DESIGN AN ONLINE CHATBOT BUS BOOKING SYSTEM WITH CARBON FOOTPRINT INSIGHTS USING NLP

Dissertation submitted in partial fulfillment of the requirements
for the award of Degree of

BACHELOR OF COMPUTER APPLICATIONS

Submitted by

ODETI MOUNIKA (22102C010226)

Under the Guidance of

DR. J. SURESH BABU,MCA, Ph.D.,
Professor, Dept of Computer Applications.

DEPARTMENT OF COMPUTER APPLICATIONS

SCHOOL OF COMPUTING

MOHAN BABU UNIVERSITY

Sree Sainath Nagar, Tirupati–517102

(2025)

MOHANBABUUNIVERSITY

DEPARTMENT OF COMPUTER APPLICATIONS

Vision

To become a center of excellence in the field of computer science and applications.

Mission

- Imparting knowledge and skills through contemporary curriculum to the diverse group of students.
- Creating a talent pool of faculty in diverse domains of computer applications through continuous training.
- Domain and transferable skill development for the holistic personality of students to inculcate values and ethics for effective professional practice and as an entrepreneur.

BACHELOR OF COMPUTER APPLICATIONS (BCA)

PROGRAM EDUCATIONAL OBJECTIVES

After few years of completion of the Program, the graduates of BCA will be able to:

PEO1. Pursue higher education in the core and allied areas of computer science by applying computing knowledge and domain-specific knowledge, demonstrating their innovative skills, and considering social and environmental concerns.

PEO2. Become professionals in industry and academia with ability to investigate, and solve complex computing problems using modern tools in evolving technologies in the core and allied areas of computer science.

PEO3. Become successful entrepreneurs to excel in diverse application skills in the core or allied area of computer science of societal importance.

PEO4. Exhibit professionalism, and uplifting health, safety, legal, environmental, ethical, and cultural diversity issues for serving the society and communicating with local, and national peers, bound within regulations and leading to lifelong learning.

PROGRAM OUTCOMES

On successful completion of the Program, the graduates of BCA will be able to:

PO1.Computational Knowledge: Apply knowledge of computing fundamentals, computing specialization, mathematics, and domain knowledge appropriate for the computing specialization to the abstraction and conceptualization of computing models from defined problems and requirements.

PO2.Problem Analysis: Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.

PO3.Design / Development of Solutions: Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.

PO4.Conduct Investigations of Complex Computing 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.

PO5.Modern Tool Usage: Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.

PO6.Professional Ethics: Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practices.

PO7.Life-long Learning: Recognize the need, and have the ability, to engage in independent learning for continual development as a computing professional.

PO8.Project management and finance: Demonstrate knowledge and understanding of the computing 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**.**

PO9.Communication Efficacy: Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.

PO10.Societal and Environmental Concern: Understand and assess societal, environmental, health, safety, legal, and cultural issues within local and global contexts, and the consequential responsibilities relevant to professional computing practices.

PO11.Individual and Team Work: Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.

PO12.Innovation and Entrepreneurship: Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society and interpretation of data, and synthesis of the information to provide valid conclusions.

PROGRAM SPECIFIC OUTCOMES

On successful completion of the Program, the graduates of BCA will be able to:

PSO1: Design, implement and test applications for complex computing problems for desired specifications using programming skills.

PSO2: Analyze and adapt managerial and domain skills of Information Management to model an application's data requirements to extract information for interpreting the datasets for Decision Making.

PSO3: Apply suitable techniques and algorithms to integrate Operating System Services, Network devices, Security mechanisms and Infrastructure to meet the requirements for the deployment of an application and to communicate on networks.

PSO4: Design and develop websites and Platforms by applying skills of Full Stack Technologies

MOHAN BABU UNIVERSITY

Sree Sainath Nagar, Tirupati 517 102

SCHOOL OF COMPUTING

DEPARTMENT OF COMPUTER APPLICATIONS

Certificate

This is to certify that the project report entitled **"DESIGN AN ONLINE CHATBOT BUS BOOKING SYSTEM WITH CARBON FOOTPRINT INSIGHTS USING NLP"** is the Bonafide work carried out and submitted by

ODETI MOUNIKA(22102C010226)

in the Department of **Computer Applications, School of Computing** of **Mohan Babu University**, Tirupati in partial fulfillment of the requirements for the award of the degree of **Bachelor of Computer Applications** during 2022-25.

SUPERVISOR

DR. J. SURESH BABU,MCA, Ph.D.,
Professor,
Dept of Computer Applications.

HEAD,DEPT.

Dr. M. Sowmya Vani,
Associate Professor & Head,
Department of CA,MBU, Tirupati

Date:

INTERNAL EXAMINER

EXTERNALEXAMINER

ACKNOWLEDGEMENT

This acknowledgement transcends the reality of formality when we would like to express my deep gratitude and respect to all those people behind the screen who guided, inspired and helped me for the completion of my project work.

I express my deep sense of gratitude to our beloved chancellor **Dr. M. Mohan Babu,** Padma Shri awardee for his encouragement throughout the program.

I express my deep sense of gratitude to our beloved vice-chancellor **Dr. K. Karunakaran,** for his encouragement throughout the Program.

I am extremely thankful to **Dr. M. Sowmya Vani, HOD**, and **Department of Computer Applications** for all provisions made and for her constant encouragement throughout my work.

I wish to express my deep sense of gratitude to my Project Coordinator **Dr. k. Sailaja, Associate Professor, Department of Computer Applications** and Project supervisor **DR J Suresh Babu, Professor, Department Of Computer Applications** for extending his valuable co-operation, moral support, kind attention, guidance, suggestions and encouragement to complete my Project Work successfully.

I thank all my beloved **Faculty**, Department of Computer applications for giving their valuable suggestions and maximum co-operation.

I owe a deep sense of gratitude to my beloved **Parents** in extending their moral support in this Endeavour. I would like to thank all my **friends** who extended their help, encouragement and moral support either directly or indirectly in completing my project work.

ODETI MOUNIKA (22102C010226)

DECLARATION

I, **ODETI MOUNIKA** hereby declare that, the project entitled "**DESIGN ONLINE CHATBOT BUS BOOKING SYSTEM WITH CARBON FOOTPRINT INSIGHTS USING NLP**" developed by me at **MOHAN BABU UNIVERSITY, Tirupati** during the Academic year 2022-2025 and submitted to The Principal, MOHAN BABU UNIVERSITY for partial fulfilment for the award of Bachelor of Computer Applications (BCA).

I also declare that, the project is resulted by my own effort and that it has not been copied from anyone and not been submitted by anybody in any of the University or Institution or Research Centre.

Place: Tirupati
Date:

ODETI MOUNIKA
(22102C010226)

ABSTRACT

The Online Chatbot Bus Booking With Carbon Footprint Insights Using NLP is a dual- module platform designed to simplify bus ticket bookings and provide carbon emission predictions for users. Admins can log in and input bus details, including departure and arrival locations, travel dates, and bus type. Based on this information, the system predicts the carbon emissions associated with the bus travel, helping admins monitor the environmental impact of each route. Admins also manage bus operations, such as adding buses, viewing bookings, and overseeing the bus schedule.Users can register, log in, and search for buses based on departure and arrival locations and travel dates. After selecting a bus, they can book tickets, make secure payments, and view their booking history. Additionally, the system features a chatbot powered by Gemini AI, offering real-time assistance and providing carbon footprint insights based on the user's travel data. The chatbot helps users with queries about their trip and environmental impact. Once tasks are completed, users can log out.This system enhances the convenience for both bus operators and travelers, offering seamless booking, real-time updates, secure payment processing, and valuable carbon footprint insights for a sustainable travel experience.

Keywords: Admin Module, User Module, Carbon Emission Prediction, Registration, Chatbot.

TABLE OF CONTENTS

List of Tables

List of Figures

CHAPTER - 1

1. Introduction

In an era marked by climate change, rapid urbanization, and digital transformation, sustainable development and intelligent technology have become central to modern innovation. Among the sectors most impacted by this shift is the transportation industry, where increasing demand for mobility is often accompanied by concerns about traffic congestion, fossil fuel dependence, and carbon emissions. Public transportation—especially bus travel—offers a compelling alternative to private vehicle use, as it significantly reduces emissions per passenger and encourages shared mobility.

Yet, the full environmental potential of public transport remains underutilized, largely due to the lack of accessible information, real-time systems, and public awareness about sustainable travel options. To bridge this gap, we propose a robust, AI-driven platform titled "Online Chatbot Bus Booking System with Carbon Footprint Insights Using NLP". This system introduces a smarter, more environmentally conscious approach to travel planning, making bus travel not only efficient and user-friendly but also transparently sustainable. The essence of the project lies in combining three core elements: automated bus booking, carbon emission tracking, and NLP-powered chatbot support.

These elements form the foundation of a dual-module platform designed to cater to both users and administrators. For passengers, the platform provides a smooth experience for searching, selecting, and booking buses across various routes. It simplifies the process by offering secure payment systems, booking history tracking, and 24/7 virtual assistance through a chatbot. For bus operators and administrators, the system introduces functionalities to manage buses, monitor booking records, and most importantly, calculate and oversee the carbon footprint of each journey based on bus type, route distance, and travel time. By leveraging the capabilities of Gemini AI for chatbot integration and utilizing Natural Language Processing (NLP) for intelligent interaction, the system ensures that communication between the user and the system is both human-like and context-aware.

The need for this platform becomes evident when examining the limitations of traditional bus booking systems. While many existing platforms support basic features such as ticket reservations and payment integration, they are largely transactional in nature. They lack personalization, do not offer real-time assistance, and almost entirely ignore the Page15 MBU-2024 environmental impact of travel. This one-dimensional approach not only weakens user engagement but also represents a missed opportunity in promoting green transportation. In contrast, our system emphasizes a holistic experience by empowering users with valuable information—such as the estimated carbon emissions of a particular route—right at the decision-making point. This makes the system not just a booking tool, but a guide for eco conscious choices. The Admin Module of the system is engineered to give full control over bus operations.

1. Background of the Study

In the age of digital transformation, artificial intelligence (AI) has revolutionized every major industry, from healthcare to finance, retail to transportation. Among these, the travel and transportation sector has witnessed significant technological advancements aimed at enhancing convenience, efficiency, and user satisfaction. Particularly, bus travel—a vital and affordable mode of transport for millions—has undergone digitalization. Traditional bus booking systems, reliant on physical counters and manual records, have given way to online reservation platforms. However, despite this progress, current systems largely prioritize ease of booking and cost-effectiveness, overlooking environmental concerns such as carbon emissions associated with travel.

Amidst escalating climate change challenges and the global shift towards sustainable living, integrating eco-conscious features into everyday systems has become imperative. The transportation sector contributes significantly to greenhouse gas emissions, necessitating solutions that not only offer convenience but also promote sustainable choices. In this context, the Smart Online Chatbot Bus Booking System with Carbon Footprint Insights using NLP emerges as a pioneering approach.

2. Need for Intelligent Systems in Transportation

Transportation systems worldwide are being reimagined under the vision of smart cities, where efficiency, environmental consciousness, and user-centricity are paramount. The traditional offline methods for bus booking, while effective in their time, are plagued with challenges—manual errors, limited accessibility, time-consuming processes, and lack of real-time information. Moreover, passengers have minimal or no insights into the environmental impact of their travel decisions.

The pressing need for intelligent transportation systems arises from several key factors. Accessibility is critical, ensuring that users can conveniently book, modify, or cancel tickets from anywhere at any time. Efficiency is equally important, aiming to reduce manual intervention, streamline the booking process, and minimize operational errors. Enhancing user engagement through personalized, interactive experiences delivered by conversational agents, such as chatbots, further strengthens the system's appeal. Moreover, there is a growing emphasis on environmental responsibility, empowering users to understand and reduce their carbon footprint by making more informed travel choices.

3. Evolution of Chatbots and NLP in Public Services:

Chatbots, once simple rule-based programs, have evolved into sophisticated AI agents capable of handling complex interactions, understanding natural language, and delivering personalized services. Their application has expanded from customer support in retail to banking, healthcare, education, and now, transportation.

Natural Language Processing (NLP) forms the backbone of modern chatbots. It allows machines to process and interpret human language, deciphering user intent and context to provide accurate and relevant responses. Recent advancements like deep learning-based NLP models (e.g., BERT, GPT) have significantly improved the conversational abilities of chatbots, making them more human-like and contextually aware.

4. Concept of Carbon Footprint Insights

The carbon footprint refers to the total amount of greenhouse gases, primarily carbon dioxide, emitted directly or indirectly by human activities. Travel, particularly motorized transportation, is a major contributor to individual and collective carbon footprints. Despite growing environmental awareness, many travelers remain unaware of the emissions linked to their journey choices.

Providing carbon footprint insights at the point of booking empowers users to make sustainable decisions. For instance, choosing a high-occupancy bus over a low-occupancy or fuel-inefficient option significantly reduces per capita emissions. Visualizing carbon emissions alongside ticket prices and travel time allows users to weigh environmental impact alongside cost and convenience, promoting more eco-friendly travel behaviors.

5. Problem Statement

While existing online bus booking systems have enhanced convenience, they fall short in two critical areas: natural, intuitive user interaction and sustainability integration. Most platforms require users to navigate structured forms and dropdown menus, which can be cumbersome and non-conversational. Moreover, they seldom provide information on the environmental impact of different travel options, depriving users of the opportunity to make greener choices.

The challenge lies in designing a system that enables natural, conversational ticket booking using NLP, offers real-time, personalized travel recommendations, calculates and displays the carbon footprint for each travel option, and actively encourages and educates users about sustainable travel choices. Addressing these needs demands a multidisciplinary approach that brings together expertise from AI, environmental science, human-computer interaction (HCI), and transportation management.

6. Research Motivation

The motivation behind this project is rooted in the dual pursuit of technological innovation and environmental stewardship. As cities become denser and the effects of climate change become more pronounced, it is essential to rethink how public transport services are delivered and consumed.

leveraging AI for social good by demonstrating how advanced technologies like NLP can be used to address global challenges such as climate change; and fostering user empowerment by providing travelers with the information they need to make environmentally conscious decisions. Through this project, we aim not only to build a better booking system but also to contribute meaningfully to the advancement of sustainable urban mobility.

7. Scope of the Project

The system offers a range of user-centric features, allowing users to search for buses through natural language queries (e.g., “Find me a bus from Hyderabad to Bangalore tomorrow morning”), view available options along with fare details, travel time, seat availability, and carbon emissions, securely book tickets while maintaining a booking history, receive real-time updates and alerts, and engage with an AI-powered chatbot for support and personalized recommendations. On the admin side, the platform enables the management of bus schedules, seat inventories, and fare structures, monitoring of booking statistics and user interactions, analysis of carbon footprint data across routes, and optimization of fleet operations to encourage eco-friendly travel choices.

The project’s technical scope includes frontend development using React.js, backend development with Python Django, database management through MySQL, AI integration via NLP models and carbon calculation APIs, and implementation of robust security frameworks, including secure payment gateways.

8. Objectives

The primary objectives guiding this project are to simplify bus booking by making it as easy as asking a question, educate users about the environmental impact of their travel choices, empower them to make decisions based on both convenience and carbon footprint, optimize operational efficiency for bus operators, and inspire sustainable travel habits among users. Achieving these objectives demands meticulous system design, robust AI integration, and the development of a user-centered interface.

The primary aim of the project titled "Online Chatbot Bus Booking System with Carbon Footprint Insights Using NLP" is to create a comprehensive, intelligent, and environmentally conscious platform for bus travel management that simplifies ticket booking, enhances user engagement, and promotes sustainable travel decisions. This system is designed to cater to the modern-day needs of travelers and transport operators by integrating automation, artificial intelligence, and real-time communication tools with an environmentally responsible approach. It moves beyond basic booking functionalities to provide a complete ecosystem where users can not only manage their travel but also understand and reduce their carbon footprint.

9. Technical Approach:

Our technical approach integrates multiple cutting-edge technologies to build a cohesive, high-performing system. Natural Language Processing (NLP) techniques, leveraging pre-trained models like BERT, are employed to parse and understand user queries, while Support Vector Machines (SVMs) are used for intent classification and response optimization. Carbon footprint calculations are handled through a combination of external APIs and custom algorithms that factor in distance, bus efficiency, and occupancy rates. The application is developed using secure and scalable web technologies, specifically Django for the backend and React.js for the frontend. MySQL is utilized for efficient storage and retrieval of user, booking, and carbon footprint data. Additionally, AI chatbot frameworks such as Gemini AI APIs are integrated to enable real-time, intelligent user interactions. Together, these components create a system architecture focused on performance, scalability, and sustainability.

It creates an ecosystem where sustainable decisions are not just possible but convenient and rewarding. This reflects a broader mission to contribute to global climate efforts, align with the United Nations Sustainable Development Goals (particularly SDG 11 and SDG 13), and inspire innovation that goes hand-in-hand with environmental responsibility.

The aim of the Online Chatbot Bus Booking System with Carbon Footprint Insights Using NLP is multifaceted and forward-thinking. It seeks to offer a practical, intelligent, and user-centered solution for bus travel while embedding a deeper understanding of environmental sustainability into the everyday choices of individuals and organizations. From enhancing booking efficiency and enabling carbon transparency to providing intelligent assistance and supporting future scalability, the project lays a strong foundation for what digital, green transportation should look like in the modern world.

Despite the advancements in online travel booking systems, a significant research gap remains in the integration of carbon footprint insights into user interfaces and booking decisions. Current systems primarily optimize for price and convenience without considering environmental sustainability. Additionally, while chatbots are increasingly used for customer support, their potential in promoting sustainable behaviors through NLP-driven interactions remains largely untapped.

