M.Kumarasamy College of Engineering
NAAC Accredited Autonomous Institution
Approved by AICTE & Affiliated to Anna University
ISO 9001:2015 Certified Institution
Thalavapalayam, Karur - 639 113, TAMILNADU.

A Project Report

on

BUS RESERVATION SYSTEM

Submitted in partial fulfillment of requirements for the award of the course

of

CGB1221-DATABASE MANAGEMENT SYSTEMS

Under the guidance of

Mrs. .S. SANDHIYA M.E.,

Assistant Professor / CSE

Submitted By

ARUN SANJEEV M S (927623BCS011)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

M.KUMARASAMY COLLEGE OF ENGINEERING
(Autonomous)

KARUR – 639 113

MAY 2025

M.Kumarasamy College of Engineering
NAAC Accredited Autonomous Institution
Approved by AICTE & Affiliated to Anna University
ISO 9001:2015 Certified Institution
Thalavapalayam, Karur - 639 113, TAMILNADU.

M. KUMARASAMY COLLEGE OF ENGINEERING

(Autonomous Institution affiliated to Anna University, Chennai)

KARUR – 639 113

BONAFIDE CERTIFICATE

This is to certify that this project report on **"BUS RESERVATION SYSTEM"** is the bonafide work of **ARUN SANJEEV M S (927623BCS011)** who carried out the project work during the academic year 2024 - 2025 under my supervision.

Signature

Mrs. S. SANDHIYA M.E.,

SUPERVISOR,

Department of Computer Science and Engineering,

M. Kumarasamy College of Engineering,

Thalavapalayam, Karur -639 113.

Signature

Dr. D. PRADEEP M.E., Ph.D.,

HEAD OF THE DEPARTMENT,

Department of Computer Science and Engineering,

M. Kumarasamy College of Engineering,

Thalavapalayam, Karur -639 113.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

VISION OF THE INSTITUTION

To emerge as a leader among the top institutions in the field of technical education

MISSION OF THE INSTITUTION

- Produce smart technocrats with empirical knowledge who can surmount the global challenges
- Create a diverse, fully-engaged, learner-centric campus environment to provide quality education to the students
- Maintain mutually beneficial partnerships with our alumni, industry, and Professional associations

VISION OF THE DEPARTMENT

To achieve education and research excellence in computer Science and Engineering

MISSION OF THE DEPARTMENT

- To excel in academic through effective teaching learning techniques.
- To promote research in the area of computer science and engineering with the focus on innovation.
- To transform students into technically competent professionals with societal and ethical responsibilities.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

- **PEO1:** Graduates will have successful Career in software industries and R&D divisions through continuous learning
- **PEO2:** Graduates will provide effective solutions for real world problems in the key domain of computer science and engineering and engage in lifelong learning.
- **PEO3:** Graduates will excel in their profession by being ethically and socially responsible

PROGRAM OUTCOMES (POs)

Engineering students will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, give and receive clear instructions.

11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

- **PSO1: Professional Skills:** Ability to apply the knowledge of computing techniques to design and develop computerized solutions for the problems.
- **PSO2: Successful career:** Ability to utilize the computing skills and ethical values in creating a successful career.

ABSTRACT

The Bus Ticket Booking System is a full-stack web application developed using PHP, MySQL, HTML, CSS, and JavaScript, aimed at modernizing and simplifying the traditional bus ticket reservation process. This system enables passengers to easily search for buses based on route and schedule, book available seats, make secure payments, and view their booking history — all within a single, user-friendly platform. To enhance safety and user control, features like emergency SOS alerts, multi-bus journey planning, and real-time booking status have been incorporated. For administrators, the system offers modules to add and manage bus details, monitor bookings, cancel reservations, and generate daily or weekly reports, making the backend operations smooth and efficient. The backend logic is powered by PHP, while MySQL is used for managing and retrieving data securely and accurately. The application follows essential DBMS principles such as data normalization, primary and foreign key constraints, and SQL joins to maintain data integrity and optimize performance.

This system demonstrates how traditional ticketing processes can be transformed into a digital, secure, and scalable platform, improving both user convenience and administrative efficiency in the public transportation sector.

ABSTRACT WITH POs AND PSOs MAPPING

ABSTRACT	COs MAPPED	POs MAPPED	PSOs MAPPED
The **Bus Ticket Booking System** is a web-based application developed using **PHP**, **MySQL**, and standard web technologies like **HTML, CSS, and JavaScript** to automate and simplify the bus ticket reservation process. This system allows users to easily **search for buses**, **book seats**, **make online payments**, and **view booking history**, all from a single platform. It enhances user convenience while ensuring fast and secure transactions. Administrators are provided with tools to **manage bus schedules**, **monitor ticket bookings**, **resolve SOS alerts**, and **generate reports** for operational oversight. The application is built using core **DBMS concepts** such as **data normalization**, **primary and foreign keys**, and **relational joins**, ensuring accurate and efficient data handling.	CO1 CO2 CO3 CO4 CO5	PO1 PO2 PO3 PO5 PO6 PO8 PO9 PO10 PO11 PO12	PSO1 PSO2

Note: 1- Low, 2-Medium, 3- High

SUPERVISOR **HEAD OF THE DEPARTMENT**

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 PROBLEM STATEMENT

Design and develop a Bus Ticket Booking System using DBMS concepts to streamline the process of booking bus tickets for passengers. The system should allow users to search for available buses based on departure and arrival locations, dates, and times. It should enable users to book tickets, make payments, and view booking history. The database will manage details such as bus schedules, passenger information, ticket availability, and transaction records. Additionally, the system should allow administrators to update bus schedules, monitor bookings, and generate reports.

