LinC

A Mini Project Report

Submitted in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

by

Vishnu Vardhan Rongali	**22L31A5484**
Kumar Ganesh	**22L31A5486**
Mekala suresh	**22L31A5460**
Reddi ruakesh	**23L31A5421**
Uppalapati Bhanu avinash varma	**22L31A54A2**

Under the Esteemed Guidance of

Mr. Gouri Sankar Nayak

Assistant Professor

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY VISAKHAPATNAM

(Autonomous)

Affiliated to JNTU GV & Approved by AICTE, NewDelhi ,Re-Accredited by NAAC (CGPA of 3.4/ 4.00)

ISO 9001:2008, ISO 14001:2004, OHSAS 18001:2007 Certified Institution

VISAKHAPATNAM – 530 039

VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY

Department of Artificial Intelligence and Data Science

CERTIFICATE

This is to certify that this project report entitled **"LinC"** is a bonafide record of the work done by **Vishnu Vardhan Rongali 22L31A5484 , Kumar Ganesh 22L31A5486, Mekala Suresh 22L31A5460, Reddi Ruakesh 23L31A5421, Uppalapati Bhanu Avinash Varma 22L31A54A2** during the academic year 2023-2024, in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Artificial Intelligence and Data Science of Jawaharlal Nehru Technological University, Vizianagaram. The results embodied in this project report have not been submitted to any other University or Institute for the award of any Degree or Diploma.

Project Guide

Mr. Gouri Sankar Nayak

Assistant Professor, AI&DS

Head of the Department

Dr.T.V Madhusudhana Rao

Professor,HoD-AI&DS

DECLARATION

We hereby declare that the project report entitled "**LinC**" has been written by us and has not been submitted either in part or whole for the award of any degree, diploma or any other similar title to this or any other university.

Vishnu Vardhan R	22L31A5484
Kumar Ganesh S	22L31A5486
Mekala suresh	22L31A5460
Reddi ruakesh	23L31A5421
Bhanu Avinash varma	22L31A54A2

DATE :

PLACE :

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to acknowledge the assistance and cooperation we have received from several persons while undertaking this B. Tech. Mini Project.

We express our gratitude to our beloved and honourable principal **Dr. Sudhakar Jyothula**, for providing necessary departmental facilities in spite of his busy schedule and guiding us in every possible way

We also take the opportunity to acknowledge the contribution of **Prof. Dr. T. V. Madhusudhana Rao**, Head of the Department of artificial intelligence and data science, for his full support and assistance during the development of the project.

We owe special debt of gratitude to **Mr. Gouri Sankar Nayak**, Assistant Professor Department of Artificial Intelligence and Data Science, for her constant support and guidance throughout the course of our work. Her sincerity, thoroughness, and perseverance have been a constant source of inspiration for us.

We would like to thank **Vignan's Institute of Information Technology**, Duvvada for providing me with the facilities and resources necessary to complete this project. Your support has been invaluable.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not least, we acknowledge our friends for their contribution to the completion of the project.

ABSTRACT

The LinC is an innovative platform meticulously crafted to bridge the ever-widening gap between students and job opportunities. Recognizing the challenges faced by job seekers, particularly those entering the workforce for the first time, LinC provides a fully integrated solution aimed at simplifying and enhancing the job search experience. The platform features a state-of-the-art Applicant Tracking System (ATS) that not only streamlines the recruitment process for employers but also empowers job seekers by offering clear insights into their application statuses and progress. A standout feature of LinC is its Resume Builder, which guides users through the process of creating professional, industry-specific resumes. This tool is designed with flexibility and customization in mind, allowing users to tailor their resumes to fit the specific requirements of different job roles and industries. Complementing the Resume Builder is the Job Tracking feature, which serves as a centralized hub for users to organize, monitor, and manage their job applications efficiently, ensuring that no opportunity slips through the cracks. In addition to these core functionalities, LinC leverages cutting-edge AI technology through its chat feature powered by Langchain. This feature offers real-time communication capabilities, providing users with instant support, guidance, and feedback throughout their job search journey. Built on a robust tech stack that includes Flask for the backend, HTML and JavaScript for the frontend, and SQLite3 for the database, LinC ensures a seamless, user-friendly experience that is both reliable and scalable. This platform is designed to be a game-changer in the job market, particularly for students and recent graduates looking to make their mark in the professional world.

CONTENTS

LIST OF FIGURES

NOMENCLATURE

- **HTML:** HyperText Markup Language
- **CSS:** Cascading Style Sheets
- **SQL:** Structured Query Language
- **SQLite**: A lightweight, serverless, self-contained SQL database engine
- **Flask:** A lightweight web framework for Python
- **JavaScript:** A programming language for creating dynamic web content
- **Python:** A high-level programming language for general-purpose programming
- **Jinja2:** A templating engine for Python, used for rendering dynamic content
- **Bootstrap:** A front-end framework for designing responsive web pages
- **Langchain:** A framework for integrating language models into applications
- **Git:** A distributed version control system for tracking changes in source code
- **VSCode:** Visual Studio Code, a source-code editor used for writing and editing code
- **ATS:** Applicant Tracking System, a tool for managing job applications and recruitment
- **AI**: Artificial Intelligence, used in the chat support feature for providing real-time assistance
- **REST API:** Representational State Transfer Application Programming Interface, a web service that allows communication between systems
- **JSON:** JavaScript Object Notation, a lightweight data-interchange format used for data exchange between a server and a client
- **HTTP:** Hypertext Transfer Protocol, the foundation of data communication on the web
- **UI/UX:** User Interface/User Experience, design principles focusing on the interface and overall experience of users

CHAPTER I

INTRODUCTION

INTRODUCTION

LinC is a dynamic platform developed to address the challenges faced by students and recent graduates in navigating the job market. By integrating essential tools such as an Applicant Tracking System (ATS), a customizable Resume Builder, and a Job Tracking feature, LinC offers a streamlined and efficient approach to job searching. Additionally, the platform harnesses AI-driven technology to provide real-time support through its chat feature, ensuring users have the guidance they need at every step. Built with a focus on user experience and reliability, LinC is designed to empower the next generation of job seekers.