CHAPTER - 2

2. Literature Survey

Urban smart public transport systems have garnered substantial attention in recent years due to their potential to revolutionize mobility[21], reduce traffic congestion, and enhance environmental sustainability. For instance, studies such as those by explore the integration of smart technologies into public transport, aiming to optimize route planning, reduce waiting times, and improve passenger satisfaction. By utilizing data analytics and real-time monitoring, smart transport systems have the capacity to improve operational efficiency and reduce energy consumption. In a similar vein, highlight the transformative role [22] of big data in driving sustainable consumption patterns.

By enabling more informed decision-making and forecasting consumer behavior, big data fosters environmentally responsible practices across industries. Additionally, discuss the development and implementation of online bus ticket reservation systems [23] demonstrating their efficiency in simplifying ticketing processes, reducing manual errors, and enhancing the overall user experience.

These advancements illustrate how digital solutions can contribute to improving transportation systems and reducing their environmental impact. Moreover, the intersection of Artificial Intelligence (AI) and environmental, social, and governance (ESG) [24] frameworks in finance is critically reviewed by , where the potential of AI in fostering sustainable investment strategies is emphasized. Finally, the concept of smart and circular cities is explored in , with a focus on the use of chatbots in waste recycling. This study illustrates how AI-driven conversational agents[25]can assist citizens in waste segregation, thus promoting sustainable urban development and contributing to circular economy goals .The integration of Natural Language Processing (NLP) in supporting Sustainable Development Goals (SDGs) [16] has gained significant traction in recent years, as highlighted by .

The authors argue that NLP models play a critical role in enhancing the understanding of complex sustainability challenges by translating semantics and visualizing interconnected issues. By leveraging text analytics and semantic mapping, NLP aids in clarifying the interactions between various stakeholders and promotes more effective communication.

In addition, explores the automation of services like bus ticket booking systems, which streamline the booking process and contribute to a reduction in the carbon footprint[17] associated with traditional booking methods. This aligns with broader efforts to utilize digital solutions to minimize environmental impact. Another significant contribution is made by , who examine the role of chatbots in improving customer support services. Their study emphasizes the advantages of automated, round-the-clock[18] support that enhances operational efficiency and user satisfaction. Furthermore, provides a comprehensive evaluation of chatbots, discussing the challenges of creating systems that adequately capture user satisfaction, conversation quality[19], and overall system performance. Finally, presents various methods for estimating carbon footprints [20], offering tools that assess the environmental impact of human activities and industries.

These studies underline the growing importance of technological solutions, such as NLP and chatbots, in facilitating sustainability efforts while simultaneously improving user engagement. The literature on carbon footprint evaluation, sustainable travel practices, and the integration of NLP and AI technologies [11] further demonstrates how environmental consciousness intersects with digital innovations. Investigates the environmental impacts of NLP models, presenting an analysis of existing tools to evaluate their carbon footprints.

The paper advocates for the development of more energy- efficient NLP methods[12], which is crucial given the increasing carbon emissions tied to computational linguistics. focuses on an automated carbon footprint calculator tailored for individuals, emphasizing its role in estimating carbon emissions related to transportation usage. The study offers a practical tool for reducing personal environmental impact by encouraging conscious travel choices. In a related domain, explores the role of chatbots in promoting sustainable travel practices among tourists, demonstrating how conversational agents can provide real-time[13] suggestions to encourage eco-friendly behaviors.

Moreover, reviews the integration of NLP in Intelligent Transportation Systems (ITS)[14], shedding light on its applications in enhancing transportation efficiency, minimizing emissions, and improving user experience within smart cities. In line with this, surveys various conversational agents designed to raise energy awareness and improve efficiency.These agents are aimed at helping users monitor and reduce energy consumption, illustrating the potential of AI-driven solutions [15] in advancing sustainable practices across sectors. Recent studies further demonstrate the growing impact of AI and NLP technologies across multiple domains.

Similarly, the work Application and Prospect of AI Technology in Low-Carbon Cities—From the Perspective of Urban Planning Content and Process (2022) explores how AI can be integrated into urban planning to optimize sustainability and support the transition to low-carbon cities. This aligns with the findings of the Chat-Bot Based Ticketing System Using Dialogflow (2021), which discusses the use of chatbots for automating ticketing services and improving customer interaction. Lastly, Travel Booking Chatbot (2022) examines how chatbots can streamline the booking process in the travel industry, improving user experience through AI-powered conversations. Together, these studies highlight the expanding scope of AI and NLP applications, particularly in enhancing user engagement, optimizing system efficiency, and fostering sustainability across a variety of sectors.

A comprehensive review of recent advancements in chatbot systems reveals significant contributions to the fields of NLP and AI [5]. explores the use of NLP in customer service, noting its ability to improve interaction efficiency and user experience by automating responses and providing timely solutions. The authors suggest that NLP technologies are revolutionizing customer support systems by enabling quick, accurate communication. Another study by emphasizes the implementation of AI and NLP in chatbot [4] systems, demonstrating how these technologies enable chatbots to process natural language input and engage in meaningful dialogues with users.

By improving service delivery and reducing operational costs, AI- powered chatbots are shown to significantly enhance both user satisfaction and system performance. The research in further investigates the design and development of chatbots , focusing on methodologies that guide the creation of effective AI-driven systems. The authors stress the importance of selecting appropriate frameworks [3] and models while addressing the challenges associated with maintaining conversational flow and incorporating adaptive learning features.

The role of chatbots in the context of smart cities is explored by , who discusses how chatbots can aid in waste recycling initiatives[2], facilitating communication, educating citizens, and fostering greater participation in environmental sustainability programs. The study underscores the importance of chatbots in enhancing community engagement and adherence to recycling efforts.

Lastly, examines the use of personalized chatbot coaches to reduce carbon footprints. By offering tailored recommendations and actionable feedback, these chatbots encourage users to make eco-conscious [1] choices, thus contributing to more sustainable lifestyles. These studies collectively emphasize the growing importance of AI and NLP in transforming various sectors, from customer service to environmental sustainability, and highlight their potential to shape the future of chatbot technologies.

CHAPTER - 3

3. Problem Definition

The problem definition for this project revolves around the challenges and inefficiencies present in traditional bus booking systems, which often lead to poor user experience, operational difficulties for bus service providers, and a lack of environmental awareness among travelers. Despite technological advancements, many public transport systems and ticketing platforms still lack intelligent automation and sustainability-focused features. Key issues identified in the current travel booking landscape include:

a) Manual Bus Ticket Booking:

Many traditional bus operators still rely on offline or semi-digital booking processes that are inefficient, time-consuming, and prone to human error. Users are required to visit physical counters or navigate poorly designed websites, which results in long wait times, booking errors, and user dissatisfaction.

b) Limited Availability of Real-Time Assistance:

Most bus booking platforms do not provide real-time customer support. Users with queries regarding routes, ticket availability, or travel policies often have to rely on call centers or FAQs, which are not always efficient or responsive. This results in frustration, especially for new users or during peak seasons.

c) Lack of Environmental Impact Awareness:

One of the major gaps in current booking systems is the lack of integration of sustainability features. Passengers are rarely informed about the carbon emissions associated with their travel choices, and there is no standard mechanism to encourage eco-friendly decisions. This contributes to a lack of awareness regarding environmental impact.

d) Fragmented Information Sources:

Information such as bus schedules, seat availability, route maps, and fare details are often scattered across various platforms or poorly integrated interfaces. This fragmentation makes it difficult for users to make informed decisions quickly and confidently.

e) Absence of Personalized and Intelligent Interfaces:

Conventional booking systems lack intelligent interfaces like chatbots that can interact in natural language. Users often face a rigid and non-intuitive user experience that does not caterto individual preferences or offer dynamic assistance throughout the booking journey.

f) Limited Integration of Natural Language Processing (NLP):

Even though NLP technology has made significant advances, most bus booking platforms have not adopted it effectively. Without NLP integration, systems fail to understand and process user queries in natural language, limiting accessibility and usability, especially for non- technical or linguistically diverse users.

g) Inefficient Admin Management Tools:

Bus operators and admins face challenges managing bookings, schedules, and route details manually or through outdated systems. This inefficiency leads to difficulties in monitoring fleet performance, handling passenger data, and maintaining service quality.

h) Lack of Secure Payment Systems and Booking Verification:

Some platforms still operate without proper encryption, secure payment gateways, or automated booking verification processes, putting users at risk of fraud, transaction failures, and identity theft. Ensuring secure and seamless payment is critical for user trust.

i) No Carbon Emission Tracking for Routes:

Traditional platforms overlook the environmental cost of travel. There are no built-in models to estimate and display the carbon emissions associated with specific bus routes, bus types (AC vs Non-AC), or distances. This lack of transparency prevents users from making informed sustainable travel choices.

j) Security and Payment Issues:

Security remains a concern in many outdated booking systems. Some platforms do not follow proper encryption standards for online transactions, making users wary of sharing their financial information. Additionally, the lack of multiple payment options and unreliable transaction confirmations further reduces trust in digital booking.

k) Absence of Integrated Feedback Mechanism: Most systems do not include robust feedback or rating features for routes, operators, or overall user experience. This prevents continuous service improvement and denies new users valuable insights from previous travelers. A lack of review integration also allows subpar services to continue unchecked.

3.1. Existing System

The existing bus booking systems, both traditional and semi-digital, typically offer basic functionalities such as searching for available buses based on source, destination, and travel date, selecting seats, and making payments either online or offline. These platforms often send booking confirmations and tickets through email or as downloadable PDFs. However, these systems fall short in terms of automation, real-time interaction, and personalized user experiences.

While some platforms provide basic mobile applications for convenience, they are often limited in terms of functionality and user-friendliness. Most existing systems rely on static interfaces and require users to manually navigate through multiple pages to complete their bookings, which is particularly frustrating for users who are not tech-savvy. Furthermore, these systems are typically devoid of advanced features such as chatbot support or intelligent assistance, meaning users must rely on traditional customer service methods, which can be inefficient.

One significant limitation of current systems is the lack of integration with Natural Language Processing (NLP), which prevents users from interacting with the system in a conversational way. This leads to a less accessible and more cumbersome user experience, especially for users who prefer informal language or need assistance in multiple languages. Additionally, there is no incorporation of carbon footprint estimation or environmental impact insights, which could encourage users to make more eco-friendly travel choices.

The absence of real-time updates regarding booking status, delays, or cancellations also limits the convenience of the booking process, leaving users to manually check their booking status or wait for delayed notifications.Another key drawback of existing systems is the outdated and inefficient admin management interfaces. Administrators often lack the tools to make data-driven decisions, optimize bus schedules, or offer personalized recommendations to travelers. This lack of predictive analytics and smart management options leads to inefficiencies in bus operation and customer service.

Security is another area of concern, as many existing systems do not adequately secure sensitive user data, leaving the platform vulnerable to potential breaches or fraud. Finally, the user interface design on many existing platforms is not optimized for different devices or screen sizes, leading to usability issues for passengers using smartphones or tablets.

Overall, while existing bus booking systems offer basic functionalities, they lack the innovation and sustainability features that could significantly enhance the user experience. The absence of NLP-powered chatbots, carbon footprint insights, personalized recommendations, and real-time updates highlights the need for a more advanced, user-centric, and eco-friendly solution, which is precisely what this proposed system aims to address.

Disadvantages:

a. **Limited Booking Management Features:** The current bus booking systems often lack comprehensive booking management features for both passengers and bus operators. For instance, passengers may find it difficult to reschedule or cancel their bookings easily, leading to potential inconvenience. Similarly, bus operators do not have. efficient tools for managing bookings dynamically, such as updating available seats in real-time or adjusting schedules based on demand. This limitation can disrupt the booking process and lead to frustrated customers, especially in situations involving sudden changes to travel plans.

b. **Lack of Real-Time Updates:** Most existing bus booking platforms do not provide real-time updates regarding changes in booking status, such as seat availability, delays, or cancellations. As a result, both users and operators may face confusion regarding their bookings, leading to missed connections, wasted time, and unnecessary stress. Without timely notifications or automated alerts, passengers may not be informed of schedule changes until they arrive at the station, further increasing the likelihood of disruptions.

c. **Dependency on Traditional Booking Interfaces:** The majority of existing bus booking platforms rely on traditional booking interfaces, which can be rigid and non-interactive. This lack of natural language processing (NLP) limits the ability of users to interact with the system in a conversational and intuitive manner. Passengers must rely on form-based input rather than being able to express their booking needs in natural language. This could be a barrier for non-technical users and those who are more comfortable with informal language, reducing accessibility and usability.

3.2. Hardware Requirements

● Processor	- I3/Intel Processor
● Hard Disk	-160GB
● Key Board	- Standard Windows Keyboard
● Mouse	- Two or Three Button Mouse
● Monitor	- SVGA
● RAM	- 4Gb/ 8Gb

a. **Processor:** The Intel Core i3 processor is suitable for the basic operations of your chatbot bus booking system. It can handle user interactions, natural language processing for query handling, and basic calculations for estimating carbon footprint. While it may not be ideal for handling large volumes of concurrent users or training complex NLP models locally, it's sufficient for development, testing, and running a lightweight production system with a pre-trained model.

b. **Hard Disk (160GB):**A 160GB hard disk offers adequate storage space for the operating system, development tools (e.g., Python, libraries), chatbot scripts, and carbon footprint datasets. With efficient storage management, it can also accommodate basic logs, user data, and booking records. However, if your application scales up (e.g., adding more detailed logs, images, or long-term booking history), you may need cloud storage or an external drive.

c. **Keyboard (Standard Windows Keyboard):**This standard input device is essential for writing code, interacting with the development environment, and manually inputting or troubleshooting user queries. It supports seamless interaction while developing and managing the chatbot interface and backend services.

d. **Mouse (Two or Three Button Mouse):** The mouse assists with navigating the development environment, selecting files, debugging visually, and operating the graphical components of the chatbot system. It enhances productivity and ease of use during development and demonstration phases.

3.3.Software Requirements

Operating System: Windows 7/8/10

Front End: HTML, CSS, React.js

Programming Languages: Python (Backend), JavaScript (Frontend)

Tool: Natural Language Processing Framework

Database: MySQL

Chatbot Framework: Django, React

Carbon Footprint Calculator: API Integration or Custom Calculation (NumPy, math)

IDE/Workbench: VS code

AI/ML: Gemini API &NLP

In the modern landscape of software development, the seamless integration of multiple technologies has become essential for delivering robust, scalable, and user-friendly applications. The project in focus was developed on the Windows operating system, with compatibility across versions 7, 8, and 10, ensuring wide accessibility for end-users. The front-end of the application leverages HTML and CSS for basic structure and styling, while React.js—a powerful JavaScript library—is employed for building dynamic, responsive user interfaces. React.js facilitates component-based architecture, which allows for efficient UI updates and reusability of code, making the application both scalable and maintainable.

The programming backbone of the project is structured with JavaScript and Python. JavaScript handles client-side operations, while Python is employed on the server side, ensuring a smooth transition of data between front-end and back-end processes. Python, known for its simplicity and versatility, also plays a critical role in processing logic, data handling, and integrating advanced functionalities such as Natural Language Processing (NLP). The integration of NLP tools has enabled the system to understand, interpret, and respond to human language effectively, enhancing the user experience, particularly in chatbot functionalities.

For natural language understanding and machine learning capabilities, the Gemini API is utilized. This cutting-edge AI/ML tool offers state-of-the-art NLP services, enabling the application to process user inputs intelligently and provide context-aware responses. Gemini API's integration allows for tasks such as sentiment analysis, keyword extraction, and intent recognition to be performed with high accuracy, which is essential in building smart conversational agents and enhancing interaction quality in chatbot interfaces.

The backend architecture is powered by the Django framework—a high-level Python web framework that promotes rapid development and clean, pragmatic design. Django offers a secure, scalable foundation for building web applications and RESTful APIs. Its built-in admin panel, ORM (Object-Relational Mapping), and robust middleware system streamline the development process. This framework, when combined with React on the front end, facilitates the development of modern full-stack web applications with clear separation of concerns and a smooth data flow between components.

The database system employed is MySQL, a reliable and high-performance relational database management system. MySQL is used to store and manage structured data efficiently, such as user information, chat logs, and application settings. Django's ORM seamlessly connects to MySQL, enabling developers to interact with the database using Python code instead of raw SQL queries. This abstraction simplifies database management while maintaining performance and scalability.

Incorporating a chatbot into the application adds a significant layer of user interaction and automation. The chatbot is developed using Django and React, allowing it to operate as a single-page application with real-time response capabilities. React handles the client-side message rendering, while Django processes user inputs on the server side, leveraging the NLP functionalities provided by the Gemini API and other language processing tools. This system enables the chatbot to provide meaningful responses, guide users through workflows, and handle queries autonomously.

An additional feature of the application is the Carbon Footprint Calculator—a tool designed to help users understand their environmental impact based on various inputs like travel habits, energy usage, and dietary preferences. This calculator either uses a predefined API for standardized carbon footprint estimations or performs custom calculations using Python libraries such as NumPy and the math module. These calculations rely on scientifically validated emission factors and algorithms to estimate a user's carbon footprint accurately. By providing immediate feedback and suggestions for greener choices, this tool not only enhances user engagement but also promotes environmental awareness and responsibility.

For the development and testing environment, Visual Studio Code (VS Code) is employed. VS Code is a lightweight but powerful source code editor with support for debugging, task running, and version control. Its rich ecosystem of extensions allows developers to customize their workflow, integrating tools for Python, JavaScript, React, Django, and MySQL seamlessly. Features like IntelliSense, Git integration, and real-time linting significantly enhance productivity and code quality throughout the development cycle.

The system as a whole is designed with modularity and extensibility in mind. Each component—from the frontend interface built with React to the NLP and AI services powered by Gemini API—is loosely coupled yet fully integrated. This architecture allows for individual parts to be updated, replaced, or scaled independently without disrupting the overall system. For instance, switching from MySQL to another relational or NoSQL database would require minimal changes in the ORM configuration, thanks to Django’s abstracted database layer.

