AIRLINE RESERVATION SYSTEM

A MINI PROJECT REPORT

SUBMITTED BY

ADHITHYA P G	**2116221701005**
AKSAYAA S V	**2116221701006**
CHARAN RAJ DK	**2116221701011**
GOWRI NANDA M	**2116221701016**

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND DESIGN

RAJALAKSHMI ENGINEERING COLLEGE

THANDALAM

CHENNAI – 602105

ANNA UNIVERSITY: CHENNAI 600625

BONAFIDE CERTIFICATE

Certified that this project report "**AIRLINE RESERVATION SYSTEM**" is the Bonafide work of **"ADHITHYA PG (2116221701005), AKSAYAA S V(2116221701006), CHARAN RAJ DK (2116221701011) AND GOWRI NANDA M (2116221701016)"** who carried out the project work under my supervision.

SIGNATURE   SIGNATURE

Mr.Uma Maheswar Rao MA.,MFA in Design .,   **Mr.Vijaykumar M.Tech.,**

Professor and Head,   Asst. Professor (SS),

Computer Science and Design,   Computer   Science   and   Design,
Rajalakshmi Engineering College,   Rajalakshmi   Engineering   College,
Thndalam, Chennai – 602105.   Thandalam, Chennai – 602105.

EXTERNAL EXAMINER   **INTERNAL EXAMINER**

ACKNOWLEGEMENT

We are highly obliged in taking the opportunity to thank our Chairman **Mr.S. Meganathan**, Chairperson **Dr.Thangam Meganathan** and our Principal **Dr.S.N. Murugesan** for providing all the facilities which are required to carry out this project work.

We are ineffably indebted to our H.O.D **Mr. Uma Maheswar Rao M.A., MFA.,** for his conscientious guidance and encouragement to make this project a recognizable one.

We are extremely thankful to our faculty **Mr.Vijaykumar M.Tech.,** for his valuable guidance and indefatigable support and extend our heartfelt thanks to all the teaching and non-teaching staff of **Computer Science & Design department** who helped us directly or indirectly in the completion of this project successfully.

At last but not least gratitude goes to our friends who helped us compiling the project and finally to god who made all things possible.

Any omission in this brief acknowledgement doesn't mean lack of gratitude.

ADHITHYA P G   **2116221701005**

AKSAYAA S V   **2116221701006**

CHARAN RAJ D K   **2116221701044**

GOWRI NANDA M   **2116221701047**

ABSTRACT

The **"Airline Reservation System"** is designed to simplify and enhance the ticket booking process for passengers. Traditional manual methods were prone to delays, errors, and miscommunication, especially during high-demand periods. This system addresses these challenges by providing a centralized digital platform for managing reservations efficiently. Key features include real-time updates on seat availability, a simple booking process, and the ability to search and filter trains based on routes. Passengers can easily view their reservations, track booking status, and cancel tickets when necessary. By ensuring accurate and instant communication, the system minimizes errors and enhances transparency. Additionally , the platform optimizes seat allocation, ensuring fair and equitable distribution among passengers. With a focus on user experience, the interface is designed to be intuitive and accessible, making it suitable for all users. The system also supports robust database management to handle multiple concurrent operations seamlessly. This modernized approach revolutionizes railway ticketing, offering convenience, reliability, and efficiency to passengers while reducing the operational burden on railway authorities.

TABLE OF CONTENT

PAGE NO.

CHAPTER – 1

INTRODUCTION

The project discussed here, is implemented using the concepts of PYTHON and SQL.

1.1PYTHON

Python is a versatile, high-level programming language known for its simplicity, readability, and broad applicability. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming, making it a popular choice for developers worldwide. Python emphasizes code readability and efficiency, allowing developers to focus on problem-solving rather than syntax. For the Airline Reservation System, Python was chosen as the primary programming language due to its robust library ecosystem and ease of use.

The system employs Flask, a lightweight, open-source Python web framework, to create a dynamic and interactive web-based interface. Flask provides flexibility and simplicity, enabling developers to build and deploy web applications efficiently while offering extensive control over application structure. Its minimalist nature allows seamless integration with additional libraries, enhancing user experience and productivity.

This system integrates Python's extensive capabilities with MySQL for database management, ensuring efficient handling of passenger data, flight schedules, and reservations. MySQL, a powerful and widely used relational database management system, offers robust performance, scalability, and reliability, making it ideal for managing large datasets and maintaining data integrity. The combination of Python, Flask, and MySQL allows for real-time updates, responsive design, and user-friendly navigation.

First introduced in 1991 by Guido van Rossum, Python has since become one of the most widely used programming languages, celebrated for its versatility and extensive community support. Its use in the Airline Reservation System demonstrates its ability to handle complex tasks, ensuring an efficient, reliable, and scalable solution for ticket booking and management.

1.2 SQL (STRUCTURED QUERY LANGUAGE)

SQL is a special-purpose programming language designed for managing data held in a relational database management system (RDBMS) or for stream processing in a relational data stream management system (RDSMS). Originally based on relational algebra and tuple relational calculus, SQL supports operations such as data insertion, querying, updating, deletion, schema creation, modification, and data access control.