1.1 MOTIVATION

The transition from academia to the professional world is often a challenging journey for students and recent graduates. Many find themselves overwhelmed by the complexities of the job market, where the lack of guidance and resources can make it difficult to navigate through the multitude of opportunities available. This challenge is further compounded by the competitive nature of the job market, where even the slightest edge can make a significant difference. Recognizing these difficulties, we were motivated to create a platform that not only simplifies the job search process but also provides the necessary tools and support to help students succeed.

A major driving force behind the development of LinC is the need to bridge the gap between education and employment. While academic institutions equip students with theoretical knowledge, there is often a lack of practical resources to prepare them for real-world job applications. We identified a clear need for a platform that can guide users in creating professional resumes, tracking job applications, and staying organized throughout their job search. LinC aims to fill this gap by offering a comprehensive solution that addresses these specific needs.

The introduction of an Applicant Tracking System (ATS) in LinC was inspired by the growing importance of ATS software in the recruitment process. Many companies now rely on ATS to manage job applications, making it crucial for candidates to understand how these systems work. By incorporating an ATS feature, LinC not only helps users improve their chances of getting noticed by employers but also educates them on the nuances of modern recruitment practices. This feature is particularly beneficial for students who may be unfamiliar with the intricacies of the hiring process.

In developing the Resume Builder, we were motivated by the realization that many job seekers struggle with creating resumes that effectively showcase their skills and experiences. A well-crafted resume is often the first impression a candidate makes on a potential employer, making it a critical component of the job application process. LinC’s Resume Builder is designed to

simplify this task by providing users with customizable templates and guidance, ensuring that they can present themselves in the best possible light.

The decision to integrate a Job Tracking feature stemmed from the recognition that managing multiple job applications can be a daunting task. Keeping track of deadlines, application statuses, and follow-ups is essential to maintaining an organized and efficient job search. LinC's Job Tracking feature was created to address this need, offering users a centralized platform to monitor their applications, reducing the likelihood of missed opportunities and helping them stay on top of their job search efforts.

In conclusion,The motivation behind LinC is deeply rooted in the desire to empower students and recent graduates as they embark on their professional journeys. By providing a platform that offers a complete suite of tools tailored to the needs of job seekers, LinC aims to make the job search process less intimidating and more manageable. Our goal is to equip users with the resources and support they need to confidently pursue their career aspirations and successfully transition from education to employment.

1.2 PROBLEM DEFINTIONS

- Existing job portals often lack comprehensive tools for tracking and managing multiple job applications, leaving users disorganized and prone to missing important deadlines.

- Many platforms do not provide customizable resume-building tools, which forces users to rely on external resources that may not align with industry standards.
- Most job portals do not offer an Applicant Tracking System (ATS) that is accessible to job seekers, leaving them unaware of how their applications are being processed.

- A significant gap in existing platforms is the lack of real-time support and guidance during the job search process.

1.3 WHAT IS A FACE BASED ATTENDANCE SYSTEM?

LinC, short for "Linking Careers," is an innovative platform designed to bridge the gap between students, recent graduates, and job opportunities. In today's competitive job market, finding the right opportunity and presenting oneself effectively can be challenging. LinC addresses these challenges by offering an integrated suite of tools that streamline the job search process,

making it easier for users to connect with potential employers and secure their desired roles. Whether it's crafting a professional resume, applying for jobs, or tracking applications, LinC provides a comprehensive solution tailored to meet the needs of modern job seekers.

At the core of LinC is a state-of-the-art Applicant Tracking System (ATS) that simplifies the recruitment process for both job seekers and employers. For job seekers, the ATS provides clear insights into the status of their applications, helping them stay informed and engaged throughout the hiring process. Employers, on the other hand, benefit from an organized and efficient system that allows them to manage and evaluate candidates with ease. This dual approach ensures that both sides of the job market can interact seamlessly and effectively, leading to better outcomes for all parties involved.

LinC also includes a powerful Resume Builder feature, which guides users through the process of creating a professional, industry-specific resume. In today's job market, a well-crafted resume is crucial for making a strong first impression. LinC's Resume Builder offers users a range of customizable templates and intuitive tools that make it easy to highlight their skills, experience, and achievements. By providing users with the ability to tailor their resumes to fit specific job roles, LinC helps them stand out in a crowded job market.

In addition to these core features, LinC leverages advanced AI technology to offer real-time support through its AI-powered Chat Support feature. This feature is designed to provide users with immediate assistance and guidance as they navigate the job search process. Whether users have questions about their resume, need help with a job application, or seek advice on how to improve their chances of success, LinC's AI Chat Support is there to provide instant, personalized feedback.