Moreover, security considerations are embedded throughout the application lifecycle. Django provides robust security features such as protection against SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF). HTTPS protocols and secure authentication mechanisms ensure that user data remains protected during transmission and storage.

User experience is further enhanced through responsive design principles, ensuring that the application works flawlessly across different screen sizes and devices. React components are styled using CSS modules and responsive layout techniques, allowing the UI to adapt dynamically to various environments, from desktop browsers to mobile devices.

In conclusion, this project exemplifies a full-stack development approach that combines modern technologies and frameworks to deliver a comprehensive solution. The synergy between frontend tools like React.js, backend frameworks like Django, powerful AI/ML integration through Gemini API, and advanced calculation modules using Python libraries, makes this system robust, intelligent, and user-centric.

With a focus on functionality, performance, and sustainability, the application not only addresses user needs effectively but also contributes to a larger goal of environmental awareness through its Carbon Footprint Calculator. The deliberate selection of tools, commitment to best practices, and emphasis on a seamless user experience position this project as a strong representation of contemporary software development capabilities.

modular integration with third-party services and APIs has been planned. For instance, integrating real-time weather APIs or transportation data could help enhance the accuracy of the Carbon Footprint Calculator, offering more precise results based on user location and activity patterns. This feature would empower users with location-aware insights, making the tool even more relevant and personalized. Additionally, language translation APIs can be integrated to support multilingual user interaction, making the chatbot accessible to a wider demographic and enhancing inclusivity.

To conclude, this multifaceted application is more than just a software project—it is a living system designed for growth, adaptability, and user empowerment. It skillfully combines state-of-the-art technologies such as React, Django, Gemini API, MySQL, and advanced Python libraries to deliver a holistic and intelligent solution. With a clear vision for the future, strong foundations in AI and web development, and a commitment to sustainability and accessibility, the system stands as a testament to the power of modern technology in addressing both digital and environmental challenges. Whether deployed in an educational context, corporate setting, or as a public tool, the application offers a meaningful blend of functionality, usability, and social impact.

CHAPTER - 4

4.Proposed System

The proposed Online Chatbot Bus Booking System with Carbon Footprint Insights using NLP represents a transformative step forward from traditional bus booking platforms. By integrating modern web technologies, Natural Language Processing (NLP), and intelligent carbon emission tracking, the system aims to deliver a user-friendly, efficient, and environmentally conscious solution. Unlike existing systems that often offer static interfaces and limited environmental data, the proposed platform emphasizes interactivity, personalization, and sustainability awareness.

At its core, the system will leverage the Flask web framework for backend development, ensuring lightweight and scalable performance. The inclusion of advanced NLP techniques—such as cosine similarity, TF-IDF vectorization, and intent recognition—enables the chatbot to understand user inputs in natural language, delivering relevant, accurate responses. This empowers users to inquire about bus availability, schedules, booking statuses, and even environmental impact in an intuitive conversational manner, reducing the reliance on complex navigation or forms.

Passengers will benefit from an accessible interface that allows quick ticket reservations by specifying their source, destination, date, and preferred travel time. The chatbot will provide intelligent assistance throughout the booking process, resolving queries and offering contextual recommendations such as the most eco-friendly route or the option with the least carbon emissions. Users can compare different travel routes based on estimated CO_2 output, distance, and time, fostering greater awareness and participation in sustainable travel choices.

The system will feature real-time updates on bus availability, route changes, and booking confirmations via automated notifications. Integration with a centralized database will allow dynamic access to bus schedules, seating availability, and route-specific emission data, ensuring a seamless and informed booking experience.

A dedicated admin panel will provide transport administrators with comprehensive tools for managing fleet information, analyzing user booking trends, updating route details, and monitoring overall system performance. Carbon footprint analytics will be visualized through dashboards.

protocols, role-based access controls, and secure data encryption will be implemented to protect user information, booking records, and environmental metrics. The system will align with data protection regulations and standards, applying best practices for information security and ethical AI deployment.

User experience will be enhanced through a responsive, mobile-friendly design using Bootstrap, ensuring compatibility across all devices. The interface will be built with accessibility in mind, incorporating intuitive navigation, voice-enabled chatbot interactions, and customizable display options to cater to diverse user needs.

Rigorous testing phases, including unit testing, usability testing, and system integration testing, will be conducted to ensure software reliability, performance, and user satisfaction. Continuous deployment pipelines and automated testing tools will be employed to maintain code quality and reduce deployment errors.

Advantages:

Efficiency: Automates the entire booking process, reducing human error and manual intervention, saving time for both users and admins.

Real-Time Updates: Provides real-time information on bus availability, schedules, and bookings, ensuring accurate and up-to-date details for users.

Secure Payments: Integrates secure payment gateways, ensuring safe and reliable transactions for users.

User-Friendly Interface: Features an intuitive, easy-to-navigate platform for users to book tickets, view history, and access customer support.

Admin Control: Empowers admins with a centralized dashboard for managing buses, schedules, and bookings, streamlining operations and improving efficiency.

4.1.Objectives:

- To develop a user-friendly online bus booking platform that simplifies the ticket reservation process for travelers.
- To enable secure user registration, login, and payment functionalities for a seamless booking experience.
- To allow users to search buses by route and date, with real-time data on availability and pricing.
- To integrate a chatbot powered by Gemini AI for real-time assistance and enhanced user interaction.
- To provide carbon emission insights for each bus journey, encouraging eco- conscious travel decisions.
- To offer a comprehensive admin dashboard for managing bus schedules, bookings, and customer data.
- To automate the tracking of carbon emissions for each route to support sustainable transport operations.
- To maintain booking history records, allowing users to view their past journeys and associated carbon footprints.
- To ensure data security and privacy through validated login and logout processes.
- To build a scalable system using Python, Django, React, and MySQL for future enhancements and maintenance.
- To implement multi-language support within the platform, making it accessible to users from diverse linguistic backgrounds.
- To integrate seat selection features with real-time visuals of bus layouts for enhanced user convenience.
- To ensure responsive web design for optimal performance across desktops, tablets, and mobile devices.
- To integrate with GPS-based tracking systems for live bus location updates and estimated arrival times.
- To include analytics and reporting features in the admin dashboard for monitoring system performance and user engagement.

4.2.Problem Formulation:

Objective: The primary objective of this project is to develop an intelligent, AI-driven chatbot system that streamlines bus ticket booking while providing users with real-time carbon footprint insights for their chosen routes. By integrating Natural Language Processing (NLP) techniques, the system aims to facilitate natural, human-like conversations to guide users through the booking process, provide travel suggestions, and promote environmentally friendly travel decisions. This system is designed to offer seamless, hassle-free bus ticket booking, reduce manual intervention, and encourage sustainable travel habits among passengers.

This project addresses the common challenges faced by users during online bus bookings, such as difficulty finding relevant routes, comparing travel options, and a lack of environmental awareness during travel planning. By offering contextual carbon emission data alongside route suggestions, the system aims to promote conscious travel decisions and contribute to broader sustainability efforts in public transportation.

Dataset: The dataset used in this project consists of historical bus travel records, including route information, schedules, fares, seat availability, and user preferences. Additionally, carbon emission data for different types of buses, distances, and routes are incorporated into the system, sourced from reliable environmental databases and APIs. This data is essential for calculating the estimated carbon footprint of each journey, ensuring accuracy and relevance when providing users with emission insights. User interactions and feedback are continuously collected to improve the chatbot's responses and enhance the accuracy of the NLP models over time.

Methodology:

The methodology combines advanced Natural Language Processing (NLP) models with machine learning techniques to create a responsive and intelligent chatbot. Initial pre-processing of user queries involves tokenization, stop-word removal, and entity recognition to ensure the chatbot can accurately interpret travel requirements. The system leverages Transfer Learning with pre-trained NLP models (such as BERT or spaCy) to enhance its ability to understand context and intent, reducing the need for extensive labeled data.

Once user intent is recognized, the chatbot interacts with the backend system to fetch route availability, fare information, and estimated carbon emissions for each travel option. By integrating an API for carbon footprint calculation or using pre-stored emission formulas, the system provides real-time emission insights alongside travel suggestions.

The chatbot employs reinforcement learning techniques to improve its performance based on user feedback, ensuring continuous optimization.

Model Training: The NLP models within the chatbot are trained using a labeled dataset of user queries, intents, and responses. These models are fine-tuned on domain-specific data to recognize various booking scenarios, such as ticket reservations, modifications, and cancellations. The machine learning models also include a component that evaluates user travel patterns and preferences to offer personalized route suggestions.

Additionally, a supervised learning model is employed to categorize different bus types (e.g., AC, non-AC) and calculate associated carbon emissions accurately. By integrating the emission calculation model with the NLP system, the chatbot can instantly provide travelers with both travel options and environmental impact data.

Model Evaluation: The performance of the chatbot and the emission calculation module is evaluated using metrics such as accuracy, precision, recall, and F1-score. These metrics ensure that the chatbot can correctly interpret user queries and provide accurate booking information. Evaluation through cross-validation ensures the model performs consistently across unseen data and user inputs. Feedback loops are integrated into the system to gather user responses, helping further refine the model and reduce false positives or misinterpretations.

Challenges: One of the key challenges faced during the development of this system is ensuring that the chatbot accurately understands the diverse language and phrasing used by users. Natural language is inherently complex and unpredictable, requiring the system to handle varying sentence structures, typos, and informal language. Another challenge is integrating real-time carbon emission data, as emissions can vary based on bus type, fuel efficiency, and route. Ensuring accuracy in carbon calculations while maintaining quick response times is a complex task**.**

Expected Outcome: The expected outcome is a highly efficient and intelligent chatbot system that simplifies the bus booking process while promoting eco-friendly travel decisions. The system will provide users with an easy-to-use interface for booking, modifying, or canceling bus tickets, supported by real-time updates and personalized travel suggestions. By including carbon footprint insights, the system will raise awareness about the environmental impact of travel and encourage users to choose greener options whenever possible.

In the long run, this AI-driven platform can serve as a blueprint for integrating sustainability into public transportation systems and other industries where environmental awareness can make a significant impact

4.3.Methodology of the Proposed Algorithm:

The Online Chatbot Bus Booking System with Carbon Footprint Insights is an intelligent, user-centric platform designed to streamline the process of reserving bus tickets while promoting environmental awareness through real-time carbon emission analysis. This system employs Natural Language Processing (NLP) to facilitate seamless human-like interactions, allowing users to search, book, modify, or cancel tickets simply by chatting with the AI-powered assistant. Using frameworks like Rasa or Dialog flow integrated with Python-based NLP libraries such as spaCy or Transformers, the chatbot interprets user queries, understands intent, and responds contextually, enhancing usability and accessibility.

The platform features a responsive web interface developed using HTML, CSS, and JavaScript frameworks like React.js or Angular, ensuring optimal user experience across devices. Real-time data on seat availability, routes, and schedules is fetched from a connected SQL (MySQL/PostgreSQL) or NoSQL (MongoDB) database, allowing accurate and dynamic interactions. One of the core innovations of the system is its carbon footprint calculator, which uses APIs or custom algorithms built with NumPy and math libraries to estimate CO_2 emissions for each journey based on travel distance, vehicle type, and occupancy.

These insights are presented to users during the booking process, encouraging eco-conscious travel decisions. On the backend,secure authentication and role-based access controls protect user data, while the admin panel offers tools for managing routes, analyzing user behavior, and monitoring emissions trends. Version control through GitHub and testing via tools like Postman and PyTest ensure code reliability and maintainability. Optional deployment on cloud platforms such as AWS or Google Cloud ensures scalability and uptime. Overall, this smart booking system not only enhances travel convenience but also supports environmental sustainability by guiding users toward greener travel choices through integrated carbon impact analytics.

Architecture:

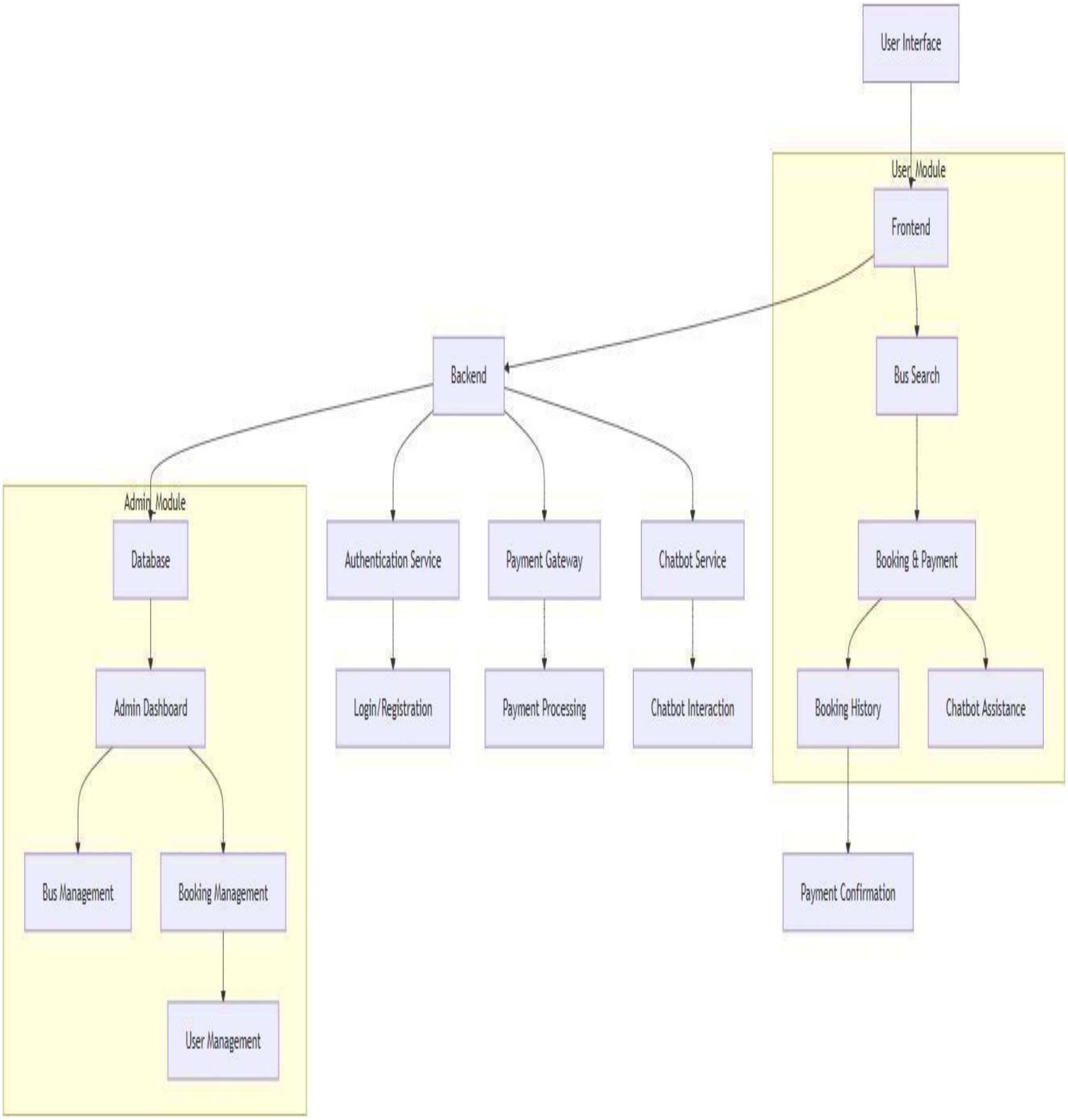


Fig 4.3: Architecture

The Online Chatbot Bus Booking System with Carbon Footprint Insights using NLP is designed with a modular architecture comprising three primary components: the User Module, the Admin Module, and Backend Services. These modules interact seamlessly to ensure a smooth, intelligent, and eco-conscious booking experience for users.

The User Module is tailored for end-users, providing an intuitive User Interface that connects to the Frontend system. Through this interface, users can search for buses via the Bus Search function and proceed to the Booking & Payment section. Here, they can make reservations, view their Booking History, and receive Payment Confirmation upon successful transactions. Additionally, users have access to Chatbot Assistance, which is powered by Natural Language Processing (NLP). This chatbot not only helps with booking queries and navigation assistance but also provides insightful data regarding the carbon footprint of different travel options, enabling users to make environmentally conscious travel decisions.

On the administrative side, the Admin Module offers robust control over system resources. At its core lies the Database, which stores essential data such as user information, bus schedules, and booking records. The Admin Dashboard acts as the central control panel, allowing administrators to manage three key areas: Bus Management, Booking Management, and User Management. This ensures that the platform remains updated with accurate schedules and user data, while also maintaining a smooth booking workflow.

The heart of the platform is its Backend Services, which support both user-facing and admin-facing modules. These services include the Authentication Service, which manages user Login/Registration processes to ensure secure access; the Payment Gateway, responsible for safe and efficient Payment Processing; and the Chatbot Service, which facilitates Chatbot Interaction. This backend chatbot integrates NLP algorithms to interpret user inputs naturally and provide relevant responses — including carbon emissions estimates based on selected travel routes.

In essence, the system architecture promotes a highly responsive and scalable environment for bus ticket booking, with an innovative edge provided by its integration of NLP and environmental awareness. It enhances user experience through automation and smart assistance while equipping administrators with powerful tools to maintain the platform’s functionality.

CHAPTER - 5

5.Dataset collection

In the rapidly advancing domain of intelligent transportation systems, the integration of deep learning, natural language processing (NLP), and real-time user interfaces has revolutionized the way public services operate. A compelling application of these technologies is found in the development of an Online Chatbot Bus Booking System with Carbon Footprint Insights, which enhances the travel planning experience while promoting environmentally responsible decisions.

This project is built around a multi-layered architecture that systematically organizes user and administrative functions along with backend services. The system is designed to offer a seamless, intelligent, and eco-conscious travel booking process. Much like constructing a robust dataset in computer vision, this project emphasizes the structured compilation of user interactions, service modules, and database integration, ensuring high reliability and functionality.