We are using MySQL for our project: MySQL (My Structured Query Language) is a powerful, open-source, server-based database management system widely used for efficient and scalable data handling. It is integrated into our Airline Reservation System to store and manage passenger details, flight schedules, reservations, and administrative information.

MySQL's robust performance, scalability, and support for concurrent operations enable real-time updates to flight availability and bookings. Its ability to handle complex queries and ensure data integrity makes it an ideal choice for a dynamic and reliable reservation system. By leveraging MySQL, the system achieves seamless database management, enhancing efficiency and delivering a consistent user experience.

CHAPTER – 2

SYSTEM SPECIFICATION

2.1 HARDWARE AND SOFTWARE SPECIFICATIONS

The following hardware and software are required:

criterion	Description
OS version	Microsoft® Windows® 7/8/10 (32-bit or 64-bit) Mac® OS X® 10.10 (Yosemite) or higher, up to 10.13 (macOS High Sierra) GNOME or KDE desktop Linux (64 bit capable of running 32-bit applications)(GNU C Library (glibc) 2.19+)
RAM	3 GB RAM minimum, 8 GB RAM recommended; plus 1 GB for the Android Emulator
Disk space	2 GB of available disk space minimum, 4 GB recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
Python version	Python 3.13.0
Screen resolution	1280×800 minimum screen resolution

CHAPTER – 3

SOFTWARE DESCRIPTION:

3.1 Python

Python is a versatile, high-level programming language known for its simplicity, readability, and broad applicability. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming, making it an ideal choice for developing web-based applications like the Airline Reservation System. Python's extensive library support ensures efficient integration with databases, frameworks, and visualization tools, enabling seamless backend and frontend development.

3.2 Flask

Flask is a lightweight, open-source Python web framework designed for building dynamic and interactive web applications. Known for its simplicity and flexibility, Flask allows developers to create robust applications with minimal effort. Unlike larger frameworks, Flask does not impose extensive boilerplate code, enabling developers to focus on functionality and customization. Flask eliminates the need for extensive HTML or JavaScript coding by integrating seamlessly with templates and libraries, simplifying the development process. Features such as its built-in development server, routing capabilities, and easy integration with databases make Flask an ideal choice for the Airline Reservation System. The framework ensures real-time booking functionalities and efficient database interactions, providing users with a seamless and responsive experience.

3.3 Visual Studio Code (VS Code)

VS Code is a lightweight yet powerful source code editor developed by Microsoft. It supports a wide range of programming languages, including Python, and offers features like intelligent code completion, debugging, and Git integration. The editor's flexibility, combined with its vast library of extensions, provides an optimal environment for developing and testing applications. It was used in the Airline

Reservation System project for writing, debugging, and maintaining Python and Flask code.

3.4 FEATURES

The following features were utilized in the development of the Airline Reservation System:

- Seamless Python and Flask integration for backend and frontend functionality.

- A user-friendly interface with real-time updates for flight schedules and seat availability.

- Simplified debugging and code management through VS Code's built-in tools.

- Database management using MySQL for efficient storage and retrieval of reservations, train details, and passenger information.

- Flexible and modular design with Flask, allowing easy customization and scalability.

- Cross-platform support, ensuring compatibility on Windows, macOS, and Linux systems.

Together, these tools and technologies provided a robust foundation for building an efficient and user-centric airplane reservation system.

CHAPTER – 4

PROJECT DESCRIPTION

The **Airline Reservation System** is a web-based application developed using **Python** and **Flask**, designed to simplify airplane ticket booking and management. It allows users to search for airplane, book tickets, view reservations, and cancel bookings with real-time updates on seat availability. The system is backed by a MySQL database that stores airplane schedules, passenger data, and reservation details.

It offers a user-friendly interface for both passengers and administrators, enabling efficient management of bookings. The platform also includes features for viewing all reservations and contacting the supervisor for assistance. The project aims to enhance user experience and optimize airplane reservation operations.

4.1 MODULE DESCRIPTION

1. Home
2. Book Ticket
3. Reservation Page
4. Cancel Ticket
5. Contact Supervisor
6. Passenger
7. Train

4.1.1 HOME

The Home module displays a list of available trains, their details, and their current seat availability. Users can easily navigate through the platform and access other functionalities like booking tickets, viewing reservations, and more.

4.1.2 BOOK TICKET

This module allows passengers to book tickets by selecting a train, entering passenger details, and choosing available seats. It enables users to complete their booking and view confirmation details such as seat number and train schedule.

4.1.3 RESERVATION PAGE

The Reservation Page provides users with an overview of all their active 12 bookings. Passengers can view detailed information about each reservation, including train name, seat number, and travel dates.

4.1.4 CANCEL TICKET

This module allows passengers to cancel an existing reservation. Users can search for their reservation by ID and remove it from the system if necessary. It also updates the seat availability in real-time.

4.1.5 CONTACT SUPERVISOR

The Contact Supervisor module provides users with the contact information of the supervisor, including name, phone number, and email, for assistance with any booking or system-related issues.

4.1.6 PASSENGER

The Passenger module allows passengers to create profiles, store personal information, and manage their bookings, including adding new passengers for ticket reservations.