Overall, LinC is more than just a job portal; it's a comprehensive career management tool designed to empower students and recent graduates as they embark on their professional journeys. By integrating essential job search functionalities into a single, user-friendly platform, LinC simplifies the job search process and helps users achieve their career goals with confidence.

1.4 OBJECTIVE OF THE PROJECT

The primary objective of LinC is to revolutionize the job-seeking experience for students and recent graduates by providing a comprehensive, integrated platform that simplifies the job application process. Through the development of a sophisticated Applicant Tracking System (ATS), LinC aims to streamline the recruitment process, making it more transparent and manageable for both job seekers and employers. The ATS is designed to enhance

communication, track application progress, and provide valuable insights into the job market, ultimately reducing the complexity and time required for job searching.

In addition to the ATS, LinC includes a Resume Builder and Job Tracking feature designed to enhance the job-seeking experience. The Resume Builder helps users create tailored, professional resumes, while the Job Tracking feature ensures users can efficiently monitor and manage their applications. The AI-driven chat functionality, powered by Langchain, offers real-time support and guidance, addressing users' needs promptly. Together, these tools aim to streamline the job search process and provide a comprehensive solution for effective career advancement.

ADVANTAGES

- **Streamlined Recruitment:** Efficiently manages the hiring process with a user-friendly Applicant Tracking System.
- **Customizable Resumes:** Allows users to create professional resumes tailored to specific industry requirements.
- **Centralized Application Management:** Keeps job applications organized and trackable in one convenient location.
- **Real-Time Support:** Offers instant guidance and assistance through AI-driven chat functionality.
- **User-Friendly Interface:** Ensures a seamless experience with a reliable and intuitive platform design.

CHAPTER II

LITERATURE SURVEY

LITERATURE SURVEY

The evolving landscape of job-seeking platforms has seen a significant shift towards integrating technology to enhance the user experience. Traditional methods of job searching often involve cumbersome processes, from manual resume submissions to fragmented tracking systems. Recent advancements have introduced more sophisticated solutions, such as Applicant Tracking Systems (ATS), which streamline recruitment by automating the tracking of applications and improving communication between employers and candidates. Studies have shown that ATS significantly reduces the time and effort involved in the hiring process, making it an essential tool for modern recruitment strategies.

The rise of AI-driven technologies has further transformed the job application process. Natural Language Processing (NLP) models, such as those integrated into chatbots, are being employed to offer real-time assistance and personalized guidance. Research indicates that AI-powered chat systems can enhance user engagement and satisfaction by providing instant responses and tailored advice, thus addressing common challenges faced by job seekers, such as navigating application requirements and preparing for interviews.

In parallel, the development of resume-building tools has become increasingly important. Customized resume creation is a key factor in job application success, as tailored resumes are more likely to meet specific job requirements and pass through automated resume screening systems. Literature highlights the effectiveness of dynamic resume builders in helping users present their qualifications in a manner that aligns with industry standards, ultimately improving their chances of securing interviews.

Job tracking systems have also gained prominence as a means to improve the organization and management of job applications. Efficient tracking solutions enable users to monitor application statuses, set reminders, and track deadlines, reducing the risk of missed opportunities. Studies have demonstrated that comprehensive job tracking can lead to a more organized and proactive job search, which is crucial for staying competitive in the job market.

In conclusion, The literature indicates that integrating advanced technologies such as ATS, AI-driven chat support, and dynamic resume builders significantly enhances the job-seeking experience. By leveraging these tools, platforms like LinC address the inefficiencies of traditional job application processes and offer a more streamlined, organized, and supportive approach to career advancement. As technology continues to evolve, the adoption of such integrated solutions is likely to become increasingly important for both job seekers and employers.

CHAPTER III

SYSTEM ANALYSIS

SYSTEM ANALYSIS

The system is a web-based platform for job seekers that leverages AI-powered tools and a comprehensive suite of features to streamline the job search process. Users can explore various job options including full-time, part-time, and freelancing roles. The platform includes a resume builder, a job tracker, and an ATS system to help users craft compelling resumes, manage applications, and prepare for interviews. The AI-powered resume generator analyzes users' resumes and provides suggestions for improvement. The platform also integrates with Google AI for resume analysis, providing insights into the relevance of a resume to specific job descriptions. The system is built on a secure infrastructure using Flask, SQLite, and LangChain, which ensures user data privacy and security.

3.1 EXISTING SYSTEM

1. **Social Networking Integration:**Many existing job-seeking platforms leverage social networking integration to enhance job search and networking opportunities. For example, platforms like LinkedIn allow users to connect with professionals, join industry groups, and participate in discussions. This social aspect helps in building a professional network, which can be crucial for job referrals and opportunities. LinC does not currently include social networking features, focusing instead on tools directly related to the job application process.

2. **Video Interviewing Tools**:Some job-seeking systems incorporate video interviewing tools as part of their offerings. Platforms like HireVue provide integrated solutions for conducting video interviews, allowing employers to assess candidates remotely. These tools often include features for scheduling, recording, and reviewing interviews. LinC does not include video interviewing capabilities, concentrating on resume building, job tracking, and real-time chat support.

3. **Advanced Analytics and Insights**: Advanced analytics and insights are features provided by some platforms to help users understand job market trends and improve their job search strategies. For instance, platforms like Glassdoor offer salary insights and company reviews based on user-generated data. These analytics can guide job seekers in making informed decisions about their career paths. LinC’s focus is on providing direct tools for job applications and real-time support, rather than in-depth market analytics.

