**Pimpri Chinchwad University**

# blueBus - Online Bus Ticket Booking

Academic Year:

2024 - 2025

Under the Guidance of:

Prof Sanjay Dandele

Department Name: Master of Computer Applications

## Developed By:


1. Bansil Pabari [ 19 ] PCU24-MCA-5201
2. Ansh Gadoya [ 14 ] PCU24-MCA-5200

**PCET's PIMPRI CHINCHWAD UNIVERSITY,**

**Sate, Maval, Pune-412106**

## Certificate

*This is to certify that Mr. / Ms. _____ bearing PRN No _____ of MCA _____ semester ____ and Subject _____ has satisfactorily completed Mini Project on _____during the academic year _____.*

Signature of Guide

Signature of HOD

Signature of Internal Examiner

Signature of External Examiner

# ACKNOWLEDGEMENT

I would like to express my gratitude and sincere regards to the following people to whom I am grateful for their support and help without which I would not have been able to do this project entitled “blueBus”.

I would like to thanks **Dr. Kamlesh Meshram**, HOD, MCA, PCU for providing us with excellent facilities valuable guidance.

My special thanks to Prof. Sanjay Dandale, Internal Guide, whose cooperation and keen interest coupled with valuable suggestions help me in the completion of the project.

Last but not least, I am thankful to all Teaching and Non-Teaching staff of the college, whose names have not been mentioned here but have directly or indirectly helped me in the completion of this project.

**Ansh Gadoya**

## Abstract

This report introduces blueBus, a modern web-based platform designed to redefine the bus ticket booking experience. With the growing demand for online services in the travel industry, traditional systems face limitations in accessibility, usability, and efficiency. The blueBus platform aims to bridge these gaps through a streamlined, feature-rich system providing:

- Real-time seat availability and dynamic pricing,
- A wide network of bus operators and routes, and
- Secure payment options integrated with modern gateways.

The project followed a four-phase methodology: planning, iterative development, testing, and deployment. Developed using Python, Django, and MySQL, it ensures scalability, reliability, and user satisfaction. Key outcomes include automation of ticket booking, improved customer experience, and operational efficiency, contributing to a stronger market position.

# Index

1) Introduction……01
1.1) Problem Statement……..01
1.2) Objectives..……...01
1.3) Scope……01
1.4) Organization of the Report...……..02

2) Literature Survey/Review.……03
2.1) Overview of existing work……...03
2.2) Projects, or products reviewed…….....03
2.3) Identification of gaps..……...……03

3) System Analysis……..……04
3.1) Requirements Specification……...04
3.1.1) Functional Requirements..…….……04
3.1.2) Non-functional Requirements…….…….....06
3.2) Feasibility Study…….……..……..07
3.3) Tools and Technologies…….……07

4) Design……...08
4.1) System Architecture……08
4.2) Diagrams……..08
4.3) Database Design……11

5) Implementation……15
5.1) Description of Modules/Components……...15
5.2) Code Snippets……15

5.3) Screenshots of the Application/System...……....23

6) Testing……...36

6.1) Test Cases and Testing Methodology……...36

6.2) Results of Testing……....37

6.3) Bug Fixes/Issues Resolved……....40

7) Results and Discussion……41

7.1) Outcomes of the project……...41

7.2) Comparison with existing solutions……41

7.3) Challenges Faced……42

8) Conclusion and Future Scope……..43

8.1) Summary of the work……...43

8.2) Limitations……...…..43

8.3) Future Enhancements…….....43

References……..45

# 1. Introduction

This chapter provides the background, problem statement, objectives, scope, and organization of the report.

## 1.1 Background/Problem Statement

Bus travel remains a widely used mode of transportation, but traditional ticket booking systems are plagued with inefficiencies. Offline methods such as physical ticket counters are inconvenient and time-consuming. While existing online systems partially address these issues, many still lack essential features such as:

- Real-time seat availability,
- Transparent pricing,
- User-friendly interfaces, and
- Secure payment systems.

blueBus was conceptualized to address these gaps and provide a comprehensive solution for both users and operators.

## 1.2 Objectives of the Project

The primary objectives of blueBus are:

- To create a user-friendly platform for seamless online ticket booking.
- To ensure real-time updates on seat availability and dynamic pricing.
- To provide comprehensive network coverage, connecting diverse routes and operators.
- To integrate secure and reliable payment gateways.
- To improve operational efficiency for bus operators and travel agencies.

## 1.3 Scope of the Project

The initial focus of the project is on domestic routes within India, catering to a broad demographic of users. The system is designed to accommodate multiple operators, flexible schedules, and real-time booking updates. Future plans include international routes and advanced AI-driven features for user personalization.

## 1.4 Organization of the Report

This report is structured to detail the development of blueBus through the following chapters:

- Introduction: Provides background and objectives.
- Literature Survey: Reviews related work and identifies gaps.
- System Analysis: Details system requirements, feasibility, and tools used.
- Design: Includes system architecture, UML diagrams, and database design.
- Implementation: Explains the coding and key modules.
- Testing: Describes testing methods and results.
- Results and Discussion: Highlights outcomes and challenges.
- Conclusion: Summarizes the work and suggests future enhancements.

# 2. Literature Survey/Review

## 2.1 Overview of Existing Work

Existing manual bus ticket booking serve a significant market. However, many of these platforms fail to address critical user pain points like:

Complex user interaction,

Delays in real-time seat availability updates, and

Lack of knowledge of regional route integration for semi-urban and rural areas.

## 2.2 Projects or Products Reviewed

**RedBus:**

- Strengths: Extensive network coverage, user-friendly interface.
- Weaknesses: Limited focus on real-time updates, issues with transaction failures.

**EaseMyTrip:**

- Strengths: Affordable pricing options.
- Weaknesses: Insufficient route options and limited scalability.

**Traditional Systems:**

- Strengths: Trusted by users accustomed to offline methods.
- Weaknesses: Time-consuming and prone to human errors.

## 2.3 Identification of Gaps

The research revealed a need for:

- Seamless integration of real-time availability and pricing.
- Enhanced security for digital transactions.
- A broader network that includes regional routes.

# 3. System Analysis

## 3.1 Requirements Specification

Functional Requirements:

- User registration and profile management.
- Search functionality for buses based on route and schedule.
- Real-time updates for seat availability and pricing.
- Dynamic fare calculation based on seat selection.
- Secure payment gateway.

Non-functional Requirements:

- Scalability for handling thousands of concurrent users.
- High availability with minimal downtime.

## 3.2 Feasibility Study

Technical Feasibility:

- The project employs Django and MySQL, ensuring a robust and scalable backend. The use of JavaScript enhances the frontend interactivity.

Economic Feasibility:

- With low initial costs and high revenue potential through booking fees and advertisements, the system is economically viable.

Operational Feasibility:

- A user-friendly interface ensures ease of adoption for all demographics.