4.1.7 AIRLINE

The Train module enables administrators to manage and update the train schedules, available seats, and other relevant details in the system. It ensures that real-time data is available for passengers to make informed decisions.

CHAPTER – 5

IMPLEMENTATION

5.1 SOURCE CODE

Home.html:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>ENVY Airlines</title>
    <link rel="icon" type="image/png" href="{{ url_for('static', filename='logo1.png') }}">
    <link href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@500;700&family=Roboto:wght@300;400;700&display=swap" rel="stylesheet">
    <style>
        body {
            margin: 0;
            font-family: 'Roboto', sans-serif;
            background-color: white;
            color: #333;
        }
        header {
            background-image: url("{{ url_for('static', filename='BG.png') }}");
            background-size: cover;
            background-position: fill;
            color: white;
```

```css
    height: 100vh;

    display: flex;

    flex-direction: column;

    justify-content: center;

    align-items: center;

    text-align: center;

    text-shadow: 0 2px 10px rgba(0, 0, 0, 0.5);

}

header h1 {

    font-family: 'Playfair Display', serif;

    font-size: 3rem;

    margin: 0;

}

header p {

    font-size: 1.2rem;

    margin-top: 10px;

}

.header-buttons {

    margin-top: 30px;

    display: flex;

    gap: 15px;

}

.header-buttons a {

    text-decoration: none;

    color: white;

    padding: 15px 30px;

    background-color: rgba(0, 0, 0, 0.7);

    border-radius: 5px;
```

```css
        transition: background-color 0.3s ease;

        font-weight: bold;

        text-transform: uppercase;

    }

    .header-buttons a:hover {

        background-color: rgba(0, 0, 0, 0.9);

    }

    section {

        padding: 50px 20px;

        text-align: center;

    }

    section h2 {

        font-family: 'Playfair Display', serif;

        font-size: 2.5rem;

        margin-bottom: 20px;

    }

    .features {

        display: flex;

        justify-content: center;

        gap: 30px;

        flex-wrap: wrap;

    }

    .feature {

        max-width: 300px;

        text-align: left;

    }

    .feature img {

        width: 100%;
```

```css
      border-radius: 10px;
    }
    .feature h3 {
      font-size: 1.5rem;
      margin: 15px 0 10px;
    }
    .feature p {
      font-size: 1rem;
      line-height: 1.6;
    }
    .footer {
      position: absolute;
      bottom: 20px;
      width: 100%;
      text-align: center;
      font-size: 0.9rem;
      color: rgb(255, 255, 255);
    }
    img{
      max-width: 30px;
    }
  </style>
</head>
<body>
  <header>
    <h1>Welcome, {{ name }}</h1>
    <p>Explore the world with ease and luxury</p>
    <div class="header-buttons">
```
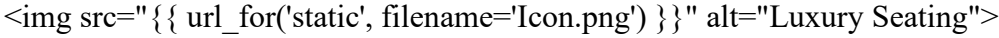
```html
        <a href="/search">Search Flights</a>
        <a href="/mybookings">My Bookings</a>
    </div>
</header>
<section>
    <h2>Your Travel Awaits</h2>
    <div class="features">
        <div class="feature">
            <img src="{{ url_for('static', filename='Icon.png') }}" alt="Luxury Seating">
            <h3>Luxury Seating</h3>
            <p>Relax in spacious and comfortable seats designed for your utmost convenience.</p>
        </div>
        <div class="feature">
            <img src="{{ url_for('static', filename='Icon2.png') }}" alt="Secure Booking">
            <h3>Secure Booking</h3>
            <p>Experience fast and secure booking options for a hassle-free journey.</p>
        </div>
        <div class="feature">
            <img src="{{ url_for('static', filename='icon1.png') }}" alt="Premium Services">
            <h3>Premium Services</h3>
            <p>Enjoy exclusive in-flight services tailored to make your journey exceptional.</p>
        </div>
    </div>
    <div class="footer">
        <p>&copy; 2024 ENVY Studioz. All rights reserved.</p>
    </div>
</section>
```
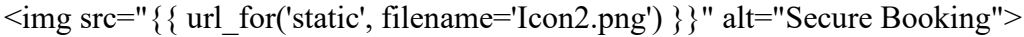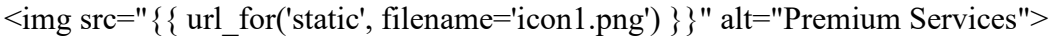
```
</body>

</html>


Landing.html:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>ENVY Airlines</title>

    <link rel="icon" type="image/png" href="{{ url_for('static', filename='logo1.png') }}">

    <link
href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@500;700&family=Robo
to:wght@300;400;700&display=swap" rel="stylesheet">

    <style>

        body {

            margin: 0;

            font-family: 'Roboto', sans-serif;

            background-image: url("{{ url_for('static', filename='BG.png') }}");

            background-size: cover;

            color: white;

            display: flex;

            flex-direction: column;

            align-items: center;

            justify-content: center;

            height: 100vh;

            text-align: center;

        }

        h1 {
```

```css
    font-family: 'Playfair Display', serif;

    font-size: 3rem;

    margin-bottom: 10px;

    text-shadow: 0 2px 10px rgba(0, 0, 0, 0.5);