4. **Employer Branding Tools**: Employer branding tools are often used by companies to enhance their reputation and attract top talent. Platforms like Indeed and Glassdoor allow

employers to showcase their company culture, benefits, and employee experiences. These features help job seekers make informed decisions about potential employers. LinC does not currently offer features for employer branding, focusing instead on tools that directly assist job seekers in their application process.

5. **Integration with External Job Boards:** Integration with external job boards is a feature available in some job-seeking systems, allowing users to access job listings from multiple sources in one place. Platforms like ZipRecruiter aggregate job postings from various websites, providing a broader search experience. LinC is designed as a self-contained system with its own job application and tracking features, without integrating with external job boards for a wider range of listings.

3.2 PROPOSED SYSTEM

Our latest LinC offers a vast range of advantages:

- **Integrated Applicant Tracking System (ATS):** LinC features an advanced Applicant Tracking System (ATS) designed to streamline the recruitment process for both employers and job seekers. This system automates the tracking of job applications, provides real-time updates on application status, and facilitates clear communication between candidates and employers. The ATS is integrated seamlessly into the platform, offering a comprehensive solution for managing recruitment activities efficiently.

- **Customizable Resume Builder**: The platform includes a powerful Resume Builder that enables users to create professional, industry-specific resumes with ease. The tool offers a range of customizable templates and formatting options, allowing users to tailor their resumes to specific job roles and industries. This feature ensures that job seekers can present their qualifications effectively and increase their chances of landing interviews.

- **Centralized Job Tracking**: LinC provides a centralized Job Tracking feature that allows users to monitor and manage their job applications in one place. This feature includes tools for tracking application statuses, setting reminders, and organizing job-related information. By consolidating job tracking tasks into a single interface, LinC helps users stay organized and proactive throughout their job search.

- **AI-Powered Chat Support**: The platform leverages AI technology through its chat feature, powered by Langchain, to offer real-time support and guidance to users. This AI-driven chat

system provides instant responses to user queries, offers personalized advice, and helps with various aspects of the job application process. The inclusion of AI support enhances the user experience by providing timely assistance and addressing common challenges.

- **Enhanced User Experience**: LinC is built with a focus on user experience, incorporating a tech stack that includes Flask for the backend, HTML and JavaScript for the frontend, and SQLite3 for the database. This robust combination ensures a reliable, scalable, and user-friendly platform. The design emphasizes ease of use and accessibility, providing a seamless experience for job seekers and employers alike.

3.3 SOFTWARE REQUIREMENT SPECIFICATION

The software requirements and specifications include Flask, HTML, CSS, JavaScript, SQLite3, Python, Langchain, Jinja2, Git, VSCode, and Bootstrap.

3.3.1. Introduction

3.3.1.1 Purpose

The purpose of this project is to create a comprehensive platform that simplifies and enhances the job-seeking process for students and recent graduates. By integrating essential tools like an Applicant Tracking System (ATS), a customizable Resume Builder, and a centralized Job Tracking feature, the platform aims to streamline the recruitment process, improve application management, and provide real-time AI-driven support, ultimately empowering users to navigate the job market with greater ease and confidence.

3.3.1.2 Scope

The LinC will include the following features and functionalities:

- **Job Application Management**: Provides tools for tracking and organizing job applications efficiently.
- **Resume Creation:** Offers a customizable Resume Builder to help users craft professional resumes tailored to specific industries.
- **Applicant Tracking System (ATS):** Streamlines the recruitment process by automating application tracking and communication between candidates and employers.
- **Real-Time AI Support:** Incorporates AI-powered chat features to deliver instant guidance and support throughout the job-seeking process.
- **User Experience Optimization:**Focuses on creating a user-friendly, scalable platform that ensures a seamless and reliable experience for both job seekers and employers.

3.3.2. Functional Requirements

3.3.2.1 Applicant Tracking System (ATS):

- Track the status of job applications in real-time.
- Facilitate communication between job seekers and employers through automated notifications.
- Filter and sort applications based on specific criteria set by employers.

3.3.2.2 Resume Builder:

- Provide customizable resume templates tailored to various industries.
- Allow users to edit and format their resumes within the platform.
- Enable users to download or directly submit their resumes to job postings.

3.3.2.3 Job Tracking Feature

- Centralize the management of all job applications in a single interface.
- Set reminders and deadlines for application follow-ups.
- Provide visual indicators of application progress and status.

3.3.2.4 AI-Powered Chat Support:

- Offer real-time assistance and guidance to users via an AI-driven chat system.
- Respond to common job search queries and provide personalized advice.
- Help users with resume tips, interview preparation, and job application strategies.

3.3.2.5 User Account Management:

- The system shall record the attendance of recognized individuals, storing the date, time, and other relevant information in a database or file system.
- The system shall maintain a comprehensive and accurate attendance log for each user.

3.3.2.6 Attendance Reports

- Allow users to create, manage, and delete their accounts securely.
- Provide options for users to save and retrieve their resumes and job application history.
- Ensure data privacy and secure handling of personal information.

3.3.3. Non-Functional Requirements

3.3.3.1 Performance

- Optimize the platform to load within three seconds under normal user conditions.
- Support simultaneous use by multiple users without significant degradation in performance.

