seat_availability entity are Train_code, Class_code, and Number of seats.

9 Class: Attributes of Class entity are Class_id(PK), coach_prefix, class_code, Class_name, seat_per_coach.

10 Zone: Attributes of Zone entity are zone_id(PK), Zone_name,Zone_code.

11 Station: Attributes of station entity are Station_id(PK),Station_code,station_name, zone_id.

12 Pay_info: Attributes of Pay_info entity are payment_id(PK), pay_mode, amount, pay_date, srl-no, PNR_no, inst_type, inst_amt

Relationships between Entities:

- 2 One-to-one relation: PAX_info to login_credentials , PAX_info to refund_rule, Train to Via_details, Class to train_fare, PAX_info to pay_info.
- 3 Many-to-one relation: PAX_info to ticket_reservation, Zone to the station, Class to seat_availability.
- 4 Many-to-many: Train to the station, Train to class.

3.2 DIAGRAMS IN TICKET RESERVATION SYSTEM

Table 3.1 : Diagrams of the Project

INDEX	DIAGRAM NAME
A	USE CASE DIAGRAM
B	CLASS DIAGRAM
C	SEQUENCE DIAGRAM
D	ACTIVITY DIAGRAM
E	COMMUNICATION DIAGRAM
F	STATECHART DIAGRAM
G	COMPONENT DIAGRAM

3.3 IDEATION MAP WORKFLOW

3.3.1 USE CASE DIAGRAM

A use case diagram is a type of UML (Unified Modeling Language) diagram that illustrates how different users or actors interact with a system or software application. It is a visual representation of the functional requirements of a system and provides an overview of the different use cases that the system supports.

A use case diag. consists of several components like Actors, Use cases and Relationships.

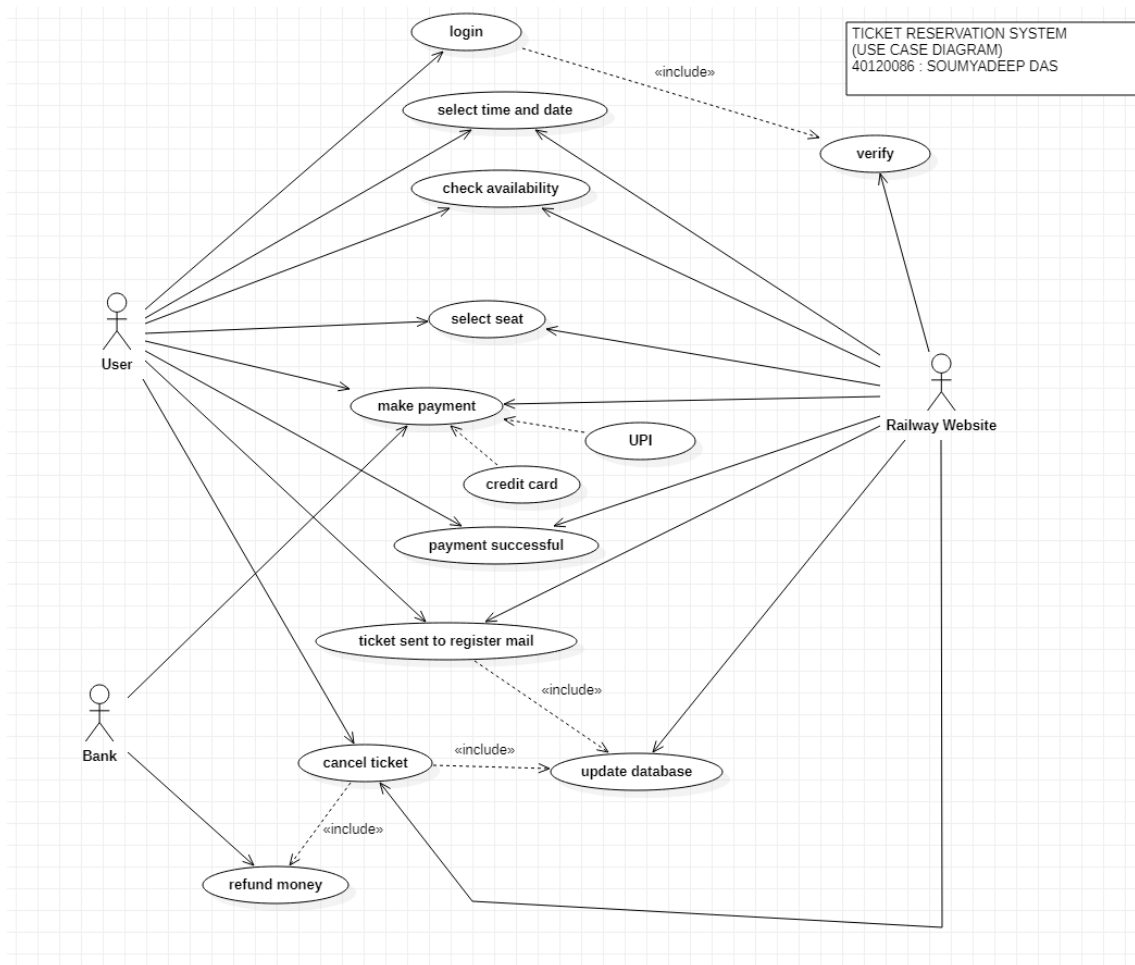


Fig 3.2 : Use Case Diagram

3.3.2 CLASS DIAGRAM

A class diagram is a type of UML diagram that represents the structure of a system or application in terms of classes, their attributes, methods, and relationships. The diagram provides a visual representation of the classes, their properties, and how they interact with each other.

In a class diagram, each class is represented as a rectangle with three compartments. The top compartment contains the class name, the middle compartment contains the class attributes, and the bottom compartment contains the class methods. The attributes and methods are listed below their respective headings, with the visibility modifier (public, private, or protected) and the data type specified for each attribute, and the return type and parameter list specified for each method.

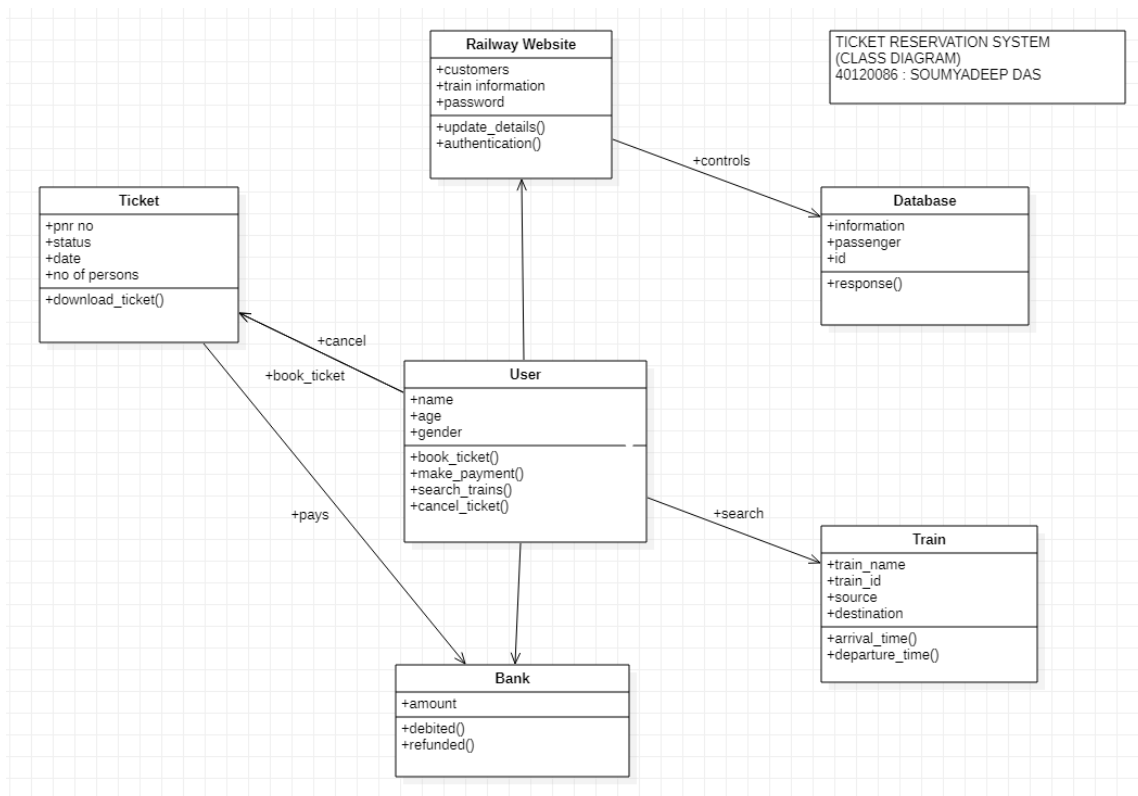


Fig 3.3 : Class Diagram

3.3.3 SEQUENCE DIAGRAM

A sequence diagram in UML (Unified Modeling Language) is a type of interaction diagram that illustrates the interactions between objects or components in a system over time. It shows the messages sent between objects or components in chronological order, and the order of execution of these messages.

