**INDEX**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Section** | **Title** |
| **1** |  | **Introduction** |
| 1.1 | Company Profile |
| 1.2 | Abstract |
| **2** |  | **Proposed System** |
| 2.1 | Need of System |
| 2.2 | Existing System and Problem Statement |
| 2.3 | User requirements |
| 2.4 | Purpose/Objective/Goals of proposed System |
| 2.5 | Literature Survey |
| 2.6 | Scope of Work |
| **3** |  | **System Analysis** |
| 3.1 | Detailed description of Technology used |
| 3.2 | Operating Environment – S/W and H/W |
| 3.3 | Project perspective and Features |
| 3.4 | Stakeholders |
| 3.5 | Requirement analysis – Functional requirements, Performance  requirements, Security requirements etc. |
| **4** |  | **System Design** |
| 4.1 | System Design |
| 4.2 | System Model – UML Diagrams – DFD0, DFD1, DFD2, Context Level Diagram |
| 4.3 | Data Models – E-R Diagram, Class Diagram, Use-Case Diagram, Activity Diagram, Sequence Diagram, Component Diagram, Module Hierarchy Diagram, Collaboration Diagram |
| 4.4 | User Interfaces |
| **5** |  | **Implementation Details** |
| 5.1 | Implementation Technologies |
| **6** |  | **Testing** |
|  | 6.1 | Test Plan, Black Box Testing or Data Validation Test Cases, White Box Testing or Functional Validation Test, Test Cases & Results |
| **7** |  | **Output** |
| **8** |  | **User Manual** |
| 8.1 | User Manual |
|  | 8.2 | Operating Manual |
| **9** |  | **Advantages and Limitations** |
| **10** |  | **Future Scope/Proposed Enhancements** |
| **11** |  | **Conclusion** |
| **12** |  | **Bibliography and References** |

**1. Introduction:**

**1.1 Company Profile:**

**SGMS INFOTECH LLP** works on Web Application Project, Academic Project & Mobile Application Development, also develop the Connection in Hardware and Software Project and Provide the Best Service to client & Student, also Provide Internship to Computer Sciences and Information Technologies Branches Student like B.E, MCA, MCS and BCA to give them an exposure to real time work environment by letting them work on live projects and enhancing their overall skills set to improve their employability level to match industry expectations.

Company provides services like software and web site development. Company motivated by its distribution channel aggressively by serving engineering need in sharp time, which saves time.

Company has excellent marketing network with technical support. Company deal with application development where they develop different application which help clients to do work efficiently and effectively. Following services are provided by Company to their clients are as…

1. Website Design & development
2. Software Development
3. Digital Marketing
4. Search Engine Optimization
5. Logo Design & Animation
6. E-commerce Web Design
7. Android App Development

**Contact Person:**

Mr. Dinesh Surange

SGMS Infotech LLP

**Mob No:** +91 8275329929

**Email ID:** [sgmsinfotech@gmail.com](mailto:sgmsinfotech@gmail.com)

**1.2 Abstract:**

The Ticket Master Travel Reservation System is a Java-based software platform that enables travelers to easily reserve and manage their travel arrangements. The proposed system is designed to be user-friendly and accessible from anywhere in the world.

The “Web-based Ticket Master Travel Reservation System” has been developed to override to problems prevailing in the practicing manual system. This software is supported to eliminate and, in some cases, reduce the hardship faced by this existing system. Moreover, this system is designed for need of the company to carry out operation in a smooth and effective manner.

The application is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is need for the user to use this system. Thus, by this all is proves it is user-friendly. Web-based Ticket Master Travel Reservation System, as described above can be led to error free, secure, reliable, and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus, it will help organization in better utilization of resources.

Every bus ticket booking system has different bus need, therefore we design exclusive employee management systems that are adapted to your managerial requirements. This is designed to assist in strategic planning and will help you ensure that your organization is equipped with the right level of information and details of your future goals. This system will ultimately allow you to better manage resources.

**2. Proposed System**

**2.1 Need of System:**

The “Web-based Ticket Master Travel Reservation System” has been developed to override to problems prevailing in the practicing manual system. This proposed system can be led to error free, secure, reliable, and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus, it will help organization in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant while able to reach the information.

The aim is to automate its existing manual system by the help of computerized equipment and full- fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. Basically, the project describes how to manage for good performance and better services for the clients.

**2.2 Existing System and Problem Statement:**

Several challenges exist within the travel industry, which can be addressed by modernizing and automating processes. The main of proposed system is to develop a system of improved facilities. The proposed system can overcome all the limitation of the existing system. The Existing System has following challenges are as…

* **Manual Tasks and Inefficiencies –** Manual processes are time-consuming, error-prone, and hinder efficient decision-making. Existing system is totally on books and thus a great amount of manual work need be done. The amount of manual work increases exponentially with increase in services. Also, needs a lot of working staff and extra attention on all the records.
* **Data unavailability and difficulty in accessing information –** In existing system, there are various problems like keeping records of items, seats available, prices of per/seat and fixing bill generation on each bill. Finding out details regarding any information is very difficult, as the user need to go through all the books manually.
* **Scalability and Adaptability –** Existing systems face challenges such as inefficiencies, lack of scalability, and difficulty in adapting to changing industry requirements.

**2.3 User Requirements:**

Developing an online bus ticket reservation system involves understanding the user requirements and implementing key features. Here are some essential functionalities you should consider:

1. **Easy Registration –** Implement a user-friendly sign-up/sign-in feature with social login functionality.
2. **User-friendly interface –** Design an intuitive and easy-to-navigate interface for a seamless user experience.
3. **Booking details –** Date and time for their outward journeys, number of travelers,
4. **Trip choice and Tour details –** After entering the necessary details, provide users with relevant trip options that meet their criteria.
5. **Passenger information –** Once users select a preferred trip, prompt them to enter passenger details.
6. **Real-time booking –** Ensure that the system updates availability in real time, preventing double bookings.
7. **Inventory System -** Keep track of available buses, seats, and routes.

**2.4 Purpose/Objective/Goals of proposed System:**

This study is aimed at developing an “Web-based Ticket Master Travel Reservation System” which allow customer easy book their tickets and cancel choice and provide information about bus. The system provides proper security and reduces manual work. Some advantages/purposes of proposed system over old/existing system are as…

* User friendliness and interactive.
* Minimize manual data entry.
* Minimum time needed for the various processing.
* Focus on core business services than manual data entry.
* Security of data.
* Ensure data accuracies.
* Reduction in processing time.
* Best in class inventory management

**2.5 Literature Surveys:**

* Paper Name: **Online Bus Booking Service in India: A Study on RedBus**
* Company Author: Dr. Ramakrishna Bandaru Dr. S. Radhakrishnan Post Doctoral Fellow (UGC)

Description: Booking of bus tickets in India is quite simple in these days. redBus is India’s largest online bus ticketing company. redBus has revolutionized bus ticketing industry in India. redBus sells seats directly to the customers/passengers via its website. The company has about 75000 point of sale outlets all over the country. The company has launched in August 2006 with two bus operators and a daily inventory of 10 seats covering two routes. Now, it has a network more than 700 operators and a daily inventory of 500000 seats across 2500 cities in 15 states in India. redBus provides bus travellers, the most uncomplicated and hassle-free booking experience ever. redBus is an online platform that allows users to book for bus tickets. It is one of the fastest growing Indian web start-ups due to their implementation of hassle-free ticket booking system.

* Paper Name: **Online Ticket Master Travel Reservation System to The National Transportation Service in Sri Lanka.**
* Author: W Fernando and SCM De S Sirisuriya

Description: The distance bus ticket reservation has been one of the major concerns in the National Transport Commission (NTC) and other transports commissions in new world. There is a growing demand and trend in moving to online Ticket Master Travel Reservation Systems which are based on national transport in Sri Lanka. User’ (passenger) self-interaction in ticket reservation has been changing in a positive way with the introduction of these systems. With the evolution of these new coming systems, it will cause the traditional bus ticket reservation space to be changed. At present the current bus transportation system use both manual tickets and digital ticketing machine that printed tickets. These novel digital machines can store and process data and thus provide the conductor with several significant statistics at the end of the day.

* Paper Name: **Mobile – Based Bus Ticketing System in Iraq**
* Author: Ahmed K. Ibrahim and Azman B Ta

Description: Public transportation has been classified as an essential mode of travelling. Public in many countries, especially in the third world, prefer to use buses and train services to travel from one location to another. “Electronic ticketing system is the ticketing system which uses self-service technology as a base of application helping the user to book a ticket by themselves” (Ferreira, Porfírio, Cunha, &amp; Silva, 2013). Recently, with the introduction and evolution of smartphones applications, the consumers’ behavioural habits have changed in the goods and services purchasing power. Consumers’ on-line purchases using mobile application had increasing globally, without boundaries.

The online booking, in general, gives the customer extra information regarding the company, availability of the services, ticketing structure, prices, online payment and booking structure and other services rendered.

* Paper Name: **Online Ticket Master Travel Reservation System**
* Author: Nwakanma Ifeanyi Cosmas, Etus C, Ajere I.U., Agomuo Uchechukwu Godswill.

Description: Online Ticket Master Travel Reservation System is a Web based application that works within a centralized network. This project presents a review on the software program. Online Ticket Master Travel Reservation System as should be used in a bus transportation system, a facility which is used to reserve seats, cancellation of reservation and different types of route enquiries used on securing quick reservations. OBTRS is built for managing and computerizing the traditional database, ticket booking and tracking bus and travel made. It maintains all customer details, bus details, reservation details. To achieve the design, Imo Transport Company (ITC) was chosen as a case study because of its strategic importance to Imo State. Structured Systems Analysis and Design Methodology (SSADM) was adopted. In addition, PHP Hypertext Preprocessor (PHP) language was used for the front-end of the software while the back end was designed using MySQL. The software achieved is capability of improving the customer relationship management in ITC operations.

**2.6 Scope of work:**

Here, are the key components to consider for scope of work as…

1. **User Registration and Authentication:**

* Implement a secure user registration and login system.
* Allow users to create accounts, log in, and manage their profiles.

1. **Bus Routes and Schedules:**

* Design a database to store information about bus routes, schedules, and availability.
* Allow users to search for buses based on origin, destination, date, and time.
* Display available buses, seat availability, and fares.

1. **Booking and Payment:**

* Develop a booking system that enables users to select seats and book tickets.
* Integrate payment gateways for secure transactions.

1. **Admin Dashboard:**

* Create an admin panel for bus operators or system administrators.
* Admins should be able to manage bus routes, schedules, booking and employees records.
* Implement features like adding new routes, updating schedules, and handling cancellations.

1. **Seat Selection and Layout:**

* Design an intuitive seat selection interface for users.
* Show bus layouts with available and booked seats.
* Handle seat reservations and cancellations.

1. **Search and Filters:**

* Implement robust search functionality for users to find buses easily.

1. **Reporting and Analytics:**

* Create reports for admins, such as daily bookings, revenue, and occupancy rates.
* Use analytics to track user behavior and improve the system.

1. **Security and Privacy:**

* Implement security measures to protect user data.

1. **Testing and Quality Assurance:**

* Thoroughly test the system to identify and fix any bugs.

1. **Documentation:**

* Prepare detailed documentation for developers, including system architecture, APIs, and database schema.

**3. System Analysis**

System analysis is a process of gathering and interpreting facts, diagnosing problems and the information about the bus booking tickets to recommend improvement on the system. It is problem solving activity that requires intensive communication between the system users and system developers. System analysis or study is an important phase of any system development process. The system is studied to the minutest detail and analyzed. In this chapter, we shall be explaining the technologies and tools, programming languages and methodology used in this project.

**3.1 Detailed description of Technology used:**

**Methodology:**

The proposed Ticket Master Travel Reservation System is designed to be user-friendly and accessible from anywhere in the world. The system will be developed using the Java programming language and CSS, HTML, JSP, JavaScript will give a visually appealing and easy-to-use user interface.

It is essential that any programmer must thoroughly know the language he or she uses when designing and analyzing. The programmer would have to analyses the program and then knows the problem he needs to solve. He would then perform the process of coding while applying the process of design which he presented previously. Finally, he or she would have to test the program in order as certain compatibility with customer requirement. The process we have mentioned, including Analysis, Design, Coding and Testing, identify the project Software Development Life Cycle (SDLC) as any Project would have to go through all these processes using the appropriate methodology. Otherwise, chaos would ensue. We shall use the Interactive and incremental development methodology to develop a prototype system. This process is characterized with flexibility and revision whenever necessary in all phases.

**Modules:**

• **Bus Management:** Used for managing the bus details.

• **Route Management:** Used for managing the detail of bus route.

• **Bus Schedule:** Used for managing the details of schedule bus.

• **Tickets Management:** Used to managing the information and details of the tickets.

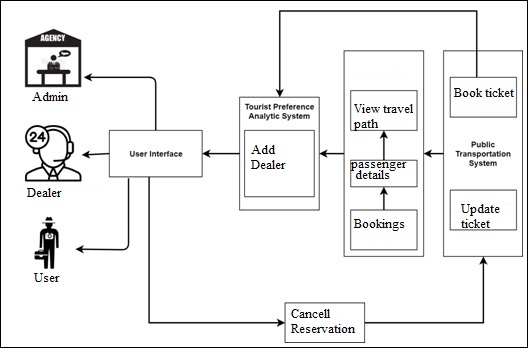
• **Booking Management:** Used for managing the booking details.

• **Customer Management:** Used for managing the customer information.

• **Login Management:** Used for managing the login details.

• **User Management:** used to managing the user of the system.

**System Architecture:**



**3.2 Operating Environment – S/W and H/W:**

**Software Requirements:**

* + Language: Java (JSP, Servlet Frameworks) [ jdk1.8 Version].
  + Frond End: HTML, CSS, JavaScript.
  + IDE: Eclipse, XAMPP, Apache tomcat
* Database: MySQL
* Server: Apache Tomcat Server

**Hardware Requirements:**

* + System Type: 64-bit or 32-bit
  + Processor: Intel core i5, 2 GHz
  + Random Access Memory (RAM): 8 GB
  + Storage Capacity: 1 TB
  + IO device: Mouse and Keyboard
  + Device Name: Laptop or Computer and Smartphone

**3.3 Project perspective and Features:**

* Full-featured talent marketplace & project management platform built for the Travel Industry.
* Easily define your requirement, our intelligent system track symptoms, and search best expert service through designed application.
* **Project Perspective:**
* User-Centric Approach – The project should prioritize user needs and convenience. Focus on creating a seamless experience for travelers, from searching for bus routes to booking tickets.
* Efficiency and Automation – Replace manual processes with automated features. Streamline ticket booking, payment, and seat allocation.
* Scalability – Design the system to handle many users simultaneously. Consider future expansion to cover more routes and bus operators.
* Security and Privacy – Maintaining Security of the application and privacy of the users.
* **Features:**
* Route Search and Selection
* Real-Time Availability
* User Profiles and Authentication
* Booking Process – Users can select preferred trips and book tickets. Choose seat preferences.
* Payment Gateway Integration – Integrate payment gateways for secure transactions.
* Booking History and Notifications – Maintain a booking history for users.
* Admin Dashboard

**3.4 Stakeholders:**

Developing an online bus ticket reservation system involves several stakeholders who play crucial roles in its success. Here are some key stakeholders:

**Users (Customers):** These are the passengers who use the system to book bus tickets. They interact with the system to check availability, select seats, and make reservations. Ensuring a user-friendly experience is essential to attract and retain customers1.