The user experience begins at the User Interface, where individuals interact with the Frontend component. Here, they can search for buses, make bookings, process payments, and access chatbot assistance. The chatbot, powered by advanced NLP techniques, understands user queries in natural language and responds with accurate travel information, booking steps, and crucially, carbon footprint insights for selected bus routes. This allows users to not only choose routes based on cost and time but also on environmental impact, fostering greener travel choices.

The system’s Admin Module plays a vital role in ensuring operational efficiency and accuracy. Through a centralized Admin Dashboard, administrators manage essential services such as Bus Management, Booking Management, and User Management. This includes updating bus schedules, verifying bookings, and monitoring user activity, all of which are stored and retrieved from a connected Database. This layered management structure guarantees that the system remains accurate and up to date.

Central to the platform is the Backend, which supports core operations like Authentication Services for secure Login and Registration, a Payment Gateway for handling transactions, and the Chatbot Service which interprets user input and delivers real-time assistance. These services operate concurrently, ensuring a responsive and secure user experience.

An innovative aspect of the system lies in its NLP-enabled chatbot, which not only assists with bookings but also educates users on the carbon emissions associated with different travel options. For instance, the chatbot can compare a chosen bus route with alternative transport modes, highlighting their respective environmental impacts. This functionality mimics the detailed, layered learning process seen in deep learning models, where the chatbot evolves to understand context and intent over time through user interaction data.

To accommodate variability and real-world usage conditions, the system is designed to handle diverse user inputs, various environmental settings, and different device interfaces. This robustness mirrors the approach used in preparing deep learning datasets, where diversity and real-world relevance are key to building generalizable models.

Moreover, the system can be enhanced through Transfer Learning, particularly within the NLP domain. By leveraging pre-trained language models (like BERT or GPT), the chatbot can understand and respond to user queries more effectively, even with limited domain-specific training data. This boosts the accuracy and responsiveness of the system while reducing the time required for development and training.

In summary, the Online Chatbot Bus Booking System with Carbon Footprint Insights using NLP is a technologically sophisticated platform that merges intelligent user interaction with sustainable travel awareness. Through a combination of NLP, user-friendly interfaces, admin control modules, and backend integration, it not only simplifies bus bookings but also empowers users to make informed, eco-conscious decisions.

CHAPTER - 6

6.System Design

The system design of the proposed Online Chatbot Bus Booking System with Carbon Footprint Insights Using NLP is meticulously crafted to integrate advanced artificial intelligence technologies for intelligent, real-time user interaction and eco-conscious decision-making. The architecture comprises several modular components, each fulfilling a specific function—from user input and interaction to route suggestions, booking, payment, and environmental impact analysis.At the foundation of the system is a robust user interface that connects to a comprehensive backend infrastructure.

The User Module allows individuals to search for buses, view route options, and complete bookings with real-time assistance from an AI-powered chatbot. This chatbot serves as the central interaction point, using Natural Language Processing (NLP) to understand and process user queries in natural, conversational language.To effectively support the chatbot's capabilities, a curated dataset of user interactions, travel routes, and carbon emission profiles has been compiled.

This dataset includes sample dialogues, user preferences, and bus route data tagged with their corresponding carbon footprint values. These labels are critical in enabling the supervised learning processes that train the chatbot to provide not only functional travel assistance but also environmental insights.The dataset undergoes several preprocessing steps including text normalization (removing noise like punctuation and irrelevant terms), tokenization, and embedding, where user inputs are converted into numerical formats suitable for machine learning models. These steps ensure that the NLP engine can interpret and respond to a wide variety of user intents effectively.

At the heart of the chatbot system is a pre-trained NLP model, such as BERT or GPT, which has been fine-tuned on domain-specific dialogue and transportation data. This technique, known as Transfer Learning, leverages the general language understanding capabilities of large models and customizes them to suit our specific task—bus booking and carbon awareness. Transfer Learning drastically reduces training time while improving response accuracy .

To ensure robust performance, the dataset is divided into training, validation, and testing sets. The training set is used to teach the model user intents and query patterns, the validation set helps fine-tune model parameters, and the testing set assesses the final performance. Techniques such as dropout and learning rate scheduling are used during training to prevent overfitting and improve generalization, especially since the amount of labeled domain-specific data may be limited.

Once trained, the NLP model is deployed within the system's backend and integrated with various services including bus schedule APIs, payment gateways, and carbon emission calculators. When a user inputs a query like "Find me a bus from City A to City B tomorrow," the chatbot processes this using the NLP engine, retrieves available bus options, and calculates the estimated carbon footprint for each. It then responds with recommendations, allowing the user to make an informed choice based on time, cost, and environmental impact.

Upon selecting a bus, the user is guided through a seamless booking and payment process. The chatbot assists with input validation, error handling, and payment confirmation. A final summary is shown, including the carbon impact of the journey and tips for greener travel where applicable.This intelligent and environmentally conscious system empowers users not just to book travel efficiently, but to make sustainable transportation decisions. By combining modern AI with real-time interaction and ecological awareness, the platform provides a forward-thinking solution to urban mobility and climate-conscious travel planning.

6.1.High-Level Design Documentation:

Use Case Diagram :

The use case diagram for the project "Online Chatbot Bus Booking with Carbon Footprint Insights Using NLP" illustrates the various interactions between users and the system, highlighting the system's core functionalities. The primary actors involved are users (passengers), administrators, and optionally bus operators. The system allows users to interact through a chatbot interface powered by Natural Language Processing (NLP), enabling a more conversational and user-friendly experience. Key use cases for users include searching for bus availability, booking bus tickets, canceling bookings, viewing booking details, and receiving personalized travel suggestions through the chatbot. An important feature of the system is its ability to calculate the carbon footprint of each journey, helping users make eco-conscious travel choices by offering greener alternatives. Administrators can manage bus routes and timings, as well as generate usage and carbon emission reports to monitor and optimize system performance. Each use case represents a specific action performed by an actor, providing a clear visualization of the system's functionality from the user's perspective

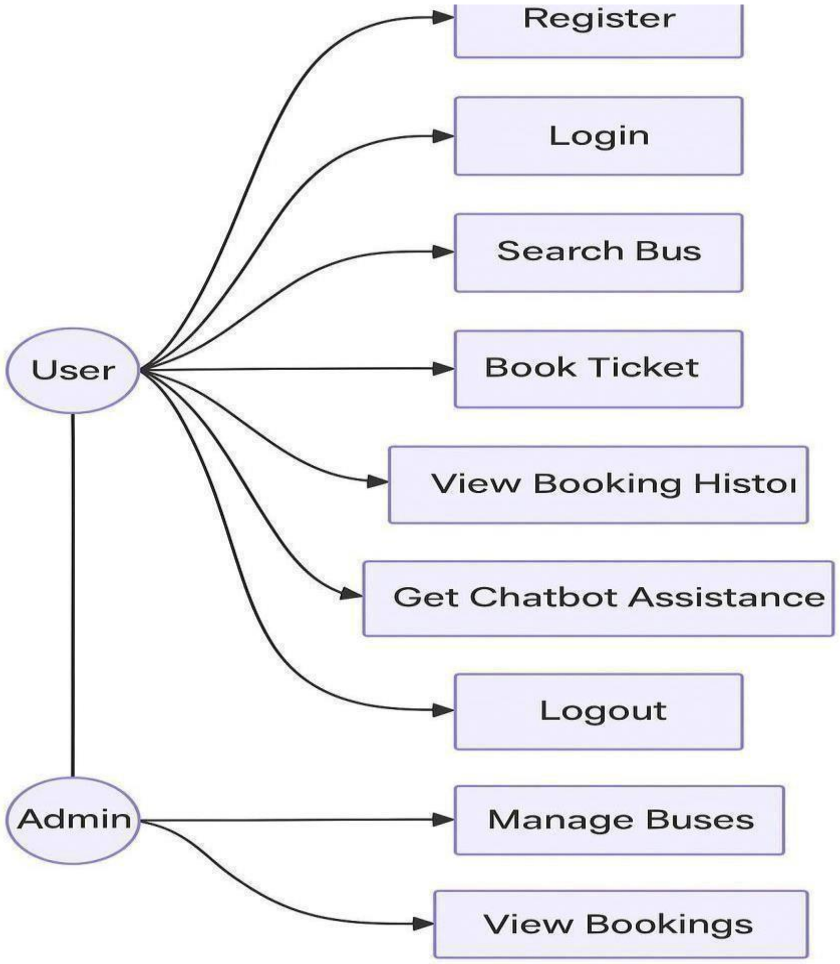


Fig.6.1: Use case diagram

Class Diagram:

The class diagram for the project depicts the various classes and their relationships within the system. It provides a structural overview of the system's components, including entities such as User, Chatbot, Bus, Booking, Route, Payment, Carbon Footprint Calculator, and Admin. Relationships between classes, such as one-to-one, one-to-many, or many-to-many associations, are represented using appropriate symbols like solid lines for associations and arrows for directional relationships. For instance, a User can make multiple Bookings, and each Booking is associated with a specific Bus and Route.

The Chatbot class interacts with the User to process natural language queries and respond with relevant actions or information. The Carbon Footprint Calculator class calculates emissions based on route distance and bus type, linking with the Booking or Route class to provide eco-friendly travel insights. Attributes and methods of each class are also specified, highlighting their roles and functionalities within the system. For example, the Booking class may include attributes such as booking ID, travel Date, and seat Number, along with methods like confirm Booking() and cancel Booking(). The class diagram serves as a blueprint for implementing the system's object-oriented design, guiding developers in organizing and modeling the system's components effectively.

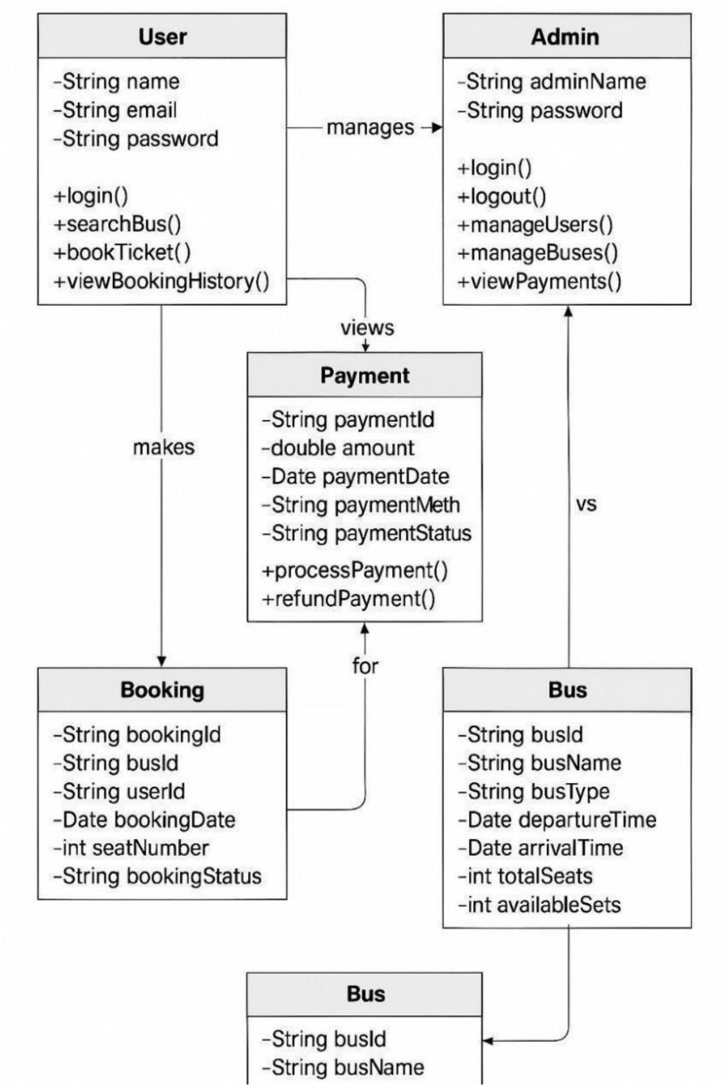


Fig.6.2: Class diagram

Sequence Diagram:

The sequence diagram for the project illustrates the interactions between various components of the system over time. It showcases the sequence of messages exchanged between objects or entities, depicting the flow of control and data during specific use cases or scenarios.

In the context of this system, a typical scenario might involve a User interacting with the Chatbot, which in turn communicates with components such as Bus Service, Booking Manager, Route Manager, Carbon Footprint Calculator, and Payment Gateway. Each message exchange is represented as a vertical lifeline for the corresponding object, with arrows indicating the direction of communication.

For example, when a user initiates a booking request via the chatbot, the chatbot parses the natural language input, fetches available bus options from the Bus Service, confirms route and availability through the Route Manager, and then initiates the booking via the Booking Manager.

After that, the Carbon Footprint Calculator estimates the environmental impact, and finally, the Payment Gateway handles the transaction. Sequence diagrams help visualize the dynamic behavior of the system, including the order of method calls, parameter passing, and responses between objects. They are invaluable for understanding the system’s runtime behavior and for identifying potential bottlenecks or inefficiencies in the communication flow.

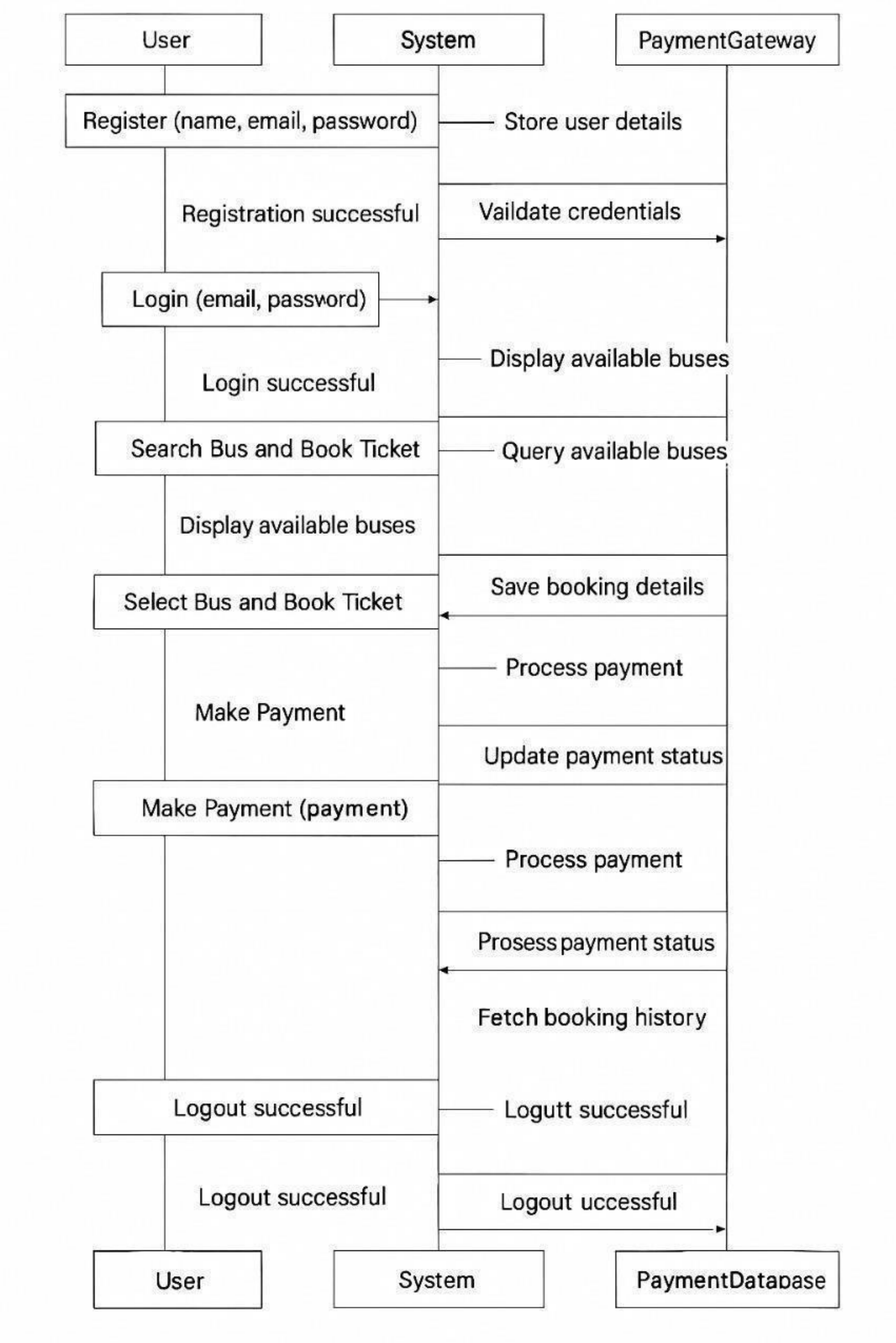


Fig.6.3: Sequence diagram

Activity Diagram:

The activity diagrams for the project provide a visual representation of the flow of activities or processes within the system. They depict the sequence of actions performed by users, system components, or external entities to accomplish specific tasks or use cases.

In the context of the chatbot bus booking system with carbon footprint insights, activity diagrams may represent processes such as booking a bus ticket, cancelling a reservation, or calculating carbon emissions. Each activity is represented as a node in the diagram, connected by arrows to show the flow of control from one activity to another.

For example, a typical booking flow may start with the user entering a query into the chatbot, followed by NLP-based query processing, bus search, display of results, user selection, carbon footprint calculation, payment processing, and confirmation. Decision points and branching paths illustrate different possible scenarios or conditions, such as route availability, payment success, or user preference for eco-friendly travel.

Activity diagrams help in understanding the workflow of the system, identifying potential bottlenecks or inefficiencies, and designing optimized processes. They are valuable for both developers and stakeholders to visualize the system’s behavior and ensure alignment with business requirements.