In a sequence diagram, objects or components are represented as rectangles, and the interactions between them are represented as vertical lifelines. A lifeline represents the object's existence and its vertical position indicates the time dimension of the sequence. Messages between objects or components are represented as arrows that connect the lifelines of the sender and the receiver. A message may be synchronous, asynchronous, or a return message.

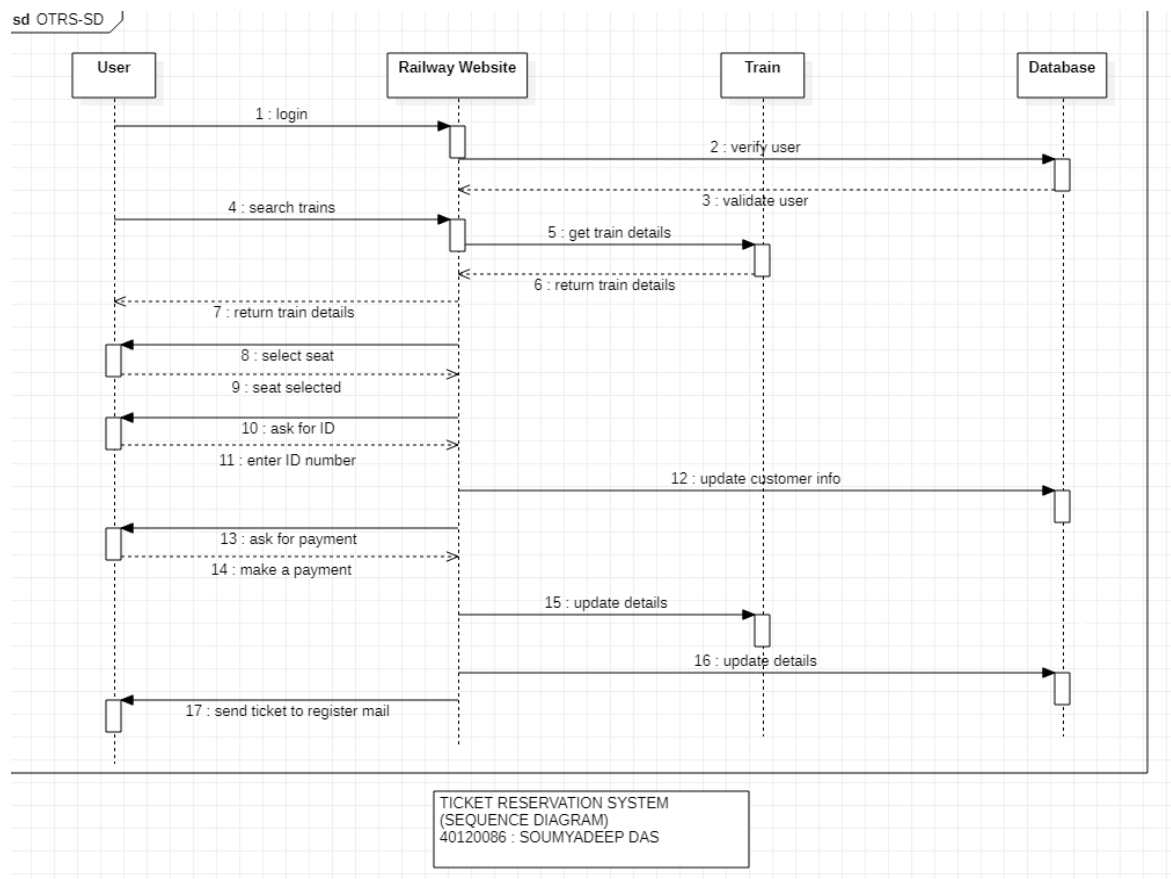


Fig 3.4 : Sequence Diagram

3.3.4 ACTIVITY DIAGRAM

An activity diagram in UML (Unified Modeling Language) is a type of behavioural diagram that illustrates the flow of activities and actions within a system or process. It is used to model the steps in a process or workflow, and to describe the logic and dependencies between these steps.

In an activity diagram, each activity or action is represented as a rounded rectangle, and the transitions between them are represented as arrows. A starting point is represented by a filled circle, and an ending point is represented by a filled circle with a hollow border. The actions can be grouped into swimlanes to indicate which entities or actors are responsible for performing each action.

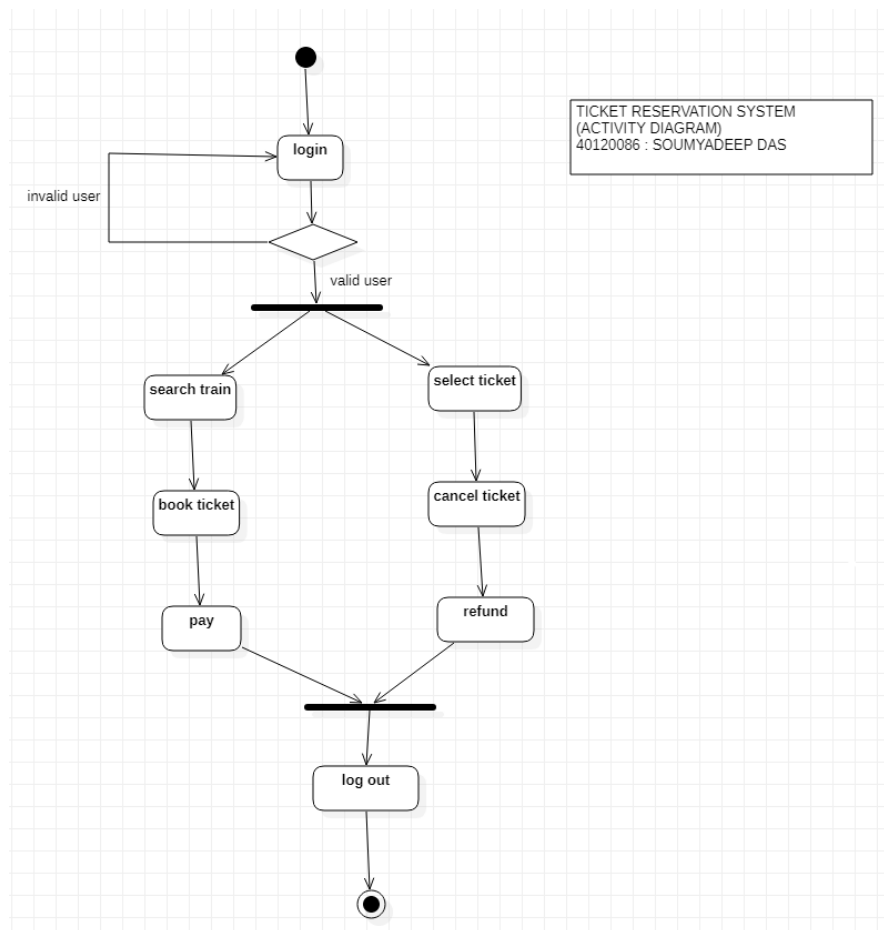


Fig 3.5 : Activity Diagram

3.3.5 COLLABORATION DIAGRAM OR COMMUNICATION DIAGRAM

A collaboration diagram in UML (Unified Modeling Language) is a type of interaction diagram that illustrates the interactions between objects or components in a system. It is also known as a communication diagram.

In a collaboration diagram, objects or components are represented as rectangles, and the interactions between them are represented as links or associations. A link represents a communication channel between two objects or components, and it can have a name and a set of properties. Messages between objects or components are represented as arrows that connect the links.