1.2 INTRODUCTION

In today's digital era, the transportation industry is embracing rapid digitalization with increasing demand for automation, accuracy, and real-time services. Traditional bus booking processes often rely on manual entries, inefficient schedule handling, and lack of user communication, leading to delays and passenger dissatisfaction. To overcome these issues, the Bus Reservation System has been developed as a web-based platform using PHP for server-side scripting and MySQL for database operations. This system aims to automate key functions such as seat booking, schedule management, and user notifications. With dedicated dashboards for Admins, Conductors, and Passengers, it ensures role-based access, proper task distribution, and secure transaction handling. The system supports real-time seat tracking, clear data flow, and timely alerts through Twilio integration, greatly enhancing operational efficiency and user satisfaction. Additionally, the interface is designed to be user-friendly, enabling passengers to book tickets with minimal steps. The admin panel allows for quick updates of routes, fares, and bus status. This modern solution aims to reduce human errors and improve the reliability of the overall booking experience.

1.3 DATABASE MANAGEMENT SYSTEMS

A Database Management System (DBMS) is a software system that enables the creation, management, and manipulation of databases. It allows users to store, retrieve, update, and delete data efficiently while ensuring data integrity, security, and consistency. DBMS provides an interface between the database and the users or applications, allowing structured access to the data using queries. It supports important operations like transaction management, concurrency control, and data backup. Examples of DBMS include MySQL, Oracle, SQL Server, and PostgreSQL.

1.4 PHP

PHP (Hypertext Preprocessor) is a widely-used open-source server-side scripting language designed for web development. It is embedded within HTML code and used to create dynamic and interactive web pages. PHP executes on the server and sends the output (usually HTML) to the client's browser. It can interact with databases like MySQL, making it ideal for building data-driven applications. PHP supports features like form handling, session management, file operations, and authentication. It is platform-independent and compatible with most servers, including Apache and Nginx. PHP is known for its simplicity, flexibility, and strong community support. It powers many popular websites and systems, including WordPress and Facebook.

1.5 OBJECTIVE

The objective of the Bus Reservation System is to develop a reliable and secure web-based platform that simplifies and streamlines the bus booking and management process. The system aims to eliminate traditional manual operations, which are often slow and error-prone, by automating critical tasks such as seat reservation, schedule updates, and user management. It ensures effective coordination among different user roles—Admins, Conductors, and Passengers—so that each user can operate efficiently within a well-structured workflow. With integrated features like real-time seat availability, instant booking confirmation, and SMS alerts via Twilio, the system is designed to enhance accuracy, user responsiveness, and overall operational transparency.