Activity Diagrams are Used for:

Modeling business workflows

Modeling operation of a system

Representing detailed steps in use cases

Visualizing logic behind complex algorithms

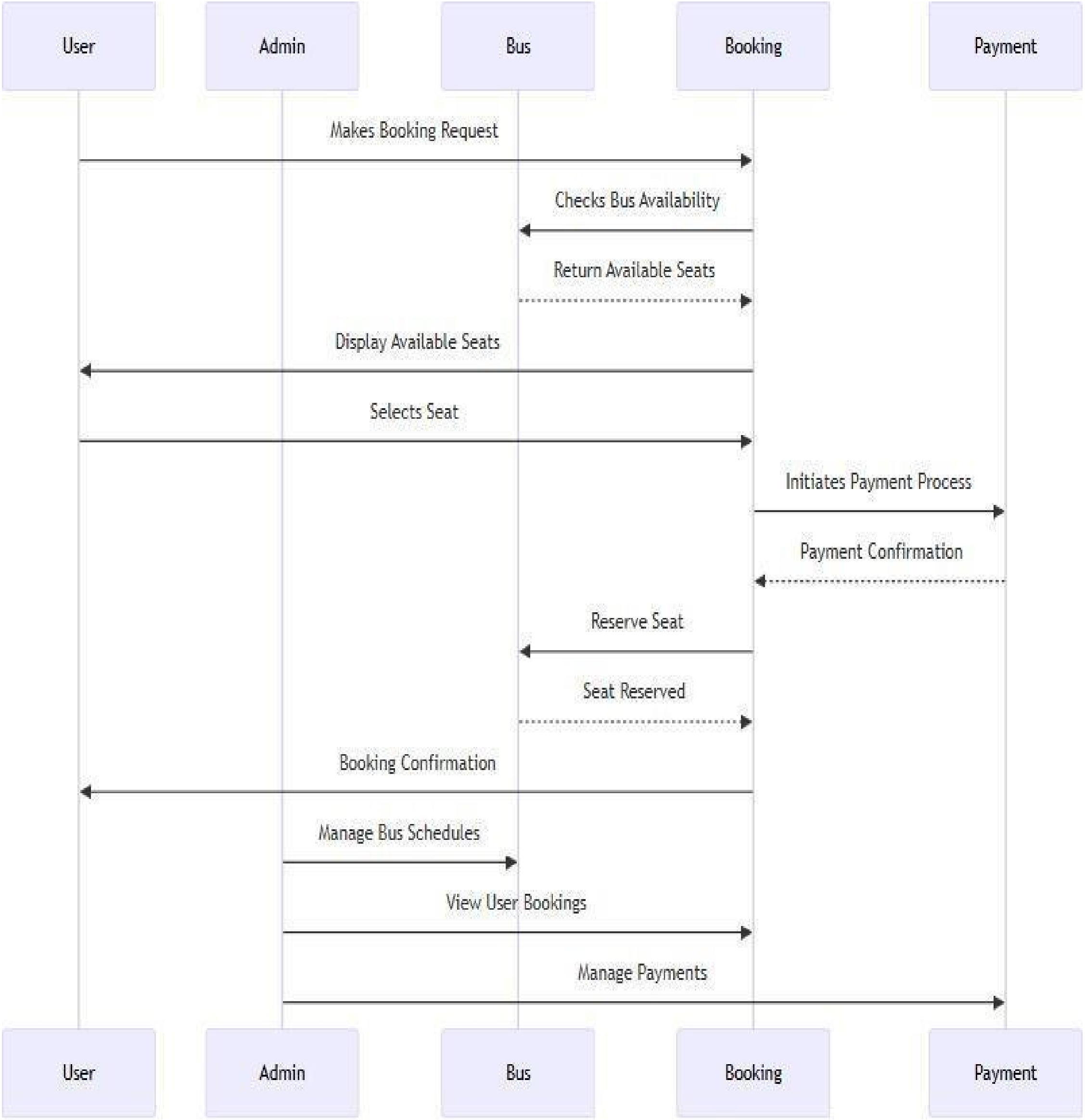


Fig.6.4: Activity Diagram

6.2.System Flow Chart

The system design of the Online Chatbot Bus Booking System with Carbon Footprint Insights using NLP is structured with a clear and logical flow to ensure seamless interactions between users and administrators. At its core, the system supports two primary roles—Users and Admins—each with distinct responsibilities and access privileges. The user journey begins with a registration process, where personal details are submitted and securely stored, enabling account creation and personalized access. Once registered, users proceed to login, granting them access to the system's main functionalities.

Upon successful authentication, users can search for buses based on criteria such as location, date, time, and type of service. The search interface is enhanced by a Natural Language Processing (NLP)-powered chatbot that assists users in formulating queries conversationally. Once a suitable bus is selected, users can book tickets directly through the platform, followed by payment processing, which supports secure gateways to complete transactions. Each transaction and booking is stored, allowing users to view their booking history, which ensures transparency and helps manage travel schedules effectively.

To support real-time interaction and resolve queries, the system incorporates a chatbot assistant. This AI-powered module leverages NLP to understand and respond to user inquiries regarding booking, schedules, cancellations, and even carbon footprint information. By estimating the carbon emissions of each journey, users are informed about their environmental impact, promoting more sustainable travel decisions.

The logout function ensures secure session termination, protecting user data and preventing unauthorized access. On the administrative side, admins log in through a dedicated interface to manage backend operations. They have access to modules such as Manage Buses, where they can update bus schedules, routes, and availability. The View Bookings feature gives administrators a comprehensive overview of all ticket sales and passenger data, which is crucial for operational planning.

Admins can also Manage Payments, ensuring all transactions are monitored and reconciled. Additionally, the Manage Users module allows for the oversight and maintenance of user accounts, handling tasks such as access control, account verification, and user support.

In terms of system intelligence, the chatbot is a centerpiece, employing NLP models to engage users conversationally. It provides human-like interaction to help with ticket booking, answering FAQs, and educating users on the carbon footprint of their selected routes. This functionality not only enhances usability but also aligns with environmental awareness goals.

Overall, the system integrates a robust, user-centric design supported by AI-driven insights and secure, scalable backend operations. From initial user onboarding to final ticket confirmation and post-travel feedback, the platform offers a smooth, informed, and eco-conscious experience tailored to modern travel needs.

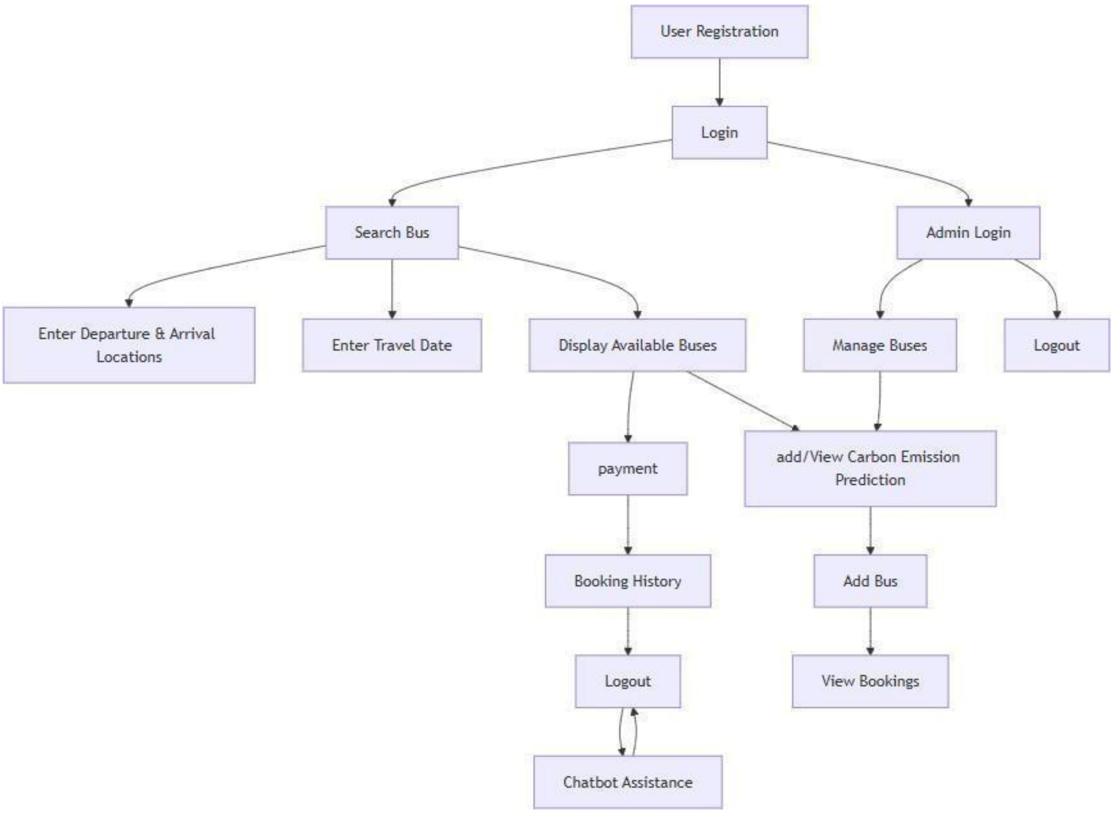


Fig: 6.2: System flow diagram

CHAPTER - 7

7.Implementation:

To implement an intelligent and eco-conscious online bus booking system enhanced with a chatbot and carbon footprint insights using NLP, the development process follows a detailed and structured approach. The journey begins with data collection, where relevant datasets are sourced from transport APIs and online platforms, including bus schedules, routes, user preferences, ticket prices, and environmental impact data such as CO_2 emissions per trip. This dataset is carefully organized and divided into training and testing sets—typically using an 80/20 or 70/30 ratio—to enable machine learning models to train efficiently while reserving a portion of the data to evaluate the system's performance. Each user query and booking record is preprocessed to standardize formats, ensure consistent input for the NLP engine, and normalize variables such as distance, fare, and emissions.

The system also benefits from data augmentation techniques in the NLP pipeline, such as paraphrasing and synonym replacement, to better understand user inputs phrased in different ways.For feature extraction, the system utilizes pre-trained NLP models, such as BERT or similar transformers, which are fine-tuned to interpret natural language queries related to bus travel— like "Find me a night bus to Chennai" or "Show eco-friendly travel options." These models are trained on general text corpora and then customized for the transportation domain. Early layers of the NLP model are kept frozen initially to retain general language understanding, while the top layers are retrained to capture context-specific meanings. Simultaneously, a machine learning component is trained to estimate carbon emissions for various bus routes based on factors such as distance, bus type, and occupancy rate.

During the training phase, these models are integrated into a backend system where the NLP engine classifies user intents (e.g., book ticket, check schedule, compare routes) and entities (e.g., cities, dates). The chatbot uses this classification to generate relevant responses and route the query appropriately—either to fetch bus options, display ticket prices, or show the carbon footprint of different routes. Softmax activation functions in the classifier ensure probability-based response accuracy for multiple user intents. The chatbot and carbon calculator models are trained iteratively using labeled data and evaluated using metrics such as precision, recall, and F1-score to refine accuracy.

Once training is complete, the classification system can understand user queries in real time, respond with contextual information, and suggest booking options. For each query, the system processes the input, predicts the intent, and provides output via the chatbot interface, including bus options and the estimated carbon footprint. The system highlights eco-friendly routes, encouraging users to choose lower-emission travel alternatives.

The evaluation phase includes testing the chatbot and carbon footprint modules with unseen queries and real-world scenarios. A confusion matrix helps identify errors in user intent prediction, guiding model improvements. Cross-validation ensures robustness across diverse user inputs and route conditions. Finally, the system is deployed as a web-based application with user-friendly interfaces for registration, login, and profile management, backed by secure databases like MySQL to store user and transaction data.

The NLP-enhanced chatbot is integrated via APIs, and the carbon footprint engine is hosted on a scalable cloud platform, enabling real-time access for travelers, transport authorities, and eco-conscious users alike.This end-to-end implementation ensures a smart, sustainable, and interactive bus booking experience that empowers users to make informed travel decisions with minimal environmental impact.

7.1Platform/Technologies used:

Python (Version 3.8.6):

Python serves as the primary programming language for developing the Online Chatbot Bus Booking with Carbon Footprint Insights system. Ensure Python is installed on the system to execute scripts and run the backend services, including the chatbot and NLP modules.

Flask Framework (Version 2.0.1)

Flask is a lightweight Python web framework used to build the backend of the bus booking application. It offers tools for URL routing, API integration, and request handling, facilitating the development and deployment of the chatbot and web-based booking platform .

Frontend Development:

In frontend development, React.js is utilized to create responsive and dynamic user interfaces for web applications. It allows for efficient component-based architecture and seamless user experiences. Alongside React, core technologies such as HTML, CSS, and JavaScript are employed to structure and style the frontend, ensuring that the design is both visually appealing and functional across various devices and screen sizes.

Backend Development:

In backend development, Python with Django is used to handle business logic, data flow, and database interactions efficiently. Flask is sometimes used for building lightweight microservices or simpler routes. These frameworks enable smooth integration with APIs and ensure robust server-side functionality.

Carbon Footprint Calculator:

Custom Calculations: Done using NumPy and math libraries. API Integration, may use external services to fetch emission factors based on travel distance, bus type, etc.

7.2.System Testing:

System testing is a critical phase in the software development lifecycle where the entire, integrated application is tested to ensure it functions as expected under real-world conditions. For the Online Chatbot Bus Booking System with Carbon Footprint Insights Using NLP, system testing validates whether the complete setup—from natural language user input to route prediction, ticket booking, and carbon footprint estimation—operates smoothly and delivers accurate results. This system integrates a variety of technologies including a Python-based backend, an NLP engine, machine learning models for carbon estimation, a user-facing web interface, and a relational database like MySQL for storing user and transactional data. As such, comprehensive system testing is essential to ensure that all components—frontend, backend, prediction model, and database—work cohesively.

TYPES OF TESTS:

Unit testing:

Unit testing is especially significant in this system as developers often follow a combined code- and-test approach. During development, each component or feature is coded and immediately tested using unit test scripts. This practice, aligned with Test-Driven Development (TDD), promotes early bug detection, better software design, and ease of future code refactoring. Immediate feedback from unit tests ensures that functions like query and emissions scoring behave accurately before they are integrated into the broader application..

Integration testing:

Integration Testing is a vital stage in the testing lifecycle where individual modules of the application are combined and tested as a group to ensure they interact correctly. In the context of the Online Chatbot Bus Booking System with Carbon Footprint Insights Using NLP, integration testing plays a crucial role in verifying that all subsystems—such as the natural language processing engine, the route and fare calculation modules, the carbon emission estimator, user authentication system, and the MySQL database—work together seamlessly. For example, when a user queries the chatbot for a bus ticket, the chatbot must correctly interpret the intent using NLP, pass the parsed information to the backend system to fetch route options, calculate carbon emissions for each route, and then return this data to the user interface in a structured and readable format.

Functional testing:

functional testing ensures that the system's features meet the functional requirements. This includes testing valid and invalid user inputs, confirming that the chatbot understands various travel-related queries, validating that the system offers appropriate bus routes, and checking that emissions insights are accurately computed and displayed. The test cases are driven by real-world user scenarios and are mapped directly to system specifications, ensuring that each function—such as ticket booking, login/logout, and CO_2 insights—is thoroughly verified.

White Box Testing:

white box testing and black box testing are employed. White box testing involves analyzing the internal workings of key components like the NLP model's classification accuracy or the CO_2 calculator's logic, while black box testing evaluates the system from the user's perspective without knowledge of the internal code, validating inputs and expected outputs for various user commands., what it is meant to do. It serves a purpose. It is employed to test regions that are inaccessible from a level of the black box

Black Box Testing:

Testing software "black box" means doing it without having any idea of the inner workings, architecture, or language of the module being tested. such the majority of other test types, black box tests also need to be written from an official source document, such a specification or requirements document. In this test, the program being tested is handled as if it were a black box. It is impossible to "see" inside. Without taking into account the operation of the software, the test generates inputs and reacts to outputs.

7.3.Testing Strategies:

Testing Strategies form the backbone of quality assurance in any software project, ensuring that the application is reliable, functional, and user-friendly. For the Online Chatbot Bus Booking System with Carbon Footprint Insights Using NLP, a combination of different testing strategies is implemented to thoroughly evaluate each component and the overall system. Unit Testing is used early in the development process to test individual components such as the NLP parser, booking logic, and carbon footprint calculator, ensuring each function performs as expected in isolation. Following this, Integration Testing is applied to check the interaction between modules—verifying that the chatbot correctly communicates with the booking engine, and the system accurately computes and returns carbon footprint data from the route selected.

System Testing is then conducted to validate the full end-to-end functionality, mimicking real-world user interactions from inputting travel queries to receiving booking confirmations and carbon insights. Acceptance Testing is carried out with real users or stakeholders to ensure the final product meets the business and user requirements. In addition, Regression Testing is implemented regularly to ensure that new updates or added features do not negatively impact existing functionalities. Both Black Box Testing (testing without knowledge of internal code) and White Box Testing (testing with internal logic visibility) are employed where appropriate to maximize test coverage. These comprehensive testing strategies collectively ensure the robustness, accuracy, and efficiency of the chatbot bus booking system.

Test objectives:

i. Verify that all functional and non-functional requirements of the chatbot bus booking system are correctly implemented..

ii. Ensure accurate NLP interpretation of user queries for effective bus booking and carbon footprint insights..

iii. Validate integration between chatbot, backend logic, database, and carbon footprint module

iv. Identify and eliminate bugs or inconsistencies across different components and user flows.

Features to be tested:

1. Verify that the entries are of the correct format
2. No duplicate entries should be allowed
3. All links should take the user to the correct page.

Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

Acceptance Testing

Acceptance Testing, on the other hand, was conducted with direct involvement from end users to validate that the system aligns with real-world needs and business requirements. This phase focused on confirming that users could perform key functions such as interacting with the chatbot, booking tickets, receiving travel options, and viewing carbon footprint insights. The user acceptance test results showed that all functionalities met user expectations and business objectives, with no issues encountered during testing.