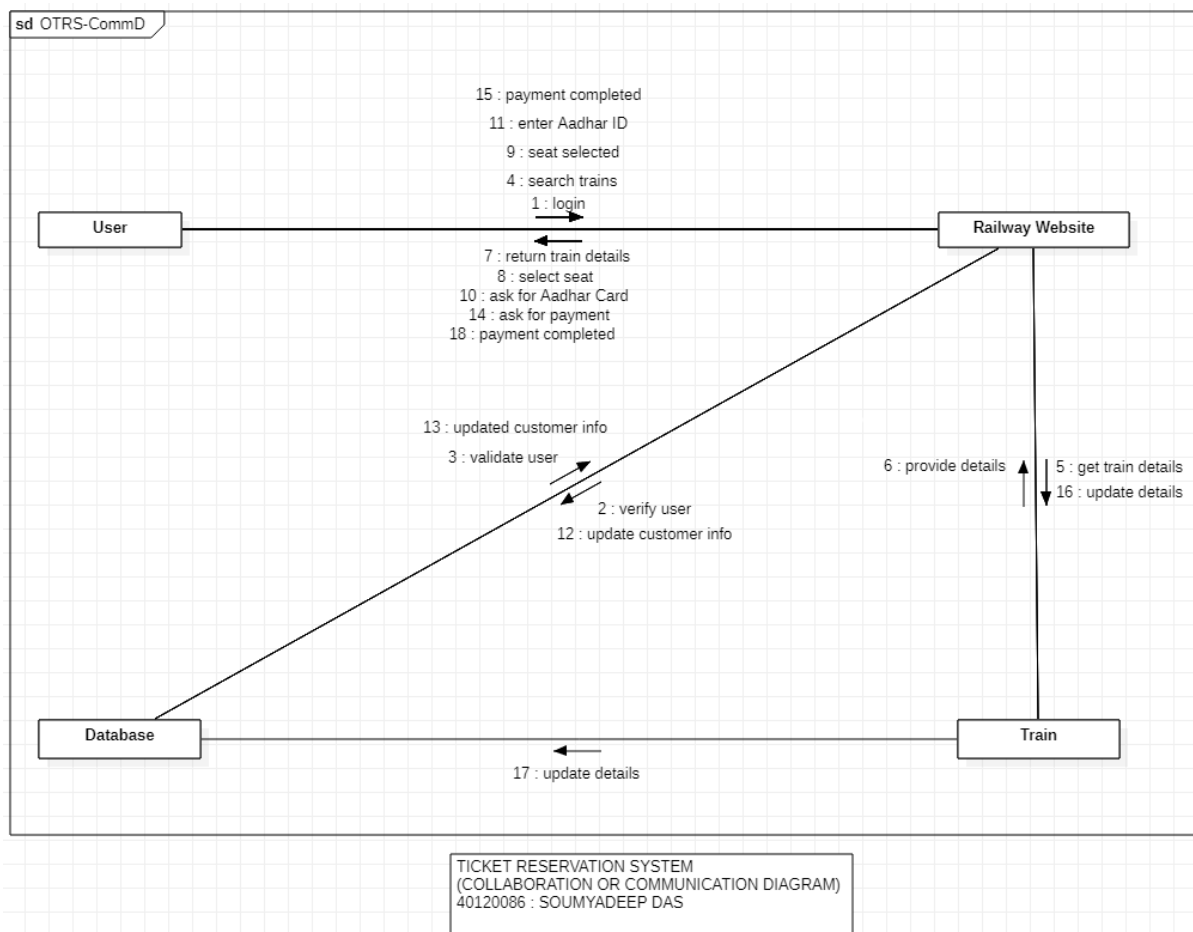


Fig 3.6 : Collaboration Diagram or Communication Diagram

3.3.6 STATECHART DIAGRAM

A statechart diagram in UML (Unified Modeling Language) is a type of behavioral diagram that illustrates the life cycle of an object or component in a system. It represents the states that an object or component can assume, and the transitions between these states.

In a statechart diagram, each state is represented as a rounded rectangle, and the transitions between states are represented as arrows. The transitions can be triggered by events, such as a change in a condition or the receipt of a message, and can include actions, such as updating the value of a variable or sending a message. Additionally, statechart diagrams can also include entry actions, exit actions, and internal actions.

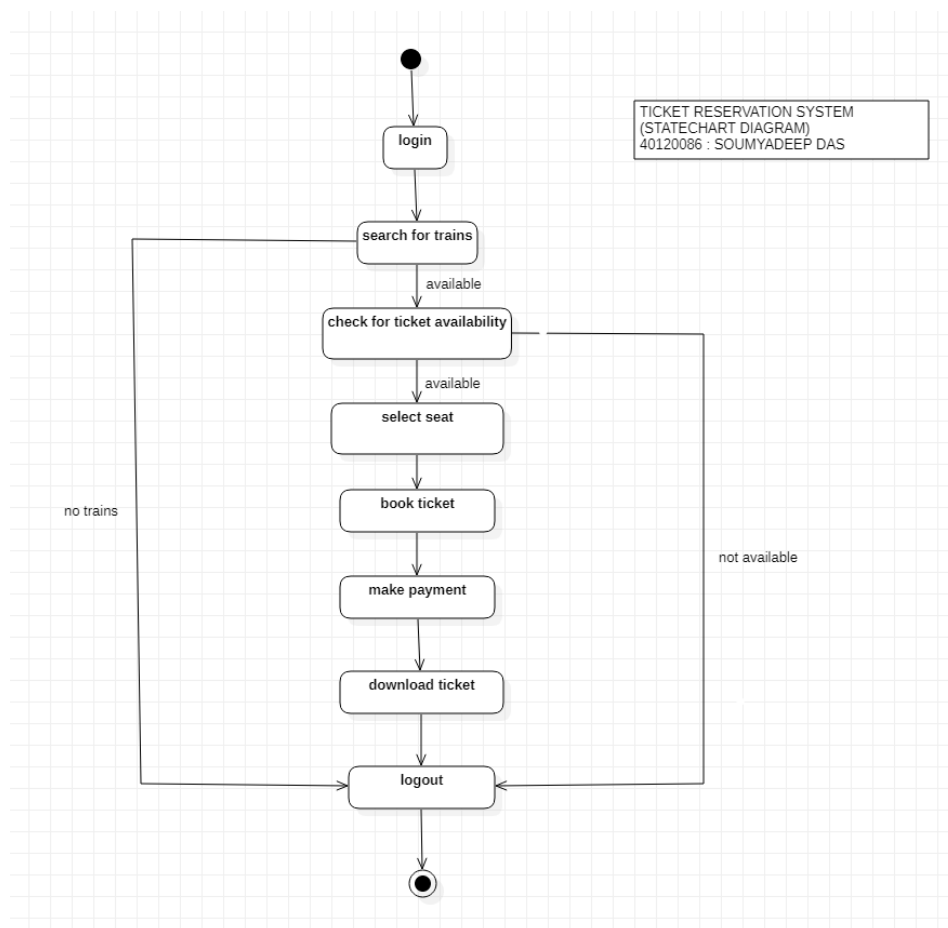


Fig 3.7 : Statechart Diagram

3.3.7 COMPONENT DIAGRAM

A component diagram in UML (Unified Modeling Language) is a type of structural diagram that illustrates the components of a system and their relationships. It is used to model the physical components of a system, such as software modules, hardware devices, and other building blocks.

In a component diagram, each component is represented as a rectangle, and the dependencies between components are represented as lines or arrows. The relationships between components can be of various types, such as a dependency, association, aggregation, or composition.

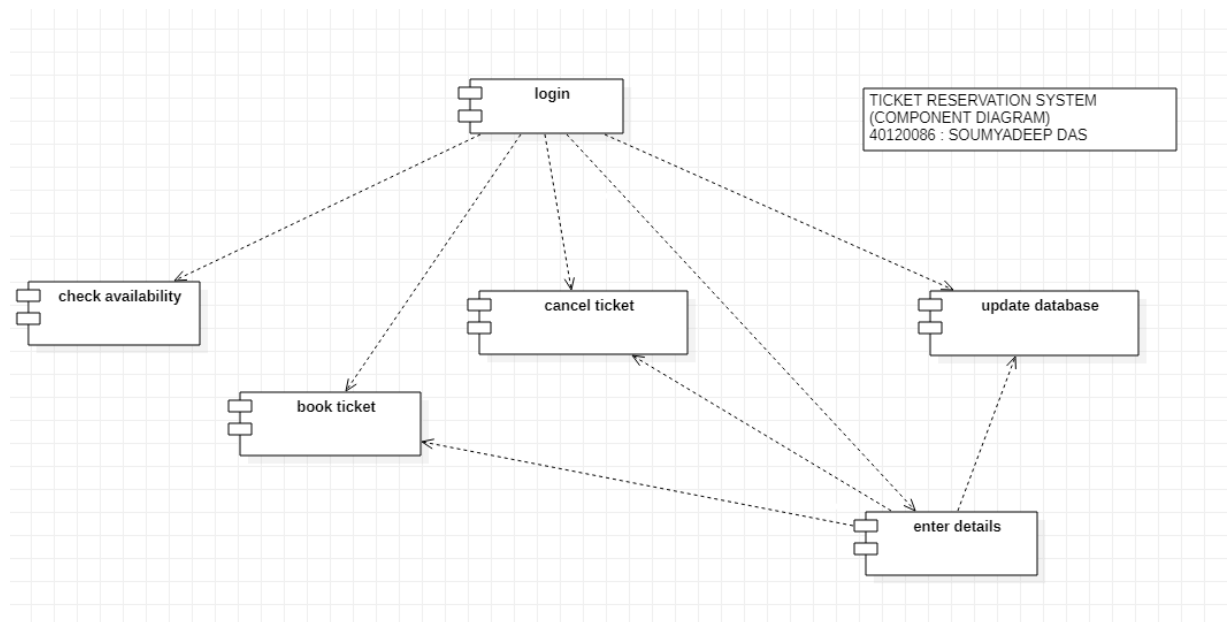


Fig 3.8 : Component Diagram

CHAPTER 4

RESULTS AND DISCUSSION

4.1 RESULT

The outcome of a ticket reservation system can vary depending on the specific industry or use case, but some common outcomes include:

Improving customer experience: To provide customers with a convenient, easy-to-use system that allows them to browse, select, and purchase tickets quickly and efficiently. This can help to improve customer satisfaction and encourage repeat business.

Maximizing revenue: Another outcome of a ticket reservation system is to maximize revenue for the business or organization. This can be achieved by offering different pricing options, promoting ticket sales through targeted marketing campaigns, and leveraging data analytics to identify sales trends and opportunities.

Streamlining operations: By automating the ticket reservation process, businesses and organizations can streamline their operations and reduce the costs associated with manual ticketing systems. This can include reducing staffing requirements, minimizing printing and distribution costs, and improving inventory management.