}

h2{

    font-family: 'Playfair Display',serif;

    font-size:1.3rem;

    text-shadow: 0 2px 10px rgba(0, 0, 0, 0.5);

}

.links {

    margin-top: 30px;

    display: flex;

    gap: 20px;

}

a {

    text-decoration: none;

    color: white;

    background-color: rgba(0, 0, 0, 0.7);

    padding: 15px 30px;

    border-radius: 5px;

    transition: background-color 0.3s ease, transform 0.2s ease;

    font-weight: bold;

    text-transform: uppercase;

    font-size: 1rem;

}

a:hover {

    background-color: rgba(0, 0, 0, 0.9);
```

```html
          transform: scale(1.05);

        }

        .footer {

          position: absolute;

          bottom: 20px;

          width: 100%;

          text-align: center;

          font-size: 0.9rem;

          color: rgb(255, 255, 255);

        }

    </style>

</head>

<body>

    <h1>Welcome to the ENVY Airlines</h1>

    <h2>Experience luxury in the skies</h2>

    <div class="links">

      <a href="/login">Login</a>

      <a href="/register">Register</a>

    </div>

    <div class="footer">

      <p>&copy; 2024 ENVY Studioz. All rights reserved.</p>

    </div>

</body>

</html>
```

Login.html:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login - ENVY Airlines</title>
    <link rel="icon" type="image/png" href="{{ url_for('static', filename='logo1.png') }}">
    <link href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@500;700&family=Roboto:wght@300;400;700&display=swap" rel="stylesheet">
    <style>
        body {
            margin: 0;
            font-family: 'Roboto', sans-serif;
            background-image: url("{{ url_for('static', filename='BG1.png') }}");
            background-size: cover;
            color: white;
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
        }
        .login-container {
            background: rgba(0, 0, 0, 0.7);
            padding: 30px;
            border-radius: 10px;
            text-align: center;
            width: 400px;
```

```css
        }
        .login-container h2 {
            font-family: 'Playfair Display', serif;
            font-size: 2rem;
            margin-bottom: 20px;
        }
        .login-container form {
            display: flex;
            flex-direction: column;
            gap: 15px;
        }
        .login-container input {
            padding: 12px;
            font-size: 16px;
            border: none;
            border-radius: 5px;
            outline: none;
        }
        .login-container input:focus {
            border-color: gold;
            box-shadow: 0 0 8px gold;
        }
        .login-container button {
            padding: 12px;
            background-color: gold;
            color: black;
            font-size: 16px;
            border: none;
```

```css
      border-radius: 5px;

      cursor: pointer;

      transition: background-color 0.3s ease;

    }

    .login-container button:hover {

      background-color: #e0b400;

    }

    .login-container a {

      color: gold;

      text-decoration: none;

      font-size: 14px;

    }

    .login-container a:hover {

      text-decoration: underline;

    }

    .error {

      color: red;

      font-size: 14px;

    }

    .footer {

      position: absolute;

      bottom: 20px;

      width: 100%;

      text-align: center;

      font-size: 0.9rem;

      color: rgb(0, 0, 0);

    }

  </style>
```

```html
    </head>
    <body>
        <div class="login-container">
            <h2>Login</h2>
            <form method="POST" action="/login">
                <input type="text" name="name" placeholder="Name" required>
                <input type="password" name="password" placeholder="Password" required>
                <button type="submit">Login</button>
            </form>
            <p>
                Don't have an account? <a href="/register">Register</a>
            </p>
        </div>
        <div class="footer">
            <p>&copy; 2024 ENVY Studioz. All rights reserved.</p>
        </div>
    </body>
</html>
```

Register.html:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Register - ENVY Airlines</title>
    <link rel="icon" type="image/png" href="{{ url_for('static', filename='logo1.png') }}">
    <link href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@500;700&family=Roboto:wght@300;400;700&display=swap" rel="stylesheet">
    <style>
        body {
            margin: 0;
            font-family: 'Roboto', sans-serif;
            background-image: url("{{ url_for('static', filename='BG1.png') }}");
            background-size: cover;
            color: white;
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
        }
        .register-container {
            background: rgba(0, 0, 0, 0.7);
            padding: 40px;
            border-radius: 10px;
            width: 400px;
            text-align: center;
```

```css
    box-shadow: 0px 4px 20px rgba(0, 0, 0, 0.6);
}
.register-container h2 {
    font-family: 'Playfair Display', serif;
    font-size: 2rem;
    margin-bottom: 20px;
}
.register-container form {
    display: flex;
    flex-direction: column;
    gap: 15px;
}
.register-container input {
    padding: 12px;
    font-size: 16px;
    border: 1px solid #ccc;
    border-radius: 5px;
    outline: none;
    background: rgba(255, 255, 255, 0.9);
    color: #333;
}
.register-container input::placeholder {
    color: #666;
}
.register-container input:focus {
    border-color: gold;
    box-shadow: 0 0 8px gold;
}
```