7.4. Test cases:

Here are some test cases you can consider for your medicinal leaf classification project:

Test Case Type	Description	Test Steps	Expected Result	Status
Functional Testing	Bus Search and Booking	1. Enter source, destination, and date. 2. search buses. 3. Book a ticket	Available buses are listed with carbon footprint data; booking completes successfully.	Pass
Functional Testing	Carbon Footprint Display	1.search buses. 2.Carbon Footprint Display	System should accept only valid image formats and reject others.	Pass
Functional Testing	Chatbot Response – Booking Help	1. Ask chatbot to book a bus.	Chatbot guides user through booking steps smoothly.	Pass

Functional Testing	Secure payment	1. Book a ticket. 2. Complete payment.	Payment is encrypted, and confirmation is Received without data leakage.	Pass
Functional Testing	Booking history view	1. Lo gin. 2. Check booking history	List of past bookings with carbon emission info shown correctly.	Pass
Security Testing	User Data Privacy	1. try accessing booking/payment info without authentication	System blocks unauthorized access.	Pass

Security Testing	User Data Privacy	1.Try accessing booking/payment info without authentication..	System blocks unauthorized access.	Pass
Usability Testing	NLP Language Understanding	1. Use casual/complex sentences with chatbot (e.g., "Book me a bus tomorrow to Mumbai").	Chatbot understands intent and responds appropriately.	Pass
Performance Testing	Chatbot Response Time	1. Submit a query. 2. Measure response time.	Chatbot replies within2 seconds.	Pass

performance Testing	System Load with Multiple Users	1. Simulate 100+ users searching and booking simultaneously.	Systemremains stable without crashing.	Pass
Integration Testing	Carbon Calculation Accuracy	1. Manually calculate carbon footprint for a journey. 2. Compare with system's prediction.	Systemcalculation should match manual results within acceptable error margins.	Pass
Performance Testing	Response Time	1. Measure system's response time for image classification.	System should classify images within an acceptable time frame.	Pass

Fig: 7.1: Test cases

7.5. Results :

Home Page:

Welcome to our intelligent online chatbot bus booking system, where users can easily search, chat, and book their bus journeys with added awareness of their carbon footprint. The home page offers an interactive experience powered by NLP, enabling real-time support and smart travel suggestions. Users are greeted with a seamless interface that combines convenience with sustainability. This page sets the tone for an eco-conscious journey by allowing travelers to track and understand the environmental impact of their bookings from the very start.

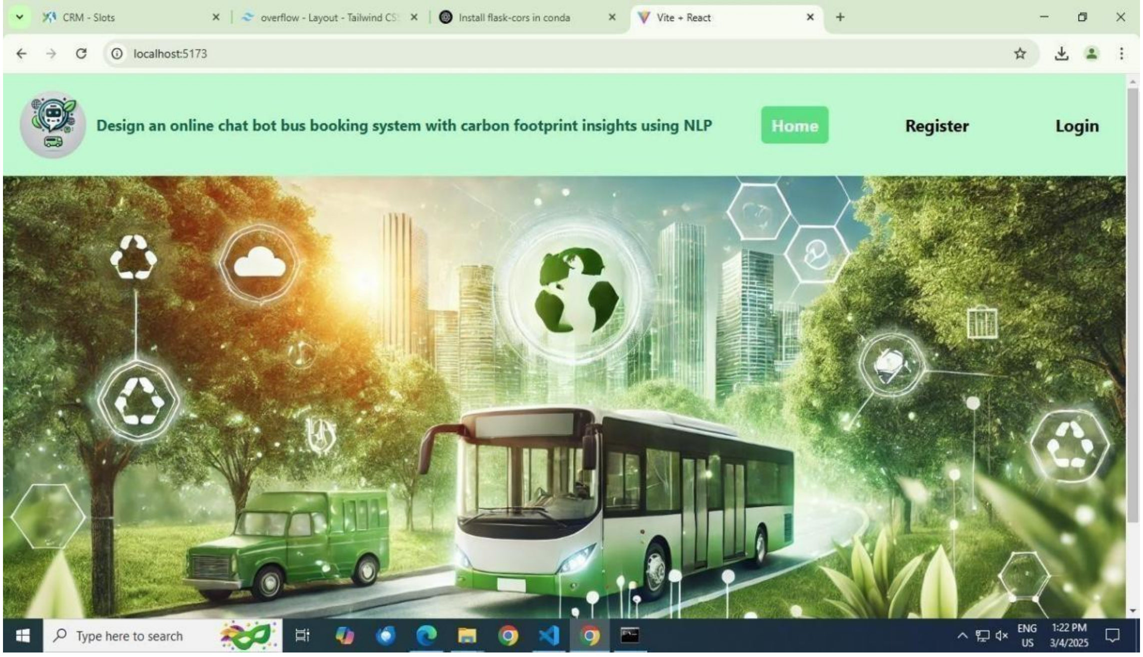

CRM - Slots
overflow - Layout - Tailwind CSS
Install flask-cors in conda
Vite + React
localhost:5173
Design an online chat bot bus booking system with carbon footprint insights using NLP
Home
Register
Login
Type here to search
ENG US
1:22 PM 3/4/2025

Fig.7.1:Home Page

Register Page:

The Register page invites users to create an account and unlock personalized features tailored to their travel needs and environmental goals. With just a few details, users can start enjoying faster bookings, access to travel history, and carbon emission insights. The NLP-powered chatbot assists during registration to ensure a smooth onboarding experience. It’s the first step toward smarter and greener travel planning.

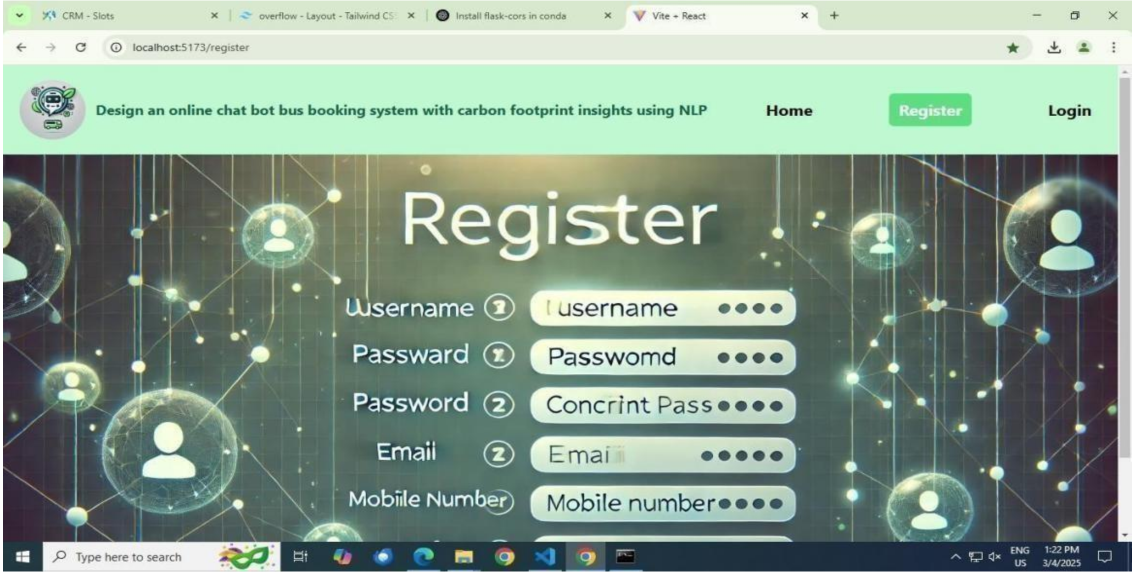


Fig.7.2:Register page

Login Page:

On the Login page, users can securely access their profiles to manage bookings, review travel stats, and track their carbon emissions. The system offers personalized travel suggestions through the integrated chatbot, helping users plan better routes. Once logged in, travelers can explore past bookings, view emission summaries, and continue their journey toward environmentally responsible travel. The page ensures quick access and data security while keeping sustainability in focus.

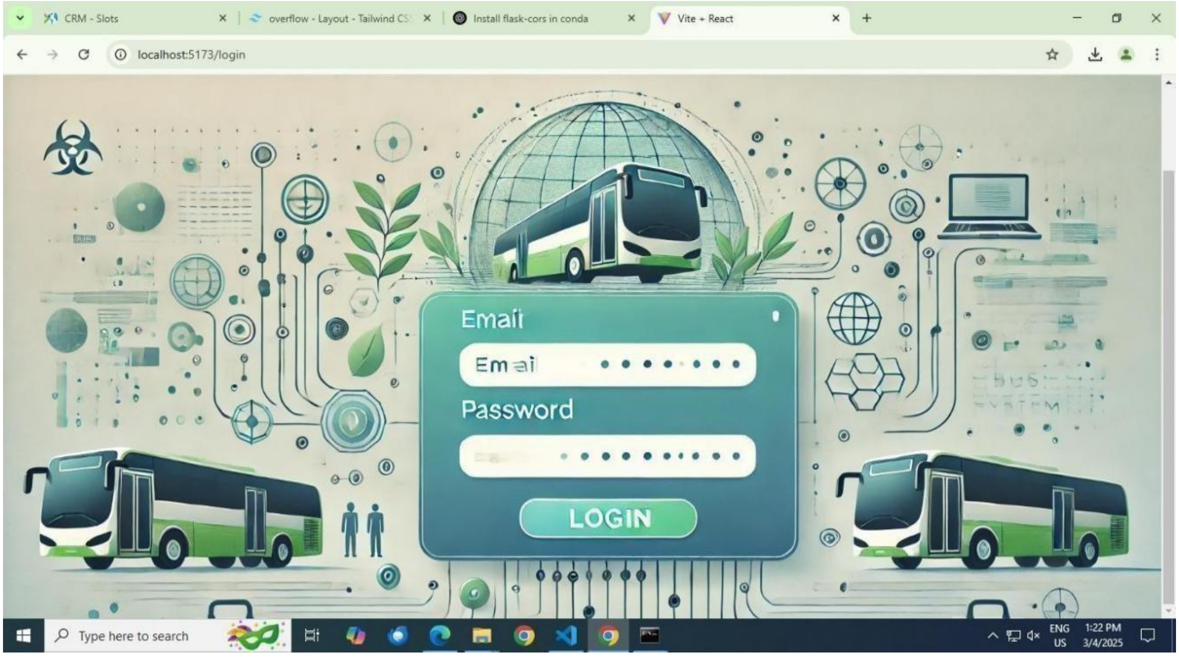
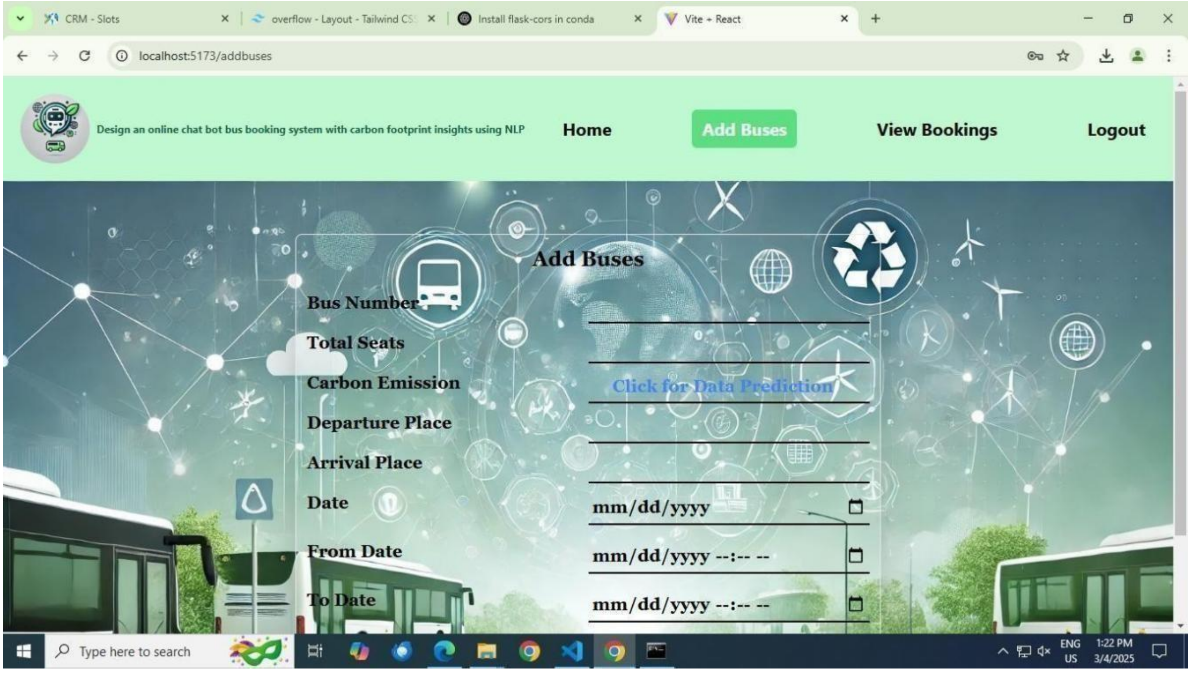


Fig.7.3: Login page

Add buses:

The Add Buses page is designed for admins to input and manage bus details including routes, capacities, and fuel types, which directly influence carbon emission calculations. This section ensures that the backend stays updated for accurate passenger information and eco-impact estimates. Admins can quickly edit or remove buses, and all changes are integrated with the chatbot to reflect updated data in real-time. It supports smooth operation and better user experiences.


CRM - Slots
overflow - Layout - Tailwind CS
Install flask-cors in conda
Vite + React
localhost:5173/addbuses
Design an online chat bot bus booking system with carbon footprint insights using NLP
Home
Add Buses
View Bookings
Logout
Type here to search
ENG US
1:22 PM 3/4/2025

Add Buses

Bus Number

Total Seats

Carbon Emission: Click for Data Prediction

Departure Place

Arrival Place

Date: mm/dd/yyyy

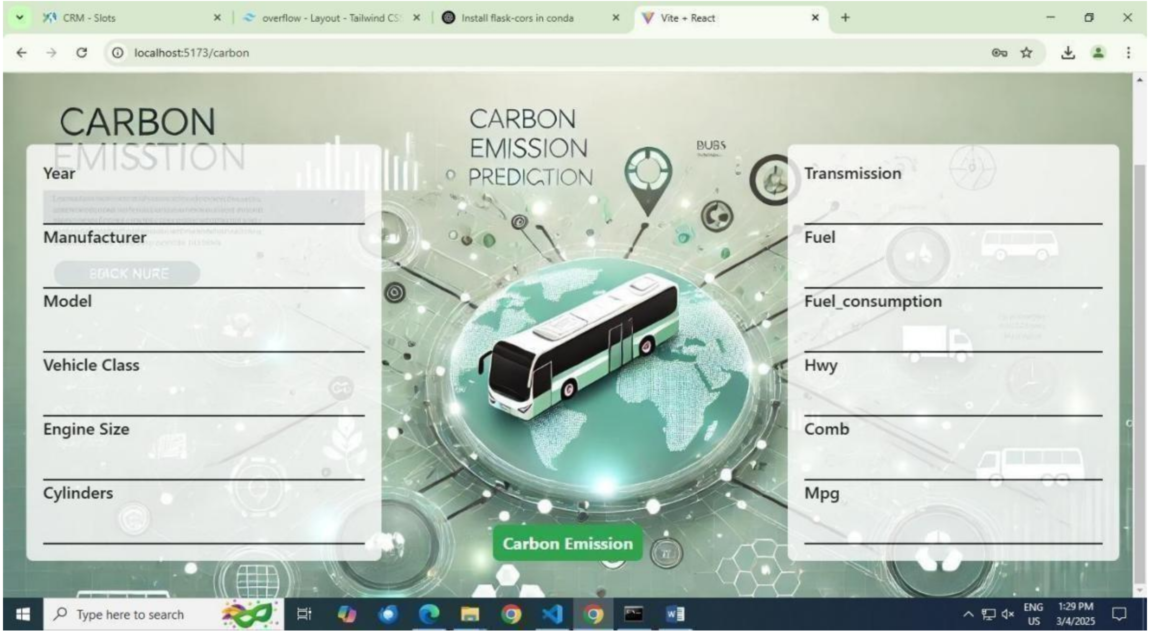
From Date: mm/dd/yyyy --:-- --

To Date: mm/dd/yyyy --:-- --

Fig.7.4: Add buses

Find Carbon Emission Page:

This page allows users to calculate and compare the carbon emissions of different bus journeys based on routes, bus types, and fuel consumption. It offers a visual and data-driven breakdown of how each journey affects the environment. Using NLP, the chatbot can interpret user queries and offer tips to reduce emissions. Travelers are empowered to make conscious decisions and choose greener alternatives before confirming their bookings.