## 3.3 Tools and Technologies Used

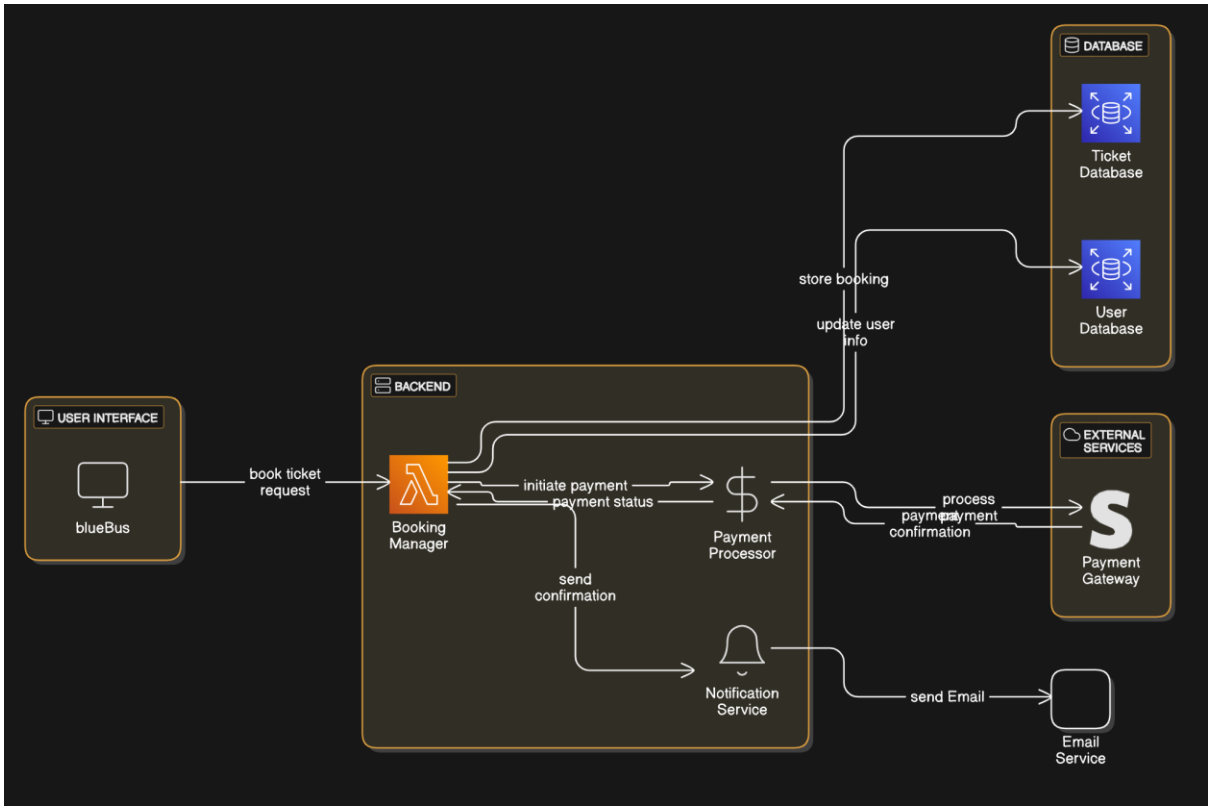
- Python and Django for backend development.
- MySQL for database management.
- HTML, CSS, and JavaScript for frontend design.

# 4. Design

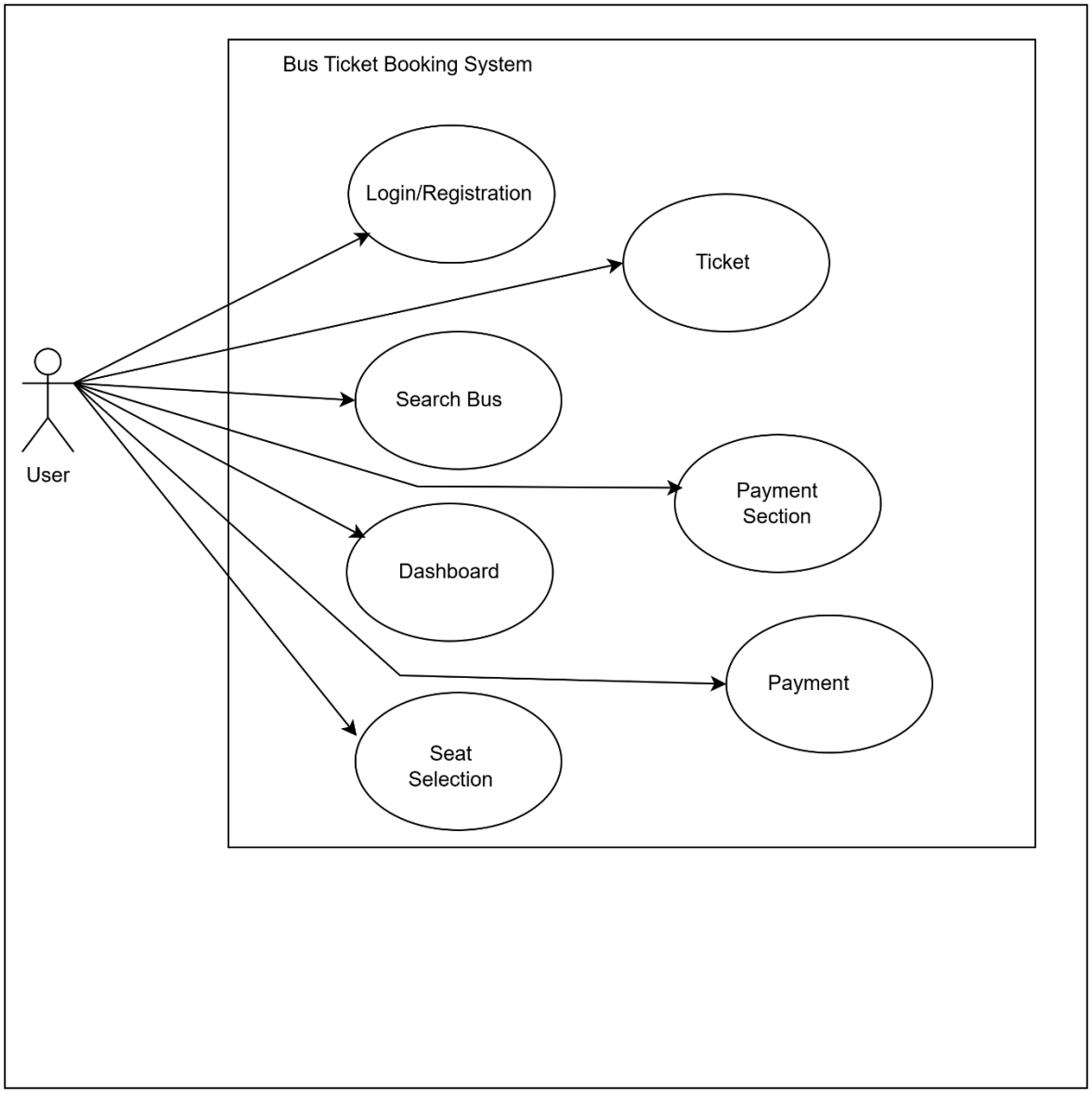
## 4.1 System Architecture

The architecture comprises:

- Frontend Layer: User interface for searching and booking tickets.
- Application Layer: Business logic for seat selection, pricing, and booking.
- Database Layer: Storage of user details, bookings, and payment data.



# 4.2 UML Diagrams – Use Case Diagram



## 4.3 Database Design

| User Login | | | | |
|---|---|---|---|---|
| Sr no. | Field Name | Datatype | Size | Constraint |
| 1 | Id | Int | 11 | primary key |
| 2 | username | varchar | 150 | not null |
| 3 | password | varchar | 250 | not null |
| 4 | is_superuser | tinyint | 1 | not null |
| 5 | email | varchar | 254 | not null |
| 6 | ast_login | datetime | 6 | not null |

| Ticket Booking | | | | |
|---|---|---|---|---|
| Sr no. | Field Name | Datatype | Size | Constraint |
| 1 | Id | bigint | 20 | primary key |
| 2 | bus_name | varchar | 50 | not null |
| 3 | from_source | varchar | 50 | not null |
| 4 | to_desti | varchar | 50 | not null |
| 5 | via_city | varchar | 50 | not null |
| 6 | from_time | time | 6 | not null |
| 7 | to_time | time | 6 | not null |
| 8 | departure_date | date | -- | not null |
| 9 | arrival_date | date | -- | not null |
| 10 | bus_ac | tinyint | 1 | not null |
| 11 | bus_nonac | tinyint | 1 | not null |
| 12 | bus_ac_check_text | varchar | 10 | not null |
| 13 | bus_sleeper_seater_text | varchar | 10 | not null |
| 14 | bus_price | int | 11 | not null |
| 15 | bus_seat | int | 11 | not null |
| 16 | bus_occupy_seat | int | 11 | not null |
| 17 | seater | tinyint | 1 | not null |
| 18 | sleeper | tinyint | 1 | not null |
| 19 | booked_seat_number | longtext | -- | |