```css
.register-container button {
    padding: 12px;
    background-color: gold;
    color: black;
    font-size: 16px;
    font-weight: normal;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    transition: background-color 0.3s ease, transform 0.2s ease;
}
.register-container button:hover {
    background-color: #e0b400;
    transform: scale(1.05);
}
.register-container a {
    color: gold;
    text-decoration: none;
    font-size: 14px;
    transition: color 0.3s ease;
}
.register-container a:hover {
    text-decoration: underline;
}
.footer {
    position: absolute;
    bottom: 20px;
    width: 100%;
```

```html
        text-align: center;

        font-size: 0.9rem;

        color: rgb(0, 0, 0);

      }

    </style>

  </head>

  <body>

    <div class="register-container">

      <h2>Create an Account</h2>

      <form action="/register" method="POST">

        <input type="text" name="UserName" placeholder="UserName" required>

        <input type="email" name="email" placeholder="Email" required>

        <input type="password" name="password" placeholder="Create a Password" required>

        <input type="password" name="confirm_password" placeholder="Confirm Password"
required>

        <button type="submit">Register</button>

      </form>

      <p>

        Already have an account? <a href="/login">Login</a>

      </p>

    </div>

    <div class="footer">

      <p>&copy; 2024 ENVY Studioz. All rights reserved.</p>

    </div>

  </body>

</html>
```

Search.html:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Search Flights</title>
    <link rel="icon" type="image/png" href="{{ url_for('static', filename='logo1.png') }}">
    <link href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@500;700&family=Roboto:wght@300;400;700&display=swap" rel="stylesheet">
    <style>
        body {
            margin: 0;
            font-family: 'Roboto', sans-serif;
            background: url("{{ url_for('static', filename='BG1.png') }}");
            background-size: cover;
            color: white;
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
        }
        .search-container {
            background: rgba(0, 0, 0, 0.7);
            padding: 40px;
            border-radius: 10px;
            text-align: center;
            width: 400px;
```

```css
    }
    .search-container h1 {
      font-family: 'Playfair Display', serif;
      font-size: 2rem;
      margin-bottom: 20px;
    }
    .search-container form {
      display: flex;
      flex-direction: column;
      gap: 15px;
    }
    .search-container input {
      padding: 12px;
      font-size: 16px;
      border: none;
      border-radius: 5px;
      outline: none;
    }
    .search-container input:focus {
      border-color: gold;
      box-shadow: 0 0 8px gold;
    }
    .search-container button {
      padding: 12px;
      background-color: gold;
      color: black;
      font-size: 16px;
      border: none;
```

```css
      border-radius: 5px;

      cursor: pointer;

      transition: background-color 0.3s ease;

    }

    .search-container button:hover {

      background-color: #e0b400;

    }

    .search-container a {

      color: gold;

      text-decoration: none;

      font-size: 14px;

    }

    .search-container a:hover {

      text-decoration: underline;

    }

    .footer {

      position: absolute;

      bottom: 20px;

      width: 100%;

      text-align: center;

      font-size: 0.9rem;

      color: rgb(0, 0, 0);

    }

  </style>

</head>

<body>

  <div class="search-container">

    <h1>Search Flights</h1>
```

```
    <form method="POST" action="/search">

        <input type="text" name="from_location" placeholder="From" required>

        <input type="text" name="to_location" placeholder="To" required>

        <input type="date" name="travel_date" required>

        <button type="submit">Search</button>

    </form>

  </div>

  <div class="footer">

    <p>&copy; 2024 ENVY Studioz. All rights reserved.</p>

  </div>

</body>

</html>
```

Flight.html:

```
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

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

    <title>Flight Results</title>

    <link rel="icon" type="image/png" href="{{ url_for('static', filename='logo1.png') }}">

    <link
href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@500;700&family=Robo
to:wght@300;400;700&display=swap" rel="stylesheet">

    <style>

        body {

            margin: 0;

            font-family: 'Roboto', sans-serif;

            background: url("{{ url_for('static', filename='BG1.png') }}");
```

```css
    background-size: cover;

    color: white;

    display: flex;

    padding: 0 15px;

    align-items: center;

    min-height: 100vh;

}

.results-container {

    background: rgba(0, 0, 0, 0.7);

    padding: 40px;

    border-radius: 10px;

    width: 100%;

    text-align: center;

}

.results-container h1 {

    font-family: 'Playfair Display', serif;

    font-size: 2.5rem;

    margin-bottom: 30px;


}

table {

    width: 100%;

    border-collapse: collapse;

    margin: 20px 0;

}

th, td {

    padding: 12px;

    text-align: left;
```

```css
    border-bottom: 1px solid rgba(255, 255, 255, 0.3);
}
th {
    font-family: 'Playfair Display', serif;
    color: gold;
    font-size: 1.2rem;
}
td {
    font-size: 1rem;
    color: white;
}
a {
    color: gold;
    text-decoration: none;
    font-size: 1rem;
}


.book-now-btn {
padding: 8px 16px; /* Reduced padding to make the button smaller */
background-color: gold;
color: black;
font-size: 14px; /* Slightly smaller text */
border: 2px solid gold; /* Added border and increased thickness */
border-radius: 5px;
cursor: pointer;
transition: background-color 0.3s ease;
}
```

```html
      .book-now-btn:hover {

        background-color: #e0b400;

      }

      .footer {

        position: absolute;

        bottom: 20px;

        width: 100%;

        text-align: center;

        font-size: 0.9rem;

        color: rgb(0, 0, 0);

      }

    </style>

  </head>

  <body>

    <div class="results-container">

      <h1>Flight Results</h1>

      <table>

        <tr>

          <th>Flight Number</th>

          <th>From</th>

          <th>To</th>

          <th>Departure</th>

          <th>Arrival</th>

          <th>Base Price</th>

          <th>Book</th>

        </tr>

        {% for flight in flights %}

        <tr>
```

```html
            <td>{{ flight.flight_number }}</td>
            <td>{{ flight.from_location }}</td>
            <td>{{ flight.arrival_location }}</td>
            <td>{{ flight.departure_date }} {{ flight.departure_time }}</td>
            <td>{{ flight.arrival_date }} {{ flight.arrival_time }}</td>
            <td>${{ flight.base_price }}</td>
            <td><a href="/checkout/{{ flight.flight_number }}" class="book-now-btn">Book Now</a></td>
        </tr>
        {% endfor %}
    </table>
  </div>
  <div class="footer">
    <p>&copy; 2024 ENVY Studioz. All rights reserved.</p>
  </div>
</body>
</html>
```