Enhancing data collection and analysis: Ticket reservation systems can provide businesses with valuable data on customer behaviour, sales trends, and other metrics. This data can be used to make informed business decisions, improve marketing strategies, and enhance the overall customer experience.

Hence, the results of a ticket reservation system are improving the customer experience, increasing revenue, streamline operations, and enhancing data collection and analysis to drive business growth and success.

A ticket reservation system can have a broad scope depending on its purpose and implementation. Here are some examples of areas where ticket reservation systems can be used:

Travel: Airlines, trains, buses, and other transportation companies use ticket reservation systems to manage seat bookings, schedules, and availability.

Entertainment: Ticket reservation systems are widely used for booking tickets for movies, concerts, theater performances, sports events, and other forms of entertainment.

Education: Schools, colleges, and universities use ticket reservation systems to manage registration and enrollment for classes, workshops, seminars, and other educational programs.

Tourism: Tour operators, travel agencies, and hotels use ticket reservation systems to manage bookings for tours, activities, and accommodations.

Healthcare: Hospitals and clinics use ticket reservation systems to schedule appointments for patients and manage their medical records.

Events: Event planners and organizers use ticket reservation systems to manage registration, attendee information, and ticket sales for conferences, trade shows, and other events like sports meet and sports organizations.

Overall, a ticket reservation system can be used in any industry or organization that requires booking or scheduling of services or events.