# 5. Implementation

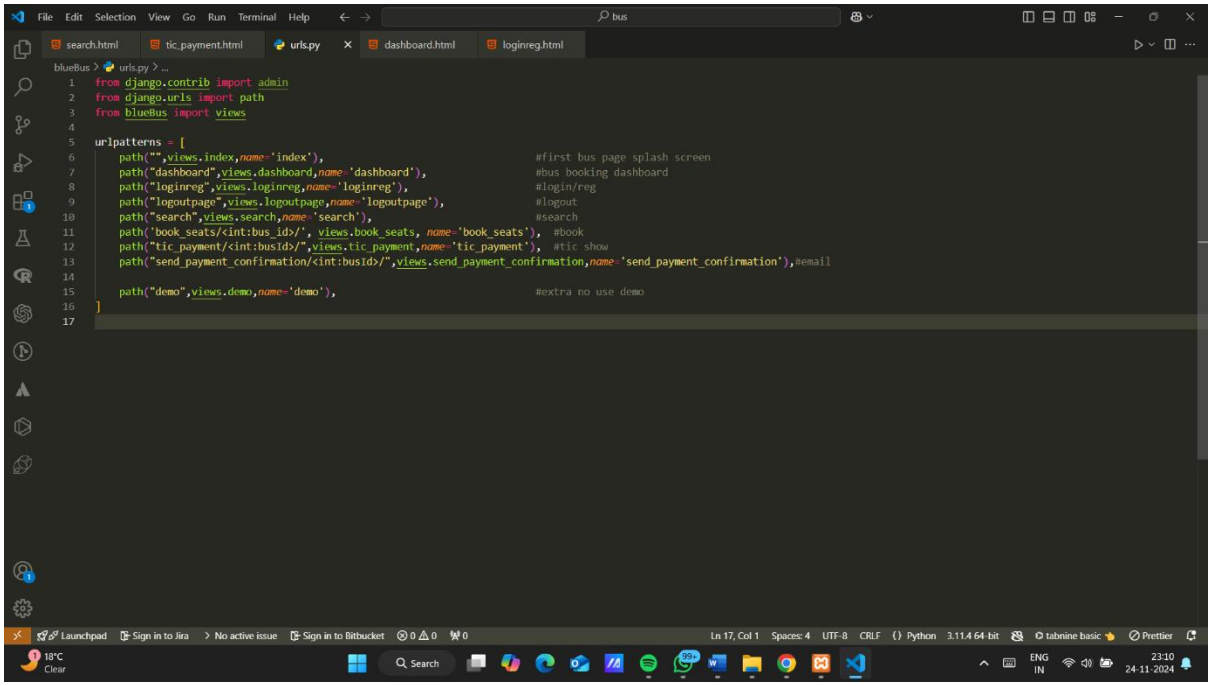
## 5.1 Description of Modules

The *blueBus* project comprises several interconnected modules designed to ensure seamless functionality and user experience:

1. User Management Module:
    - Handles user registration, login.
    - Ensures data security by encrypting sensitive information like passwords.
2. Search and Filter Module:
    - Users can search for buses by specifying origin, destination, and travel date.
    - Advanced filters enable narrowing down results by bus type AC / Non-AC , sleeper / seater.
3. Booking Module:
    - Real-time seat selection using interactive seat maps.
    - Dynamic fare calculation based on selected seats.
    - Automated email confirmation upon successful booking.
4. Payment Module:
    - Supports payment options like UPI.
    - Implements transaction tracking and error handling for failed payments.