CHAPTER 2

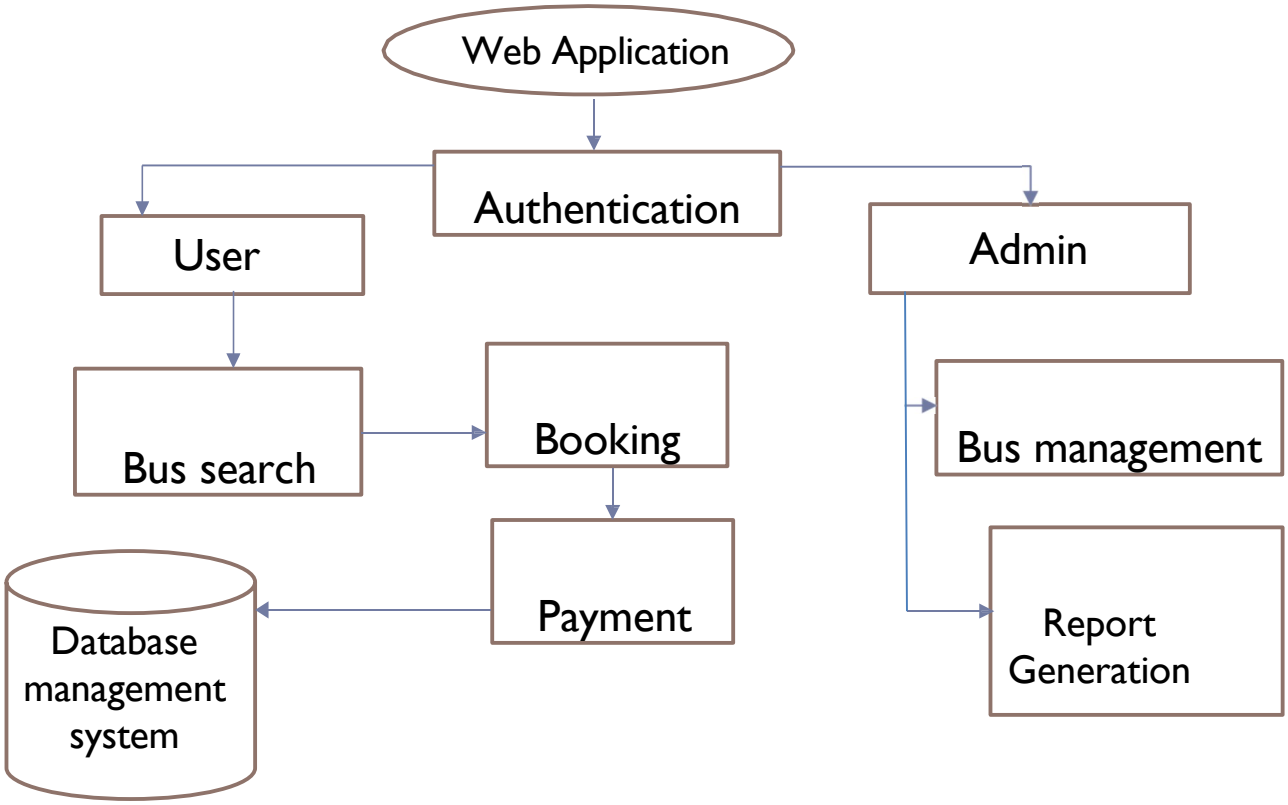
PROJECT METHODOLOGY

2.1 PROPOSED WORK

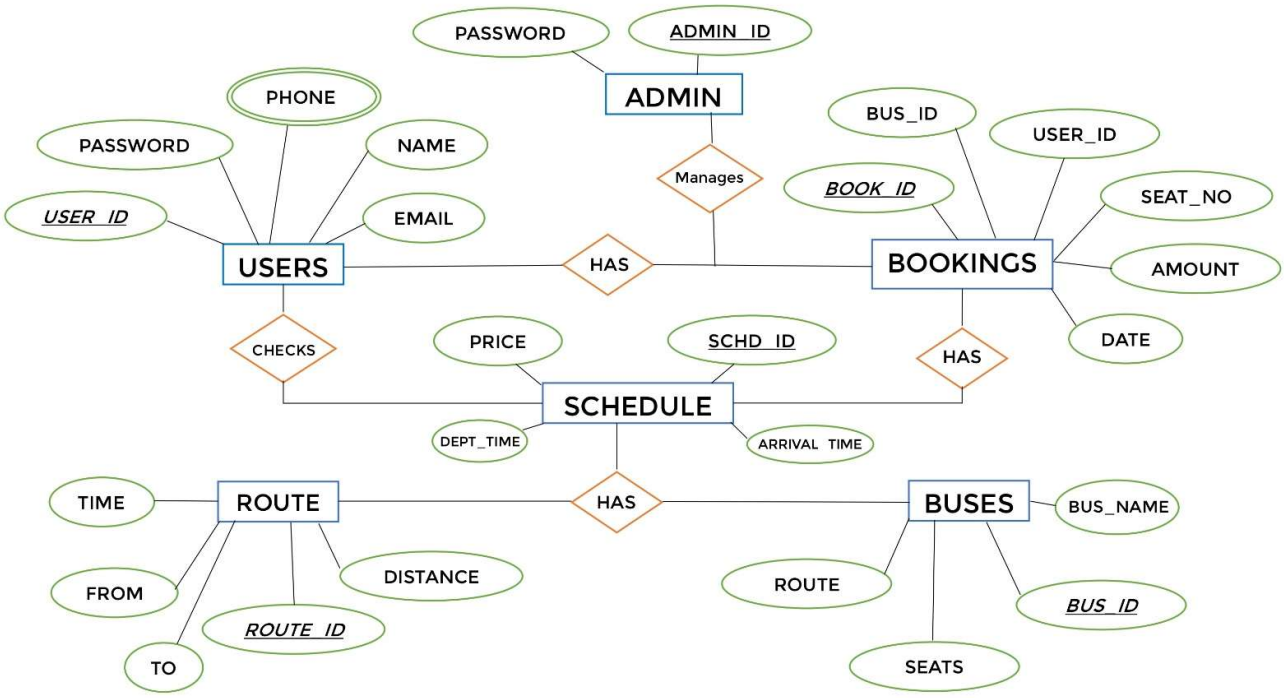
The proposed work focuses on designing and developing a web-based Bus Ticket Booking System that simplifies the process of bus reservations for users and provides efficient administrative control for operators. The system will allow users to register/login securely, search buses by routes and schedules, book seats, and make payments online. It will support dynamic seat availability checks, store booking history, and allow cancellations or refunds based on policies.

Admins will be able to add/manage buses and schedules, monitor bookings, resolve SOS alerts, and generate analytical reports on bookings and revenue. The platform will integrate emergency contact listings and a smart multi-bus journey planner, suggesting alternatives if no direct route is available. The backend will use PHP and MySQL, and the frontend will be built using HTML, CSS, and JavaScript, offering a user-friendly and responsive interface. The database design will follow DBMS principles to ensure data integrity, security, and scalability.