Year

Manufacturer

Model

Vehicle Class

Engine Size

Cylinders

Transmission

Fuel

Fuel_consumption

Hwy

Comb

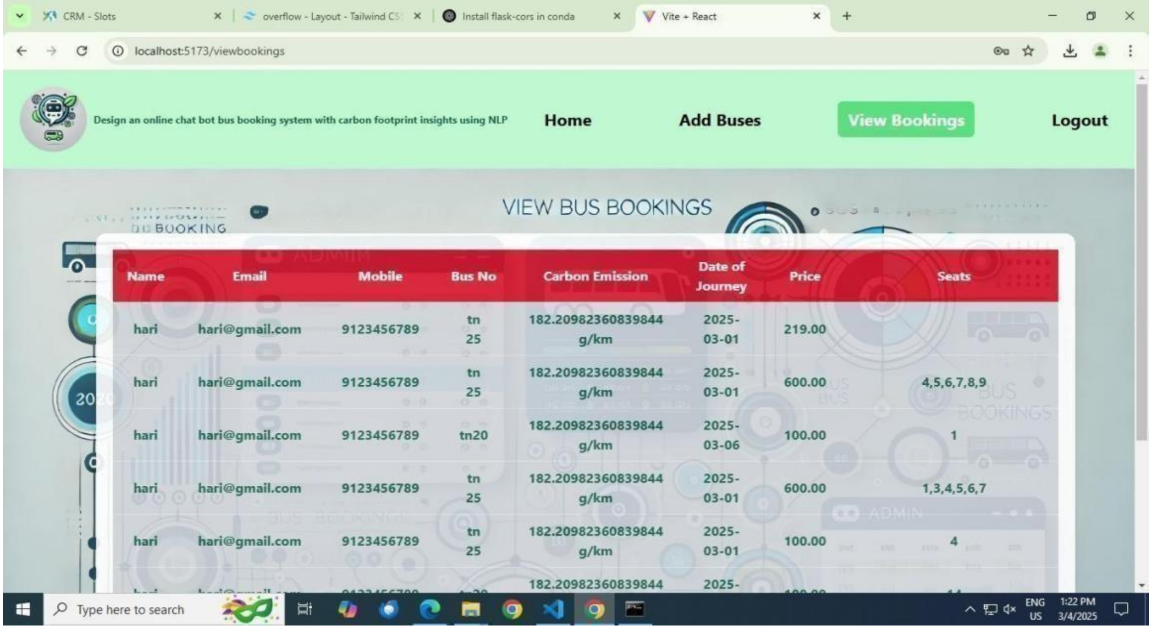
Mpg

Carbon Emission

Fig.7.5:Find carbon emission page

View Bus Bookings Page:

The View Bus Bookings page provides a comprehensive overview of all current and past bookings made by the user, along with detailed insights into the carbon emissions generated by each trip. Users can interact with the chatbot to get summaries or modify their bookings instantly. The platform emphasizes eco-consciousness by clearly displaying the impact of travel. It serves as a dashboard for both journey planning and environmental tracking.


CRM - Slots
overflow - Layout - Tailwind CS
Install flask-cors in conda
Vite + React
localhost:5173/viewbookings
Design an online chat bot bus booking system with carbon footprint insights using NLP
Home
Add Buses
View Bookings
Logout
BOOKING
Type here to search
ENG
US
1:22 PM
3/4/2025

VIEW BUS BOOKINGS

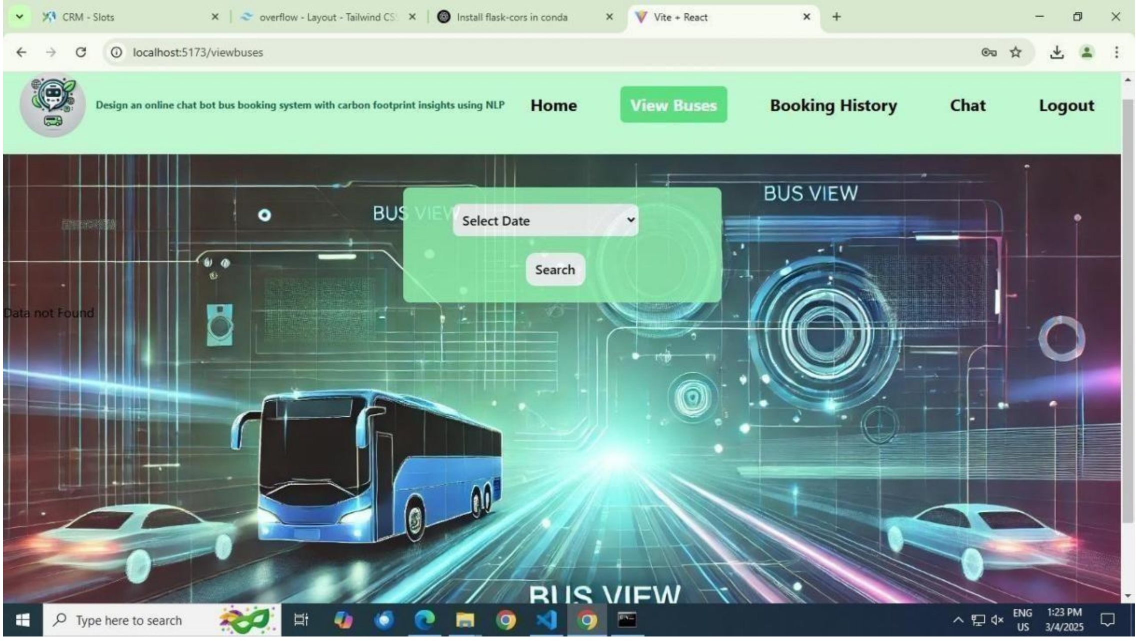
Name	Email	Mobile	Bus No	Carbon Emission	Date of Journey	Price	Seats
hari	hari@gmail.com	9123456789	tn 25	182.20982360839844 g/km	2025-03-01	219.00	
hari	hari@gmail.com	9123456789	tn 25	182.20982360839844 g/km	2025-03-01	600.00	4,5,6,7,8,9
hari	hari@gmail.com	9123456789	tn20	182.20982360839844 g/km	2025-03-06	100.00	1
hari	hari@gmail.com	9123456789	tn 25	182.20982360839844 g/km	2025-03-01	600.00	1,3,4,5,6,7
hari	hari@gmail.com	9123456789	tn 25	182.20982360839844 g/km	2025-03-01	100.00	4
				182.20982360839844	2025-		

Fig.7.6: View Bus Bookings Page

User Results :

Search Bus using Date:

This page allows users to input a date and instantly search for available buses using the chatbot for faster, context-aware results. The system presents various options filtered by time, route, and environmental efficiency. With real-time data and user-friendly search functionality, travelers can quickly find buses that align with their schedules and values. It’s an intuitive way to book with both convenience and carbon awareness in mind.

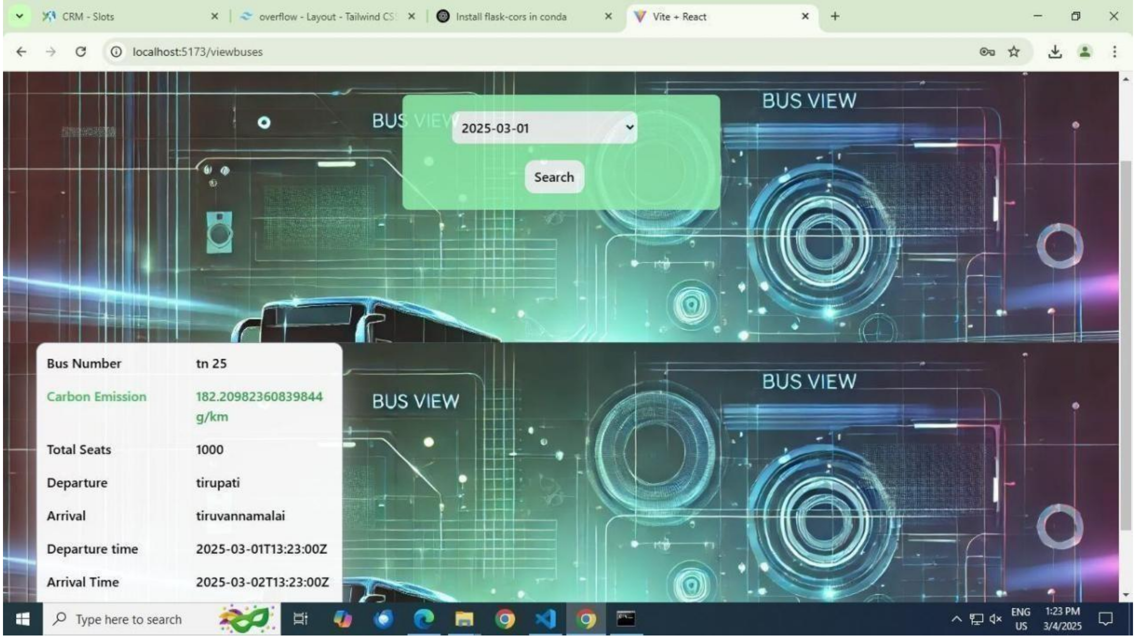


Data not Found

Fig.7.7:search bus using date

View Buses:

The View Buses page displays all available buses along with details such as routes, timings, seat availability, and carbon emissions per trip. The integrated chatbot helps users explore options, compare emissions, and find the most eco-friendly choice. Ratings and feedback from previous passengers add another layer of insight. This page empowers users to book responsibly while staying informed about their travel impact.

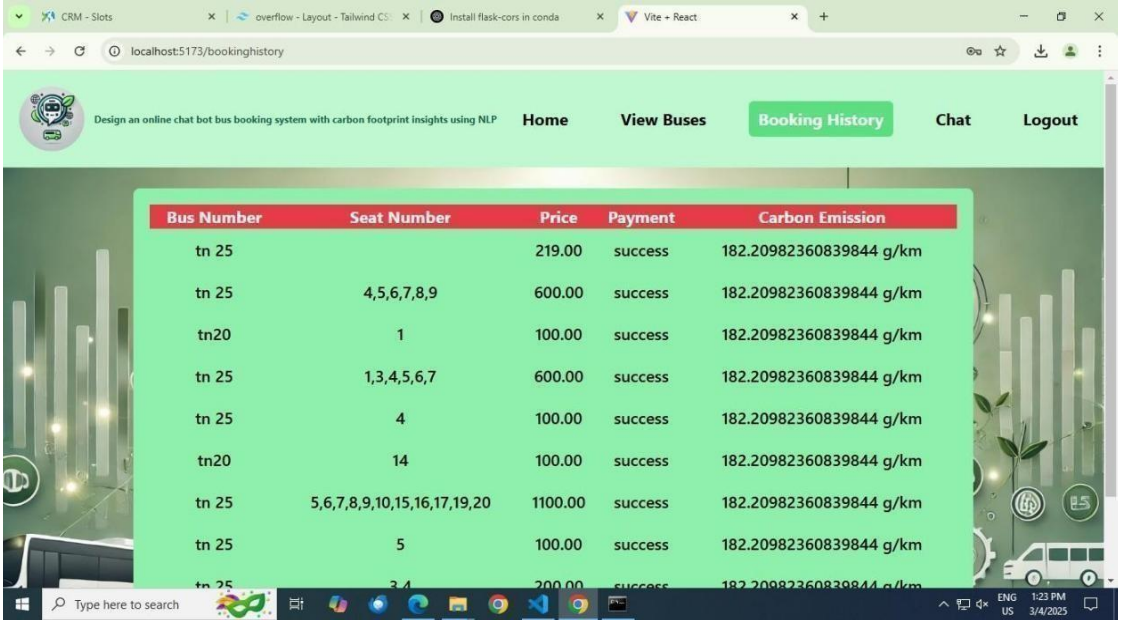


Bus Number	tn 25
Carbon Emission	182.20982360839844 g/km
Total Seats	1000
Departure	tirupati
Arrival	tiruvannamalai
Departure time	2025-03-01T13:23:00Z
Arrival Time	2025-03-02T13:23:00Z

Fig.7.8:View buses

Booking History:

The Booking History page archives all previous bookings and highlights key details such as journey date, fare, and carbon emissions. Users can review their travel patterns, understand their cumulative carbon footprint, and receive suggestions for greener alternatives in the future. The chatbot summarizes this data and even offers monthly or yearly reports for eco-conscious travelers. It promotes reflection and responsibility in travel habits.


CRM - Slots
overflow - Layout - Tailwind CS
Install flask-cors in conda
Vite + React
localhost:5173/bookinghistory
Design an online chat bot bus booking system with carbon footprint insights using NLP
Home
View Buses
Booking History
Chat
Logout
Type here to search
ENG
US
1:23 PM
3/4/2025

Bus Number	Seat Number	Price	Payment	Carbon Emission
tn 25		219.00	success	182.20982360839844 g/km
tn 25	4,5,6,7,8,9	600.00	success	182.20982360839844 g/km
tn20	1	100.00	success	182.20982360839844 g/km
tn 25	1,3,4,5,6,7	600.00	success	182.20982360839844 g/km
tn 25	4	100.00	success	182.20982360839844 g/km
tn20	14	100.00	success	182.20982360839844 g/km
tn 25	5,6,7,8,9,10,15,16,17,19,20	1100.00	success	182.20982360839844 g/km
tn 25	5	100.00	success	182.20982360839844 g/km
tn 25	3,4	200.00	success	182.20982360839844 g/km

Fig.7.9:Booking History

CHAPTER - 8

8.1.Conclusion

The Online Chatbot Bus Booking With Carbon Footprint Insights Using NLP aims to modernize and simplify the bus travel experience for both users and administrators. By providing a user-friendly interface for bus search, ticket booking, and payment processing, it ensures convenience, efficiency, and environmental awareness for travelers. The system integrates carbon footprint insights, allowing users to make eco-friendly decisions based on the carbon emissions of their chosen bus routes, thus contributing to sustainable travel.The admin module provides an intuitive dashboard for seamless management of bus schedules, bookings, and user data, along with the ability to track and monitor carbon emissions..Overall, this system improves operational efficiency, enhances user satisfaction, and provides a scalable solution for the future. Future enhancements, such as AI-based recommendations, real-time tracking, carbon emission optimization, and multi- language support, could further elevate its functionality, making it a robust and comprehensive solution for bus travel management and sustainability.

8.2 Future Work:

In envisioning the future scope and potential advancements of the Online Chatbot Bus Booking System with Carbon Footprint Insights using NLP, several key areas for enhancement and development emerge.One significant direction is the integration of advanced AI and NLP algorithms to further refine the chatbot's understanding of complex user queries and provide more context-aware and human-like interactions. Leveraging deep learning techniques can improve the system’s ability to interpret natural language, handle ambiguity, and respond intelligently in real- time.

Additionally, the incorporation of predictive analytics could allow the system to forecast user demand, recommend optimal travel routes based on historical patterns, and even suggest off-peak travel times to help reduce carbon emissions and ease congestion. These capabilities can improve booking efficiency and support sustainable transport planning.The potential use of blockchain technology also offers promising avenues, particularly for secure and transparent ticketing systems, tamper-proof transaction records, and user data protection—enhancing both data privacy and trust among users

References

1. Sharma, R., & Gupta, A. (2017). "A Web-Based Bus Ticket Booking System." International Journal of Advanced Research in Computer Science and Software Engineering, 7(2), 12-18.
2. Kumar, V., & Patel, S. (2019). "E-Commerce Integration in Bus Ticket Booking Systems." International Journal of Engineering and Technology, 8(4), 23-30.
3. Jain, N., & Kapoor, M. (2018). "An Intelligent Mobile App for Bus Ticketing." Journal of Mobile Computing, 16(3), 45-52.
4. Singh, R., & Mehta, V. (2020). "Blockchain Technology for Secure Bus Ticketing Systems." International Journal of Blockchain Applications, 4(1), 67-74.
5. Buhalis, D. (2003). "eTourism: Information Technology for Strategic Tourism Management." Tourism Management, 24(5), 153-158.
6. Nakamoto, S. (2008). "Bitcoin: A Peer-to-Peer Electronic Cash System." Bitcoin.org. Retrieved from https://bitcoin.org/bitcoin.pdf
7. Sahoo, P., & Mishra, P. (2015). "Online Bus Ticket Reservation System Using Cloud Computing." International Journal of Advanced Computer Science and Applications, 6(12), 34-39.
8. Soni, A., & Jain, M. (2017). "Bus Reservation and Ticketing System." International Journal of Computer Applications, 167(6), 30-34.
9. Patel, M., & Shah, J. (2020). "Cloud-Based Bus Reservation System." International Journal of Cloud Computing and Services Science, 8(4), 213-22.
10. Kaur, G., & Singh, H. (2016). "Development of Online Bus Ticket Booking System Using Android Platform." International Journal of Innovative Research in Computer and Communication Engineering, 4(1), 89-94.
11. Ahmed, S., & Khan, M. (2018). "Design and Implementation of a Real-Time Bus Tracking System." Journal of Transport Technologies, 5(2), 101-110.
12. Raj, P., & Menon, A. (2019). "Leveraging Cloud Services for Scalable Bus Reservation Systems." International Journal of Computer Engineering & Applications, 13(4), 44-50.
13. Thomas, R., & George, L. (2017). "Integration of Payment Gateways in Online Ticketing Systems." International Journal of Financial Technology, 2(1), 17-23.
14. Yadav, D., & Sharma, L. (2016). "Enhancing User Experience in Bus Ticket Apps through UI/UX Design." Journal of Human-Computer Interaction, 9(3), 70-76.

15. Deshmukh, P., & Kulkarni, A. (2020). "Smart Ticketing Using QR Code and NFC." International Journal of Computer Science Trends and Technology, 8(2), 65-71.

16. Verma, S., & Gupta, K. (2015). "A Review on Web-Based Ticket Booking System." International Journal of Engineering Trends and Technology, 23(3), 112-115.