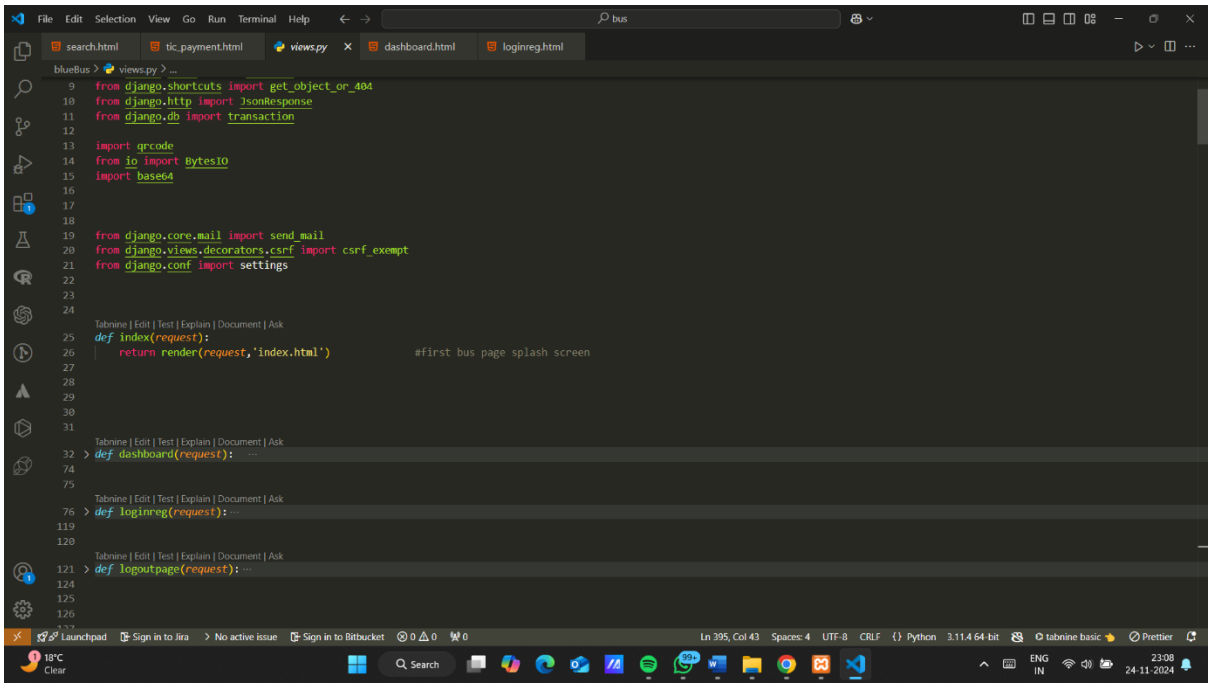
# 5.2 Code Snippets

## 1. urls.py



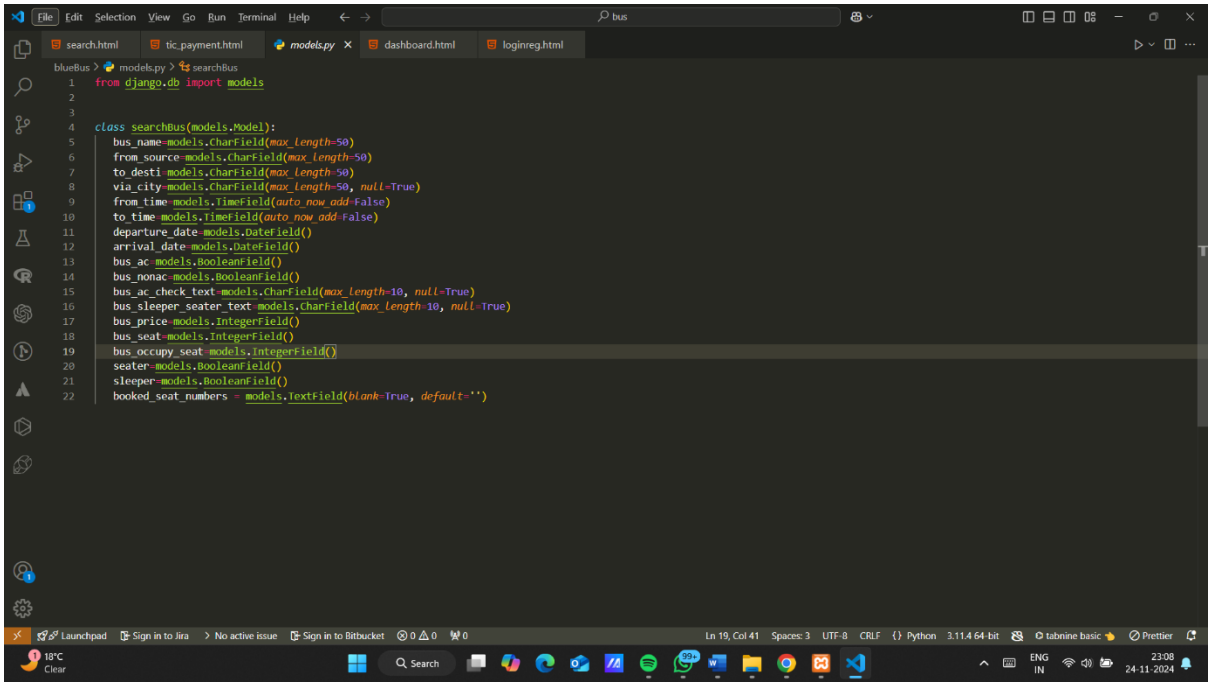
```python
from django.contrib import admin
from django.urls import path
from blueBus import views

urlpatterns = [
    path("",views.index,name='index'),                          #first bus page splash screen
    path("dashboard",views.dashboard,name='dashboard'),         #bus booking dashboard
    path("loginreg",views.loginreg,name='loginreg'),            #login/reg
    path("logoutpage",views.logoutpage,name='logoutpage'),      #logout
    path("search",views.search,name='search'),                  #search
    path('book_seats/<int:bus_id>/', views.book_seats, name='book_seats'),  #book
    path("tic_payment/<int:busId>/",views.tic_payment,name='tic_payment'),  #tic show
    path("send_payment_confirmation/<int:busId>/",views.send_payment_confirmation,name='send_payment_confirmation'),#email

    path("demo",views.demo,name='demo'),                        #extra no use demo
]
```

## 2. views.py



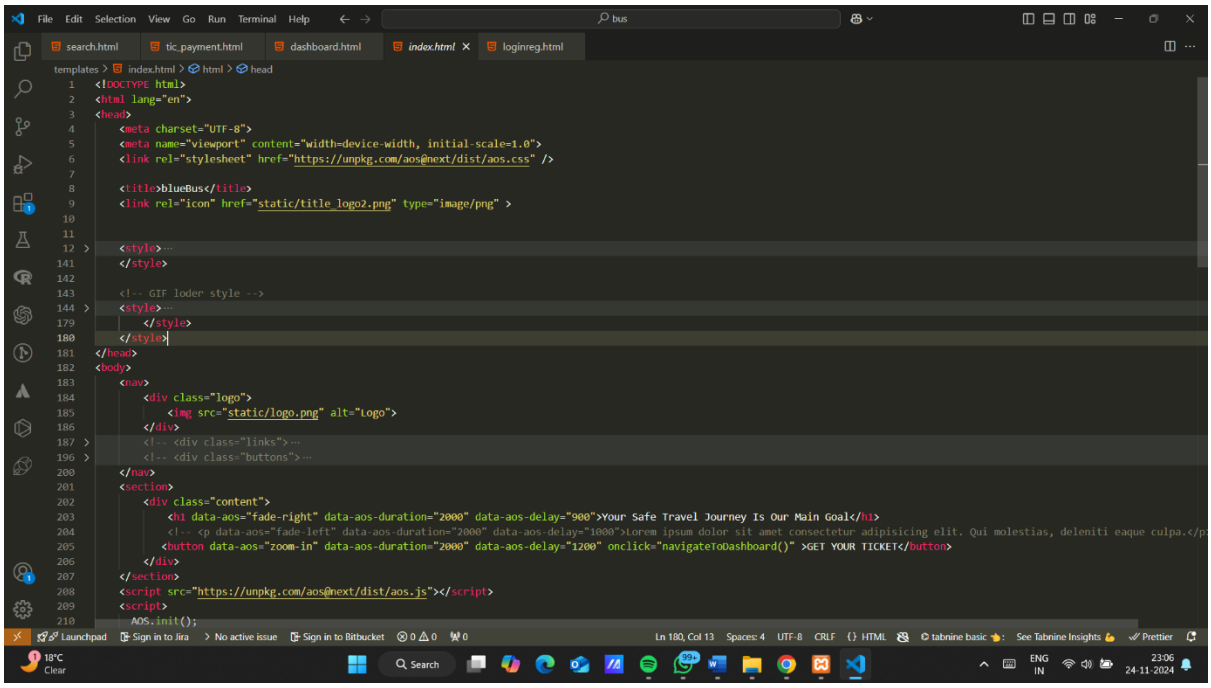
```python
from django.shortcuts import get_object_or_404
from django.http import JsonResponse
from django.db import transaction

import qrcode
from io import BytesIO
import base64



from django.core.mail import send_mail
from django.views.decorators.csrf import csrf_exempt
from django.conf import settings


Tabnine | Edit | Test | Explain | Document | Ask
def index(request):
    return render(request,'index.html')            #first bus page splash screen



Tabnine | Edit | Test | Explain | Document | Ask
def dashboard(request): ...


Tabnine | Edit | Test | Explain | Document | Ask
def loginreg(request): ...


Tabnine | Edit | Test | Explain | Document | Ask
def logoutpage(request): ...
```

## 3. models.py



```python
from django.db import models

class searchBus(models.Model):
    bus_name=models.CharField(max_length=50)
    from_source=models.CharField(max_length=50)
    to_desti=models.CharField(max_length=50)
    via_city=models.CharField(max_length=50, null=True)
    from_time=models.TimeField(auto_now_add=False)
    to_time=models.TimeField(auto_now_add=False)
    departure_date=models.DateField()
    arrival_date=models.DateField()
    bus_ac=models.BooleanField()
    bus_nonac=models.BooleanField()
    bus_ac_check_text=models.CharField(max_length=10, null=True)
    bus_sleeper_seater_text=models.CharField(max_length=10, null=True)
    bus_price=models.IntegerField()
    bus_seat=models.IntegerField()
    bus_occupy_seat=models.IntegerField()
    seater=models.BooleanField()
    sleeper=models.BooleanField()
    booked_seat_numbers = models.TextField(blank=True, default='')
```