**Administrators:** Administrators manage the system. They handle tasks such as adding new bus routes, updating schedules, managing seat inventory. Their role is critical for maintaining the system’s efficiency and accuracy.

**Bus Operators/Transport Companies:** These stakeholders provide the bus services. They need access to the system to manage their fleet, allocate buses to routes, and monitor bookings. Integrating with bus operators’ existing systems is essential for seamless operations.

**Payment Gateway Providers:** Since online bookings involve financial transactions, payment gateway providers are crucial stakeholders. They ensure secure payment processing for customers.

**Software Developers and IT Team:** The team responsible for developing and maintaining the system is essential. They design, code, test, and deploy the application.

**3.5 Requirement Analysis**

**Functional Requirements:**

Requirement Analysis is the first phase of software development process. This phase focuses to understand the problem. Requirement Analysis is on identifying what is need from these systems, not how the system will achieve its goals. In this phase often at least two parties are involved in Software Development-a client and a developer. The developer must develop the system to satisfy the clients’ needs. The developer and client arrange a meeting and discuss his/her own views. The developer asks the clients for his/her needs. After a meeting the developer understands what the requirements of the client are. According to those requirements the developer starts development process. Hence the developer needs a user’s problem.

In the software requirement we are dealing with the requirements of the proposed system, that’s the capabilities of that system, which is yet to be developed, should have. The software requirement specification (SRS) is a document that completely describes what the proposed software should do without describing how the software will do it. So, the basic goal of Requirement Phase is to produce the SRS, which describes the complete external behavior of the proposed software. This involves a meeting of user and developers. The developer may ask the following questions to users.

1. Who will use the developed software?
2. What types of characteristics may have the software?

The above questions are to be answered by us. For the first question our answer will be that the system will be used by Bus Operators/Transport Companies. These stakeholders provide the bus services. They need access to the system to manage their fleet, allocate buses to routes, and monitor bookings. Integrating with bus operators’ existing systems is essential for seamless operations.

For the second question our answer is that the characteristics of our system are user friendliness and interactive system environment which minimize manual data entry, saves time, which will focus on core business services than manual data entry. Environment which manages security of data, ensures data accuracies, and best in class inventory management.

# **Performance Requirements:**

The following performance characteristics were taken care of in developing the systems:

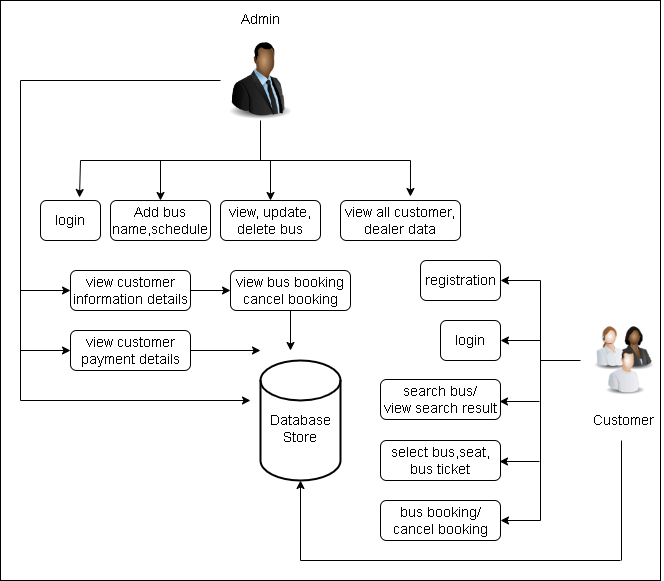
* **User Friendliness**: The system is easy to learn and understand. A native user can also use the system effectively, without any difficulty.
* **User Satisfaction**: The system is such that it stands unto the user’s expectation.
* **Error Handling**: Response to user errors and undesired situations has been taken care of to ensure that the system operations without halting in case of such situation and proper are given to user.
* **To Validations**: This results in a thorough testing of the details, and the system changes and modifications can be easily incorporated in it.
* **Portability:** The system can move to a new hardware/ operating system after making minor modifications to it.
* **Exception Handling**: To ensured that the system does not halt in case of undesired situation or events exception conditions are taken care of providing the corresponding exception responses while developing the system.
* **Code Optimization:** Web-based Ticket Master Travel Reservation System project explains about concept of Ticket Master Travel Reservation System Project. Most of codes are reused to reduce repeated coding and the result set are reused where needed due to its inheritance concept, it is possible to reduce coding. In built function organization are traced to the various processes.

**Security Requirements:**

* **Website Security:** Ensure that your website is properly secured. Implement HTTPS (SSL/TLS) to encrypt data transmitted between users and the server.
* **Sensitive Data Protection:** Encrypt sensitive cardholder information (such as credit card details) to prevent unauthorized access.
* **Antivirus Software:** Install and maintain up-to-date antivirus software on all systems involved in the reservation process.
* **Access Control:** Implement role-based access control (RBAC) to restrict access to authorized personnel only.
* **Secure Payment Gateways:** Integrate with reliable and secure payment gateways to handle financial transactions. Ensure compliance with Payment Card Industry Data Security Standard (PCI DSS) requirements.
* **Data Encryption:** Encrypt sensitive data at rest (stored in databases) using strong encryption algorithms.
* **Audit Trails and Logs:** Maintain detailed logs of system activities, including user actions, login attempts, and transaction history. Conduct periodic security audits to identify vulnerabilities.

**4 System Design**

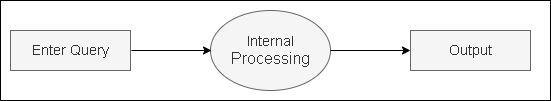
**4.1 System Design:**

****

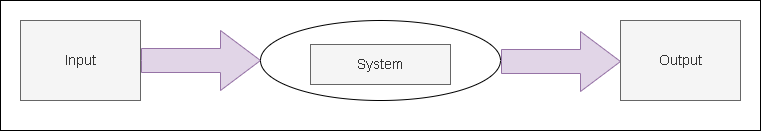
**4.2 System Model:**

**UML (Unified Modeling Language) diagrams:**

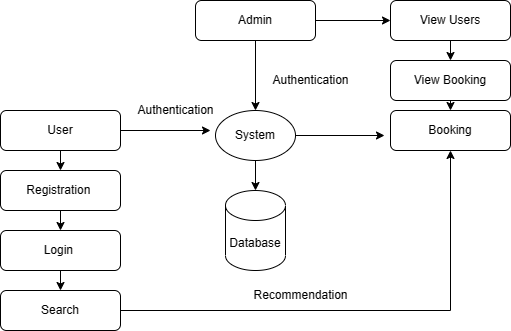
**DFD 0:**

****

**DFD 1:**

****

**DFD 2:**

****

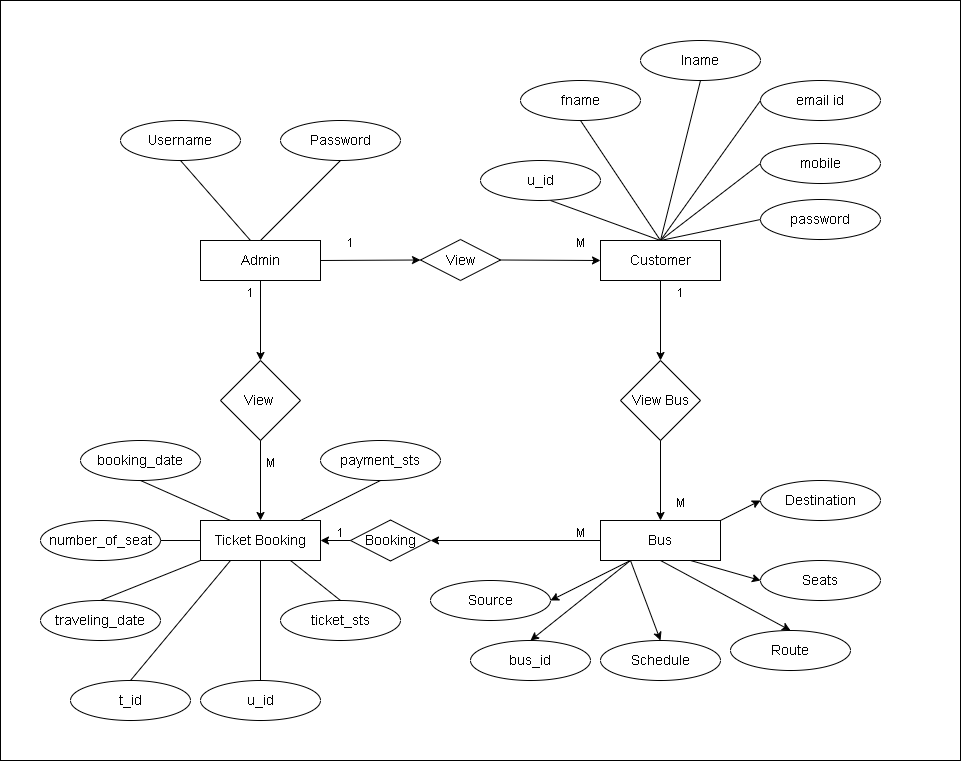
**Context Level Diagram:**

****

**4.3 Data Models:**

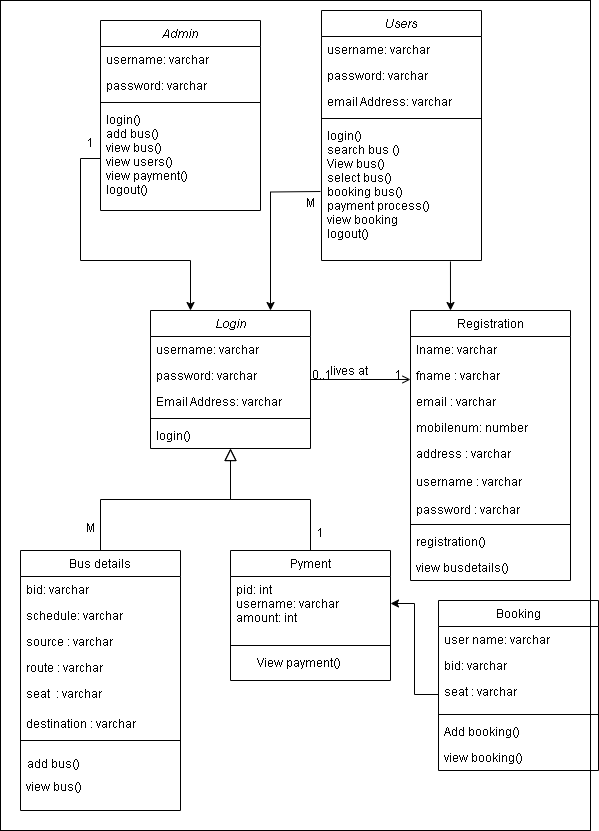
**Entity Relationship (E-R) Diagram:**

An ER diagram shows the relationship among entity sets. An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship among tables and their attributes, ER diagram shows the complete logical structure of a database.



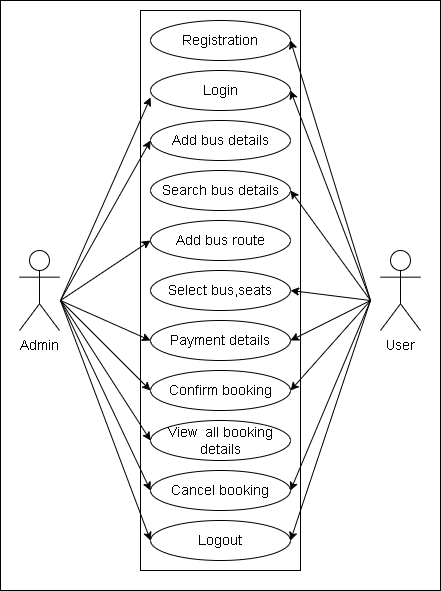
# **Class Diagram:**

Class diagrams are the blueprints of your system or subsystem. You can use class diagrams to model the objects that make up the system, to display the relationships between the objects, and to describe what those objects do and the services that they provide. Class diagrams are useful in many stages of system design.



# **Use-Case Diagram**

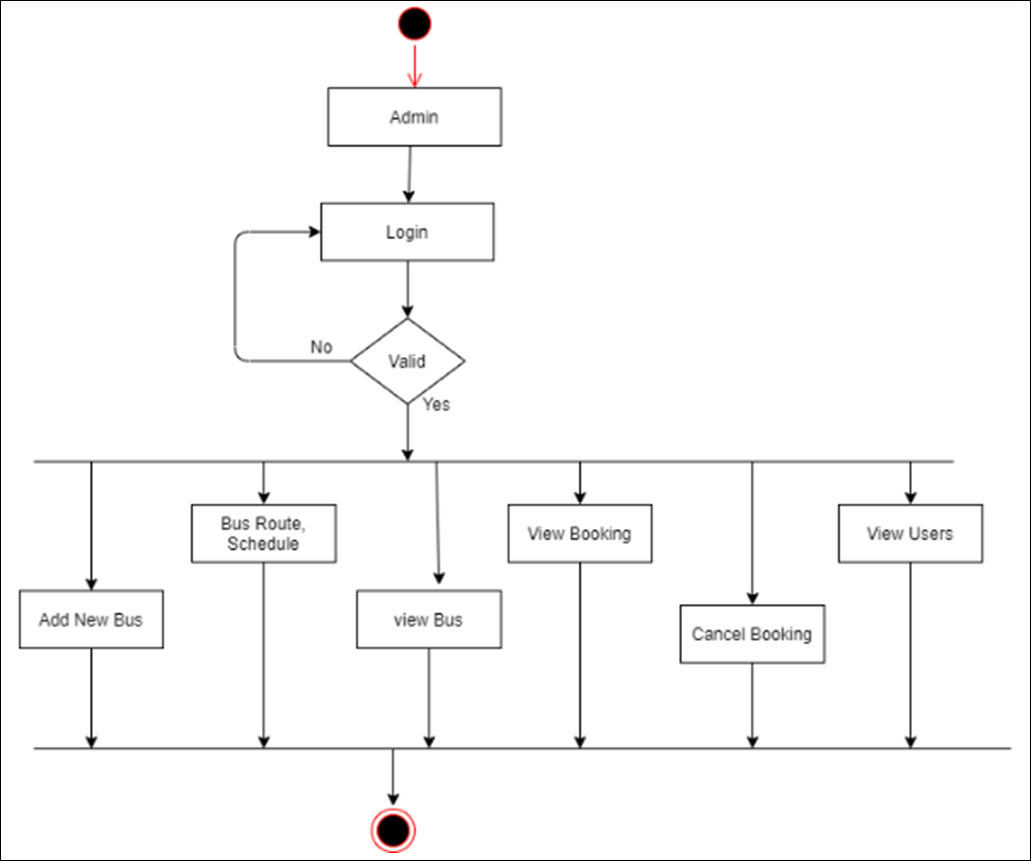
A use case diagram is a dynamic or behavior diagram in UML. Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform.