Checkout.html:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Checkout</title>
    <link rel="icon" type="image/png" href="{{ url_for('static', filename='logo1.png') }}">
    <link href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@500;700&family=Roboto:wght@300;400;700&display=swap" rel="stylesheet">
    <style>
        body {
            background: url("{{ url_for('static', filename='BG1.png') }}");
            color: #f7f7f7;
            font-family: 'Roboto', sans-serif;
            padding: 0;
            margin: 0;
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
        }
        .checkout-container {
            background: rgba(0, 0, 0, 0.7);
            padding: 40px;
            border-radius: 10px;
            text-align: center;
            width: 500px;
```

```css
    }
    h1 {
      font-family: 'Playfair Display', serif;
      color: white;
      font-size: 2rem;
      margin-bottom: 20px;
    }
    .flight-details {
      background: linear-gradient(135deg, #222, #444);
      padding: 20px;
      margin-bottom: 30px;
      border-radius: 8px;
      box-shadow: 0 4px 8px rgba(0, 0, 0, 0.3);
      color: #ffd700;
      font-size: 1.1rem;
      text-align: left;
    }
    .flight-details p {
      margin: 8px 0;
    }
    .flight-details .label {
      font-weight: bold;
      color: gold;
    }
    .flight-details .detail {
      color: white;
    }
    form {
```

```css
    display: flex;

    flex-direction: column;

    gap: 5px;

}

label {

    font-weight: bold;

    color: #f7f7f7;

    margin-top: 8px;

    text-align: left;

}

input, select, button {

    padding: 12px;

    font-size: 16px;

    border-radius: 5px;

    outline: none;

    border: none;

}

input, select {

    background-color: #222;

    color: #f7f7f7;

}

input:focus, select:focus {

    border-color: gold;

    box-shadow: 0 0 8px gold;

}

button {

    padding: 12px;

    background-color: gold;
```

```
            color: black;

            margin-top:25px;

            font-size: 16px;

            border-radius: 5px;

            cursor: pointer;

            transition: background-color 0.3s ease;

        }

        button:hover {

            background-color: #e0b400;

        }

        .footer {

            position: absolute;

            bottom: 20px;

            width: 100%;

            text-align: center;

            color: #000000;

            font-size: 0.9rem;

        }

    </style>

</head>

<body>

    <div class="checkout-container">

        <h1>Checkout</h1>


        <div class="flight-details">

            <p><span class="label">From:</span> <span class="detail">{{ flight[0] }}</span></p>

            <p><span class="label">To:</span> <span class="detail">{{ flight[1] }}</span></p>

            <p><span class="label">Departure Date:</span> <span class="detail">{{ flight[2] }}</span></p>
```

```html
        <p><span class="label">Departure Time:</span> <span class="detail">{{ flight[3]
}}</span></p>
        <p><span class="label">Base Price:</span> <span class="detail">${{ flight[6]
}}</span></p>
    </div>

    <form method="POST">
        <label for="num_passengers">Number of Passengers:</label>
        <input type="number" id="num_passengers" name="num_passengers" min="1" required>

        <label for="travel_class">Class of Travel:</label>
        <select id="travel_class" name="travel_class" required>
            <option value="Economy">Economy</option>
            <option value="Business">Business</option>
            <option value="First">First</option>
        </select>