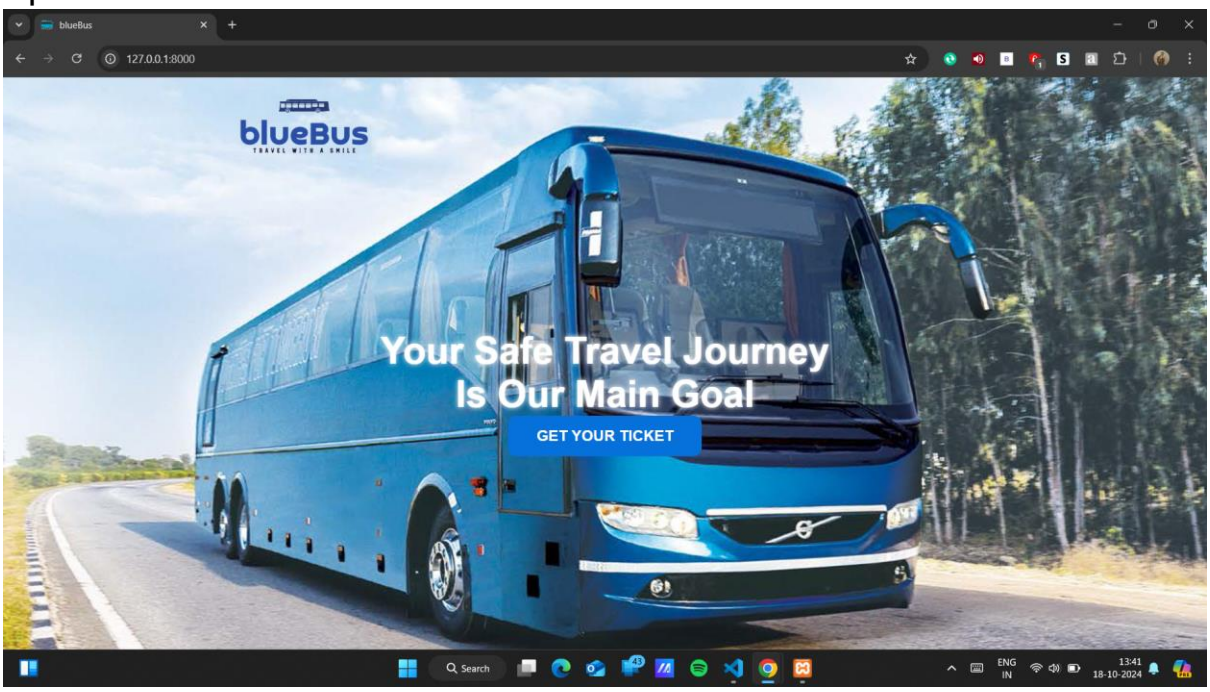
## 4. index.html



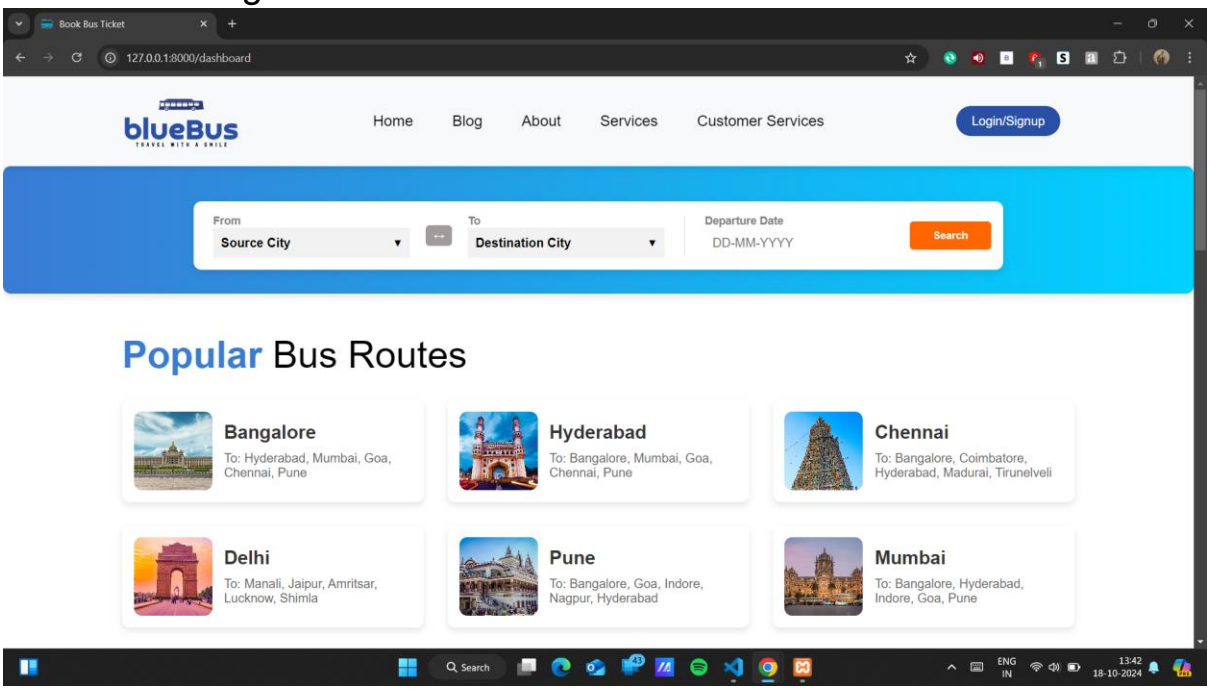
```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="https://unpkg.com/aos@next/dist/aos.css" />

    <title>blueBus</title>
    <link rel="icon" href="static/title_logo2.png" type="image/png" >


    <style>...
    </style>

    <!-- GIF loder style -->
    <style>...
        </style>
    </style>
</head>
<body>
    <nav>
        <div class="logo">
            <img src="static/logo.png" alt="Logo">
        </div>
        <!-- <div class="links">...
        <!-- <div class="buttons">...
    </nav>
    <section>
        <div class="content">
            <h1 data-aos="fade-right" data-aos-duration="2000" data-aos-delay="900">Your Safe Travel Journey Is Our Main Goal</h1>
            <!-- <p data-aos="fade-left" data-aos-duration="2000" data-aos-delay="1000">Lorem ipsum dolor sit amet consectetur adipisicing elit. Qui molestias, deleniti eaque culpa.</p>
            <button data-aos="zoom-in" data-aos-duration="2000" data-aos-delay="1200" onclick="navigateToDashboard()" >GET YOUR TICKET</button>
        </div>
    </section>
    <script src="https://unpkg.com/aos@next/dist/aos.js"></script>
    <script>
        AOS.init();
```

## 5.3 Screenshots of the Website

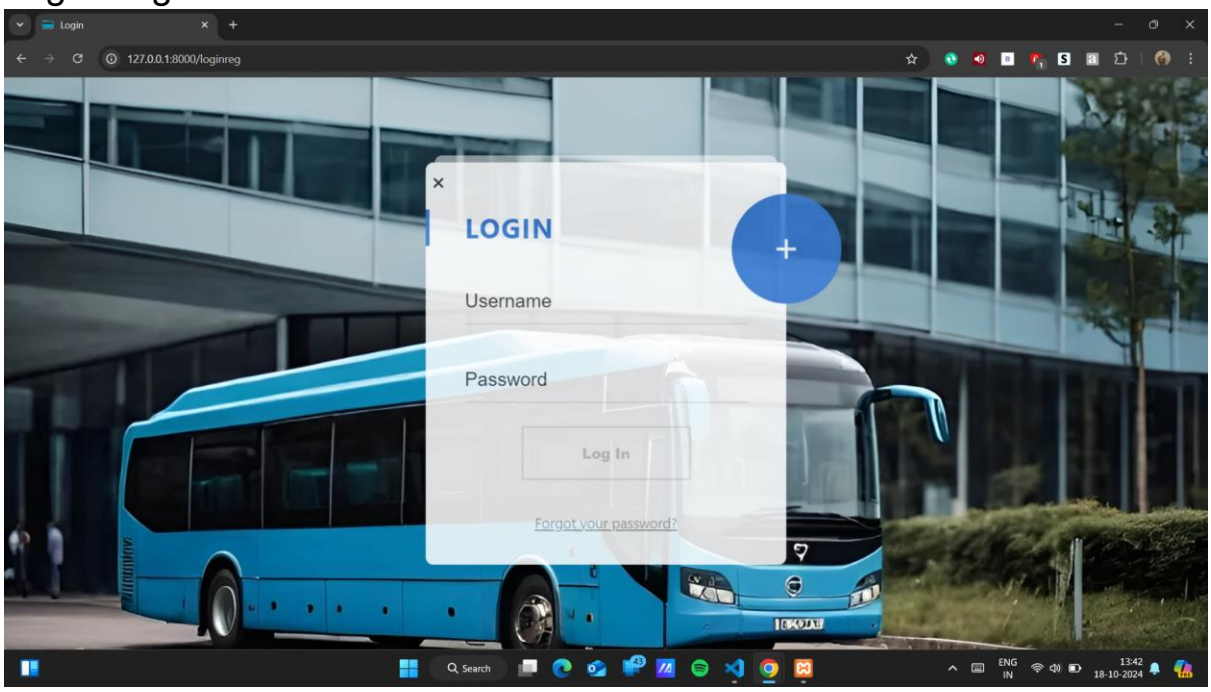
1. Splash :