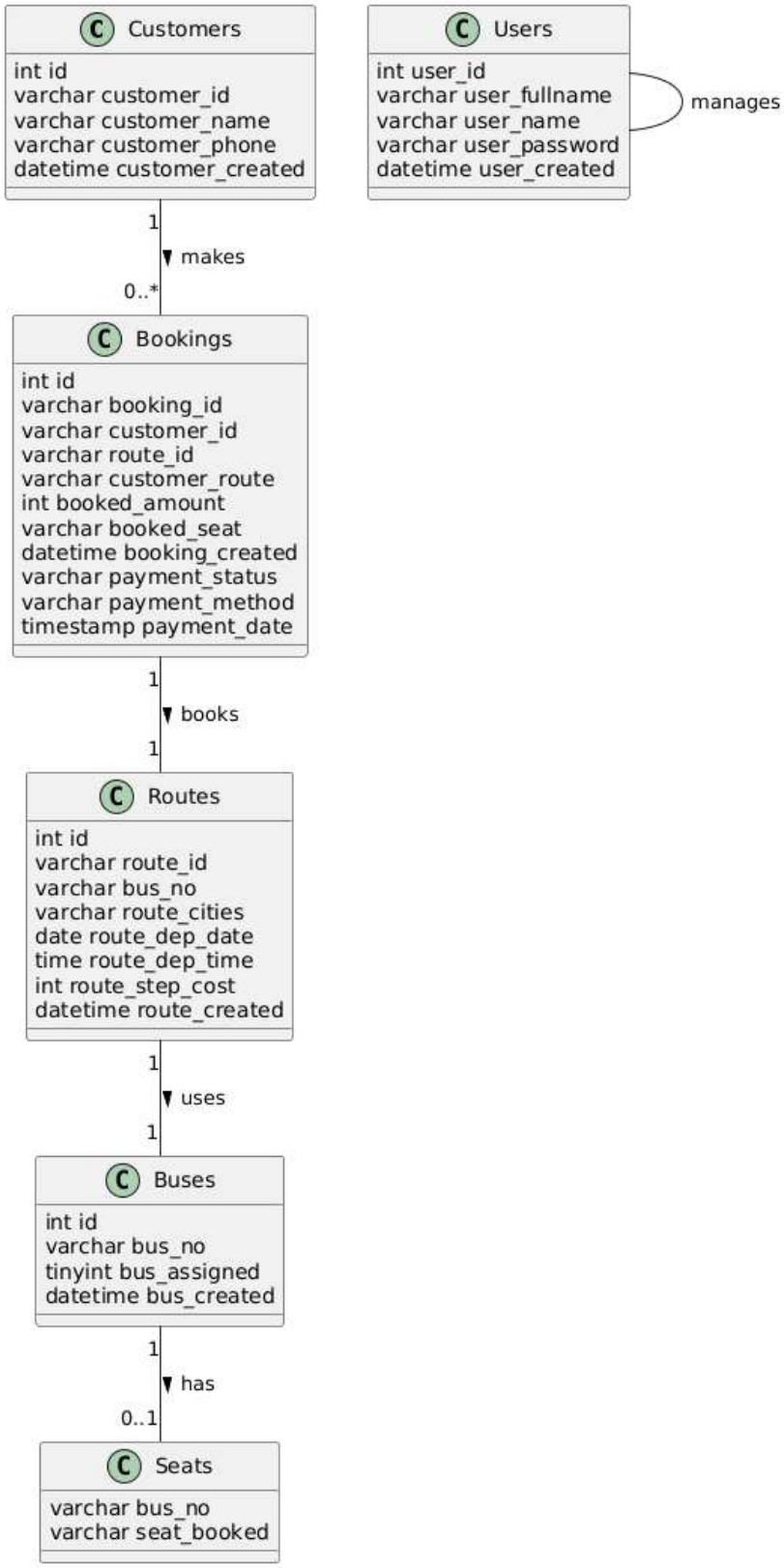
2.2 PROPOSED ARCHITECTURE



2.3 E-R DIAGRAM



2.4 SCHEMA DIAGRAM



CHAPTER 3

SOFTWARE REQUIREMENTS

3.1 Front End

The frontend of the Bus Reservation System is developed using HTML and CSS, which together define the structure and design of the web interface. HTML is used to build the layout and core elements such as booking forms, navigation menus, schedule displays, and seat selection panels, enabling users to interact with the system easily. CSS enhances these elements by applying visual styles, color themes, fonts, and responsive layouts, making the interface clean and user-friendly. The design ensures that all users—whether Admins, Conductors, or Passengers—can effortlessly navigate the system, book tickets, check availability, or manage schedules with minimal effort and maximum clarity.

3.2 Back End

The backend of the Bus Reservation System is developed using PHP, which serves as the connection between the frontend interface and the MySQL database. PHP manages essential operations like user login, form handling, trip scheduling, and seat reservation logic. It securely processes user requests—such as booking a seat, canceling a reservation, or viewing travel history—and ensures accurate data is fetched or updated in the database.

PHP scripts are also responsible for generating dynamic pages, managing errors, and enforcing role-based access so that Admins, Conductors, and Passengers can access only the features relevant to them. This server-side logic enables smooth, secure, and consistent functioning of the overall system. The backend also handles Twilio API integration to send real-time booking alerts via SMS. This not only enhances communication but also boosts user trust and reliability in the system.

CHAPTER 4

MODULE DESCRIPTION

4.1 User Authentication Module

- Handles user registration, login, and session management.
- Distinguishes between regular users and admin users using roles.

4.2 Bus Management Module (Admin)

- Allows admin to add, update, and delete buses.
- Stores bus types, amenities, and status.

4.3 Route and Scheduling Module (Admin)

- Manages routes between cities and their associated schedules.
- Defines distance, time, and pricing for each route.

4.4 Seat Booking Module

- Allows users to select seats and book tickets.
- Ensures real-time seat availability checks.
- Stores booking history with status tracking (Confirmed/Cancelled)

4.5 Payment Module

- Handles payment status tracking and refund management.
- Stores payment method, transaction ID, and amount.

4.6 Admin Dashboard

- Admins can view bookings, cancel tickets, and generate reports.
- Revenue tracking based on booking records.