4.2 PROJECT OUTPUT

4.2.1 HOME PAGE OF TICKET RESERVATION SYSTEM

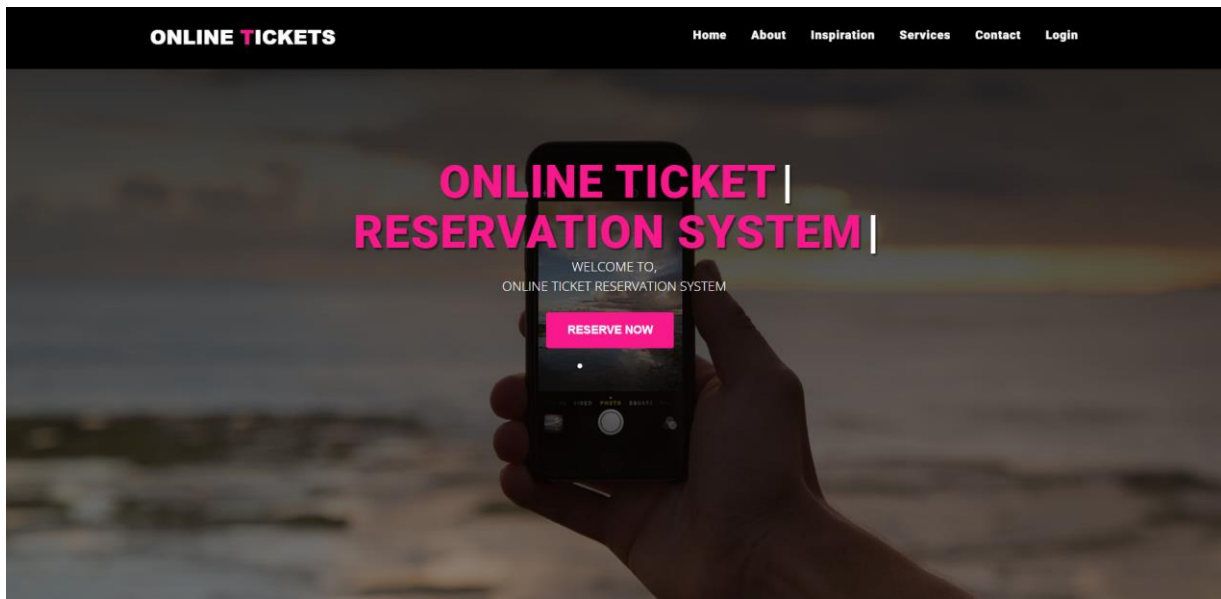


Fig 4.1 : Home Page of Ticket Reservation System

4.2.2 ABOUT PAGE OF TICKET RESERVATION SYSTEM

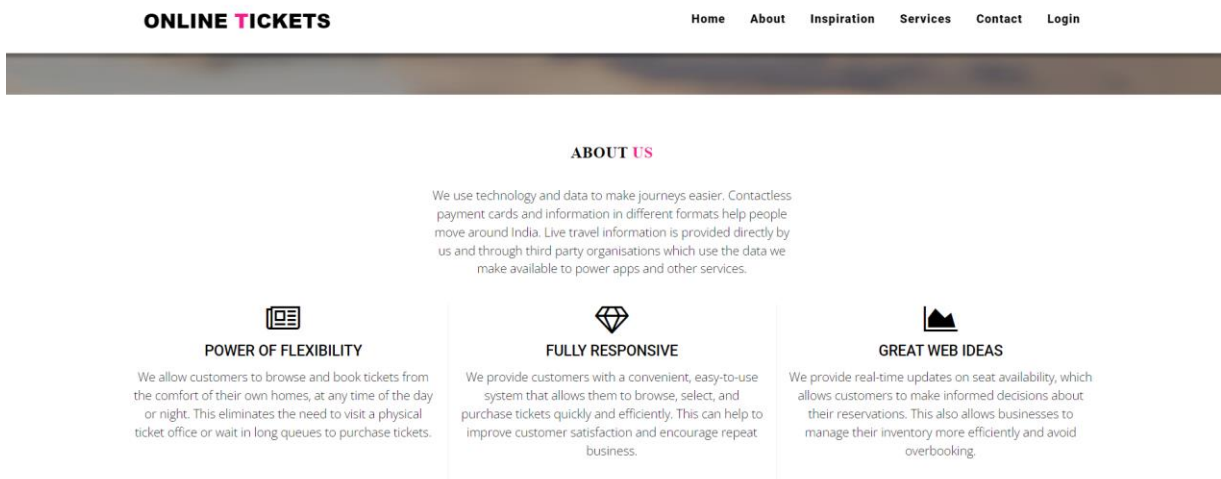


Fig 4.2 : About Page of Ticket Reservation System

4.2.3 INSPIRATION PAGE OF TICKET RESERVATION SYSTEM

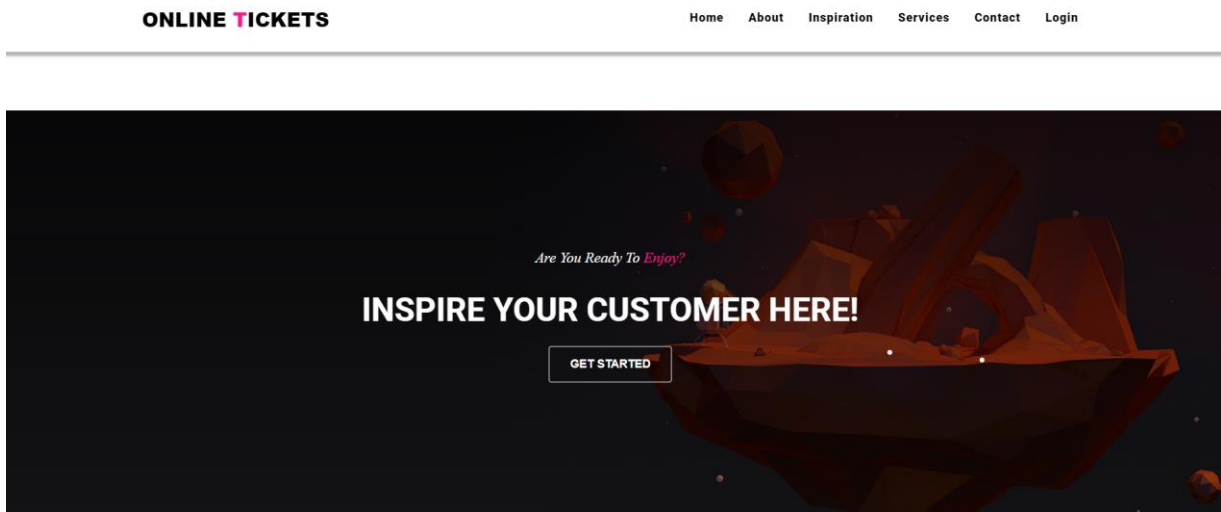


Fig 4.3 : Inspiration Page of Ticket Reservation System

4.2.4 SERVICES PAGE OF TICKET RESERVATION SYSTEM

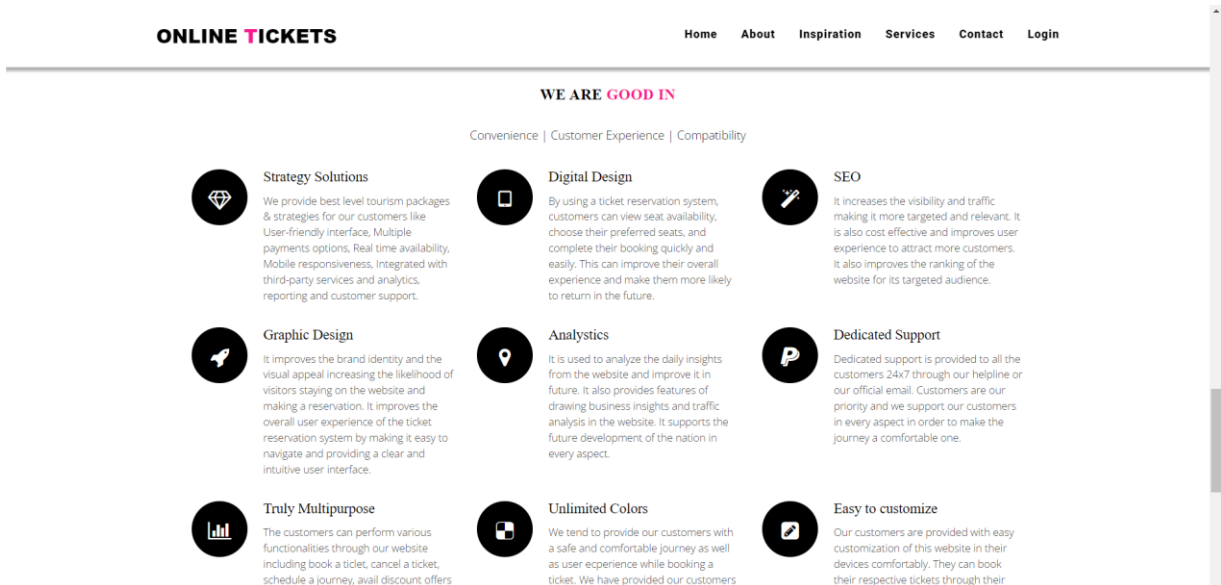
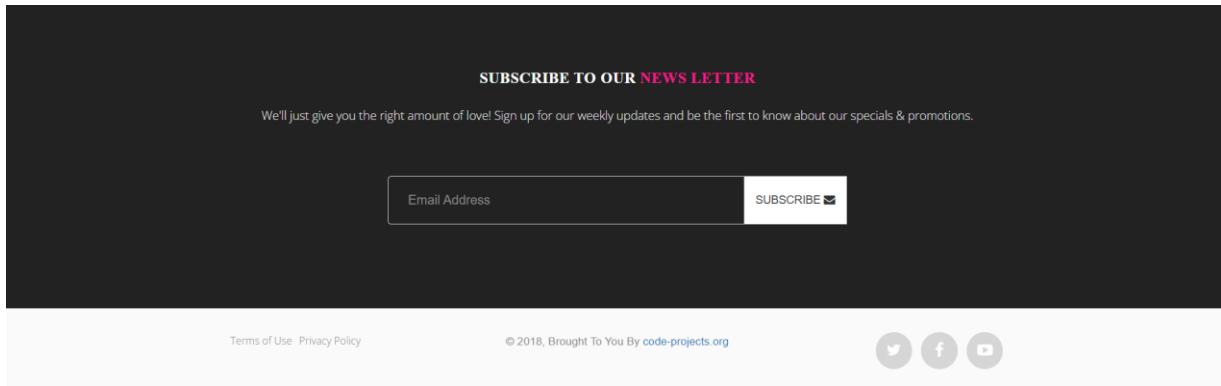


Fig 4.4 : Services Page of Ticket Reservation System

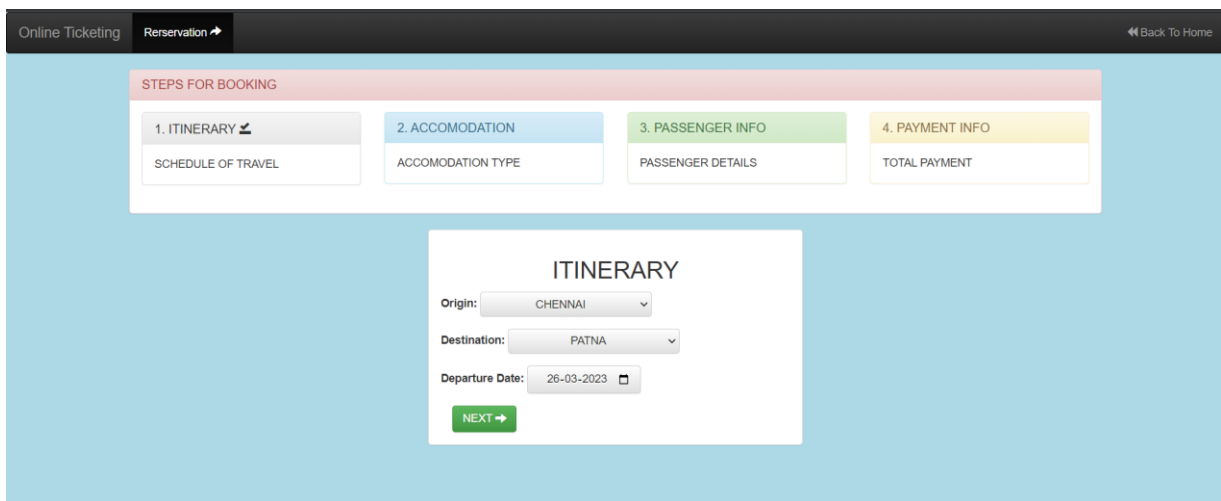
4.2.5 CONTACT PAGE OF TICKET RESERVATION SYSTEM



The image shows a dark-themed contact page. At the top, it says "SUBSCRIBE TO OUR NEWS LETTER" in white and pink text. Below this is a small line of text: "We'll just give you the right amount of love! Sign up for our weekly updates and be the first to know about our specials & promotions." In the center, there is a white input field labeled "Email Address" and a white button labeled "SUBSCRIBE" with an envelope icon. At the bottom, there is a light gray footer containing links for "Terms of Use" and "Privacy Policy", a copyright notice "© 2018, Brought To You By code-projects.org", and social media icons for Twitter, Facebook, and YouTube.

Fig 4.5 : Contact Page of Ticket Reservation System

4.2.6 ITINERARY PAGE OF TICKET RESERVATION SYSTEM



The image shows a light blue itinerary page. At the top, there is a dark header with "Online Ticketing" and "Reservation" (with a right arrow) on the left, and "Back To Home" (with a left arrow) on the right. Below the header, there is a section titled "STEPS FOR BOOKING" with four steps: 1. ITINERARY (selected, with a right arrow icon), 2. ACCOMODATION, 3. PASSENGER INFO, and 4. PAYMENT INFO. Each step has a sub-label: "SCHEDULE OF TRAVEL", "ACCOMODATION TYPE", "PASSENGER DETAILS", and "TOTAL PAYMENT" respectively. In the center, there is a white box titled "ITINERARY" containing a form with "Origin:" (CHENNAI), "Destination:" (PATNA), and "Departure Date:" (26-03-2023) fields, each with a dropdown arrow. Below these fields is a green "NEXT" button with a right arrow icon.

Fig 4.6 : Itinerary Page of Ticket Reservation Page

4.2.7 ACCOMODATION PAGE OF TICKET RESERVATION SYSTEM

STEPS FOR BOOKING

1. ITINERARY
SCHEDULE OF TRAVEL

2. ACCOMODATION
ACCOMODATION TYPE

3. PASSENGER INFO
PASSENGER DETAILS

4. PAYMENT INFO
TOTAL PAYMENT

ACCOMODATION

Accommodation	Slots	Fare
☐ Sitting	30	350
☐ Economy A	30	390
☐ Economy B	30	390
☐ Tourist	30	490
☐ Cabin	30	600
☒ Deluxe	30	700

Total # of Passenger:

2

NEXT →

Fig 4.7 : Accomodation Page of Ticket Reservation Page

4.2.8 ACCOMODATION PAGE OF TICKET RESERVATION SYSTEM (BOOKED SUCCESSFULLY)

localhost says
Booked Successfully!

OK

Booked(1)

Full Name (1):
Soumyadeep Das

Age: (1):
21

Gender: (1): Male

Booked(2)

Full Name (2):
Abitha Sri Kolli

Age: (2):
19

Gender: (2): Female

NEXT →

Fig 4.8 : Accomodation Page of Ticket Reservation Page (Booked Successfully)

4.2.9 PAYMENT INFO PAGE OF TICKET RESERVATION SYSTEM

PAYMENT INFO

DEPARTURE

Medallion Transport, M/V Lady of All Nations
CHENNAI - PATNA
Departure Date: 2023-03-26 @9:00AM
Estimated Arrival Time: The Next Day @3:00PM
Deluxe

CONTACT INFO

Book By: Soumyadeep Das
Contact #: 9113768420
Address: Chennai, Tamil Nadu

PASSENGERS - Booked ID: 641bdb43b70f8

Name	Age	Gender	Departure Price
Soumyadeep Das	21	Male	700
Abitha Sri Kolli	19	Female	700
		TOTAL:	1400

Return Home

Fig 4.9 : Payment Info Page of Ticket Reservation System

4.2.10 LOGIN PAGE OF TICKET RESERVATION SYSTEM

Online Ticketing

Return Home

Please Login Here

Username:

Enter Username

Password:

Enter password

☐ Remember me

Login

Fig 4.10 : Login Page of Ticket Reservation System

4.2.11 RESERVATION PAGE OF TICKET RESERVATION SYSTEM

Online Ticketing Admin Panel Reserved Transaction \$ Logout

Show 10 entries Search:

Book ID	Book By	Contact	Address	Departure Date	Action
No data available in table					

Showing 0 to 0 of 0 entries Previous Next

Fig 4.11 : Reservation Page of Ticket Reservation System

4.2.12 DISCOUNT ON RESERVATION PAGE OF TICKET RESERVATION SYSTEM

Online Ticketing Admin Panel Reserved Transaction \$ Logout

Show 10 entries Search:

Book ID	Book By
641bdb43b70f8	Soumyadeep Das

Showing 1 to 1 of 1 entries

Book By: Soumyadeep Das
Departure Date: 2023-03-26
Contact: 9113768420
Address: Chennai, Tamil Nadu

Name	Age	Gender	Discount	Price
Soumyadeep Das	21	Male	Student 10%	700
Abitha Sri Kolli	19	Female	Student 10%	700
TOTAL:				1400

Compute Accept Payment

Cancel Accept Payment Previous 1 Next

Fig 4.12 : Transaction Page of Ticket Reservation System

4.2.13 TRANSACTION PAGE OF TICKET RESRVATION SYSTEM

Online Ticketing Admin Panel

Reserved

Transaction \$

Logout

Show 10 entries

Search:

Name	Age	Gender	Accomodation	Paid	Action
Abitha Sri Kolli	19	Female	Deluxe	560	10 % 20 %
Soumyadeep Das	21	Male	Deluxe	560	10 % 20 %

Showing 1 to 2 of 2 entries

Previous

1

Next

Fig 4.13 : Transaction Page of Ticket Reservation System

4.3 PERFORMANCE ANALYSIS

To analyze the performance of a ticket reservation system, you can use the following **Key Performance Indicators** (KPIs):

Conversion rate: This KPI measures the percentage of website visitors who actually purchase a ticket. A low conversion rate may indicate issues with the user interface, pricing, or marketing efforts.

Website traffic: This KPI measures the number of visitors to the website. A high volume of traffic may indicate that marketing efforts are successful, but if conversion rates are low, it may indicate issues with the website or the ticket purchasing process.

Page load speed: The time it takes for web pages to load is critical for user experience. Slow loading times can lead to frustrated users and abandoned purchases.

Error rate: The error rate measures the number of times users encounter errors or issues while using the system. This includes errors during the purchasing process, as well as errors while accessing the website or using the mobile app.

Ticket availability rate: This KPI measures the percentage of time that tickets are available for purchase. A low availability rate can lead to frustrated customers and lost revenue.

Average order value: This KPI measures the average value of each ticket purchase. A high average order value can indicate that customers are buying premium tickets or purchasing add-ons, while a low average order value may indicate that pricing strategies need to be adjusted.

Customer satisfaction: This KPI measures the satisfaction of customers with the ticket reservation system. This can be measured through surveys or reviews.

By tracking these KPIs, businesses can identify areas of the ticket reservation system that require improvement and make data-driven decisions to improve the system's performance

CHAPTER 5

CONCLUSION AND FUTURE WORK

5.1 CONCLUSION

In conclusion, a ticket reservation system is a critical component of any business that sells tickets or manages bookings. The system provides a convenient and efficient way for customers to browse, select, and purchase tickets online. To ensure the success of a ticket reservation system, businesses should prioritize user experience, availability, and security. The system should have an easy-to-use interface, support multiple payment options, provide real-time availability, and offer 24/7 customer support. It should also be scalable, reliable, and secure to protect user data and prevent unauthorized access. By analyzing performance metrics and continuously optimizing the system, businesses can improve their ticket reservation system and provide a better experience for their customers.

5.2 FUTURE WORK

Some potential areas for future work on ticket reservation systems include:

Artificial Intelligence (AI) and Machine Learning (ML): The integration of AI and ML into ticket reservation systems can help businesses personalize their offerings, recommend relevant events, and improve pricing strategies based on customer behavior.

Blockchain technology: Blockchain technology can provide secure and transparent ticketing solutions that prevent fraud, reduce ticket scalping, and improve trust between ticketing stakeholders.

Virtual and augmented reality: Virtual and augmented reality technologies can create immersive experiences for customers, allowing them to visualize the event before purchasing tickets.

Voice-enabled interfaces: Voice-enabled interfaces can provide a hands-free and convenient way for customers to search and purchase tickets using voice commands.

Integration with social media: Social media integration can help businesses leverage social media platforms to promote events and reach a wider audience.

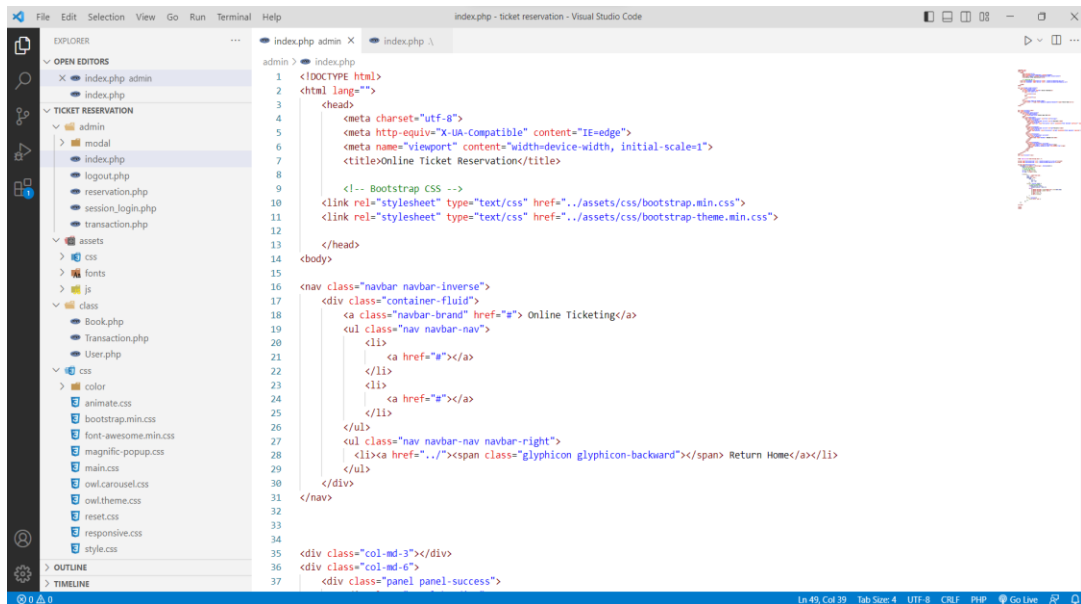
Big data analytics: Big data analytics can help businesses understand customer behavior, preferences, and trends, allowing them to optimize pricing strategies, promotions, and marketing campaigns.

Multi-language support: Multi-language support can make the ticket reservation system accessible to customers who prefer to use languages other than English.

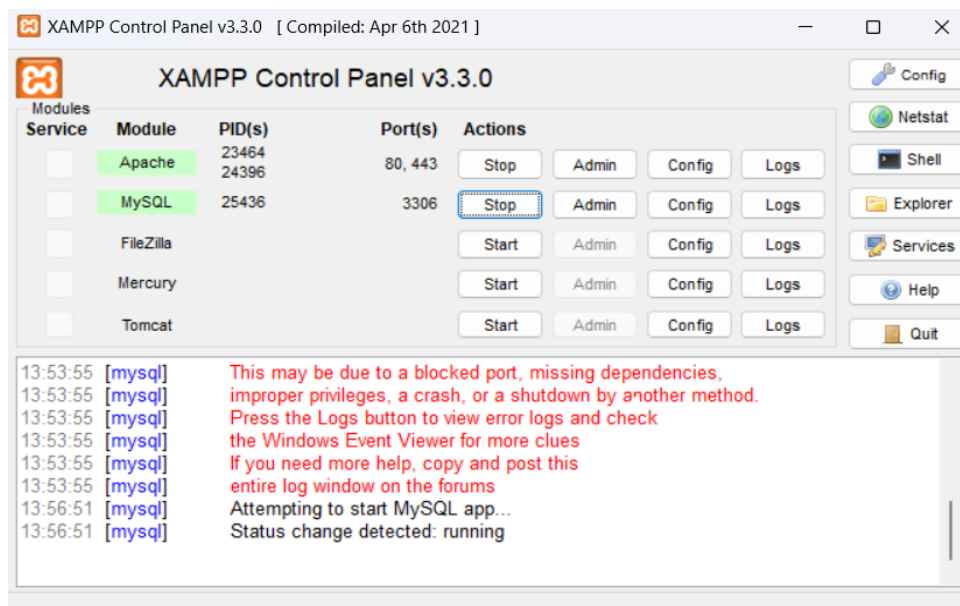
By incorporating these advanced technologies and features, ticket reservation systems can continue to evolve and provide better experiences for customers, while helping businesses increase sales, reduce costs, and stay competitive in a rapidly changing market.