****

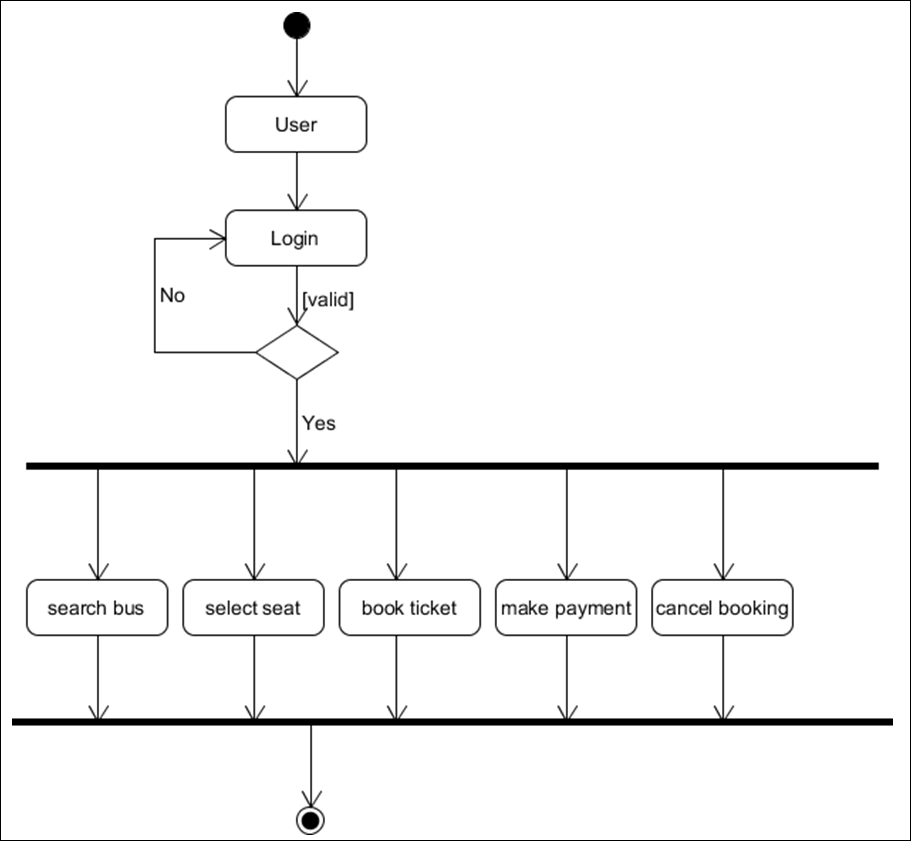
# **Activity diagram:**

An activity diagram is a behavioral diagram i.e. it depicts the behavior of a system. An activity diagram portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

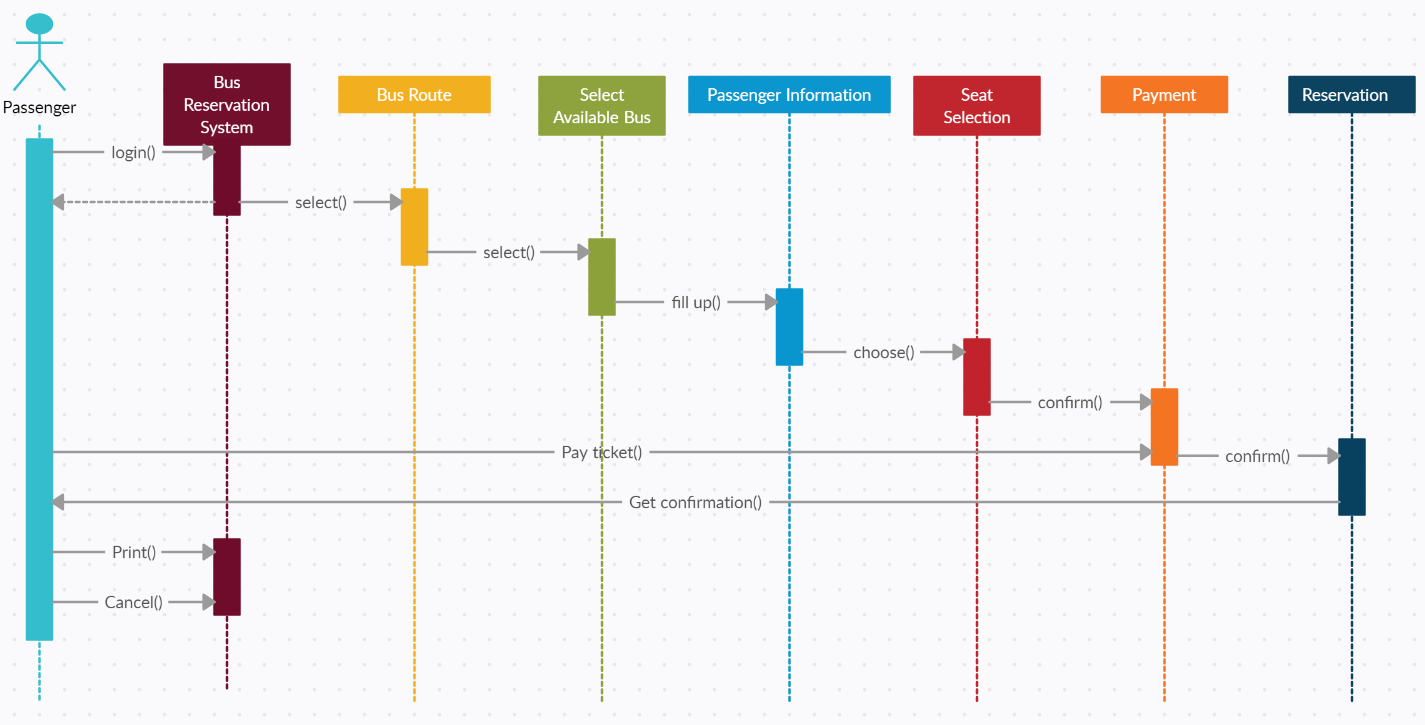
1. **Admin Activity Diagram:**

****

# **User activity Diagram:**

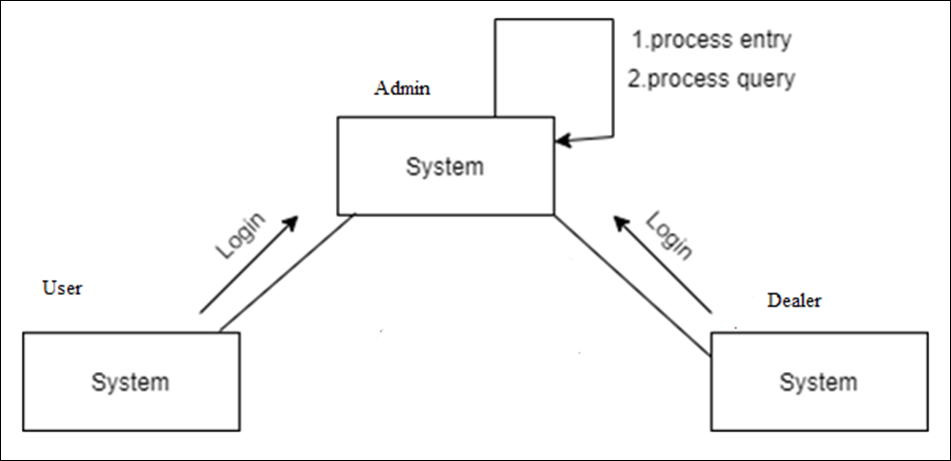
****

**Sequence Diagram:**

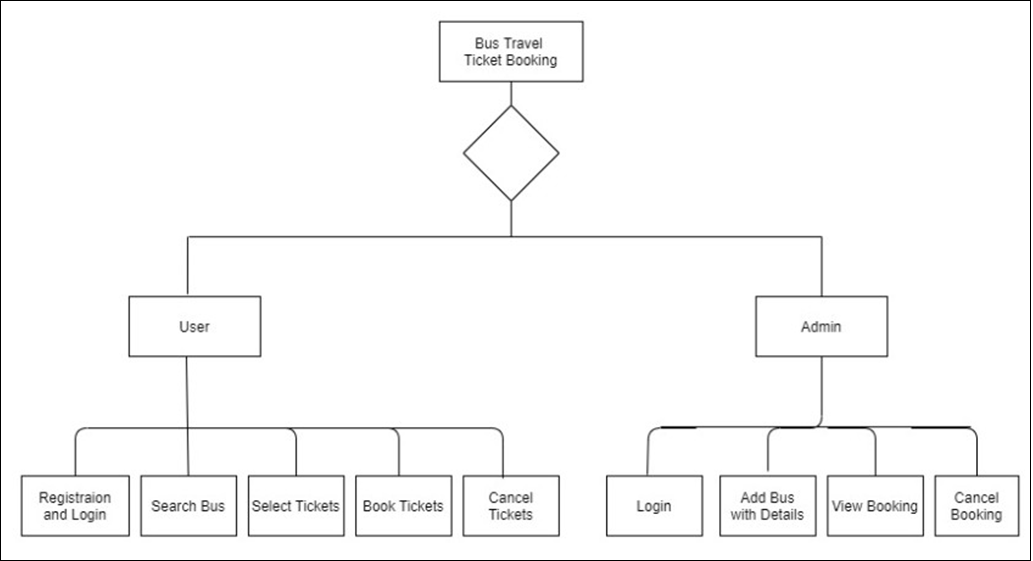


**Component Diagram:**

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required functions is covered by planned development.

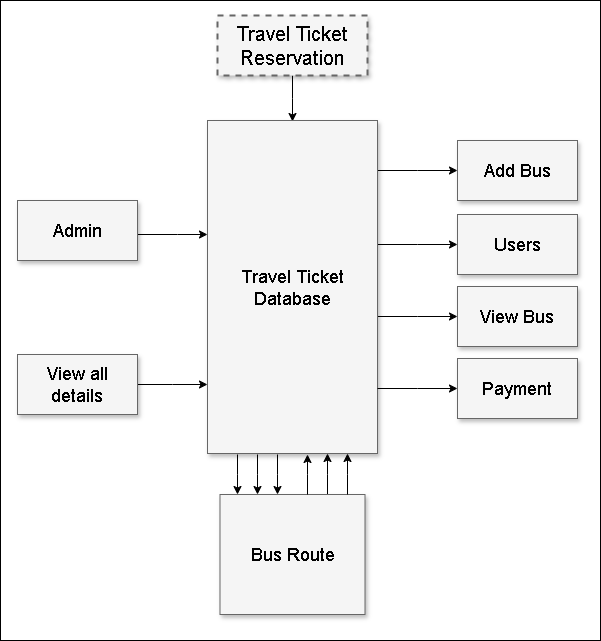


# **Module Hierarchy Diagram:**

****

**Collaboration Diagram:**

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

****

**4.4 User Interfaces:**

**Admin:**

* Admin login. like, username, password.
* Add bus route, bus name details.
* View, update, delete bus details.
* View all customer details.
* View all customer payment details. Like, advanced & pending payment details.
* View feedback details.
* Logout.

**Customer:**

* Customer registration details.
* Customer login. Like, username, password.
* Search bus.
* View search bus, route result details.
* Select bus, seat bus ticket details.
* Bus booking, cancel booking details.
* Make payment details.
* View payment summary details. Like, online, offline.
* View ticket receipt.
* Logout.

**5. Implementation Details**

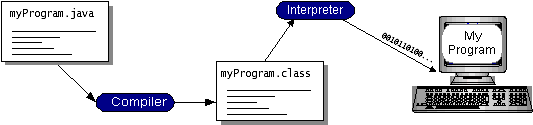
## **5.1 Implementation Technologies:**

## **JAVA:**

**Java as a Programming language:**

### The Java programming language is a high-level language that can be characterized by – Simple, Architecture Neutral, Object Oriented, Portable, Distributed, High Performance, Interpreted, Multithreaded, Robust, Dynamic and Secure.

### With most programming languages, you either compile or interpret a program so that you can run it on your computer. The Java programming language is unusual in that a program is both compiled and interpreted. With the compiler, first you translate a program into an intermediate language called Java byte codes —the platform-independent codes interpreted by the interpreter on the Java platform. The interpreter parses and runs each Java byte code instruction on the computer. Compilation happens just once; interpretation occurs each time the program is executed. The following figure illustrates how this works.



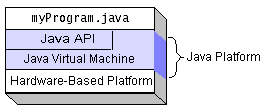
You can think of Java byte codes as the machine code instructions for the Java Virtual Machine (Java VM). Every Java interpreter, whether it’s a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make “write once, run anywhere” possible.

**Java as a Platform**

A platform is the hardware or software environment in which a program runs. Some of the most popular platforms like Windows 2000, Linux, Solaris, and MacOS. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it’s a software-only platform that runs on top of other hardware-based platforms.

The Java platform has two components: The Java Virtual Machine (Java VM), The Java Application Programming Interface (Java API).

Java VM is the base for the Java platform and is ported onto various hardware-based platforms. The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and interfaces; these libraries are known as packages. The following figure depicts a program that’s running on the Java platform. As the figure shows, the Java API and the virtual machine insulate the program from the hardware.



## **What Can Java Technology Do?**

The most common types of programs written in the Java programming language are applets and applications.

An application is a standalone program that runs directly on the Java platform. A special kind of application known as a server serves and supports clients on a network. Examples of servers are Web servers, proxy servers, mail servers, and print servers. Another specialized program is a servlet. Every full implementation of the Java platform gives you the following features:

* **The essentials**: Objects, strings, threads, numbers, input and output, data structures, system properties, date, and time, and so on.
* **Applets**: The set of conventions used by applets.
* **Networking**: URLs, TCP (Transmission Control Protocol), UDP (User Data gram Protocol) sockets, and IP (Internet Protocol) addresses.
* **Internationalization**: Help for writing programs that can be localized for users worldwide. Programs can automatically adapt to specific locales and be displayed in the appropriate language.
* **Security**: Both low level and high level, including electronic signatures, public and private key management, access control, and certificates.
* **Software components**: Known as JavaBeansTM, can plug into existing component architectures.
* **Object serialization**: Allows lightweight persistence and communication via Remote Method Invocation (RMI).
* **Java Database Connectivity (JDBCTM)**: Provides uniform access to a wide range of relational databases.