        <button type="submit">Book Now</button>
    </form>
  </div>

  <div class="footer">
    <p>&copy; 2024 ENVY Studioz. All rights reserved.</p>
  </div>
</body>
</html>
```

Mybookings.html:

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>My Bookings</title>
    <link rel="icon" type="image/png" href="{{ url_for('static', filename='logo1.png') }}">
    <link href="https://fonts.googleapis.com/css2?family=Playfair+Display:wght@500;700&family=Roboto:wght@300;400;700&display=swap" rel="stylesheet">
    <style>
        body {
            margin: 0;
            font-family: 'Roboto', sans-serif;
            background: url("{{ url_for('static', filename='BG1.png') }}");
            background-size: cover;
            color: white;
            display: flex;

            align-items: center;
            min-height: 100vh;
            padding: 0 15px; /* Added padding to prevent overflow */
            box-sizing: border-box; /* Ensures padding doesn't cause overflow */
        }
        .results-container {
            background: rgba(0, 0, 0, 0.7);
            padding: 30px;
            border-radius: 10px;
```

```css
    width: 100%;

    text-align: center;

    box-sizing: border-box;

    overflow: hidden; /* Prevents any overflow beyond the container */

}

.results-container h1 {

    font-family: 'Playfair Display', serif;

    font-size: 2.5rem;

    margin-bottom: 30px;

}

table {

    width: 100%;

    border-collapse: collapse;

    margin: 20px 0;

}

th, td {

    padding: 12px;

    text-align: left;

    border-bottom: 1px solid rgba(255, 255, 255, 0.3);

}

th {

    font-family: 'Playfair Display', serif;

    color: gold;

    font-size: 1.2rem;

}

td {

    font-size: 1rem;

    color: white;
```

```
        }
        .footer {
            position: absolute;
            bottom: 20px;
            width: 100%;
            text-align: center;
            font-size: 0.9rem;
            color: rgb(0, 0, 0);
        }
    </style>
</head>
<body>

    <div class="results-container">
        <h1>My Bookings</h1>
        <table>
            <tr>
                <th>Flight Number</th>
                <th>From</th>
                <th>To</th>
                <th>Departure</th>
                <th>Arrival</th>
                <th>Passengers</th>
                <th>Class</th>
                <th>Total Price</th>
            </tr>
            {% for booking in bookings %}
            <tr>
```

```html
        <td>{{ booking.flight_number }}</td>
        <td>{{ booking.from_location }}</td>
        <td>{{ booking.to_location }}</td>
        <td>{{ booking.departure_date }} {{ booking.departure_time }}</td>
        <td>{{ booking.arrival_date }} {{ booking.arrival_time }}</td>
        <td>{{ booking.num_passengers }}</td>
        <td>{{ booking.travel_class }}</td>
        <td>${{ booking.total_price }}</td>
      </tr>
      {% endfor %}
    </table>
  </div>

  <div class="footer">
    <p>&copy; 2024 ENVY Studioz. All rights reserved.</p>
  </div>

</body>
</html>
```

PYTHON:

```python
from flask import Flask, render_template, request, redirect, session
from flask_mysqldb import MySQL
import MySQLdb.cursors
from datetime import datetime

app = Flask(__name__)
app.config['MYSQL_HOST'] = 'localhost'
app.config['MYSQL_USER'] = 'root'
app.config['MYSQL_PASSWORD'] = 'charanraj@envy@14'
app.config['MYSQL_DB'] = 'airline_reservation'

mysql = MySQL(app)

@app.route('/')
def landing_page():
    return render_template('landing.html')

@app.route('/register', methods=['GET', 'POST'])
def register():
    if request.method == 'POST':
        name = request.form['UserName']
        password = request.form['password']
        cursor = mysql.connection.cursor()
        cursor.execute('INSERT INTO USER (name, password) VALUES (%s, %s)', (name, password))
        mysql.connection.commit()
        return redirect('/')
```

```python
    return render_template('register.html')


@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        name = request.form['name']
        password = request.form['password']
        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
        cursor.execute('SELECT * FROM USER WHERE name = %s AND password = %s', (name,
password))
        user = cursor.fetchone()
        if user:
            session['loggedin'] = True
            session['user_id'] = user['user_id']
            session['name'] = user['name']
            return redirect('/home')
        else:
            return "Invalid Credentials"
    return render_template('login.html')


@app.route('/home')
def home():
    if 'loggedin' in session:
        return render_template('home.html', name=session['name'])
    return redirect('/')


@app.route('/search', methods=['GET', 'POST'])
def search():
    if request.method == 'POST':
```

```python
        from_location = request.form['from_location']

        to_location = request.form['to_location']

        travel_date = request.form['travel_date']

        cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)

        cursor.execute('SELECT * FROM Flight WHERE from_location = %s AND arrival_location
= %s AND departure_date = %s',

                    (from_location, to_location, travel_date))

        flights = cursor.fetchall()

        return render_template('flight_results.html', flights=flights)

    return render_template('search.html')


@app.route('/checkout/<flight_number>', methods=['GET', 'POST'])

def checkout(flight_number):

    if 'user_id' not in session:

        return redirect('/login')


    user_id = session['user_id']


    if request.method == 'POST':

        num_passengers = int(request.form['num_passengers'])

        travel_class = request.form['travel_class']

        cursor = mysql.connection.cursor()

        cursor.execute(

            "SELECT from_location, arrival_location, departure_date, departure_time, arrival_date,
arrival_time, base_price "

            "FROM Flight WHERE flight_number = %s", (flight_number,)

        )

        flight = cursor.fetchone()
```

```python
    if not flight:
        return "Flight not found", 404

    base_price = flight[6]  # Index of base price in the query result
    total_price = base_price * num_passengers

    if travel_class == 'Business':
        total_price *= 2
    travel_class = request.form['travel_class']
    if travel_class == 'First':
        total_price *= 3
    travel_class = request.form['travel_class']
    cursor.execute(
        "INSERT INTO Bookings (user_id, flight_number, from_location, to_location, departure_date, departure_time, arrival_date, arrival_time, base_price, num_passengers, travel_class, total_price) "
        "VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)",
        (user_id, flight_number, flight[0], flight[1], flight[2], flight[3], flight[4], flight[5], base_price, num_passengers, travel_class, total_price)
    )
    mysql.connection.commit()
    cursor.close()
    return redirect('/home')
else:
    cursor = mysql.connection.cursor()
    cursor.execute(
        "SELECT from_location, arrival_location, departure_date, departure_time, arrival_date, arrival_time, base_price "
        "FROM Flight WHERE flight_number = %s", (flight_number,)
    )
```

```python
        flight = cursor.fetchone()
        cursor.close()
        if not flight:
            return "Flight not found", 404
        return render_template('checkout.html', flight=flight)
@app.route('/mybookings')
def my_bookings():
    if 'user_id' not in session:
        return redirect('/login')
    user_id = session['user_id']
    cursor = mysql.connection.cursor()
    cursor.execute(
        "SELECT flight_number, from_location, to_location, departure_date, departure_time, "
    arrival_date, arrival_time, base_price, num_passengers, travel_class, total_price "
        "FROM Bookings WHERE user_id = %s", (user_id,)
    )
    raw_bookings = cursor.fetchall()
    cursor.close()
    bookings = [
        {
            "flight_number": row[0],
            "from_location": row[1],
            "to_location": row[2],
            "departure_date": row[3],
            "departure_time": row[4],
            "arrival_date": row[5],
            "arrival_time": row[6],
            "base_price": row[7],
            "num_passengers": row[8],
```

```
        "travel_class": row[9],

        "total_price": row[10],

    }

    for row in raw_bookings

]

return render_template('mybookings.html', bookings=bookings)

if __name__ == '__main__':

    app.run(debug=True)
```