5.3 WORKING ENVIRONMENT

1) VISUAL STUDIO CODE

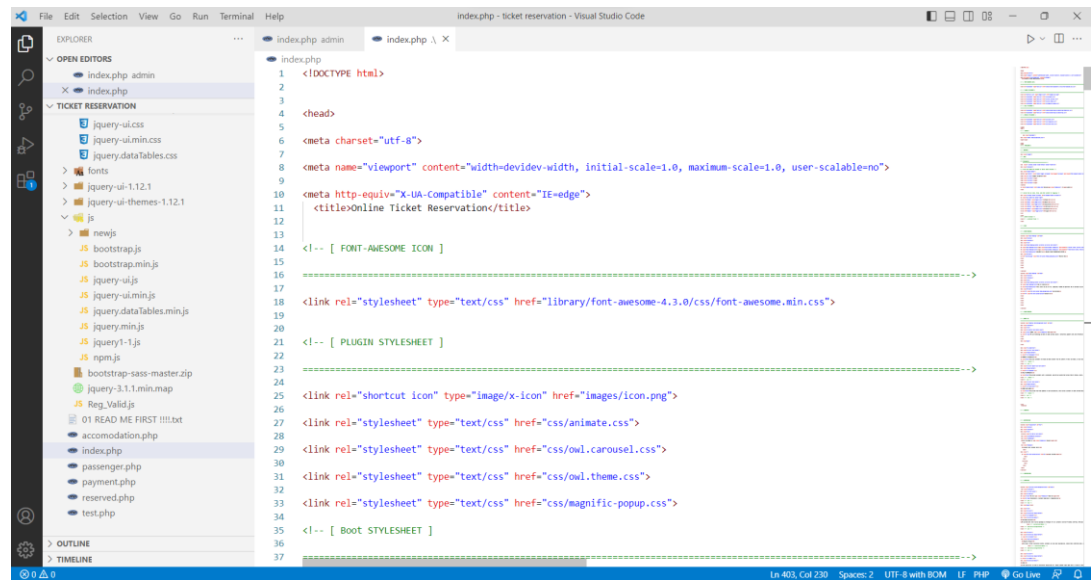


2) XAMPP



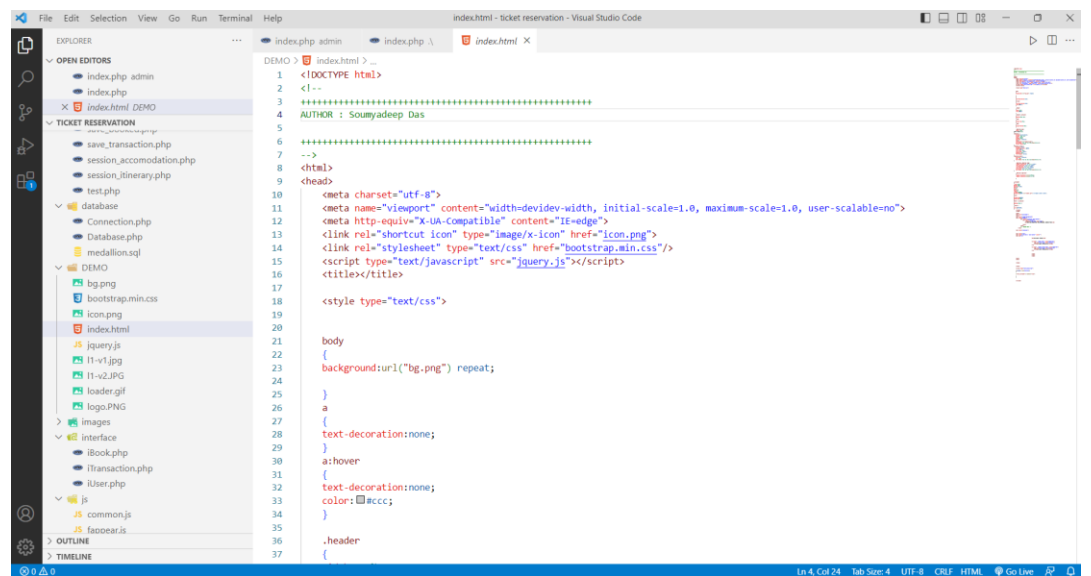
5.4 APPENDIX : SOURCE CODE

1) Index.php



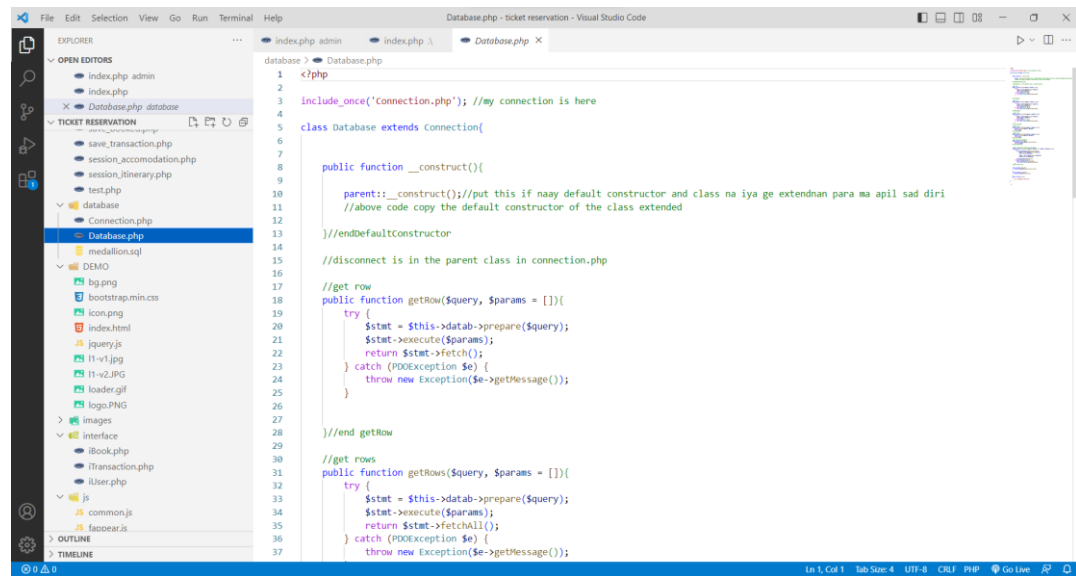
```
1 <!DOCTYPE html>
2
3
4 <head>
5
6 <meta charset="utf-8">
7
8 <meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">
9
10 <meta http-equiv="X-UA-Compatible" content="IE=edge">
11 <title>Online Ticket Reservation</title>
12
13
14 <!-- [ FONT-AWESOME ICON ]
15
16
17
18 <link rel="stylesheet" type="text/css" href="library/font-awesome-4.3.0/css/font-awesome.min.css">
19
20
21 <!-- [ PLUGIN STYLESHEET ]
22
23
24
25 <link rel="shortcut icon" type="image/x-icon" href="images/icon.png">
26
27 <link rel="stylesheet" type="text/css" href="css/animate.css">
28
29 <link rel="stylesheet" type="text/css" href="css/owl.carousel.css">
30
31 <link rel="stylesheet" type="text/css" href="css/owl.theme.css">
32
33 <link rel="stylesheet" type="text/css" href="css/magnific-popup.css">
34
35 <!-- [ BOOT STYLESHEET ]
36
37
```

2) Index.html



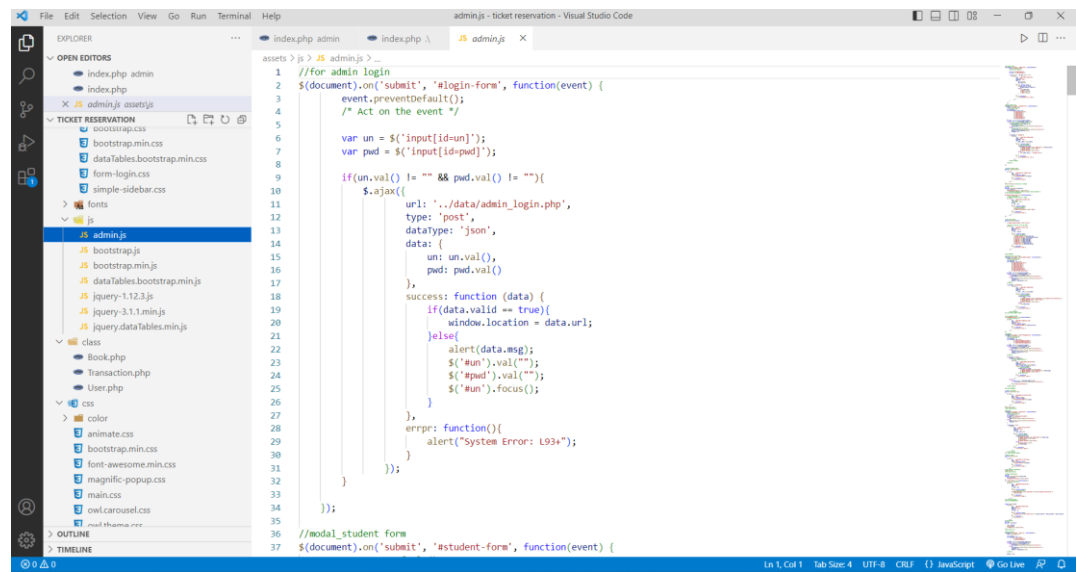
```
1 <!DOCTYPE html>
2 <!--
3 *****
4 AUTHOR : Soumyadeep Das
5 *****
6
7 -->
8 <html>
9 <head>
10
11 <meta charset="utf-8">
12 <meta name="viewport" content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user-scalable=no">
13 <meta http-equiv="X-UA-Compatible" content="IE=edge">
14 <link rel="shortcut icon" type="image/x-icon" href="icon.png">
15 <link rel="stylesheet" type="text/css" href="bootstrap.min.css"/>
16 <script type="text/javascript" src="jquery.js"></script>
17 <title></title>
18
19 <style type="text/css">
20
21 body
22 {
23 background:url("bg.png") repeat;
24
25 }
26
27 a
28 {
29 text-decoration:none;
30
31 a:hover
32 {
33 text-decoration:none;
34 color:#ccc;
35 }
36
37 .header
38 {
```