The Java platform also has APIs for 2D and 3D graphics, accessibility, servers, collaboration, telephony, speech, animation, and more.

**What is a Java Web Application?**

A Java web application generates interactive web pages containing various types of markup language (HTML, XML, and so on) and dynamic content. It is typically comprised of web components such as JavaServer Pages (JSP), servlets and JavaBeans to modify and temporarily store data, interact with databases, and web services, and render content in response to client requests.

Because many of the tasks involved in web application development can be repetitive or require a surplus of boilerplate code, web frameworks can be applied to alleviate the overhead associated with common activities.

**JavaScript and Ajax Development**

JavaScript is an object-oriented scripting language primarily used in client-side interfaces for web applications. Ajax (Asynchronous JavaScript and XML) is a Web 2.0 technique that allows changes to occur in a web page without the need to perform a page refresh. JavaScript toolkits can be leveraged to implement Ajax-enabled components and functionality in web pages.

**Web Server and Client:**

Web Server is a software that can process the client request and send the response back to the client. For example, Apache is one of the most widely used web server.

A web client is a software that helps in communicating with the server. Some of the most widely used web clients are Firefox, Google Chrome, Safari etc. When we request something from server (through URL), web client takes care of creating a request and sending it to server and then parsing the server response and present it to the user.

**HTML and HTTP:**

Web Server and Web Client are two separate software, so there should be some common language for communication. HTML is the common language between server and client and stands for Hypertext Markup Language.

Web server and client needs a common communication protocol, HTTP (Hypertext Transfer Protocol) is the communication protocol between server and client. HTTP runs on top of TCP/IP communication protocol.

Some of the important parts of HTTP Request are:

* HTTP Method – action to be performed, usually GET, POST, PUT etc.
* URL – Page to access.
* Form Parameters – like arguments in a java method, for example user, password details from login page.

Sample HTTP Request:

|  |  |
| --- | --- |
| 1  2  3 | GET /FirstServletProject/jsps/hello.jsp HTTP/1.1  Host: localhost:8080  Cache-Control: no-cache |

Some of the important parts of HTTP responses are:

* Status Code – an integer to indicate whether the request was success or not. Some of the well-known status codes are 200 for success, 404 for Not Found and 403 for Access Forbidden.
* Content Type – text, html, image, pdf etc. Also known as MIME type
* Content – actual data that is rendered by client and shown to user.

**MIME Type or Content Type**:

If you see above sample HTTP response header, it contains tag “Content-Type”. It’s also called MIME type and server sends it to client to let them know the kind of data it’s sending. It helps client in rendering the data for user. Some of the mostly used mime types are text/html, text/xml, application/xml etc.

### **Understanding URL:**

URL is acronym of Universal Resource Locator and it’s used to locate the server and resource. Every resource on the web has its own unique address. Let’s see parts of URL with an example.

**http://localhost:8080/FirstServletProject/jsps/hello.jsp**

**http://** – This is the first part of URL and provides the communication protocol to be used in server-client communication.

**localhost** – The unique address of the server, most of the times it’s the hostname of the server that maps to unique IP address. Sometimes multiple hostnames point to same IP addresses and web server virtual host takes care of sending request to the server instance.

**8080** – This is the port on which server is listening, it’s optional and if we don’t provide it in URL then request goes to the default port of the protocol. Port numbers 0 to 1023 are reserved ports for well-known services, for example 80 for HTTP, 443 for HTTPS, 21 for FTP etc.

**FirstServletProject/jsps/hello.jsp** – Resource requested from server. It can be static html, pdf, JSP, servlets, PHP etc.

### **Why we need Servlet and JSPs?**

Web servers are good for static contents HTML pages, but they don’t know how to generate dynamic content or how to save data into databases, so we need another tool that we can use to generate dynamic content. There are several programming languages for dynamic content like PHP, Python, Ruby on Rails, Java Servlets and JSPs.

Java Servlet and JSPs are server-side technologies to extend the capability of web servers by providing support for dynamic response and data persistence.

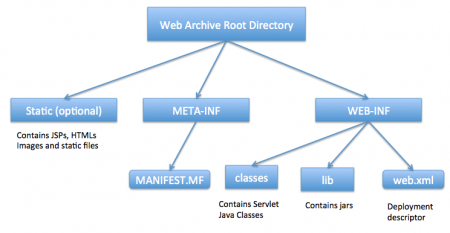
### **Web Container: Tomcat**

Here, **Tomcat** is a web container, when a request is made from Client to web server, it passes the request to web container and it’s web container job to find the correct resource to handle the request (servlet or JSP) and then use the response from the resource to generate the response and provide it to web server. Then web server sends the response back to the client.

When web container gets the request and if it’s for servlet then container creates two Objects HTTPServletRequest and HTTPServletResponse. Then it finds the correct servlet based on the URL and creates a thread for the request. Then it invokes the servlet service() method and based on the HTTP method service() method invokes doGet() or doPost() methods. Servlet methods generate the dynamic page and write it to response. Once servlet thread is complete, container converts the response to HTTP response and send it back to client.

### **Web Application Directory Structure:**

Java Web Applications are packaged as Web Archive (WAR), and it has a defined structure. You can export above dynamic web project as WAR file and unzip it to check the hierarchy. It will be something like below image.

[](http://www.journaldev.com/wp-content/uploads/2013/08/WAR-directory-structure.png)

### **Deployment Descriptor:**

**web.xml** file is the deployment descriptor of the web application and contains mapping for servlets (prior to 3.0), welcome pages, security configurations, session timeout settings etc.

### **MySQL:**

MySQL, the most popular Open-Source SQL database management system, is developed, distributed, and supported by Oracle Corporation. The MySQL Web site (http://www.mysql.com/) provides the latest information about MySQL software.

* **MySQL is a database management system:**

A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or the vast amounts of information in a corporate network. To add, access, and process data stored in a computer database, you need a database management system such as MySQL Server.

* **MySQL databases are relational:**

A relational database stores data in separate tables rather than putting all the data in one big storeroom. The database structures are organized into physical files optimized for speed. The logical model, with objects such as databases, tables, views, rows, and columns, offers a flexible programming environment. You set up rules governing the relationships between different data fields, such as one-to-one, one-to-many, unique, required, or optional, and “pointers” between different tables.

The SQL part of “MySQL” stands for “Structured Query Language”. SQL is the most common standardized language used to access databases. Depending on your programming environment, you might enter SQL directly (for example, to generate reports), embed SQL statements into code written in another language, or use a language-specific API that hides the SQL syntax.

* **MySQL software is Open Source:**

Open source means that it is possible for anyone to use and modify the software, anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License), http://www.fsf.org/licenses/, to define what you may and may not do with the software in different situations. You can buy a commercially licensed version from us. See the MySQL Licensing Overview for more information (http://www.mysql.com/company/legal/licensing/).

* **The MySQL Database Server is Fast, Reliable, Scalable, and Easy to Use:**

MySQL Server can run comfortably on a desktop or laptop, alongside your other applications, web servers, and so on, requiring little or no attention.

MySQL Server was originally developed to handle large databases much faster than existing solutions and has been successfully used in highly demanding production environments for several years. Its connectivity, speed, and security make MySQL Server highly suited for accessing databases on the Internet.

* **MySQL Server works in Client/Server or Embedded systems:**

The MySQL Database Software is a Client/Server system that consists of a multi-threaded SQL server that supports different backends, several different client programs and libraries, administrative tools, and a wide range of Application Programming Interfaces (APIs).

**6. Testing**

**6.1 Test Plan:**

A test plan is a crucial document in software testing that outlines the overall strategy and approach for testing a software application. It provides a roadmap for the testing team, ensuring that all components are thoroughly tested before the release.

**Unit testing:**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code own should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications.

**Integration testing:**

Integration tests are designed to test integrated software components to determine if they run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

**System Test:**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and owns, emphasizing pre-driven process links and integration points.

**White Box Testing/ Functional Validation Test:**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure, and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

**Black Box Testing/ Data Validation Test:**

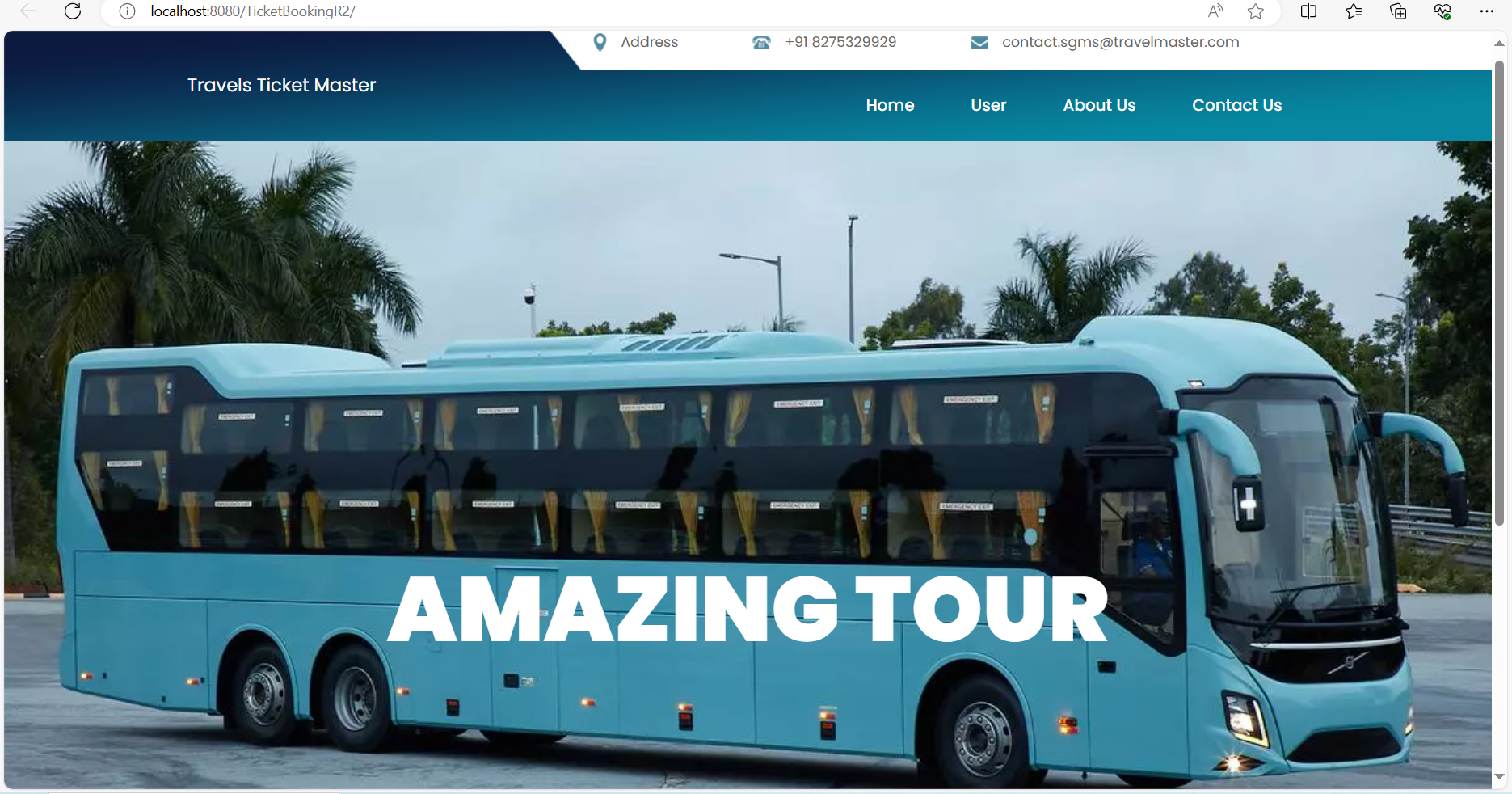
Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box you cannot see into it. The test provides inputs and responds to outputs without considering how the software works.

**Test Cases & Results:**

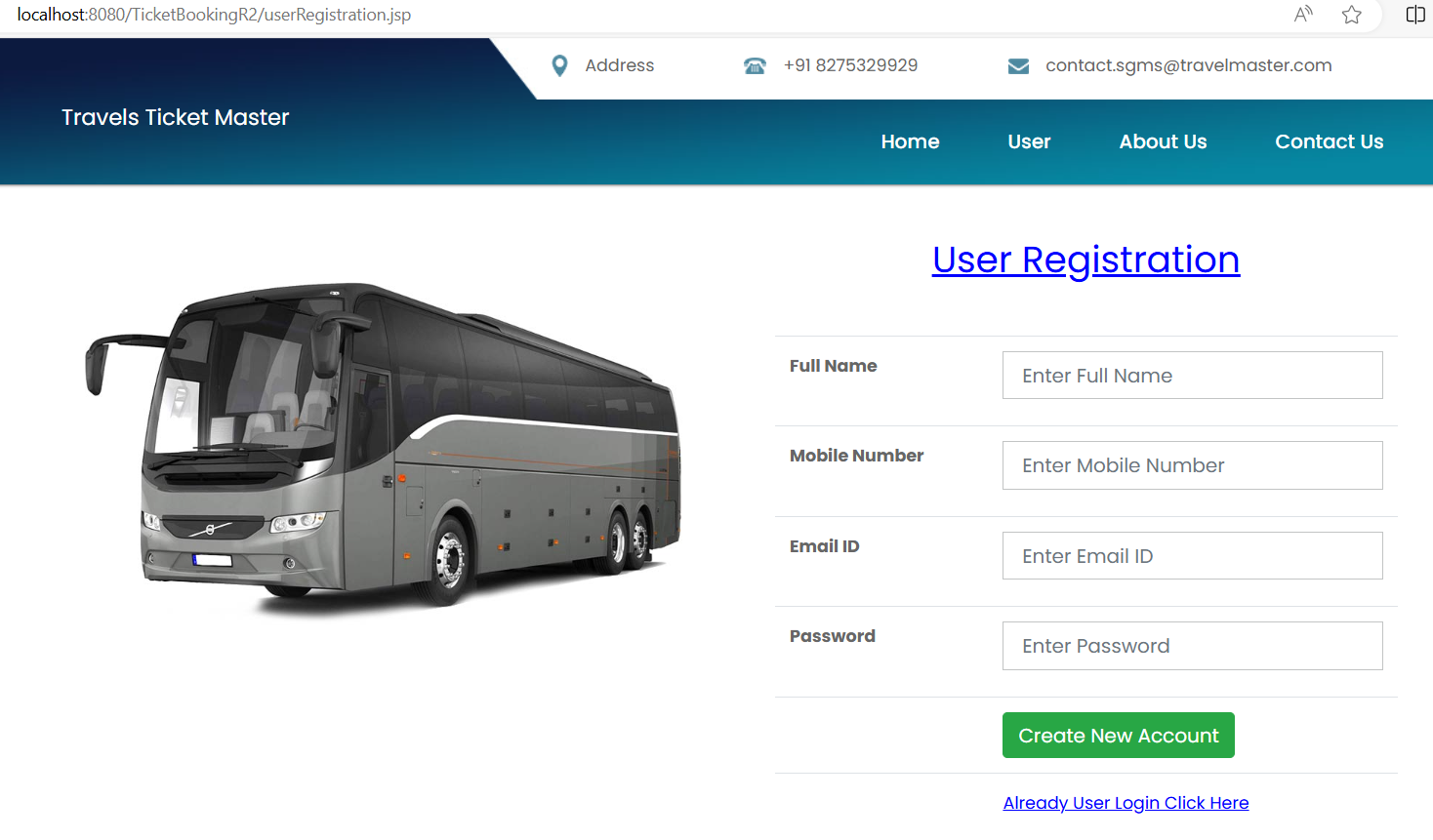
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test case\_ID** | **Description** | **Test case I/P** | **Actual Result** | **Expected Result** | **Test case criteria(pass/fail)** |
| 01 | Home page | Enter application URL | Open Welcome screen | Open welcome screen | Pass |
| 02 | Admin Login validation | 1.enter application URL  2.enter valid username  3.enter valid password  4.click on login button | Successfully login | Successfully login | Pass |
| 03 | Admin Login validation | 1.enter application URL  2.enter valid username  3.enter valid password  4.click on login button | Popup message shows an error “invalid credentials | Popup message shows an error “invalid credentials | Pass |
| 04 | User | 1.enter valid name  2.enter valid username  3.enter valid mobile no.  4. enter valid email id  5.enter valid address  6.enter valid password  7.enter valid bus route  8.click on register button | Successfully registered | Successfully registered | Pass |
| 05 | Travels Registration | 1.enter bus name  2.enter valid bus route  3.enter valid bus number | Successfully registered | Successfully registered | Pass |
| 06 | User Logout | 1.enter the application URL  2.click on logout | Logout successfully | Logout successfully | Pass |