IMPLEMENTATION

LOGIN.HTML

```
<?php
   require 'assets/partials/_functions.php';
   $conn = db_connect();

   if(!$conn)
      die("Connection Failed");
?>

<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Bus Ticket Bookings</title>
   <!-- google fonts -->
   <link rel="preconnect" href="https://fonts.gstatic.com">
   <link
href="https://fonts.googleapis.com/css2?family=Montserrat:wght@100;200;300;400;500
&display=swap" rel="stylesheet">
   <!-- Font-awesome -->
   <script src="https://kit.fontawesome.com/d8cfbe84b9.js"
```

```
crossorigin="anonymous"></script>
    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.1/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
+0n0xVW2eSR5OomGNYDnhzAbDsOXxcvSN1TPprVMTNDbiYZCxYbOOl7+AMvy
TG2x" crossorigin="anonymous">
    <!-- CSS -->
    <?php
        require 'assets/styles/styles.php'
    ?>
</head>
<body>
    <?php

    if(isset($_GET["booking_added"]) && !isset($_POST['pnr-search']))
    {
        if($_GET["booking_added"])
        {
            echo '<div class="my-0 alert alert-success alert-dismissible fade show"
role="alert">
                <strong>Successful!</strong> Booking Added, your PNR is <span style="font-
weight:bold; color: #272640;">'. $_GET["pnr"] .'</span>
                <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
                </div>';
```

```
    }
    else{
        // Show error alert
        echo '<div class="my-0 alert alert-danger alert-dismissible fade show"
role="alert">
        <strong>Error!</strong> Booking already exists
        <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
        </div>';
    }
  }

  if($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST["pnr-
search"]))
  {
    $pnr = $_POST["pnr"];

    $sql = "SELECT * FROM bookings WHERE booking_id='$pnr'";
    $result = mysqli_query($conn, $sql);

    $num = mysqli_num_rows($result);


      <!DOCTYPE html>
      <html lang="en">
      <head>
        <meta charset="UTF-8">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login | Online Book Store</title>
    <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">
    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
    <style>
        body {background: url('https://media.istockphoto.com/id/1219786897/photo/bookcase-with-
books-on-a-smartphone-screen-on-a-desktop-electronic-library-in-a-mobile-
phone.jpg?s=612x612&w=0&k=20&c=Ybx2tve7aLjUpvgYHy6It4S9IHmE_opuwoQDoRQuuFk='
) no-repeat center center fixed;
            background-size: cover;
            height: 100vh;
            display: flex;
            justify-content: center;
            align-items: center;
        }
        .overlay {
            background: rgba(34, 129, 218, 0.5);
            position: absolute;
            width: 100%;
            height: 100%;
            top: 0;
            left: 0;
```

```
.login-box {
    background: rgba(255, 255, 255, 0.2);
    backdrop-filter: blur(10px);
    padding: 40px;
    border-radius: 10px;
    width: 350px;
    text-align: center;
    color: white;
    box-shadow: 0 4px 8px rgba(0, 0, 0, 0.3);
}
.form-control {
    background: rgba(255, 255, 255, 0.2);
    border: none;
    color: white;
}
.form-control::placeholder {
    color: rgba(255, 255, 255, 0.7);
}
.btn-custom {
    background: #4800ffad;
    color: white;
    transition: 0.3s;
}
.btn-custom:hover {
    background: #4800ffad;
}
.toggle-buttons {
    display: flex;
    justify-content: space-around;
    margin-bottom: 10px;
}
```

```
            background: #ffffff30;
            border: none;
            color: white;
            padding: 10px;
            cursor: pointer;
            font-weight: bold;
            width: 48%;
        }
        .toggle-btn.active {
            background: #fe13ea  }
    </style>
</head>
<body>
    <div class="login-box">
        <h2 id="login-title">User Login</h2>
        <div class="toggle-buttons">
            <button id="user-btn" class="toggle-btn active">User</button>
            <button id="admin-btn" class="toggle-btn">Admin</button>
        </div>
        <form id="loginForm" action="login.php" method="POST">
            <input type="hidden" id="isAdmin" name="isAdmin" value="false">
                <input type="text" id="username" name="username" class="form-control"
placeholder="Username (e.g., user123, admin007)" required>
            </div>
            <div class="mb-3">
" placeholder="Password (e.g., userpass, adminxyz)" required>
            </div>
            <button type="submit" class="btn btn-custom w-100">Login</button>
        </form>
        <p class="mt-3"><a href="signup.html">Don't have an account? Sign Up</a></p>
    </div>
```

LOGIN.PHP

```
<section id="home">
  <div id="route-search-form">
    <h1>Bus Reservation System</h1>

    <p class="text-center">Welcome to Simple Bus Ticket Booking System. Login now to manage bus tickets and much more. OR, simply scroll down to check the Ticket status using Passenger Name Record (PNR number)</p>

    <center>
      <button class="btn btn-danger " data-bs-toggle="modal" data-bs-target="#loginModal">Administrator Login</button>

    </center>

    <br>
    <center>
    <a href="http://localhost/BusBook/customer_booking.php"><button class="btn btn-primary"> Book Now !! <i class="fa fa-arrow-down"></i></button></a>
    </center>

  </div>
</section>
<div id="block">
  <section id="info-num">
    <figure>
      <img src="assets/img/route.svg" alt="Bus Route Icon" width="100px"
```

```
height="100px">
            <figcaption>
                <span class="num counter" data-target="<?php echo count($routeData);
?>">999</span>
                <span class="icon-name">routes</span>
            </figcaption>
        </figure>
        <figure>
            <img src="assets/img/bus.svg" alt="Bus Icon" width="100px"
height="100px">
            <figcaption>
                <span class="num counter" data-target="<?php echo count($busData);
?>">999</span>
                <span class="icon-name">bus</span>
            </figcaption>
        </figure>
        <figure>
            <img src="assets/img/customer.svg" alt="Happy Customer Icon"
width="100px" height="100px">
            <figcaption>
                <span class="num counter" data-target="<?php echo count($customerData);
?>">999</span>
                <span class="icon-name">happy customers</span>
            </figcaption>
        </figure>
        <figure>
```

```
            <img src="assets/img/ticket.svg" alt="Instant Ticket Icon" width="100px"
height="100px">
            <figcaption>
                <span class="num"><span class="counter" data-target="20">999</span>
SEC</span>
                <span class="icon-name">Instant Tickets</span>
            </figcaption>
        </figure>
    </section>
    <section id="pnr-enquiry">
        <div id="pnr-form">
            <h2>PNR ENQUIRY</h2>
            <form action="<?php echo $_SERVER["REQUEST_URI"]; ?>"
method="POST">
                <div>
                    <input type="text" name="pnr" id="pnr" placeholder="Enter PNR">
                </div>
                <button type="submit" name="pnr-search">Submit</button>
            </form>
```