3) Database.php



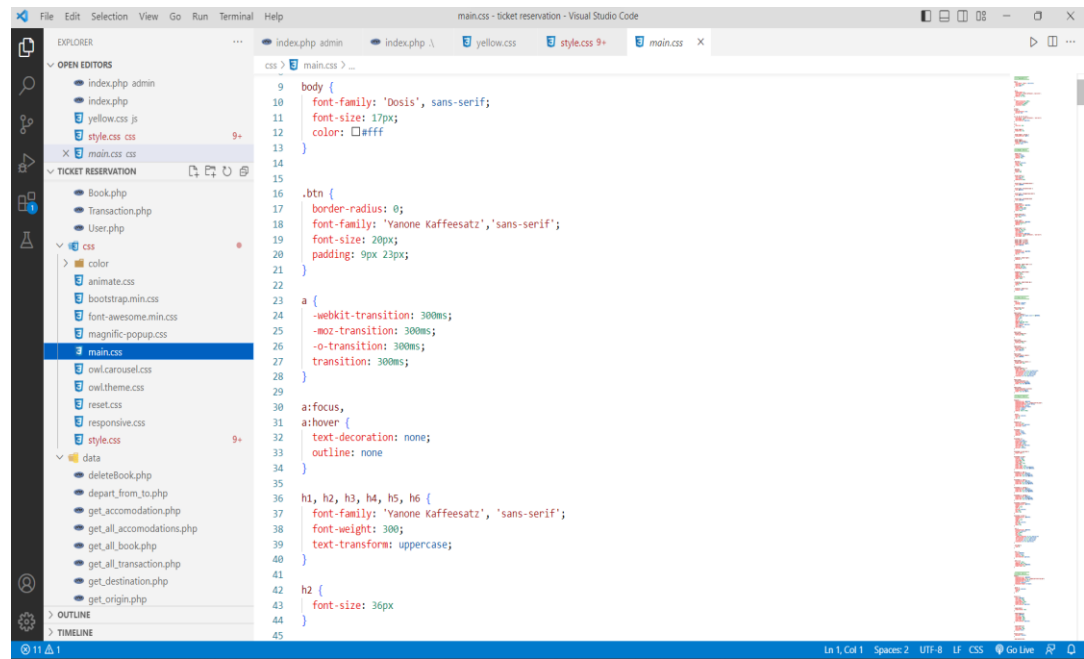
```
1 <?php
2
3 include_once('Connection.php'); //my connection is here
4
5 class Database extends Connection{
6
7
8     public function __construct(){
9
10         parent::__construct();//put this if naay default constructor and class na iya ge extendnan para ma apil sad diri
11         //above code copy the default constructor of the class extended
12     }
13 }
14
15 //disconnect is in the parent class in connection.php
16
17 //get row
18 public function getRow($query, $params = []){
19     try {
20         $stmt = $this->data->prepare($query);
21         $stmt->execute($params);
22         return $stmt->fetch();
23     } catch (PDOException $e) {
24         throw new Exception($e->getMessage());
25     }
26 }
27
28 //end getRow
29
30 //get rows
31 public function getRows($query, $params = []){
32     try {
33         $stmt = $this->data->prepare($query);
34         $stmt->execute($params);
35         return $stmt->fetchAll();
36     } catch (PDOException $e) {
37         throw new Exception($e->getMessage());
38     }
39 }
```

4) Admin.js



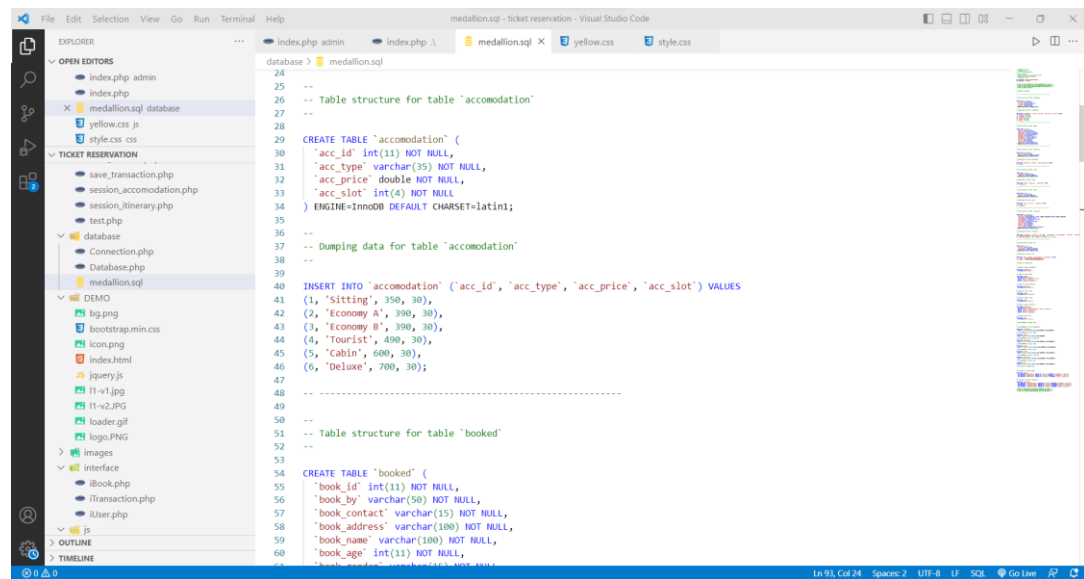
```
1 //for admin login
2 $(document).on('submit', '#login-form', function(event) {
3     event.preventDefault();
4     /* Act on the event */
5
6     var un = $('#input[id=un]');
7     var pwd = $('#input[id=pwd]');
8
9     if(un.val() != "" && pwd.val() != ""){
10         $.ajax({
11             url: './data/admin_login.php',
12             type: 'post',
13             dataType: 'json',
14             data: {
15                 un: un.val(),
16                 pwd: pwd.val()
17             },
18             success: function (data) {
19                 if(data.valid == true){
20                     window.location = data.url;
21                 }else{
22                     alert(data.msg);
23                     $('#un').val("");
24                     $('#pwd').val("");
25                     $('#un').focus();
26                 }
27             },
28             error: function(){
29                 alert("System Error: L934");
30             }
31         });
32     }
33 });
34
35 //modal student form
36 $(document).on('submit', '#student-form', function(event) {
37
```

5) Main.css



```
9 body {
10   font-family: 'Dosis', sans-serif;
11   font-size: 17px;
12   color: #fff;
13 }
14
15
16 .btn {
17   border-radius: 0;
18   font-family: 'Vanone Kaffeesatz', 'sans-serif';
19   font-size: 20px;
20   padding: 9px 23px;
21 }
22
23
24 a {
25   -webkit-transition: 300ms;
26   -moz-transition: 300ms;
27   -o-transition: 300ms;
28   transition: 300ms;
29 }
30
31 a:focus,
32 a:hover {
33   text-decoration: none;
34   outline: none;
35 }
36
37 h1, h2, h3, h4, h5, h6 {
38   font-family: 'Vanone Kaffeesatz', 'sans-serif';
39   font-weight: 300;
40   text-transform: uppercase;
41 }
42
43 h2 {
44   font-size: 36px;
45 }
```

6) Medallion.sql



```
24 database > medallion.sql
25 --
26 -- Table structure for table `accommodation`
27 --
28
29 CREATE TABLE `accommodation` (
30   `acc_id` int(11) NOT NULL,
31   `acc_type` varchar(35) NOT NULL,
32   `acc_price` double NOT NULL,
33   `acc_slot` int(4) NOT NULL
34 ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
35
36 --
37 -- Dumping data for table `accommodation`
38 --
39
40 INSERT INTO `accommodation` (`acc_id`, `acc_type`, `acc_price`, `acc_slot`) VALUES
41 (1, 'Sitting', 350, 30),
42 (2, 'Economy A', 390, 30),
43 (3, 'Economy B', 390, 30),
44 (4, 'Tourist', 490, 30),
45 (5, 'Cabin', 600, 30),
46 (6, 'Deluxe', 700, 30);
47
48
49
50 --
51 -- Table structure for table `booked`
52 --
53
54 CREATE TABLE `booked` (
55   `book_id` int(11) NOT NULL,
56   `book_by` varchar(50) NOT NULL,
57   `book_contact` varchar(15) NOT NULL,
58   `book_address` varchar(100) NOT NULL,
59   `book_name` varchar(100) NOT NULL,
60   `book_age` int(11) NOT NULL,
```