1. **Output**

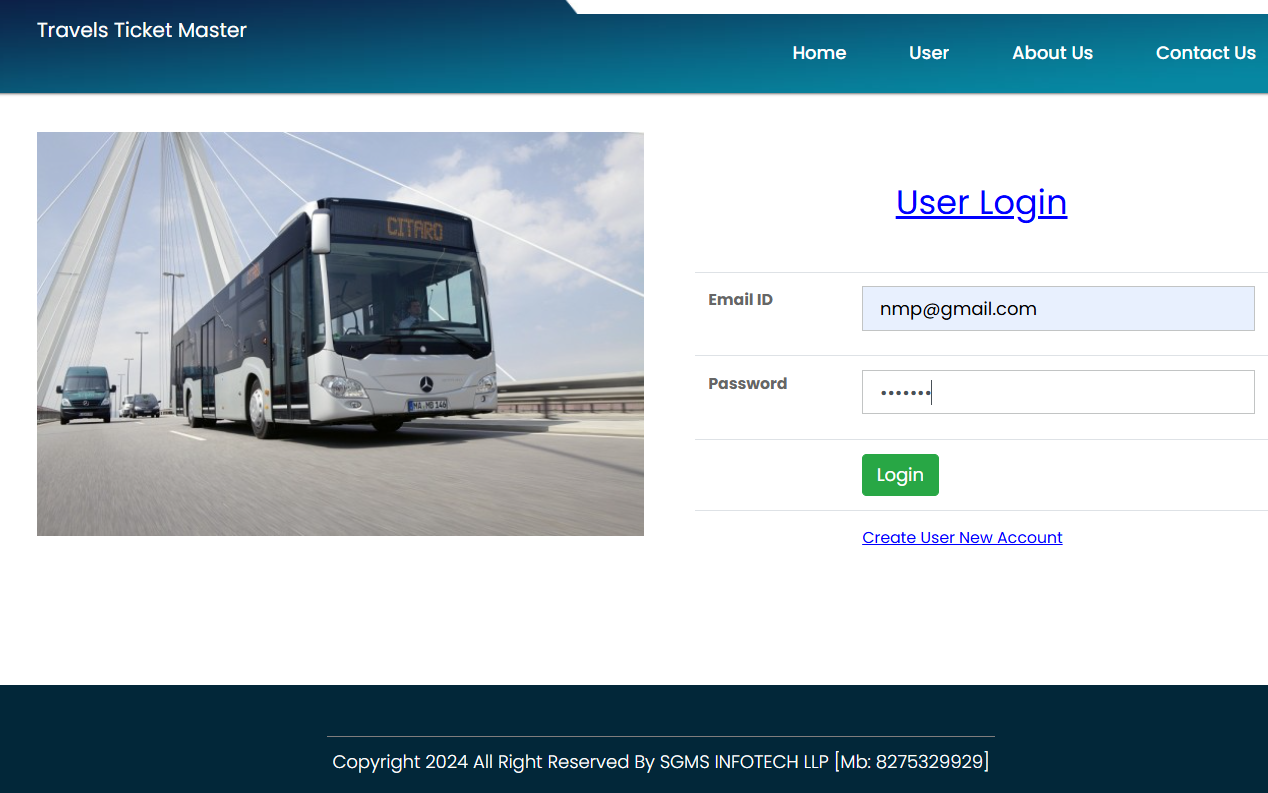
**User/Customer Homepage:**



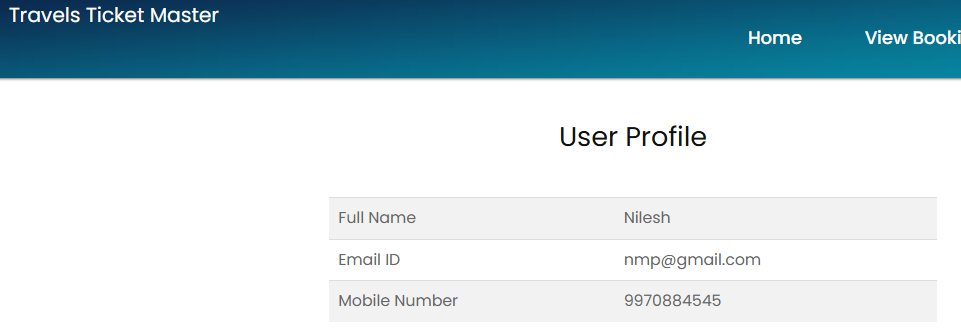
**User Registration:**



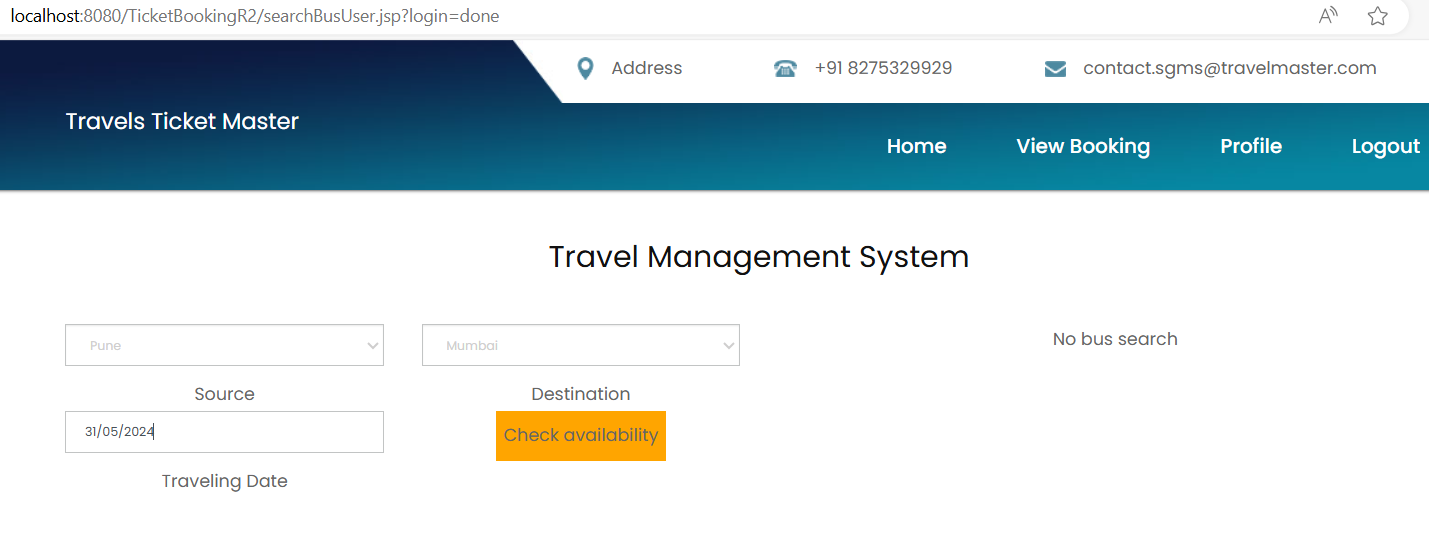
**Existing User login:**



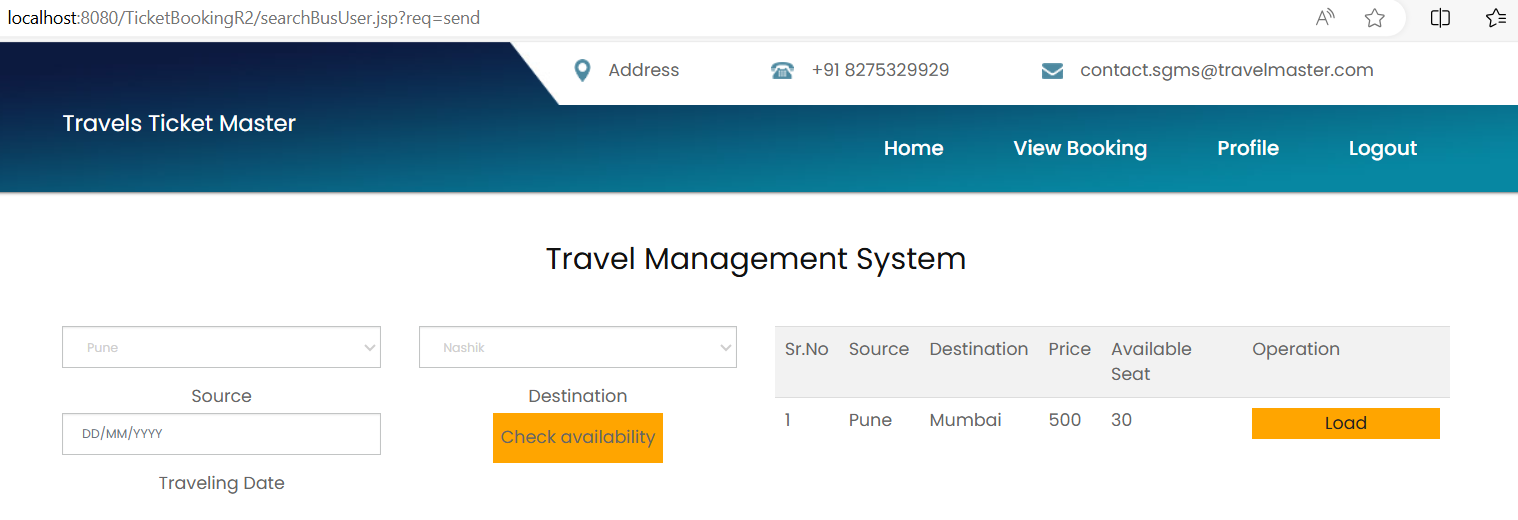
**User Profile page:**



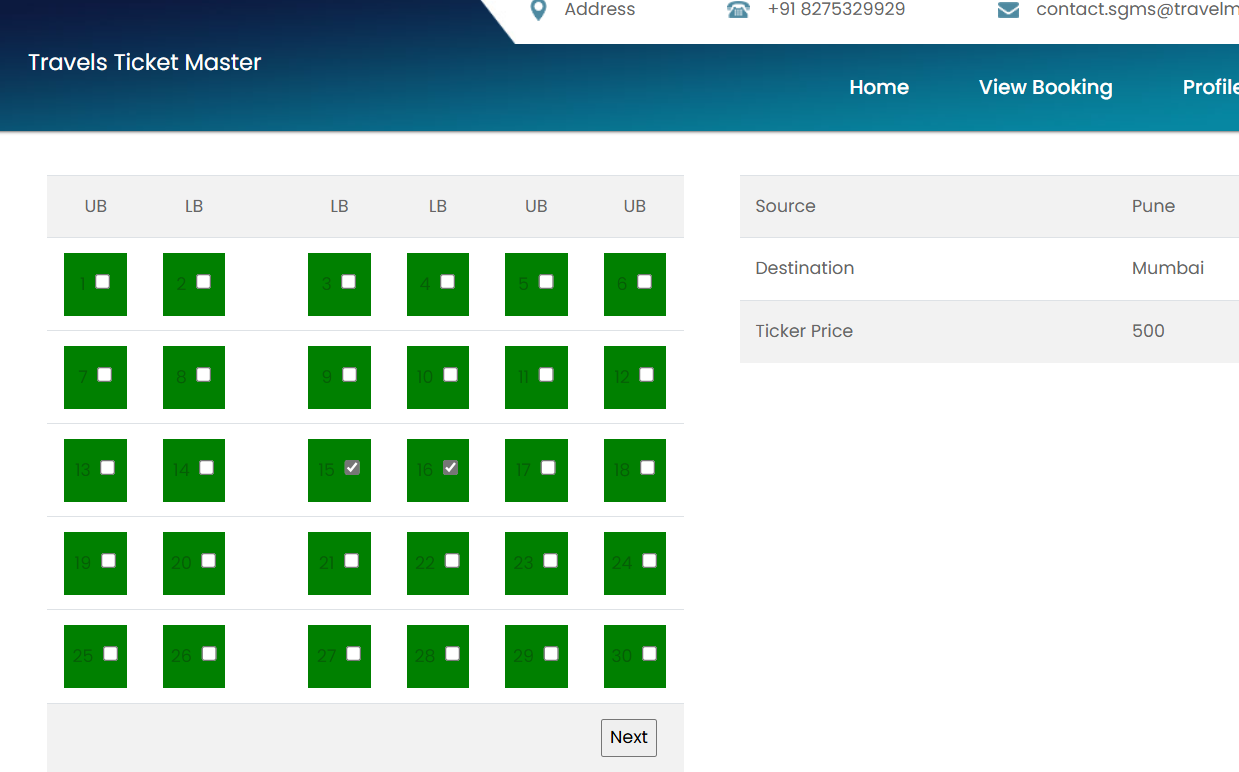
**Ticket Booking homepage:**



**Route Selection page:**



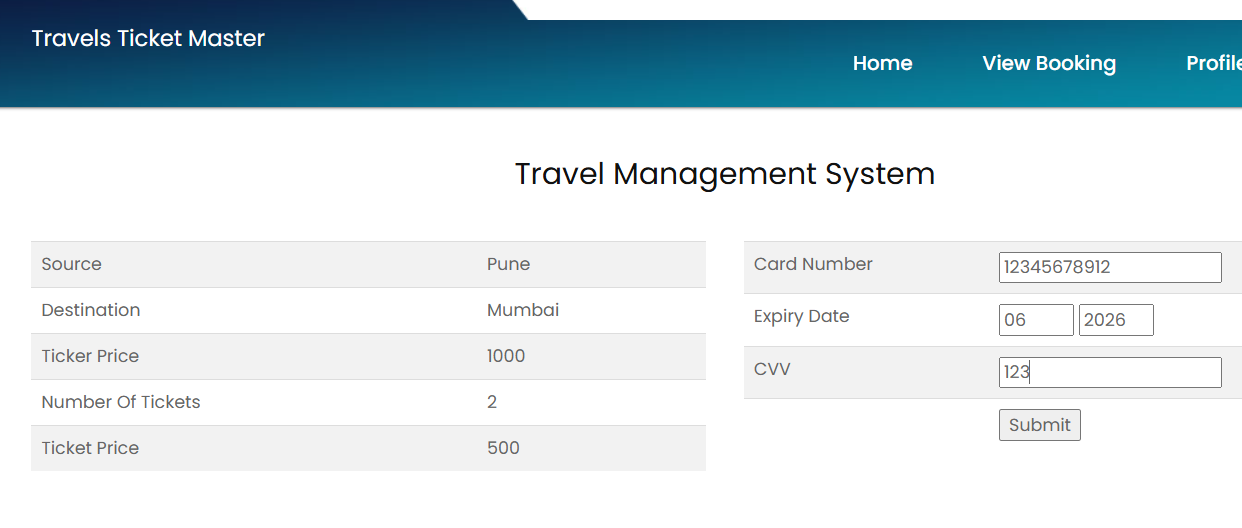