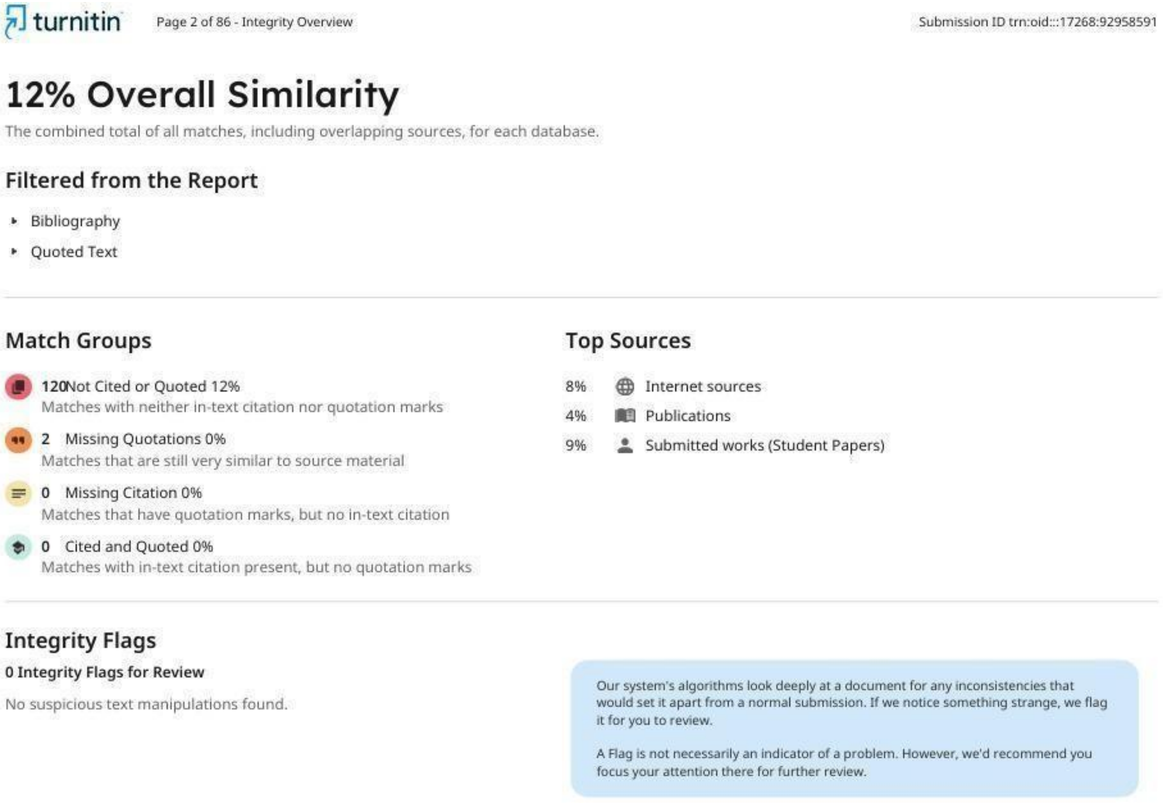
17. Chopra, R., & Das, T. (2018). "Security Challenges in Online Bus Ticketing Platforms." Journal of Cybersecurity and Information Systems, 7(1), 29-36.

18. Prakash, M., & Reddy, S. (2019). "Adoption of Mobile Apps for Public Transport Ticketing: A Case Study." Transportation Research Procedia, 37, 455-462.

19. Alvi, F., & Bano, R. (2020). "IoT-Based Smart Bus Ticketing and Monitoring System." International Journal of Internet of Things, 9(2), 12-19.

20. Khan, A., & Ali, Z. (2021). "AI-Powered Chatbots for Customer Service in Bus Ticketing Systems." Journal of Artificial Intelligence Research, 13(1), 88-95.

Appendix-A: Plagiarism Report :



turnitin Page 2 of 86 - Integrity Overview Submission ID trn:oid:::17268:92958591

12% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text

Match Groups

120Not Cited or Quoted 12%
Matches with neither in-text citation nor quotation marks

2 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

8%	Internet sources
4%	Publications
9%	Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Appendix-B: Time and Cost Analysis of your project work

Time Analysis:

Month 1: Analysis & Planning

Activities Involved:

Researching existing chatbot and carbon emission calculators. Researching existing chatbot and carbon emission calculators. Defining system objectives and architecture.

Estimated Time: 2 weeks (1 week for planning, 1 week for research and scope finalization).

Month 2: Chatbot & NLP Model Development

Activities Involved:

Developing the chatbot using Rasa/Dialogflow for handling user queries. Integrating NLP for natural language understanding

Training chatbot to recognize user intents for booking and carbon footprint queries. Estimated Time: 4 weeks (3 weeks for training and integration, 1 week for refinement).

Month 3: Carbon Footprint Estimator & Backend

Activities Involved:

Designing a carbon footprint calculator based on route distance and bus type.

Backend development using Flask/Django for chatbot communication and booking logic.

Estimated Time: 2 weeks.

Month 4: Frontend Development

Activities Involved:

Designing a simple and intuitive UI for users to chat, view bookings, and see emissions.

Integrating the frontend with the chatbot and backend APIs.

Estimated Time: 3 weeks (1 week for UI design and 2 weeks for frontend integration with the backend).

Month 5: Testing & Debugging

Activities Involved:

Performing unit, integration, and user testing for chatbot interactions and footprint accuracy.

Collecting feedback and making improvements.

Estimated Time: 3 weeks (2 weeks for thorough testing and 1 week for debugging).

Month 6: Deployment & Documentation

Deploying the chatbot to a web server or cloud platform.

Writing user manual, technical documentation, and presentation materials. Estimated Time: 2 weeks (1 week for deployment, 1 week for documentation). Total Development Time: 4 months (16 weeks).

Cost Analysis:

Development Tools & IDE

Details:Tools like VS Code, Google Colab, Flask, TensorFlow, scikit-learn, Rasa/Dialogflow – all free.

Estimated Cost: ₹0.

Hosting Server

Details Deployed using free-tier platforms like Heroku, Render, or Google Cloud/AWS (limited use).

Estimated Cost: ₹0.

NLP Libraries & Frameworks

Details:

Open-source libraries such as spaCy, NLTK, Rasa, or Dialog flow (free tier). Estimated Cost: ₹0.

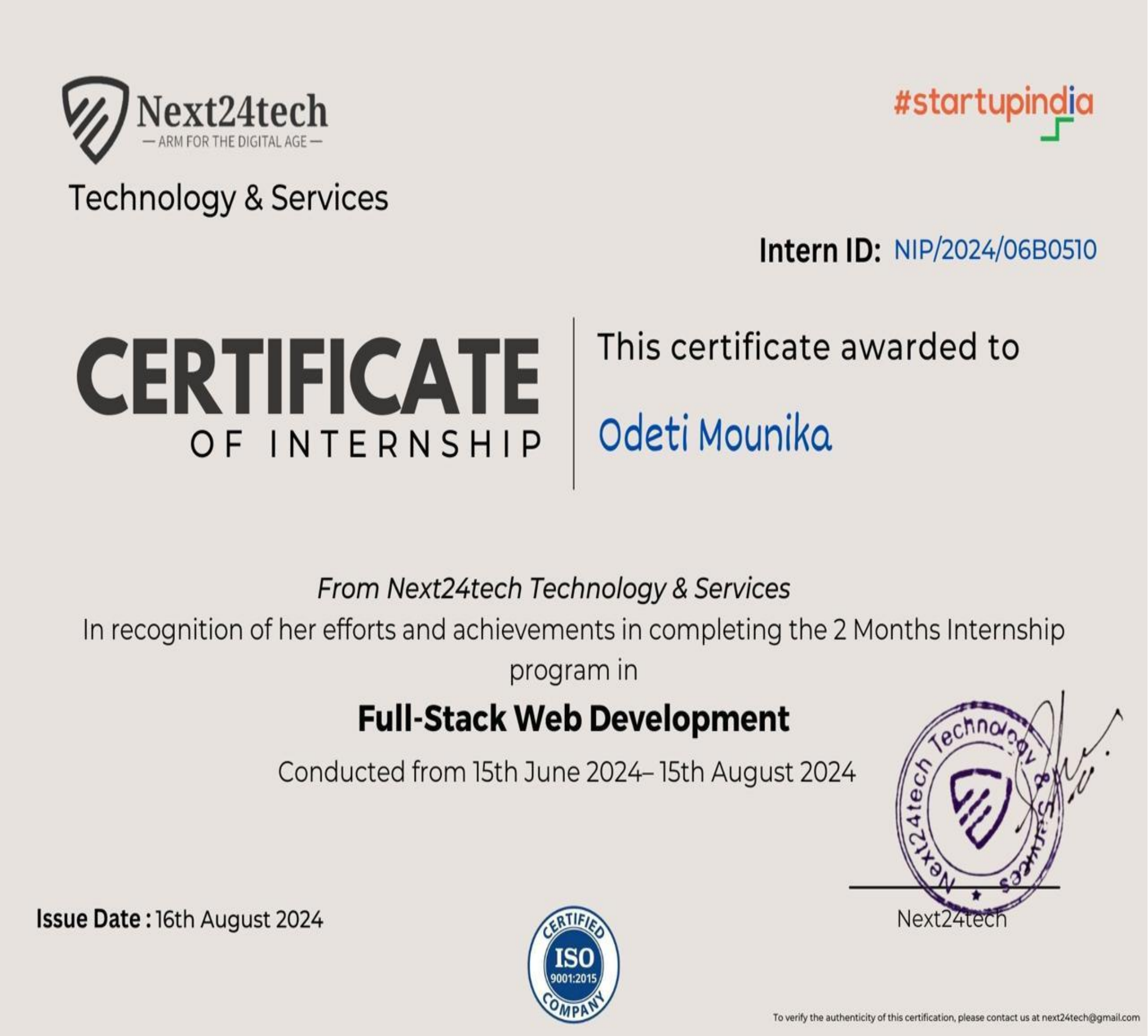
Database Email Services

Mailtrap or Firebase Cloud Messaging (FCM) for sending confirmations Estimated Cost: ₹0.

Miscellaneous Expenses

Assumed academic setting with no charge for internet/electricity. Estimated Cost: ₹0

Appendix-C: Internship Certificate:



Next24tech
— ARM FOR THE DIGITAL AGE —
Technology & Services

#startupindia

Intern ID: NIP/2024/06B0510

CERTIFICATE
OF INTERNSHIP

This certificate awarded to

Odeti Mounika

From Next24tech Technology & Services
In recognition of her efforts and achievements in completing the 2 Months Internship program in

Full-Stack Web Development

Conducted from 15th June 2024– 15th August 2024

Next24tech Technology & Services

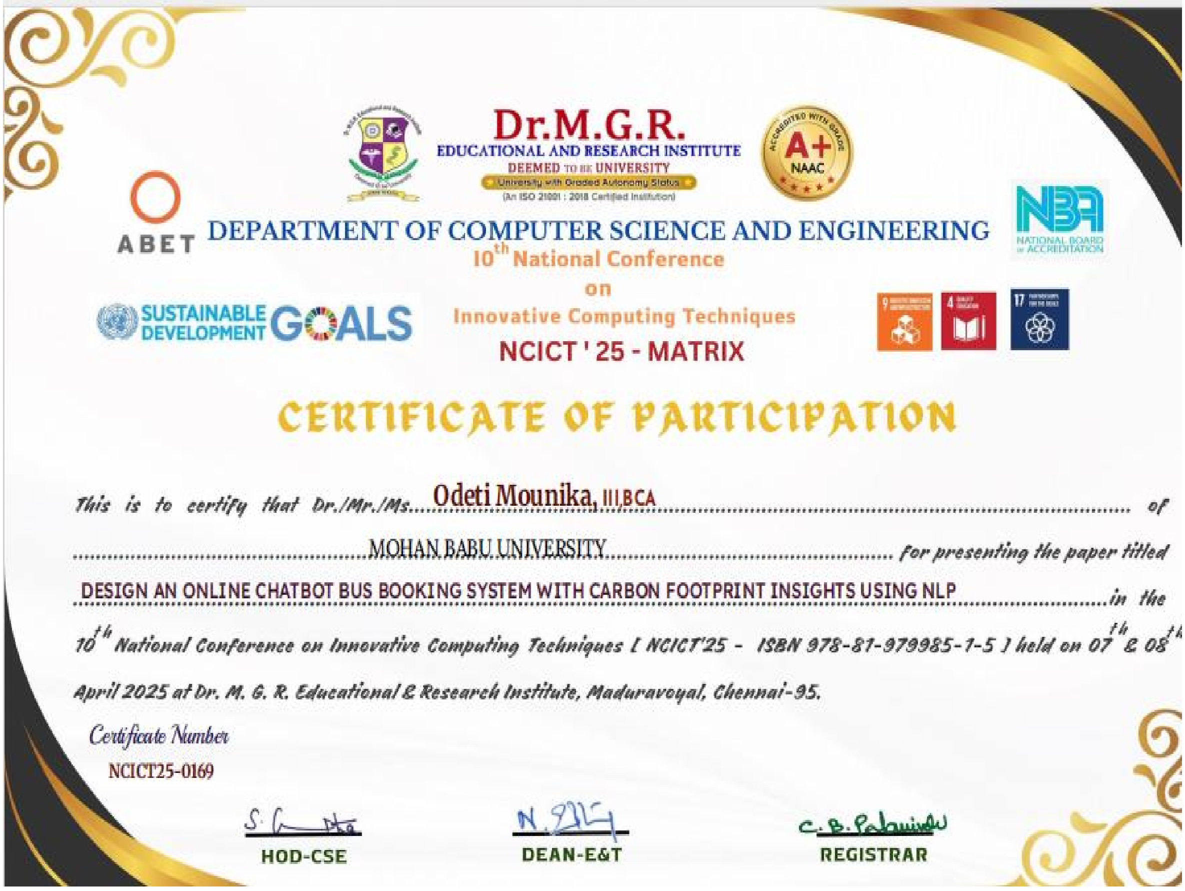
Next24tech

Issue Date : 16th August 2024

CERTIFIED ISO 9001:2015 COMPANY

To verify the authenticity of this certification, please contact us at next24tech@gmail.com

Appendix-D: Conference certificates:



Dr.M.G.R.
EDUCATIONAL AND RESEARCH INSTITUTE
DEEMED TO BE UNIVERSITY
University with Graded Autonomy Status
(An ISO 21001 : 2018 Certified Institution)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

10th National Conference
on
Innovative Computing Techniques
NCICT ' 25 - MATRIX

CERTIFICATE OF PARTICIPATION

This is to certify that Dr./Mr./Ms. Odeti Mounika, III,BCA of MOHAN BABU UNIVERSITY for presenting the paper titled DESIGN AN ONLINE CHATBOT BUS BOOKING SYSTEM WITH CARBON FOOTPRINT INSIGHTS USING NLP in the 10th National Conference on Innovative Computing Techniques [NCICT'25 - ISBN 978-81-979985-1-5] held on 07th & 08th April 2025 at Dr. M. G. R. Educational & Research Institute, Maduravoyal, Chennai-95.

Certificate Number
NCICT25-0169

HOD-CSE DEAN-E&T REGISTRAR

Appendix-E: Sample Source Code:

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

<title>Online Bus Booking with Carbon Footprint</title>

   <linkrel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css">

<style>

body {

background-color: #f8f9fa;

}

.navbar-brand {

font-weight: bold;

}

.card {

border-radius: 20px;

box-shadow: 0 4px 8px rgba(0,0,0,0.1);

}

</style>

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-dark bg-primary">

<div class="container-fluid">

<a class="navbar-brand" href="#">BusBook AI</a>

<div class="collapse navbar-collapse">

<ul class="navbar-nav me-auto mb-2 mb-lg-0">

<li class="nav-item"><a class="nav-link" href="#">Home</a></li>

<li class="nav-item"><a class="nav-link" href="#">Bookings</a></li>

<li class="nav-item"><a class="nav-link" href="#">Carbon Insights</a></li>
<li class="nav-item"><a class="nav-link" href="#">Chatbot</a></li>

</ul>

</div>
```

```
</div>

</nav>



<div class="container mt-5">

<div class="card p-4">

<h2 class="mb-4">Book Your Bus</h2>

<form id="bookingForm">
<div class="mb-3">

<label for="source" class="form-label">From</label>

<input type="text" class="form-control" id="source" required>

</div>

<div class="mb-3">

<label for="destination" class="form-label">To</label>

<input type="text" class="form-control" id="destination" required>

</div>

<div class="mb-3">

<label for="date" class="form-label">Journey Date</label>

<input type="date" class="form-control" id="date" required>

</div>

<button type="submit" class="btn btn-primary">Search Buses</button>

</form>



</div>



<div class="card mt-4 p-4" id="carbonResult" style="display: none;">
<h4>Estimated Carbon Emission</h4>

<p id="carbonInfo">Calculating...</p>

</div>

</div>



<script>
```

```
const form = document.getElementById('bookingForm');

const carbonResult = document.getElementById('carbonResult'); const carbonInfo =

document.getElementById('carbonInfo');


form.addEventListener('submit',  function(e)

{ e.preventDefault();


const source = document.getElementById('source').value;

const destination = document.getElementById('destination').value; const date =

document.getElementById('date').value;


// Dummy carbon footprint logic

const distance = Math.floor(Math.random() * (500 - 50 + 1)) + 50; // random 50-500 km const

carbon = (distance * 0.1).toFixed(2); // 0.1 kg/km per passenger

carbon Result.style.display = 'block';
carbonInfo.innerText = `Distance: ${distance} km\nEstimated Emission: ${car
});


</script>

</body>

</html>
```

CO-PO-Psomapping Table:

COURSE DESCRIPTION: Identification of topic for the project work; Literature survey; Collection of preliminary data; Identification of implementation tools and methodologies; Performing critical study and analysis of the topic identified; Time and cost analysis; Implementation of the project work; Preparation of thesis and presentation.

COURSE OUTCOMES: After successful completion of this course, the students will be able to:

CO1. Create/Design computer science engineering systems or processes to solve complex computer science engineering and allied problems using appropriate tools and techniques following relevant standards, codes, policies, regulations and latest developments.

CO2. Consider society, health, safety, environment, sustainability, economics and project management in solving complex computer science engineering and allied problems.

CO3. Perform individually or in a team besides communicating effectively in written, oral and graphical forms on computer science engineering systems or processes.

CO-PO-PSO Mapping Table:

Course Outcome	**Program Outcomes**												**Program Specific Outcomes**			
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**	**PSO4**
CO1	3	3	3	3	3			3				3				
CO2						3	3				3					
CO3									3	3						

Correlation Level: 3-High; 2-Medium; 1-Low

Short Bio-data of the Student:

Name	: Odeti Mounika
Education	: BCA
University	: Mohan Babu University
Study Period	: 2022-2025
Roll Number	: 22102C010226
Gmail	: mounikaodeti2003@gmail.com