3.3.3.2 Accuracy

- Ensure that job application statuses and user data are updated in real-time.
- Provide reliable and consistent AI-generated responses in the chat feature.

3.3.3.3 Security

- Implement encryption for user data storage and transmission.
- Ensure secure authentication and authorization processes to protect user accounts.

3.3.3.4 Usability

- Design an intuitive and user-friendly interface accessible to users of all skill levels.
- Provide clear instructions and feedback for all user interactions on the platform.

3.3.3.4 Portability

- Ensure the platform is accessible on various devices, including desktops, tablets, and smartphones.
- Support compatibility across different web browsers without loss of functionality.

3.3.3.5 Scalability

- Design the system architecture to handle increasing numbers of users and job applications efficiently.
- Allow for easy integration of additional features and services as the platform grows.

3.3.4 Glossary

Flask: A web framework used for developing web applications in Python.

HTML (HyperText Markup Language): The standard markup language used to create and design web pages.

CSS (Cascading Style Sheets): A stylesheet language used to describe the presentation of a document written in HTML or XML.

JavaScript: A programming language that enables interactive web features such as animations, form validations, and dynamic content updates.

SQLite3: A lightweight, self-contained database engine used for storing and managing data within the application.

Python: A high-level programming language used for developing server-side logic, including

web application backends.

Langchain: A framework that enables the integration of large language models into applications, used for creating the AI-powered chat feature in the platform.

Jinja2: A templating engine for Python that helps in rendering dynamic content in web pages by combining HTML with Python data.

Git: A version control system that tracks changes in source code during software development.

VSCode (Visual Studio Code): A source-code editor used for writing and editing code, supporting a variety of programming languages and development tools.

Bootstrap: A front-end framework used for designing responsive and mobile-first websites, providing pre-built CSS and JavaScript components.

CHAPTER IV

SYSTEM DESIGN

SYSTEM DESIGN

System design is a critical phase in the development of your project that focuses on translating the requirements gathered during the system analysis phase into a well-defined and structured solution. It involves designing the architecture, components, and interfaces of the system to ensure its functionality, reliability, and maintainability.

4.1 DATA FLOW DIAGRAM

A data flow diagram is a graphical view of how data is processed in a system in terms of input and output. Data flow diagram is a graphical representation of flow of data through an information system modelling as its process aspects. A DFD shows what kind of information will be input to and output from the system, how data will advance through the system and where the data will be stored. The Data flow diagram (DFD) contains some symbols for drawing the data flow diagram.

Data flow diagram symbol: -

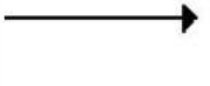
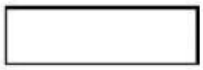
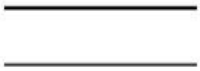
Symbol	Description
	**Data Flow** – Data flow are pipelines through the packets of information flow.
	**Process:** A Process or task performed by the system.
	**Entity:** Entity is object of the system. A source or destination data of a system.
	**Data Store:** A place where data to be stored.