**Seat Selection page:**



**Pricing page:**



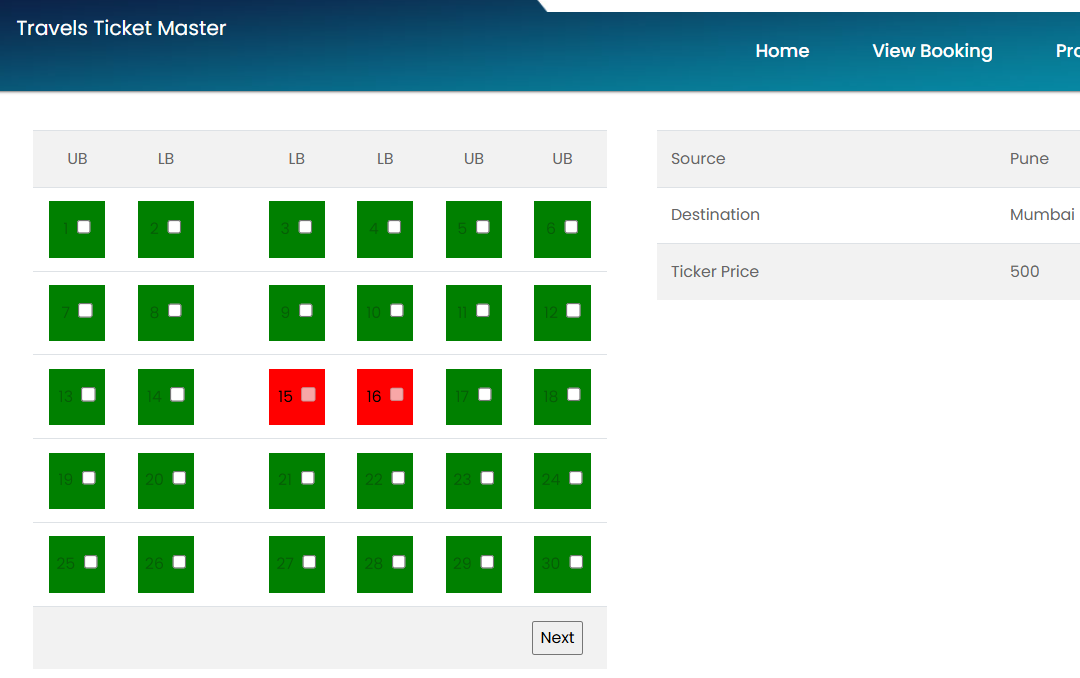
**Payment Page:**



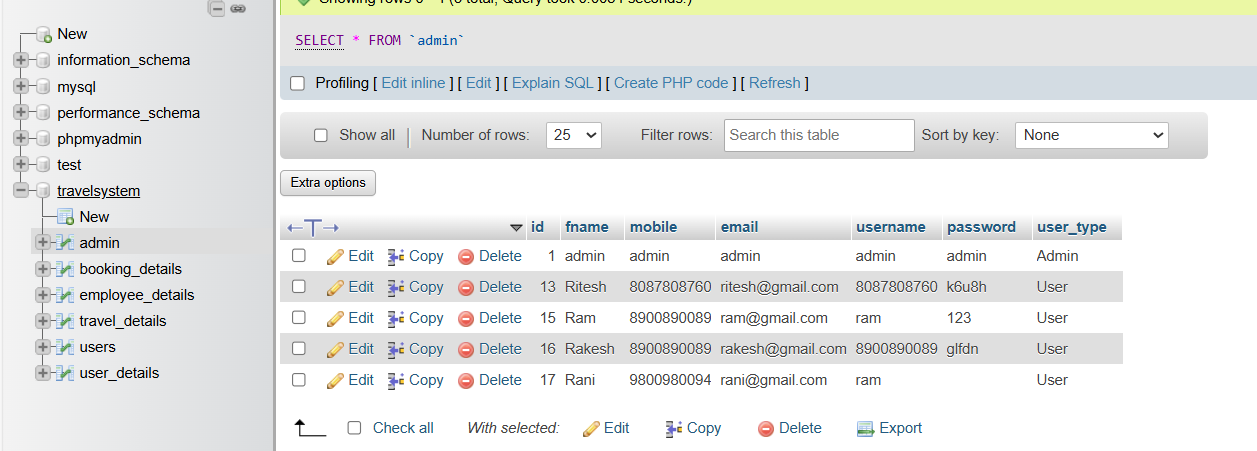
**View User Bookings:**

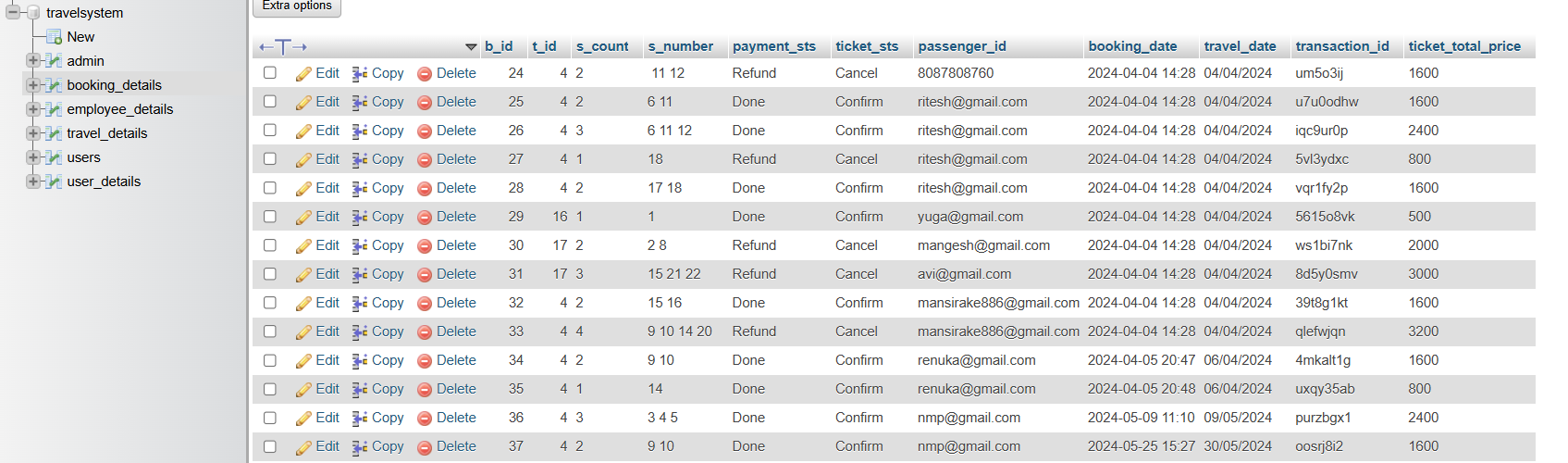


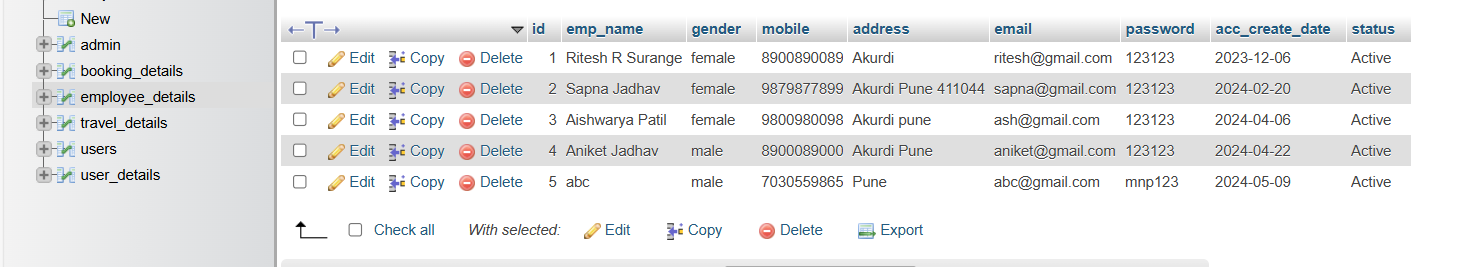
**Booked seat layout page:**

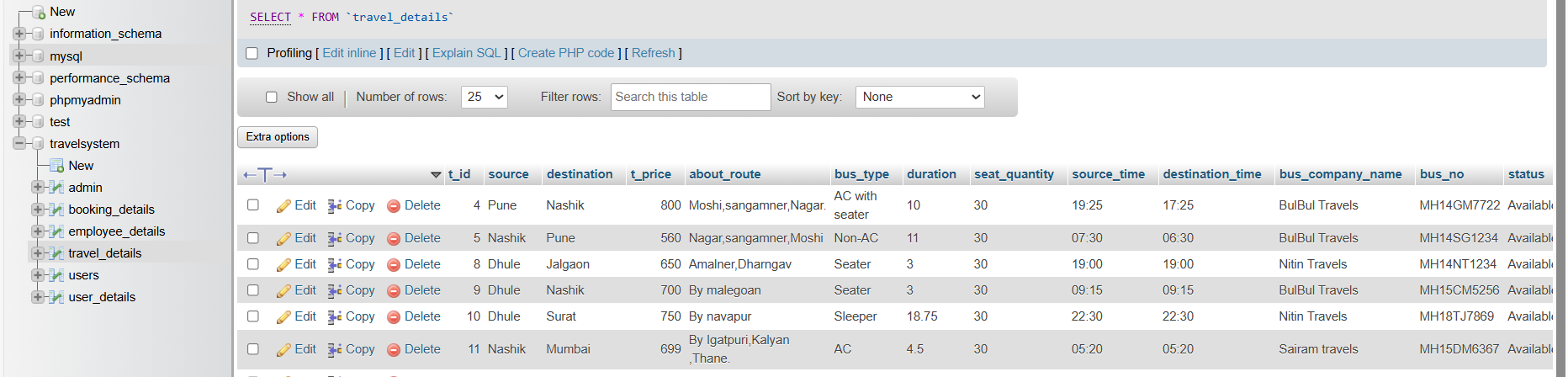


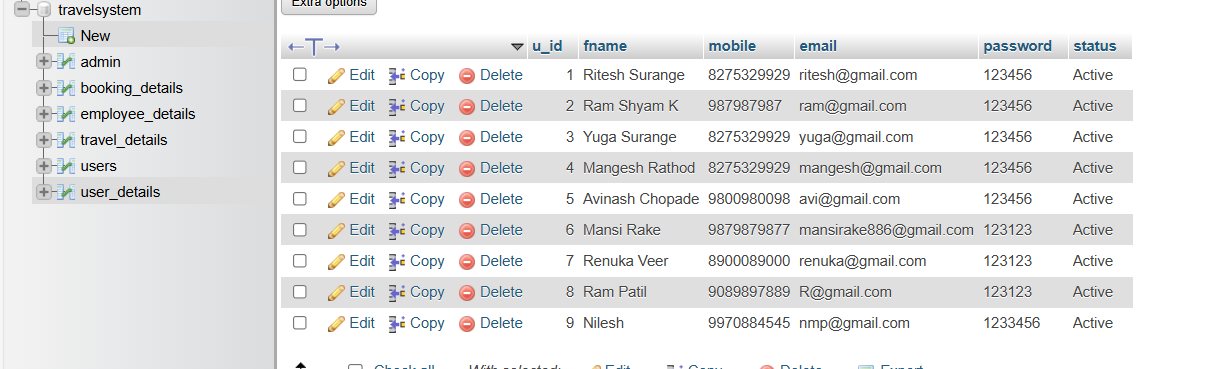
**MySQL Database showing Admin users, Booking details, Employee details, Travel details, User details Databases:**