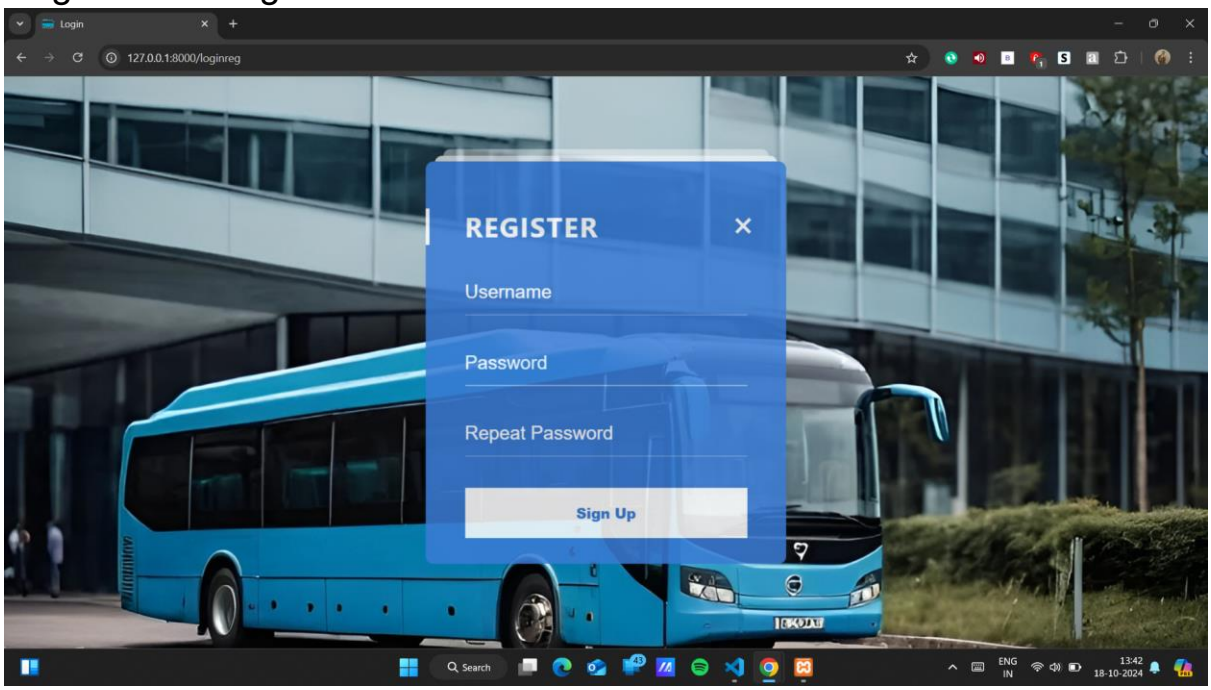

2. Dashboard Page:



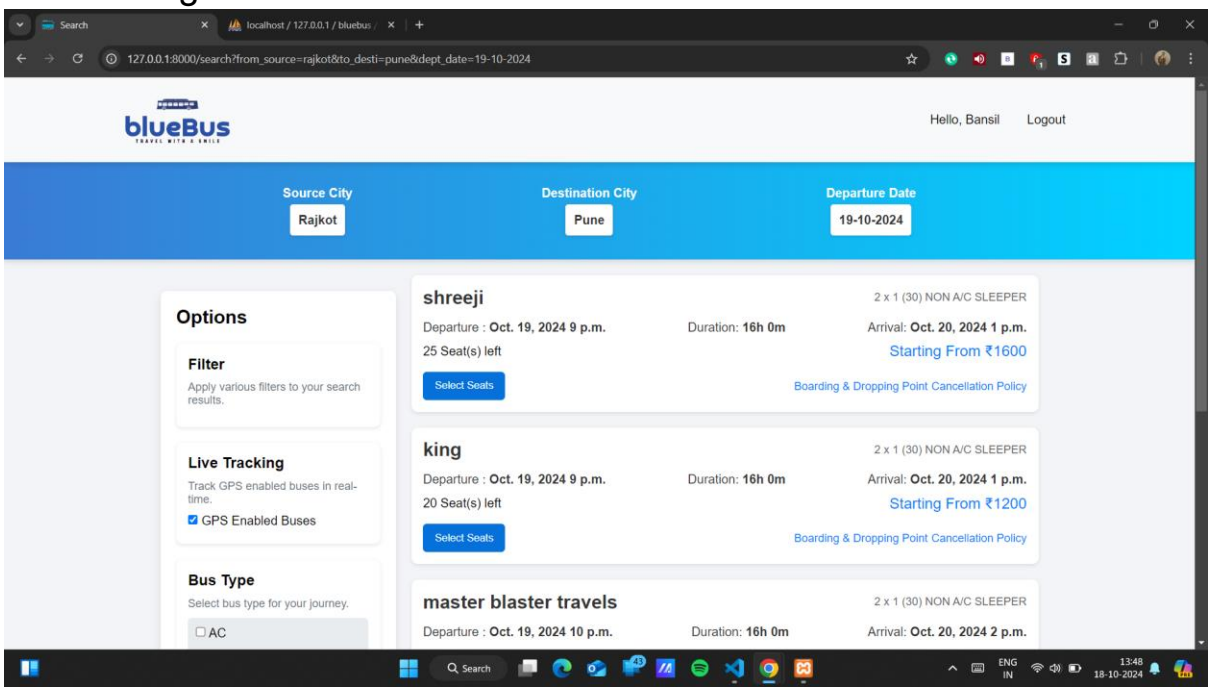
3. Login Page:



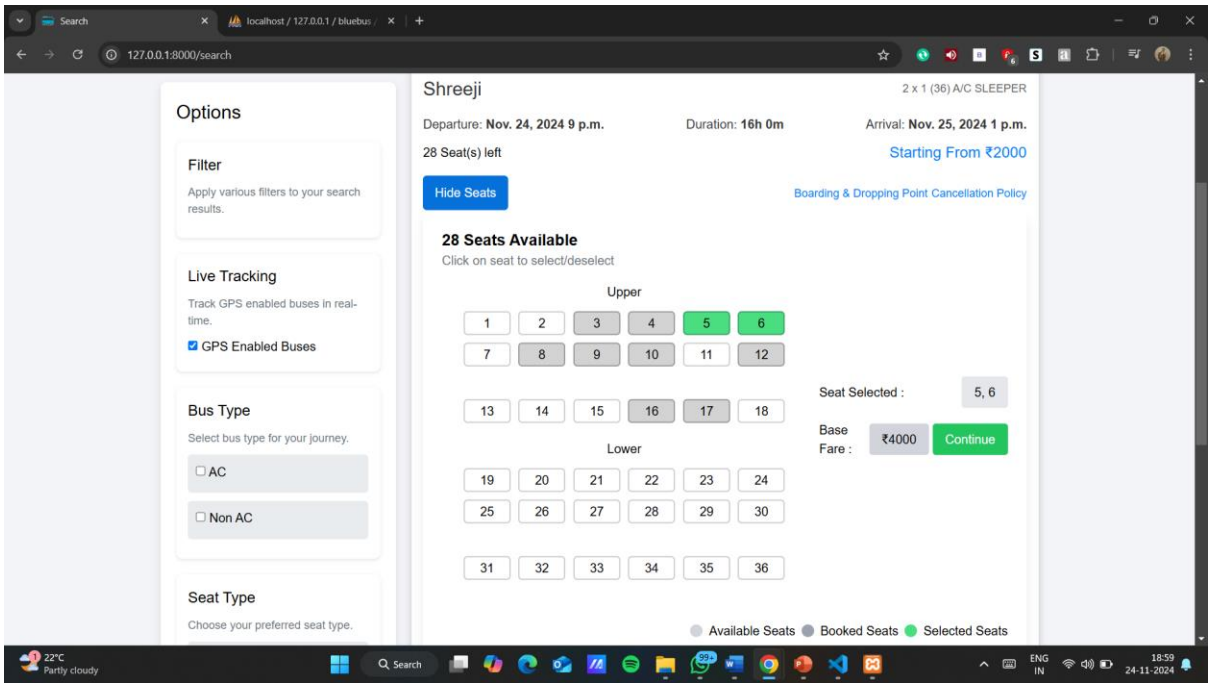

4. Registration Page:



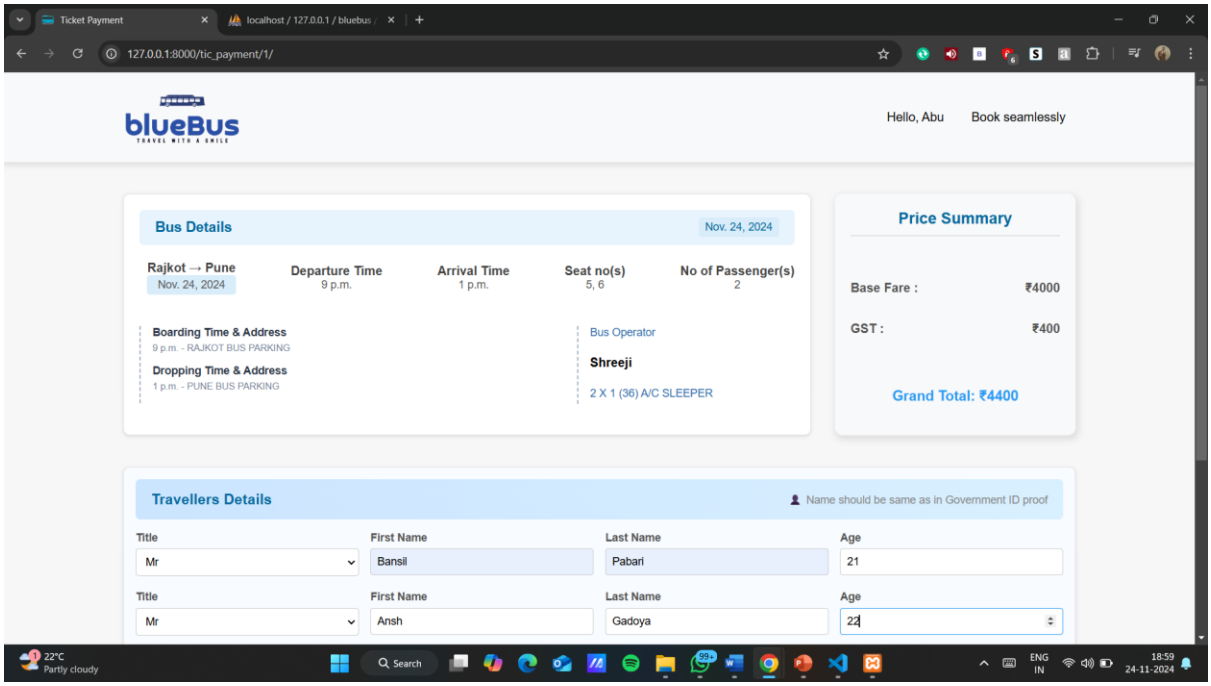
5. Search Page:

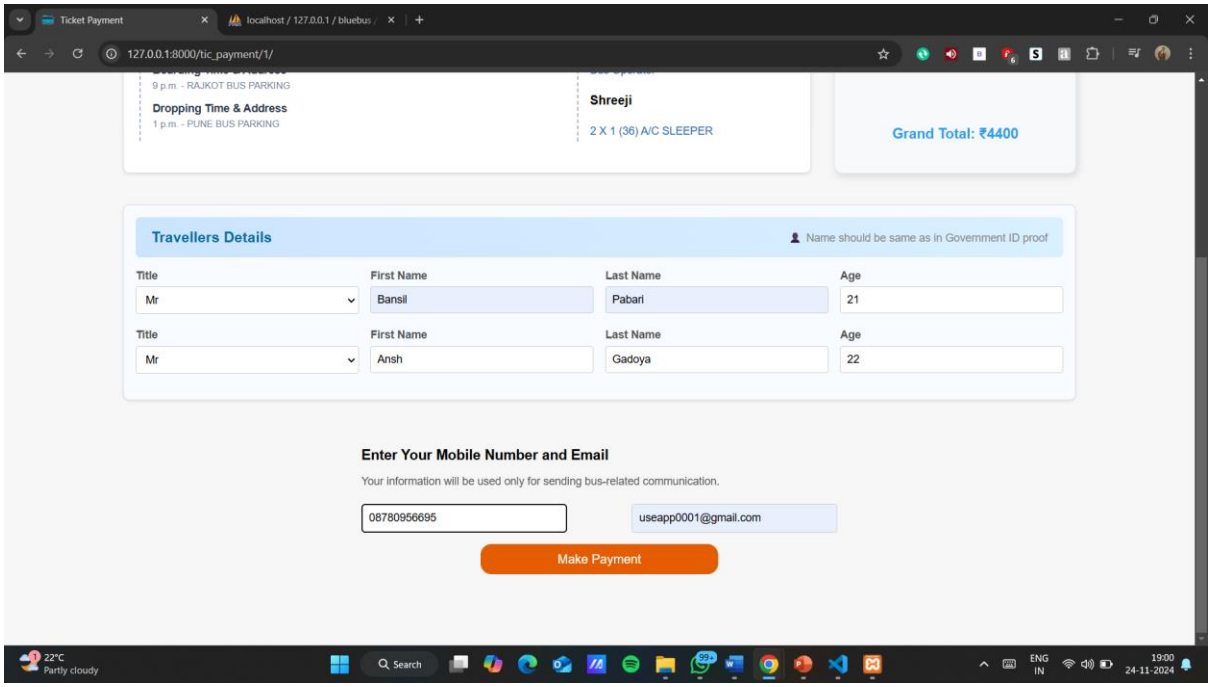



6. Seat Selection Page:

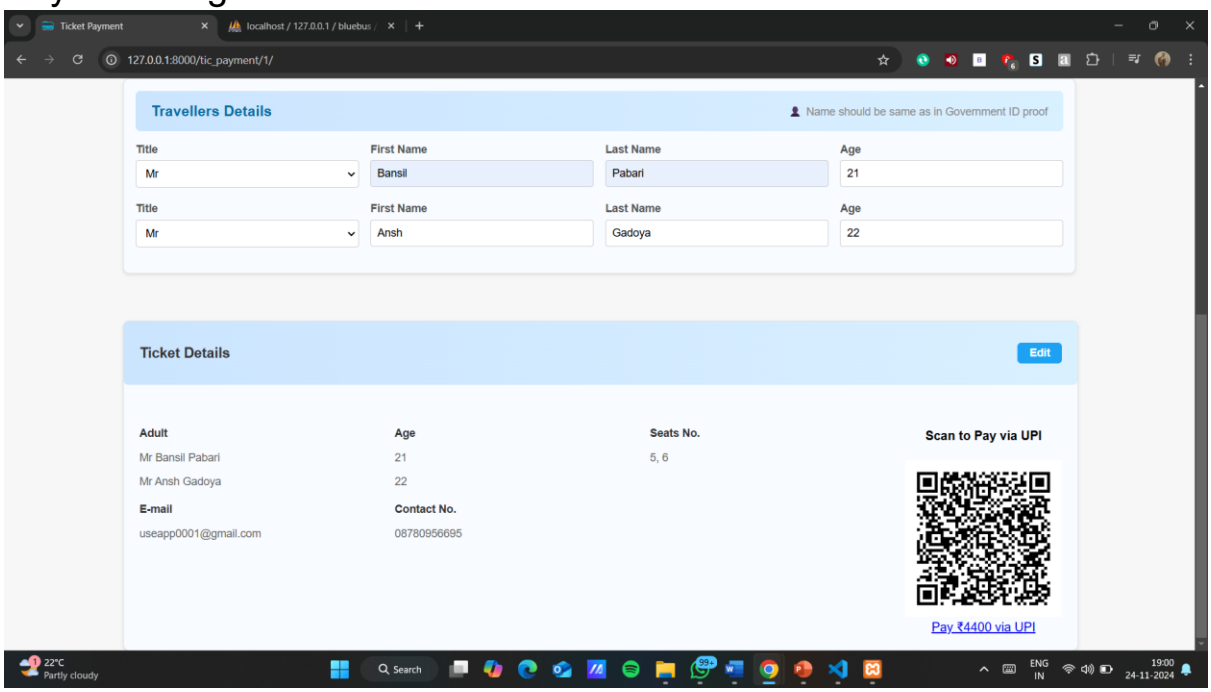


7. Ticket Payment Page:

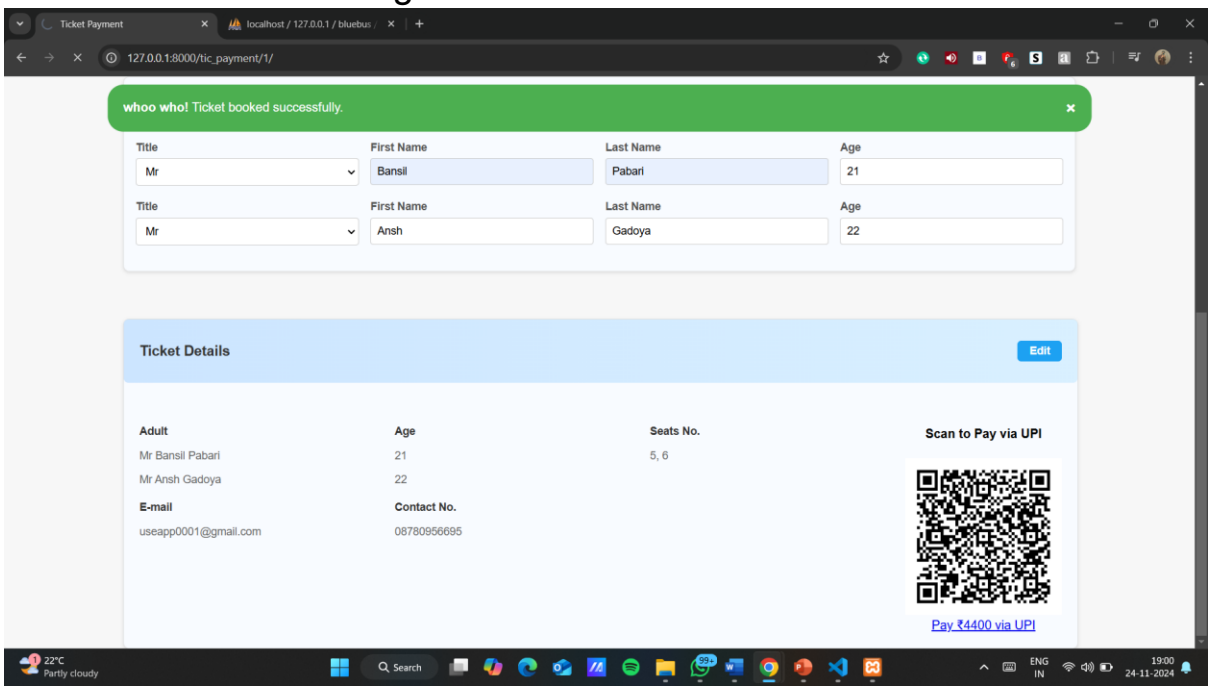


8. Payment Page:




9. Ticket Conformation Page:



# 6.Testing

## 6.1 Test Cases and Testing Methodology

| Project Name: blueBus | Module Name: User Login, Seat Booking, Mail Confirmation | Project Code: blueBus |
|---|---|---|
| Total No. of Test Cases: 6 | Total Passed: 6 | Total Failed: 0 |
| Total Executed: 6 | | |

User Login Module

| Test Case ID | Test Case Procedure | Input Data | Expected Output | Actual Output | Test Status |
|---|---|---|---|---|---|
| 1 | Check functionality of User Login with valid credentials. | 1. Enter valid Username.<br>2. Enter valid Password.<br>3. Click Login button. | Dashboard page should display. | Dashboard page displayed. | Pass |
| 2 | Check functionality of User Login with invalid credentials. | 1. Enter invalid Username or Password.<br>3. Click Login button. | Error message: "Invalid Username or Password." | Error message: "Invalid Username or Password." | Pass |
| 3 | Check functionality when both Username and Password fields are empty. | Leave both fields empty.<br>Click Login button. | Error message: "Please enter Username and Password." | Error message: "Please enter Username and Password." | Pass |

Seat Booking Module

| Test Case ID | Test Case Procedure | Input Data | Expected Output | Actual Output | Test Status |
|---|---|---|---|---|---|
| 1 | Check functionality of seat selection when selecting available seats. | Select available seats and click "Book Now." | Seats confirmed, booking summary displayed. | Seats confirmed, booking summary displayed. | Pass |
| 2 | Check functionality when attempting to select already booked seats. | Select a seat marked as booked and click "Book Now." | Error message: "Selected seat is unavailable." | Error message: "Selected seat is unavailable." | Pass |

Mail Confirmation Module

| Test Case ID | Test Case Procedure | Input Data | Expected Output | Actual Output | Test Status |
|---|---|---|---|---|---|
| 1 | Check functionality of mail confirmation after successful booking. | Complete seat booking process. | Confirmation email with booking details sent. | Confirmation email with booking details sent. | Pass |

## 6.2 Testing Methodology

1. Unit Testing: Focused on individual modules like seat selection and payment processing.
2. Integration Testing: Verified communication between modules, such as user input, search results, and bookings.
3. User Acceptance Testing (UAT): Gathered feedback from potential users to refine functionality and usability.

## 6.3 Results of Testing

All core functionalities, including search, booking, and payments, were tested rigorously under various scenarios. The system demonstrated stability, responsiveness, and accuracy, with minor bugs resolved promptly.

# 7. Results and Discussion

## 7.1 Outcomes of the Project

The *blueBus* project achieved the following objectives:

1. User-Friendly Booking Process: Simplified interface enabling users to search and book tickets effortlessly.
2. Real-Time Data Updates: Accurate seat availability and dynamic pricing ensured transparency.
3. Secure Transactions: Payment gateways offered reliability and user trust.
4. Operational Efficiency: Automation reduced manual workload for operators and minimized errors.

## 7.2 Comparison with Existing Solutions

| Feature | blueBus | Competitors (e.g., RedBus) |
|---|---|---|
| Real-Time Updates | Fully implemented | Partially implemented |
| Interactive Seat Layouts | Yes | Limited |
| Payment Security | Advanced encryption | Standard options |
| Regional Route Coverage | Comprehensive | Limited |

## 7.3 Challenges Faced

1. Data Synchronization: Managing real-time updates for multiple operators was challenging but addressed through efficient backend logic.
2. Payment Failures: Implemented robust error-handling mechanisms to ensure user-friendly resolutions.

# 8. Conclusion and Future Scope

## 8.1 Summary of the Work

*blueBus* successfully demonstrates the potential of leveraging modern web technologies to address inefficiencies in bus ticket booking systems. By prioritizing user experience, transparency, and security, the platform delivers a superior travel booking solution.

## 8.2 Limitations

1. Currently limited to domestic routes within state like Gujarat and Maharashtra.
2. Requires consistent data inputs from operators for accuracy.

## 8.3 Future Enhancements

1. Wide Route Integration: Expand the network to include around nation wide bus services.
2. AI-Driven Personalization: Use machine learning algorithms to recommend routes based on user preferences.
3. Mobile App Development: Launch Android and iOS applications for wider accessibility.
4. Dynamic Offers and Discounts: Introduce a loyalty program to retain users.

# 9. References


- RedBus Official Website.
  https://www.redbus.in
- EaseMyTrip Features and Overview.
  https://www.easemytrip.com
- Django Documentation.
  https://docs.djangoproject.com
- Python Official Website.
  https://www.python.org