CHAPTER 6

SNAPSHOTS

6.1 Home Page

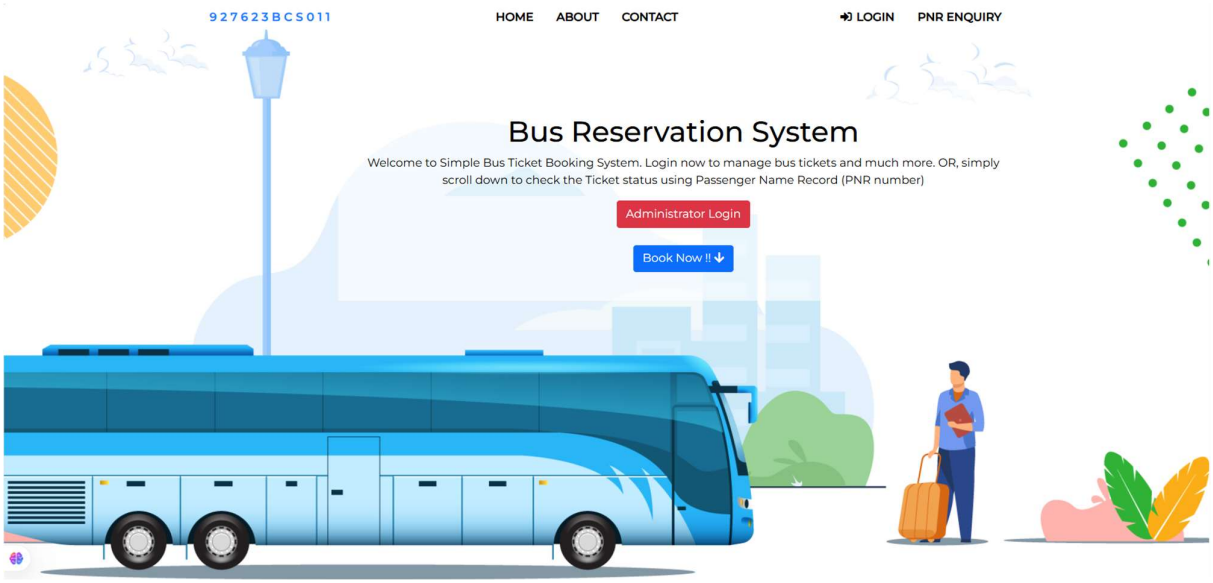


Fig.6.1 : A Web Based Login Page for Booking Tickets and Login

6.2 Login Page (Admin)

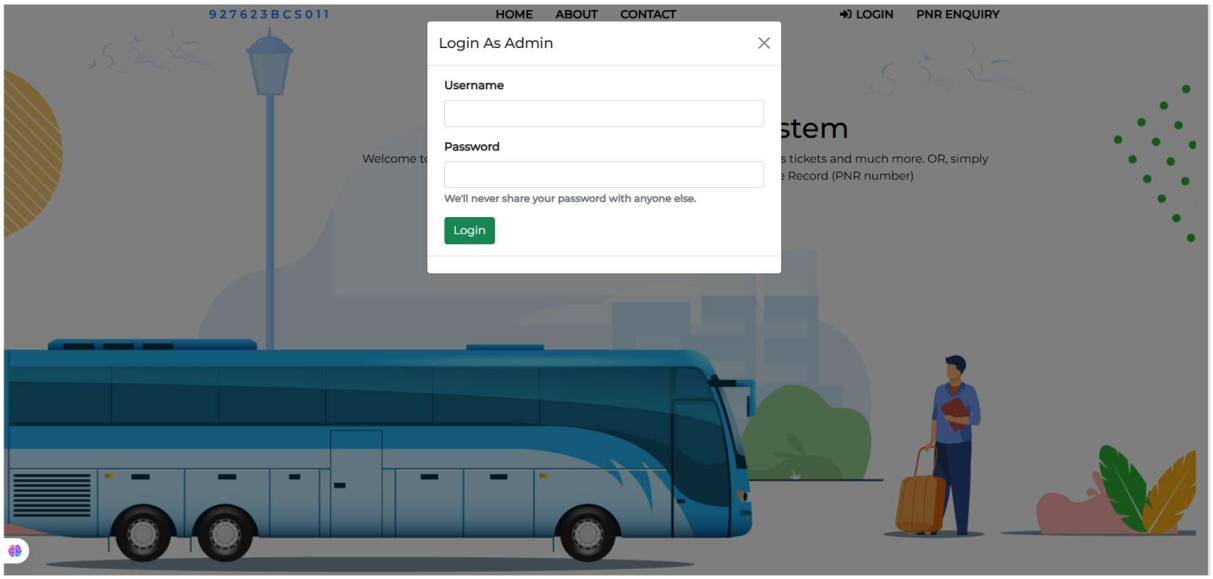


Fig.6.2 : Login Page for Administrator

6.3 User's Booking Page

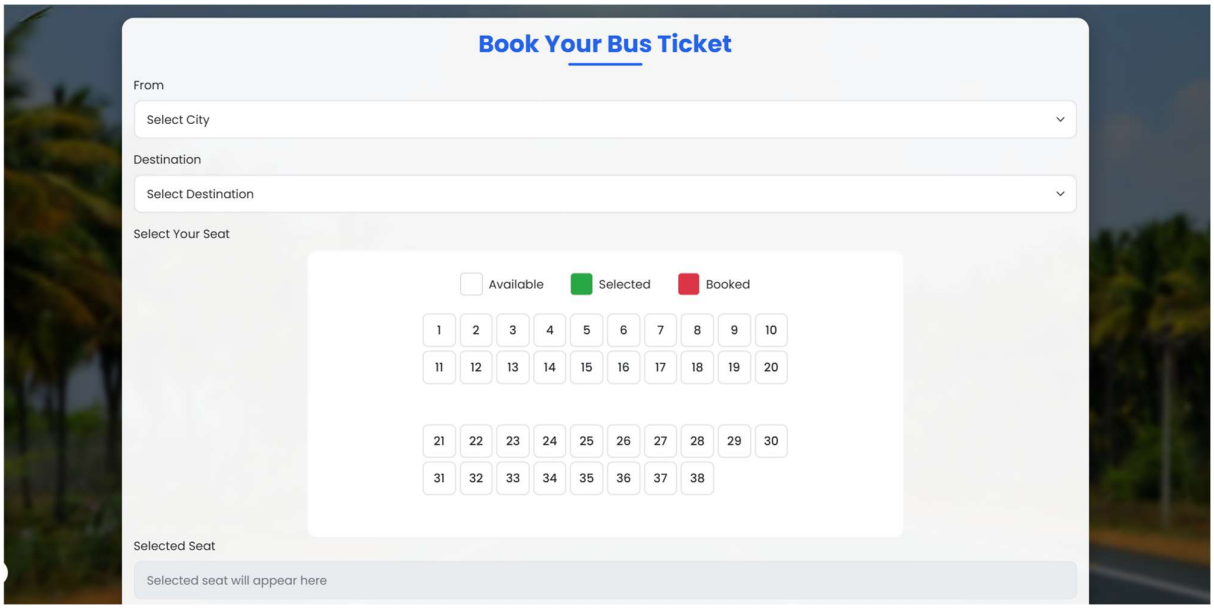


Fig.6.3 : User Ticket Booking Page where users can book ticket

6.4 Payment Page

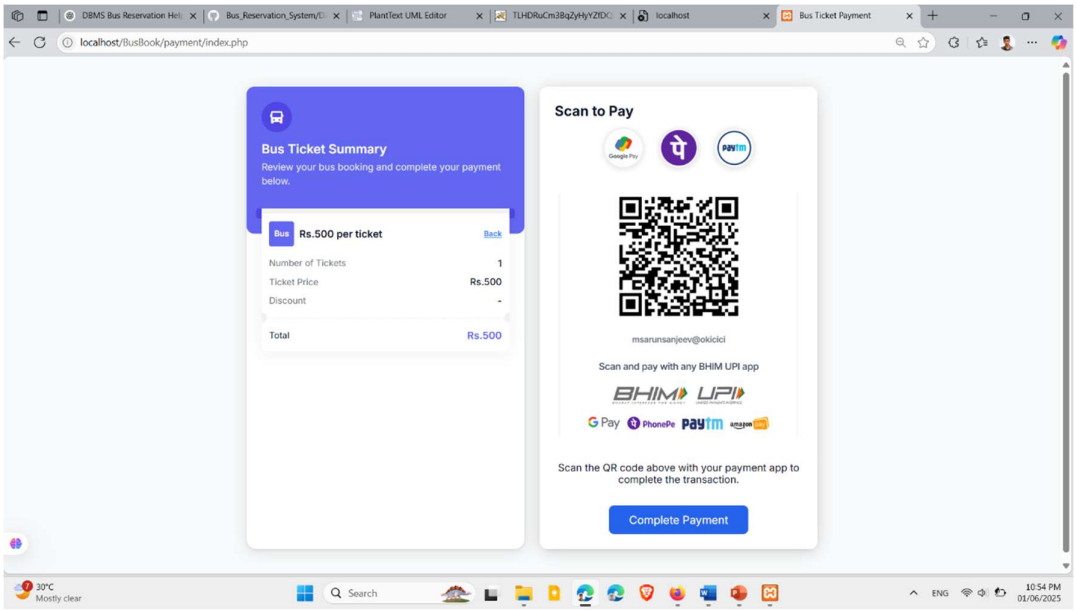


Fig.6.4: Payment Page where users can pay using UPI Apps' QR Payment

6.5 Ticket Page

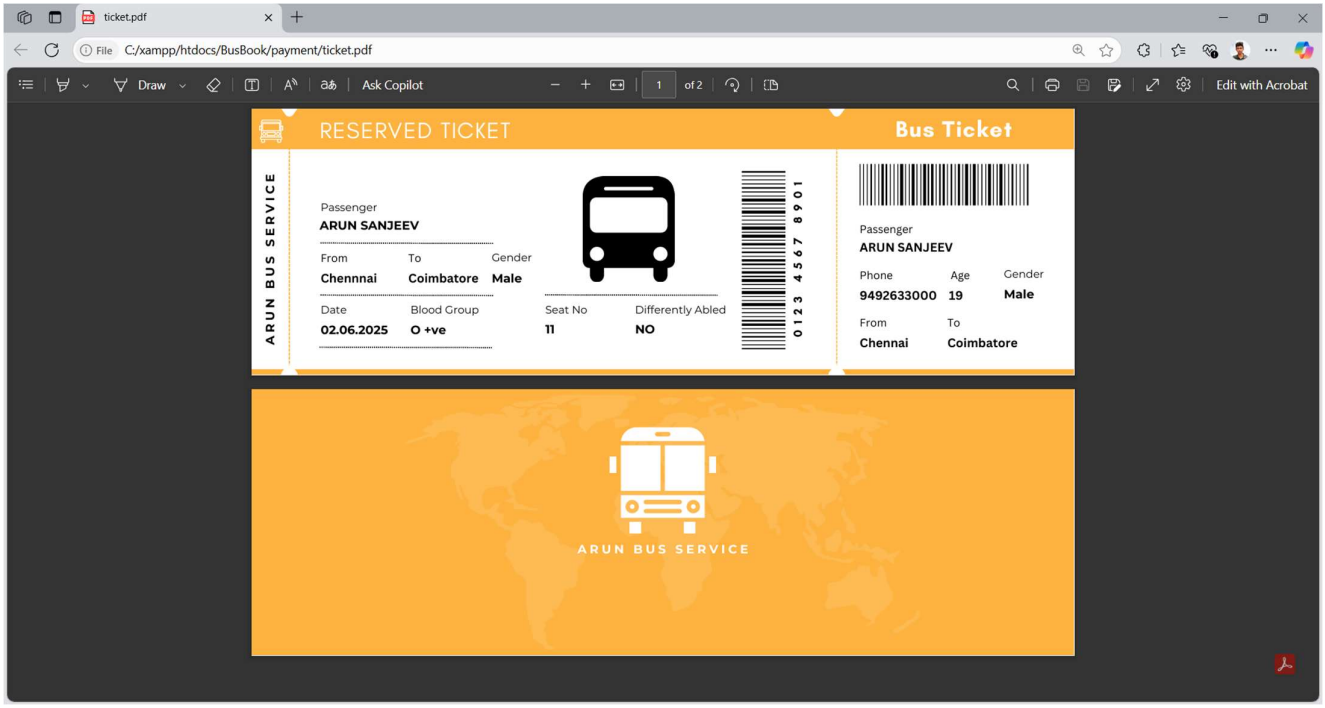


Fig.6.5 : Generated Ticket PDF where users can print with BarCode

6.6 Admin Dashboard

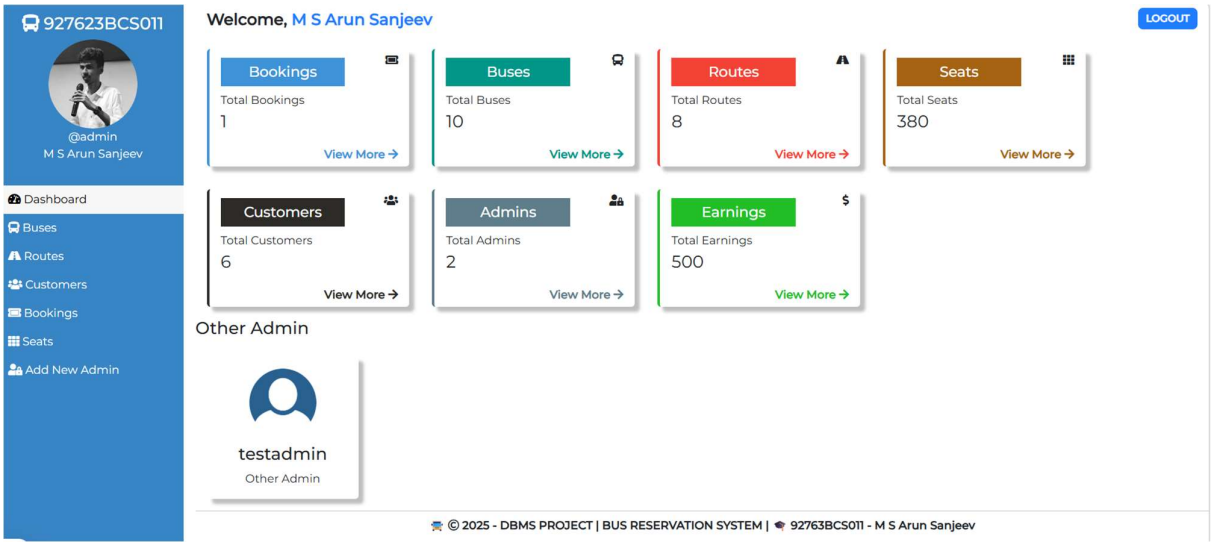


Fig.6.6: Detailed Dashboard where Admins can view Reports and Mange the Buses and Bookings

CONCLUSION

The Bus Ticket Booking System has been successfully developed as a user-friendly and efficient solution to the challenges faced by passengers and bus operators in