Fig – 4.1 : Data Flow Diagrams

4.2 FLOW DIAGRAM

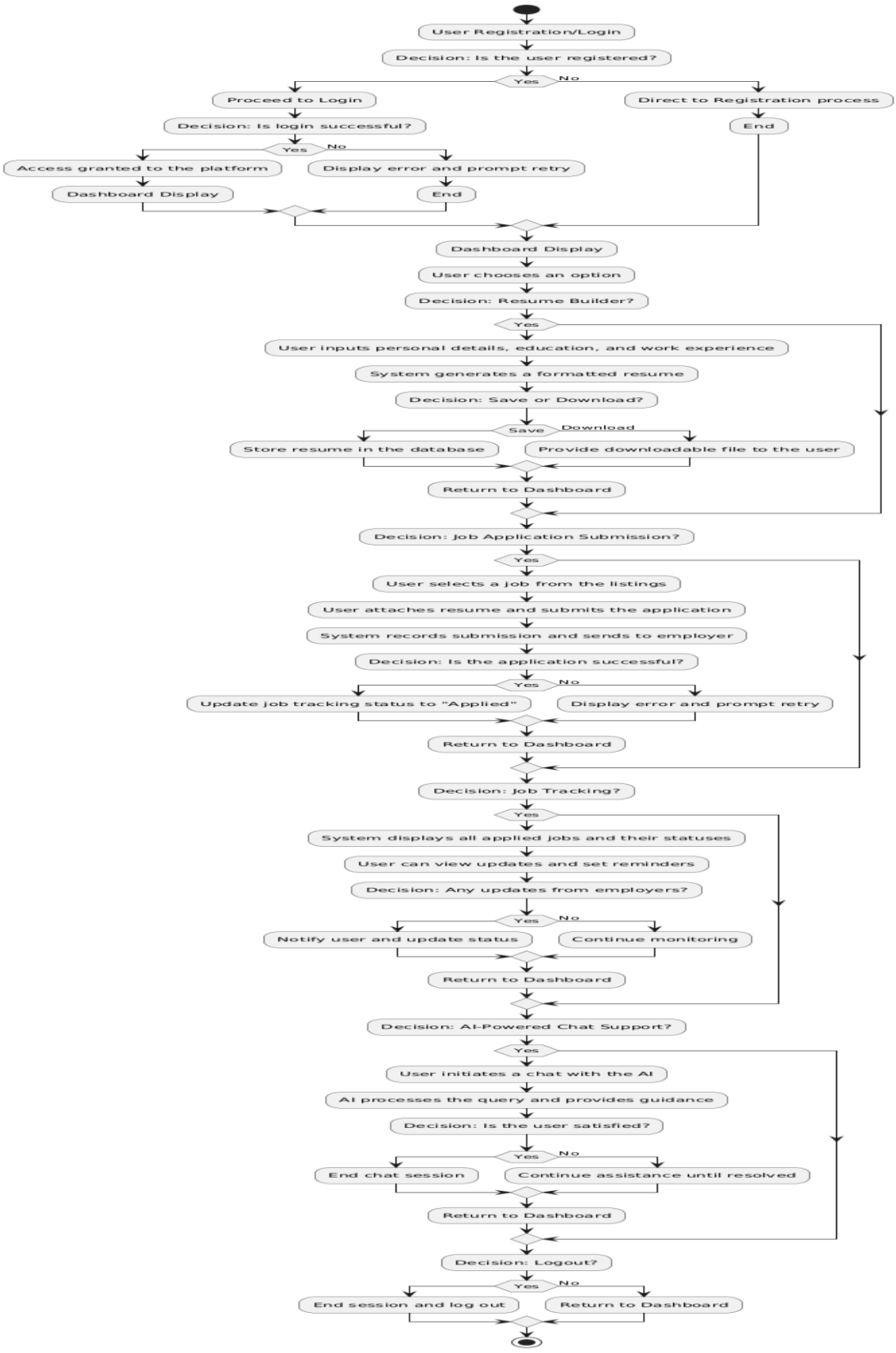


Fig - 4.2 : Flow Chart

4.3 UML DIAGRAMS

UML means Unified Modelling Language. UML is used for visualizing, constructing and documenting the artifacts of software intensive systems. UML makes a clear conceptual distinction between models, views and diagrams. UML can be used to develop diagrams and provide users with ready-to-use, expressive modelling examples. Some UML tools generate program language code from UML.

UML can be used for modelling a system independent of a platform language. Unified Modelling Language (UML) is a non-proprietary specification language for object modelling. UML is a general-purpose modelling language that includes a standardized graphical notation used to create an abstract model of a system, referred to as a UML model.

UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.UML was created by the Object Management Group (OMG) and UML 1.0 specification draft was proposed to the OMG in January 1997. OMG is continuously making efforts to create a truly industry standard.

- UML stands for Unified Modelling Language.
- UML is different from the other common programming languages such as C++, Java, COBOL, etc.
- UML is a pictorial language used to make software blueprints.
- UML can be described as a general-purpose visual modelling language to visualize, specify, construct, and document software systems.
- Although UML is generally used to model software systems, it is not limited within this boundary. It is also used to model non-software systems as well. For example, the process flows in a manufacturing unit, etc.

4.3.1 USE CASE DIAGRAM

A use case diagram shows a set of use cases and actors and their relationships. Use case is represented as an eclipse within a name inside it. Use cases are used to capture high level functionalities of a system .Each use case should provide some observable and valuable result to the actors or other stakeholders of the system. Use case diagrams have a specialization of class diagrams and class diagrams are structured diagrams.

Use case diagrams are in fact two fold they are both behavior diagrams because they describe behavior of the system