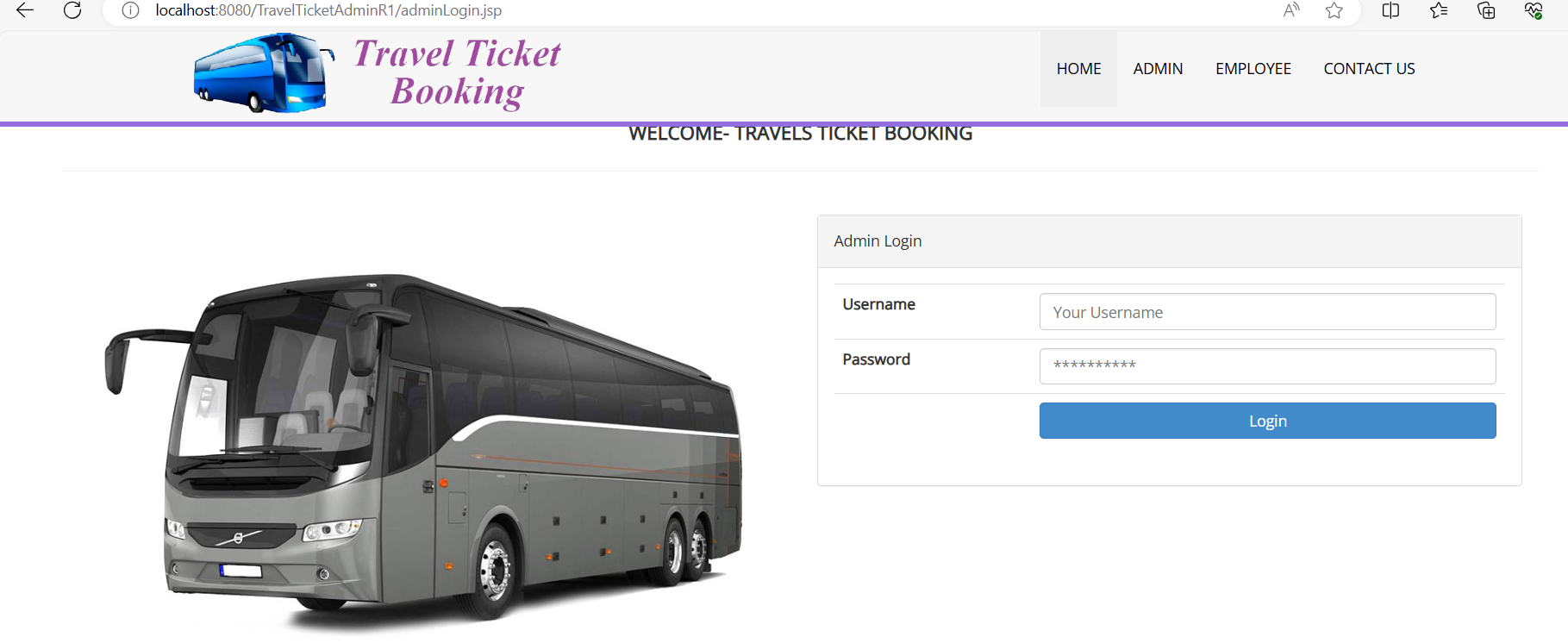




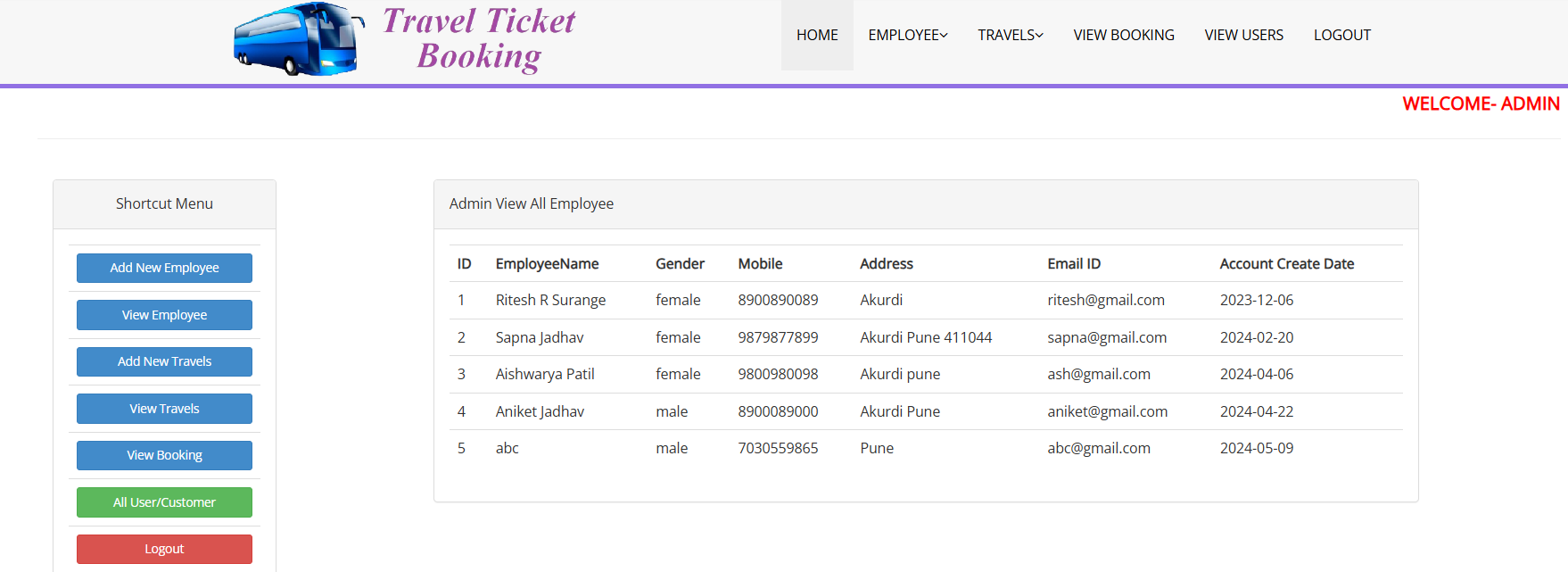




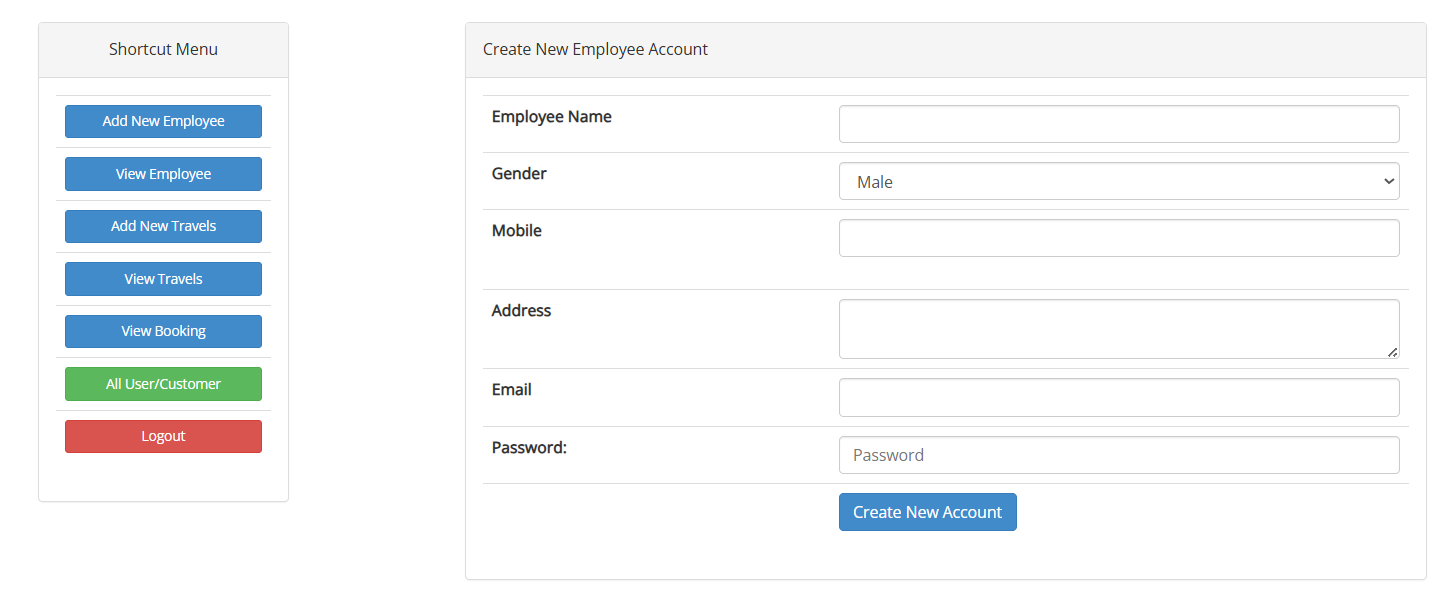
**Admin Login page:**



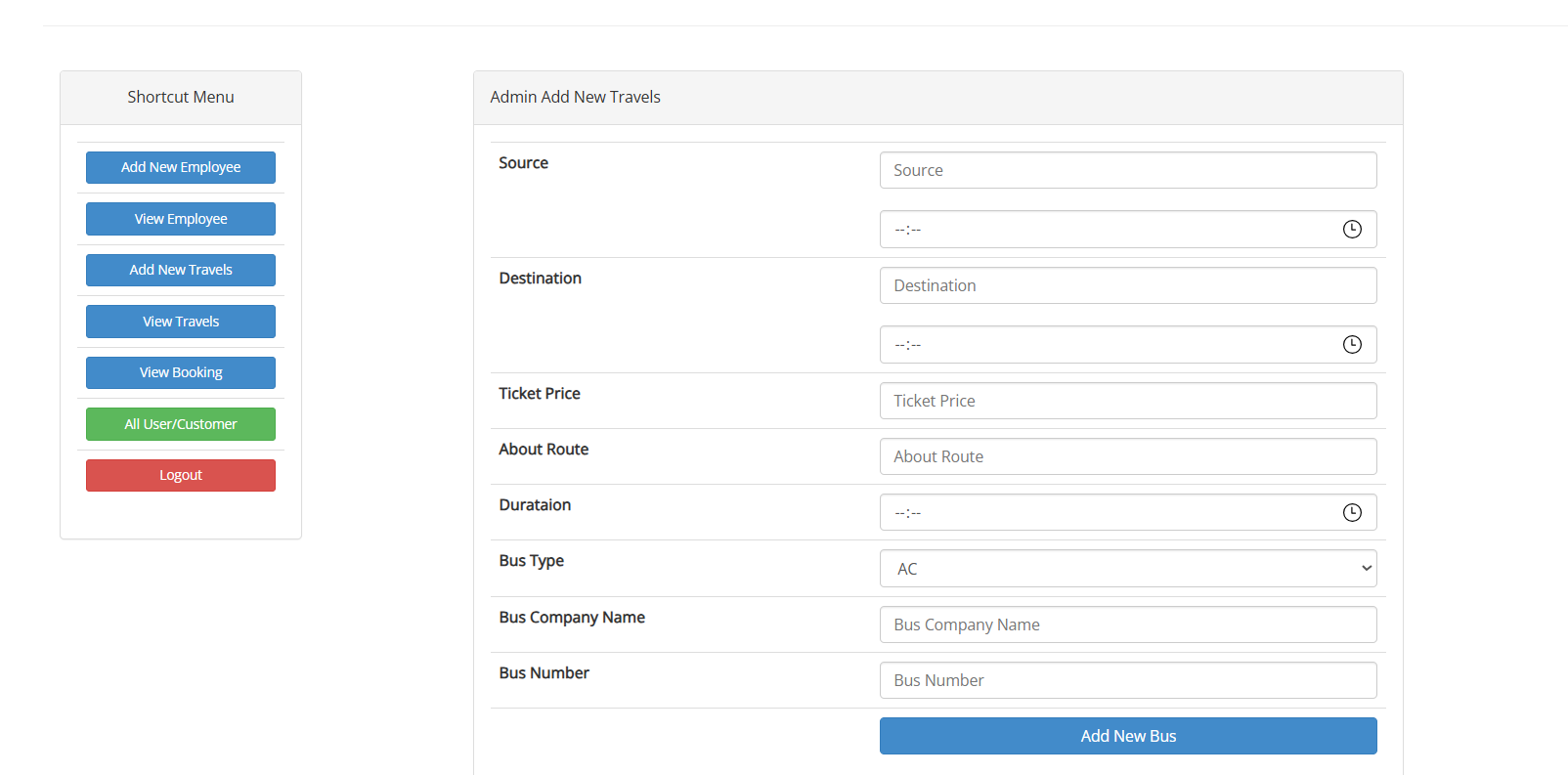
**Admin homepage:**



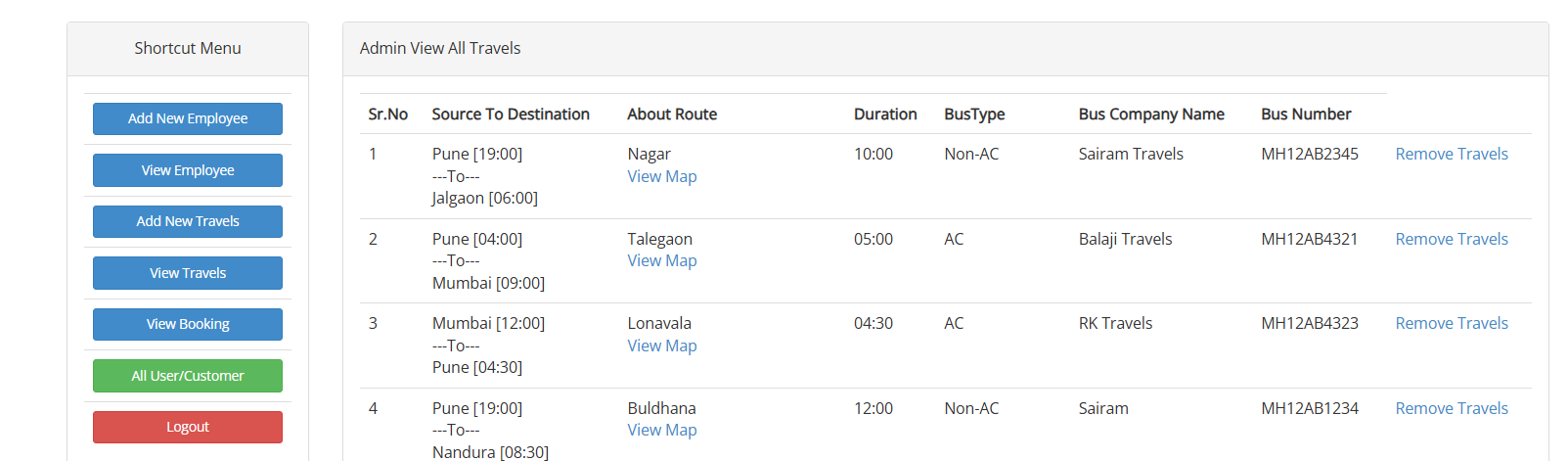
**Creating employee account page:**



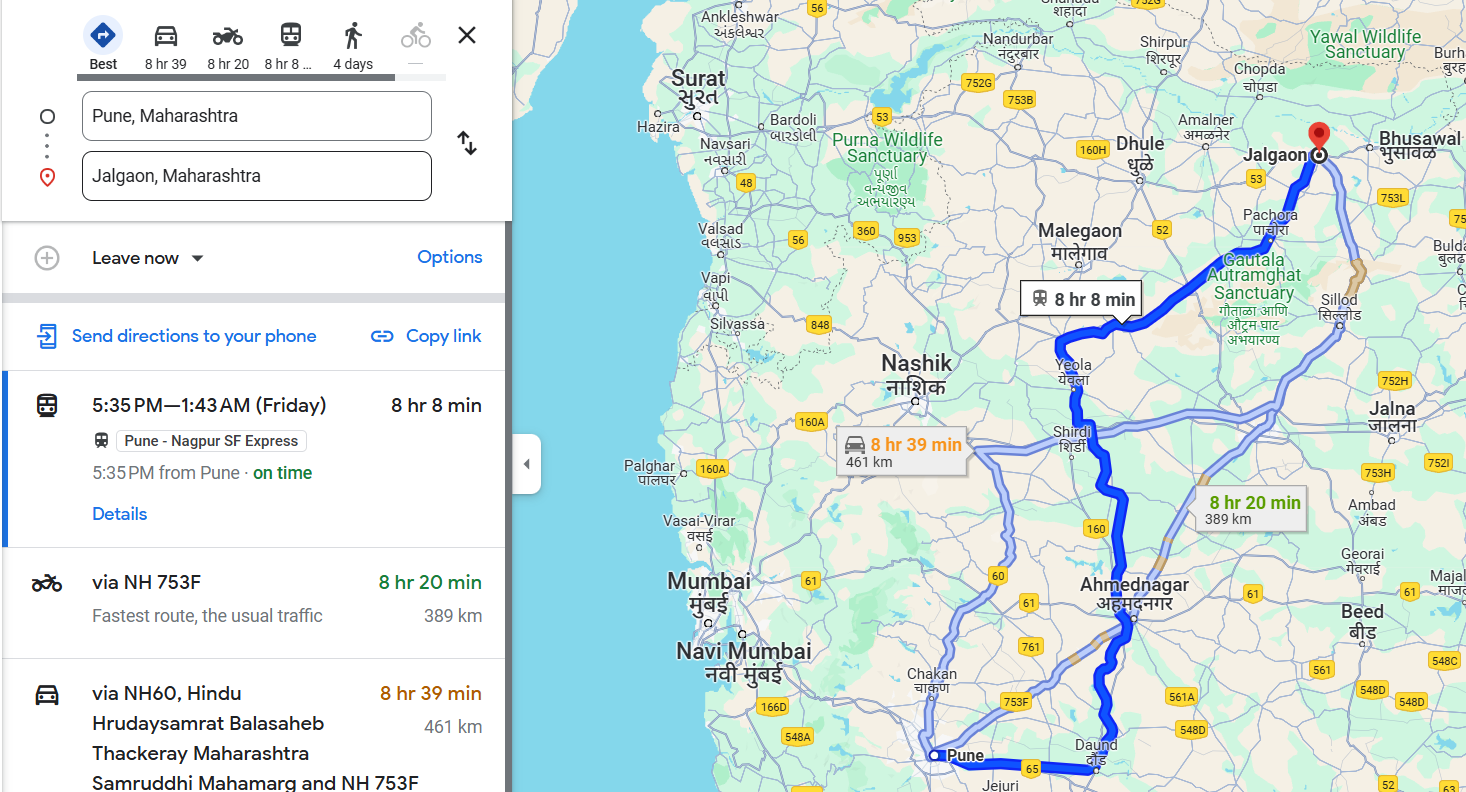
**Admin add new Travels page:**



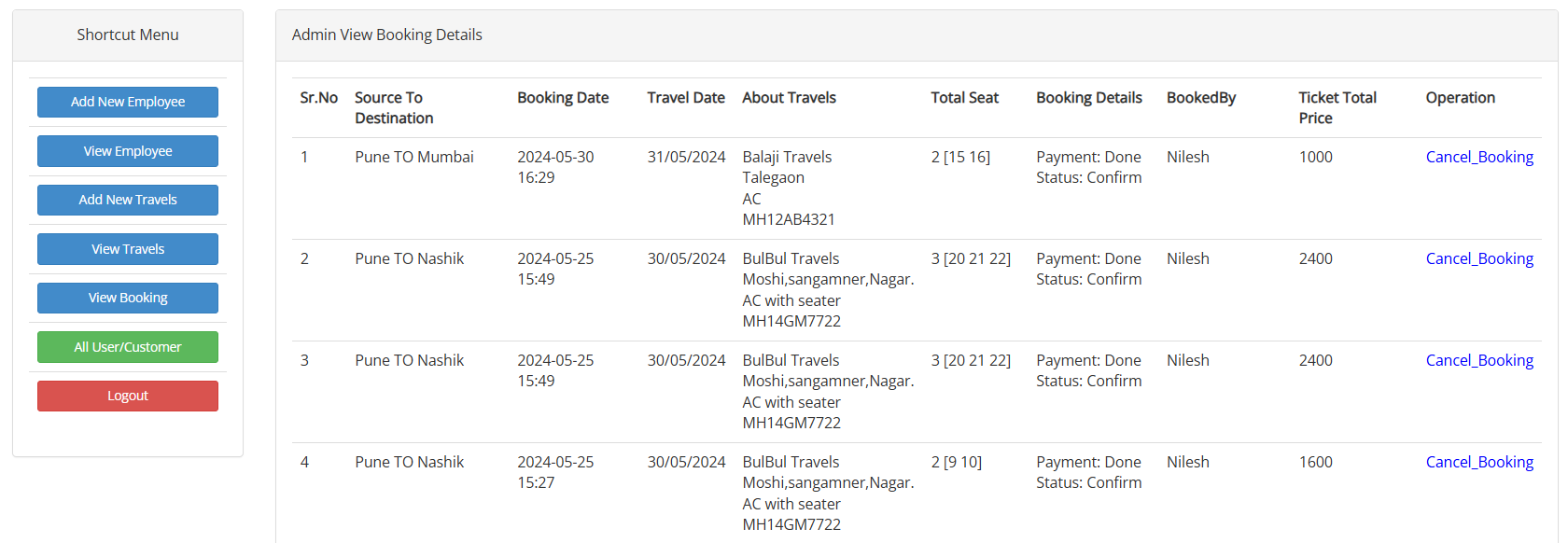
**Admin view all Travels:**



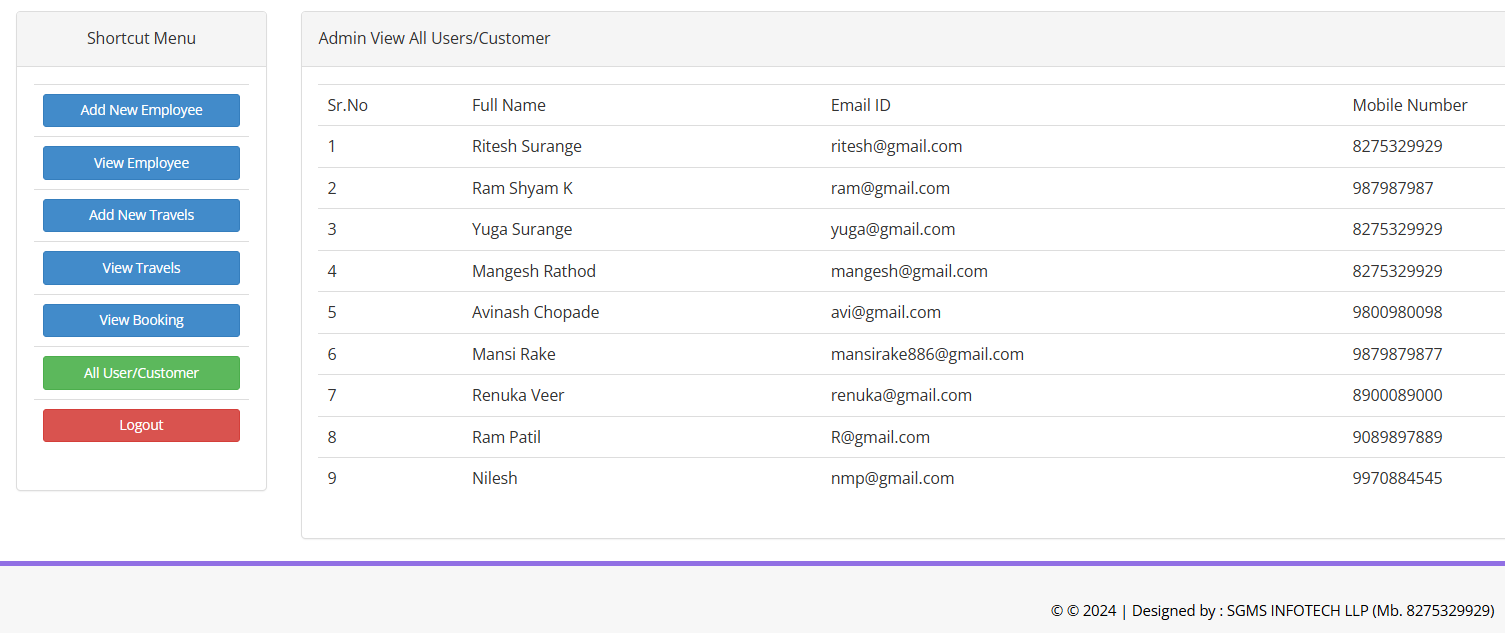
**Admin view map routes:**



**Admin view booking details:**



**Admin view all Users/Customers page:**



1. **User Manual**

Developing a comprehensive user manual and operating manual for online bus ticket reservation system is essential to ensure smooth usage and efficient management.

* 1. **User Manual**

1. **Introduction –** Briefly describe the purpose of the system and its benefits. Explain how users can access the system (web or mobile app).
2. **Registration and Login –** Provide step-by-step instructions on how users can create an account. Explain the login process, including social login options.
3. **Booking a Ticket –** Describe how users can search for available buses based on their preferred route, date, and time. Explain the seat selection process. Include screenshots or diagrams to illustrate the booking flow.
4. **Payment –** Guide users through the payment process (credit card, net banking, etc.). Highlight security measures to protect users’ financial information.
5. **Ticket Confirmation and Cancellation –** Explain how users receive their e-tickets after successful booking. Provide instructions on canceling or modifying a reservation.
6. **Account Management –** Describe how users can update their profile, change passwords, and view booking history.
7. **Contact Information –** Provide customer support details (email, phone number) for any issues users may encounter.
   1. **Operating Manual**
8. **System Overview –** Describe the architecture of the system (frontend, backend, database).
9. **Installation and Setup –** Provide step-by-step instructions for deploying the system on a server.
10. **Configuration –** Explain how to configure system parameters (e.g., bus routes, fare rules). Document any environment variables or configuration files.
11. **Database Management –** Describe database schema and relationships. Explain backup and restore procedures.
12. **Security Measures –** Detail security practices (encryption, authentication, authorization). Address potential vulnerabilities and how they are mitigated.
13. **Monitoring and Maintenance –** Explain how to monitor system performance (logs, metrics). Document routine maintenance tasks (e.g., database cleanup, server updates).
14. **Troubleshooting –** List common issues and their solutions. Include error codes and troubleshooting steps.
15. **Emergency Procedures –** Outline steps to follow during system failures or security breaches.
16. **Advantages and Limitations:**

**Advantages:**

* **Reduced Administrative Burden –** Automation eliminates manual record-keeping tasks, reducing the workload for bus company staff.
* **Streamlined Operations for Bus Operators –** Real-time updates ensure that passengers receive accurate information about bus availability and departure times. Online platforms allow bus operators to promote their services through digital marketing channels.
* **24/7 Availability and accessibility –** An online bus reservation system provides 24/7 accessibility for users. Customers can make reservations or book tickets at any time, from anywhere with an internet connection.
* **Improved Customer Service –** Online systems provide real-time updates on bus schedules, delays, and cancellations. Customers can easily search for available buses, select preferred seats, and customize their travel plans.
* **Flexibility and error free system –** Managing reservations manually can be time-consuming and error prone. With an online system, bus operators can efficiently handle bookings, seat availability, and schedules.