traditional booking systems. With the growing demand for digital services, this platform automates the entire booking process, offering both passengers and administrators a seamless experience.

For users, the system allows easy registration, login, bus search, schedule viewing, ticket booking, and secure online payments. The ability to access booking history, download or print e-tickets, and receive instant confirmation adds convenience and transparency. Administrators benefit from powerful backend controls, including bus schedule management, booking monitoring, revenue tracking, and emergency SOS alerts. The admin dashboard provides key metrics for informed operational decisions.

The database design follows DBMS principles such as normalization and indexing to ensure data integrity, efficiency, and scalability.

The user interface is designed for simplicity, responsiveness, and accessibility across devices, providing an intuitive experience for all users, whether travelers or administrators.

In conclusion, this Bus Ticket Booking System effectively combines modern technologies with sound database design to deliver a robust and scalable solution. It has the potential for further enhancements, such as real-time bus tracking, SMS/email notifications, and multilingual support, providing a strong foundation for a fully-featured commercial system.

REFERENCES

- Silberschatz, A., Korth, H. F., & Sudarshan, S. (2019). Database System Concepts (7th ed.). McGraw-Hill Education.
- Welling, L., & Thomson, L. (2009). PHP and MySQL Web Development (4th ed.). Addison-Wesley.
- Ullman, L. (2018). PHP for the Web: Visual QuickStart Guide (5th ed.). Peachpit Press.
- MySQL Documentation - https://dev.mysql.com/doc/
- PHP Manual - https://www.php.net/manual/en/

- Research Paper – Online Bus Ticket Reservation System (Use Case, Architecture & Design)

- SRS Document – Software Requirements Specification for Online Bus Ticket Reservat