5.2 SCREENSHOTS

Welcome to the ENVY Airlines

Experience luxury in the skies

LOGIN   **REGISTER**

Login

Name

Password

Login

Don't have an account? Register

Create an Account

UserName

Email

Create a Password

Confirm Password

Register

Already have an account? Login

ENVY
STUDIOZ

Welcome, Chrn_raj

Explore the world with ease and luxury

SEARCH FLIGHTS   **MY BOOKINGS**
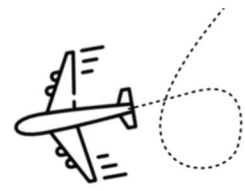
Search Flights

From

To

dd - mm - yyyy

Search
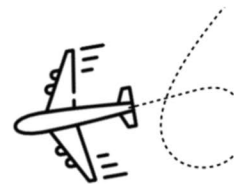
Flight Results

Flight Number	From	To	Departure	Arrival	Base Price	Book
AI104	Paris	Rome	2024-12-15 9:00:00	2024-12-15 11:30:00	$150.00	Book Now

Checkout

From: Paris
To: Rome
Departure Date: 2024-12-15
Departure Time: 9:00:00
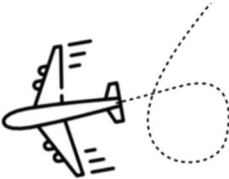**Base Price:** $150.00

Number of Passengers:

3

Class of Travel:

Economy

Book Now

My Bookings

Flight Number	From	To	Departure	Arrival	Passengers	Class	Total Price
AI103	Delhi	Mumbai	2024-12-10 12:00:00	2024-12-10 14:00:00	3	First	$900.00
AI103	Delhi	Mumbai	2024-12-10 12:00:00	2024-12-10 14:00:00	3	Business	$600.00
AI103	Delhi	Mumbai	2024-12-10 12:00:00	2024-12-10 14:00:00	4	First	$1200.00
AI103	Delhi	Mumbai	2024-12-10 12:00:00	2024-12-10 14:00:00	5	First	$1500.00
AI104	Paris	Rome	2024-12-15 9:00:00	2024-12-15 11:30:00	6	First	$2700.00
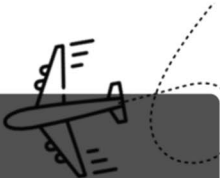
CHAPTER – 6

CONCLUSION AND FUTURE ENHANCEMENT

In conclusion, the Airline Reservation System simplifies and enhances the process of booking and managing flight tickets for customers and administrators. With the features developed in this project, users can check the availability of flights, register themselves, and book tickets efficiently. The system maintains a comprehensive database of customer details, bookings, and flight schedules, making the management process more streamlined and reliable.

In the future, this system can be enhanced with additional functionalities, such as real-time flight tracking, integration with payment gateways, and personalized recommendations based on user preferences. These improvements will further improve user experience and cater to the evolving needs of the airline industry. Thus, this project provides a foundation for a robust and scalable reservation system that can benefit both customers and airline operators.

CHAPTER-7

BIBLIOGRAPHY:

http://www.youtube.com

http://www.wikipedia.com

http://w3schools.com