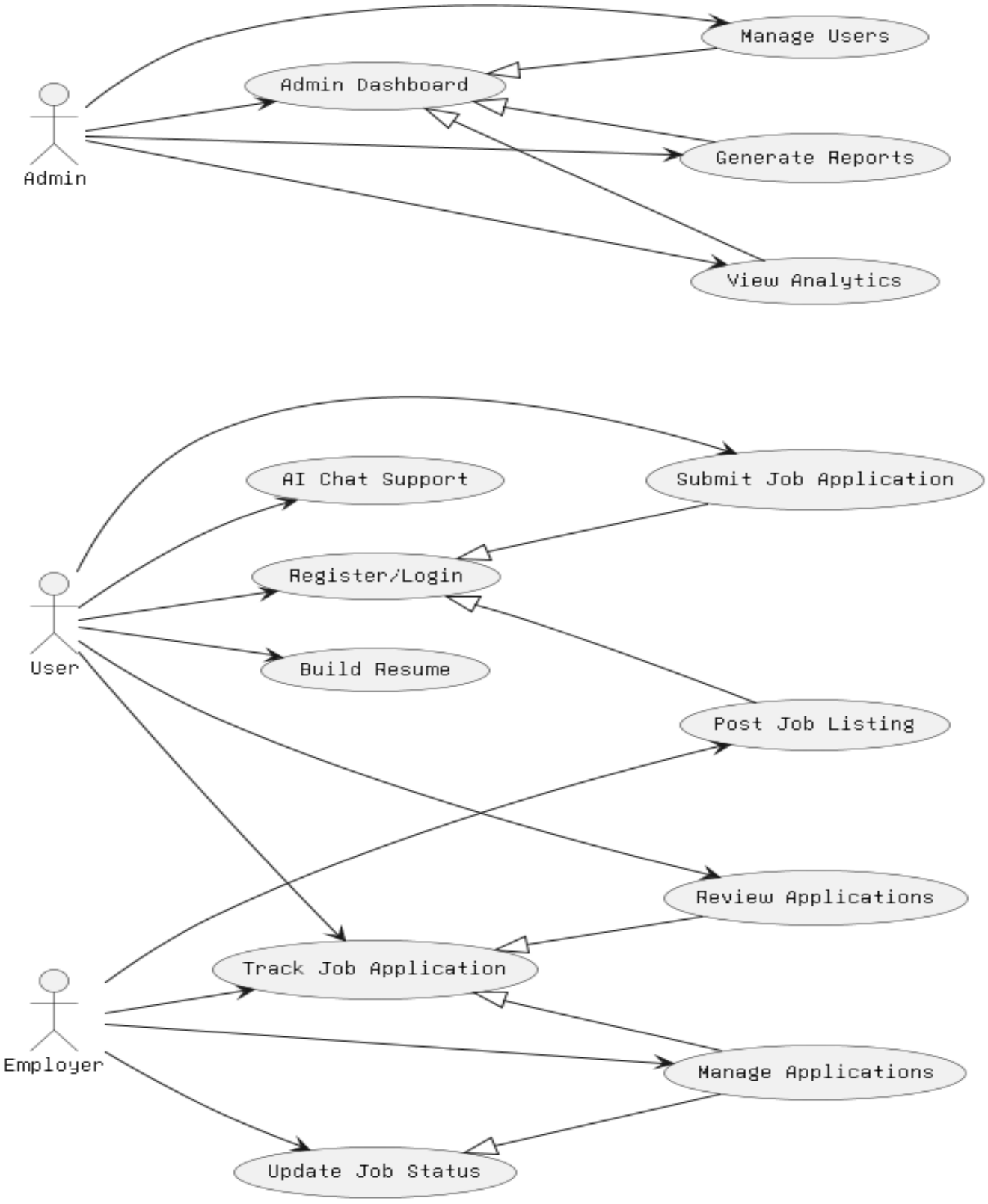


Fig - 4.3 Use case Diagram

4.3.2 ACTIVITY DIAGRAM

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all types of flow control by using different elements such as fork, join, etc.

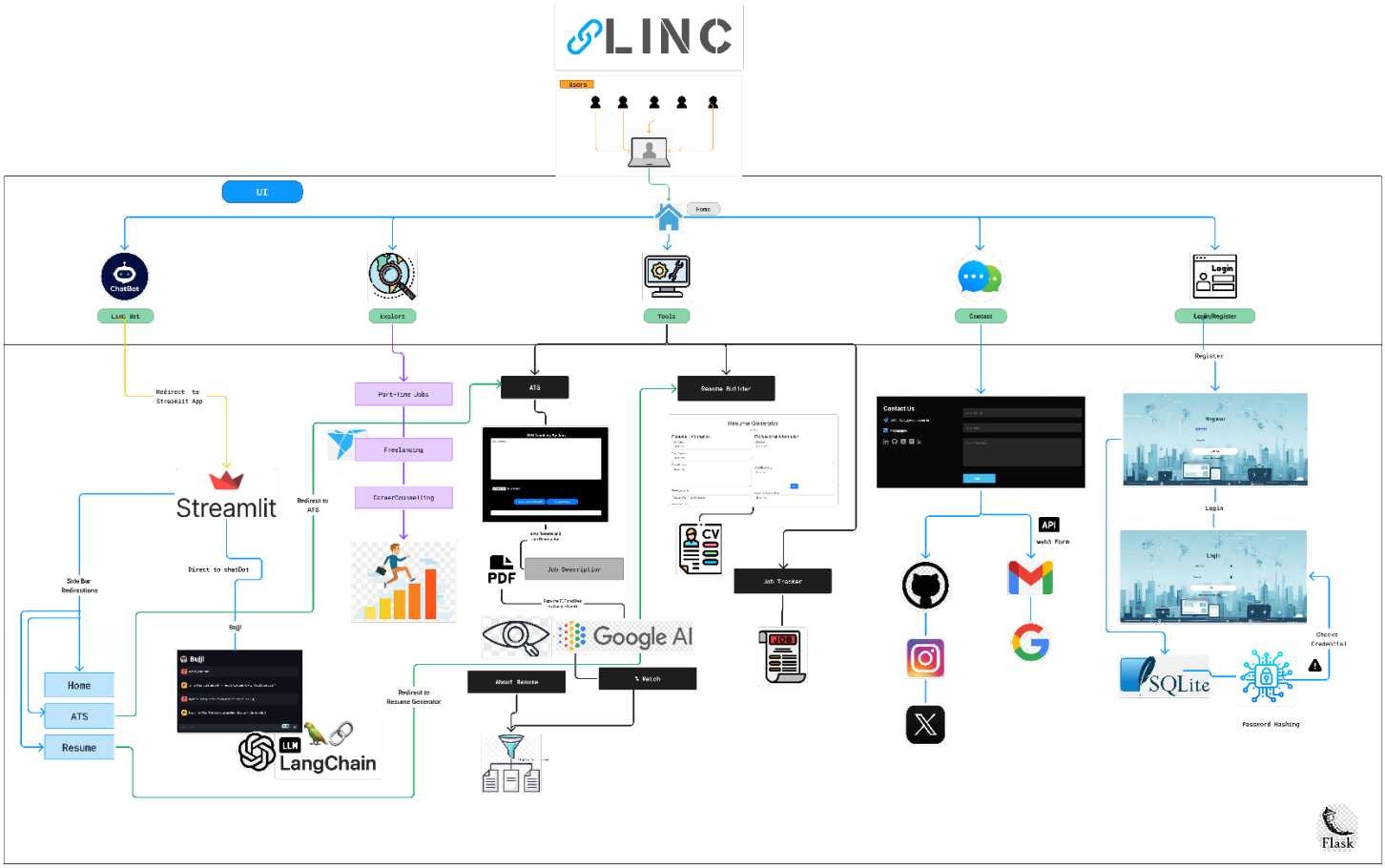


Fig – 4.4 : Activity Diagram

CHAPTER V

IMPLEMENTATION

IMPLEMENTATION

During the implementation phase, the development team will translate the design specifications into actual code. They will follow software development best practices, such as writing clean and modular code, conducting unit tests, and ensuring proper documentation. The modules and implementation process play a crucial role in bringing the system design to life and creating a functional and reliable scheduling platform for students and organisations.