* **Secure Payment Options –** Online systems offer secure payment gateways, allowing customers to pay for their tickets using various methods (credit/debit cards, net banking, etc.).
* **Self-Service Capability –** Self-service features enhance user satisfaction and reduce the need for customer support.

**Limitations:**

* **Dependency on Technology –** Online systems rely heavily on technology infrastructure, including servers, databases, and internet connectivity.
* **Initial Setup and Maintenance Costs –** Developing and implementing an online reservation system requires an upfront investment in software development, hosting, and security measures.
* **Customer digital and language illiteracy –** User needs have basic knowledge of Computer/Smart phone. User should have knowledge of online payment using credit card or debit card. User should know English language as interface is provided in English.
* **User Learning Curve –** Some passengers, especially older individuals or those unfamiliar with technology, may find it challenging to navigate online booking platforms. The learning curve can lead to deter potential customers.
* **Security Concerns –** While online payment gateways are secure, there is always a risk of data breaches or unauthorized access. Integrating with external services (e.g., payment gateways, SMS providers, map APIs) can be complex.
* **Dependency on electricity and internet –** Online systems require electricity and internet connectivity for its functioning. Passengers in remote areas with poor internet and passengers with no internet connectivity and electricity may struggle to use online booking services.
* **Fees for third-party booking platforms -** Some bus operators rely on third-party booking platforms, which charge commissions for each booking. These fees can add up, impacting the overall revenue for the bus company.

1. **Future Scope/Proposed Enhancements**

* **Mobile Apps (Android and iOS) –** Developing dedicated mobile apps for Android and iOS platforms is a crucial step. These apps will enhance user engagement and accessibility.
* **Integration with Other Travel Services –** Combining bus reservations with flights, trains, and hotels creates a comprehensive travel ecosystem.
* **Geolocation Services –** Use GPS to provide real-time bus tracking and estimated arrival times. Notify passengers about delays, diversions, or route changes.
* **Customer Support Chatbots –** IntegrateAI-powered chatbots to handle common queries, cancellations, and refunds.
* **Dynamic Pricing Algorithms –** Implement pricing strategies based on demand, seasonality, and availability. Offer last-minute deals and discounts for filling empty seats closer to departure.

1. **Conclusion**

Throughout the project, we successfully achieved most of the objectives, resulting in an efficient, user-friendly platform. We addressed the limitations of the manual system by creating an error-free, secure, and reliable online reservation system. Improved customer convenience, streamlined operations, and secure payment options. In conclusion, our online bus reservation system is poised to revolutionize the travel industry. By embracing technology, enhancing user experiences, and collaborating with small travel companies, you’ve set the stage for continued success.

**12. Bibliography and References**

**Book References:**

* JSP and servlets – A computer study
* Programming with Java – by E Balagurusamy
* Murach’s MYSQL

# **Web References:**

* [**https://www.tutorialspoint.com/**](https://www.tutorialspoint.com/)
* [**https://www.guru99.com/**](https://www.guru99.com/)
* [**https://www.w3schools.com/java/**](https://www.w3schools.com/java/)
* [**https://www.javatpoint.com/java-tutorial**](https://www.javatpoint.com/java-tutorial)
* [**https://www.mysqltutorial.org/**](https://www.mysqltutorial.org/)
* [**www.google.co.in/**](http://www.google.co.in/)