5.1MODULES

- **get_gemini_response(input, pdf_content, prompt):** This function generates a response from the Gemini model based on the input text, PDF content, and prompt.
- **input_pdf_setup(uploaded_file):** This function converts an uploaded PDF file into a format that can be used by the Gemini model.
- **login() :**This function handles the login process for users, checking their username and password against the database.
- **evaluate():** This function evaluates a resume against a job description using the Gemini model and returns a response.
- **create_tables():** This function creates the database tables for the application.
- **addNewWEField():** This function adds a new textarea field for Work Experience.
- **addNewAQField():** This function adds a new textarea field for Academic Qualifications.
- **generateCV() :**This function generates a CV by extracting values from input fields and populating a template with the data.
- **printCV() :**This function prints the generated CV by hiding certain elements and triggering the print function.
- **submitForm(action):** This function submits a form to the /evaluate endpoint with the specified action and displays the response in the responseContainer element.
- **document.getElementById('tellMeButton').addEventListener('click',** ...) This event listener calls the submitForm function with the action 'Tell Me About the Resume' when the tellMeButton is clicked.
- **spinner() :**This function removes the spinner element after a short delay (1ms) using setTimeout.
- **$(".testimonial-carousel").owlCarousel({ ... }):** This line initializes the Owl Carousel plugin for the testimonials carousel.
- **__main__():** This is the main module that runs the Flask application when executed. It starts the Flask app with debug mode enabled**.**

5.2 MODULES DESCRIPTION

- **User Management Module:** This module handles user registration, login, and account management. It ensures secure user authentication and authorization, allowing users to create and manage their profiles. The module also provides functionalities for password recovery and account settings, ensuring a personalized and secure experience.
- **Resume Builder Module:** The Resume Builder module enables users to create and customize professional resumes. It provides a range of templates and formatting options to suit various industries and job roles. Users can input personal details, education, and work experience, and generate resumes that are tailored to meet industry standards and job requirements.
- **Applicant Tracking System (ATS) Module:** This module streamlines the recruitment process by managing job applications and tracking their statuses. It allows users to submit applications for job postings and provides real-time updates on application progress. For employers, the ATS facilitates application management, candidate evaluation, and communication with applicants.
- **Job Tracking Module:** The Job Tracking module provides users with a centralized interface to monitor and manage their job applications. It allows users to view the status of each application, set reminders for follow-ups, and track deadlines. This module ensures that users stay organized and informed throughout their job search journey.
- **AI-Powered Chat Support Module:** Utilizing Langchain, this module offers real-time support and guidance through an AI-driven chat interface. It provides users with instant answers to their queries, assistance with resume tips, interview preparation, and job application strategies. The chat support enhances user experience by offering personalized and timely help**.**
- **Job Posting and Management Module (Employer Side):** This module allows employers to post new job listings and manage their recruitment processes. Employers can create job postings, set application criteria, and review received applications. The module supports effective candidate management and helps employers streamline their hiring workflow.
- **Analytics and Reporting Module:** This module provides analytical insights and generates reports on job application trends, user engagement, and platform performance. It helps administrators and employers make data-driven decisions by analyzing application data, user behavior, and system usage**.**
- **Database Management Module:** The Database Management module oversees the storage, retrieval, and management of user data, resumes, job applications, and system configurations. It uses SQLite3 for efficient data handling and ensures data

integrity and security.

- **Notification System Module:** This module handles notifications and alerts for users regarding application status updates, job posting changes, and system messages. It ensures that users receive timely information and reminders relevant to their job search activities.

INTRODUCTION TO TECHNOLOGIES USED

5.2.1 PYTHON

Python plays a crucial role in the above code for the Face Recognition Attendance System. Here are the key reasons why Python is important in this code:

- **Backend Development:**
  Python serves as the backbone of LinC's server-side logic, powering the backend with its efficient and easy-to-write code. Flask, a micro web framework in Python, is used to handle the routing, request processing, and API integration, ensuring that the application is both robust and scalable.

- **Data Handling:**
  Python's extensive libraries, such as Pandas and SQLAlchemy, facilitate efficient data management and manipulation. These tools enable smooth interaction with the SQLite3 database, allowing for complex queries, data processing, and storage with minimal effort.

- **AI and Automation:**
  The AI-powered chat feature of LinC leverages Python’s machine learning libraries like TensorFlow and Langchain. These tools are used to train and deploy models that provide real-time assistance to users, automating responses and improving the overall user experience.

- **Scripting and Automation:**
  Python is also used for writing scripts that automate various backend tasks, such as data migration, testing, and server management. This automation reduces manual effort and ensures consistent performance across the platform.

In summary, Python's versatility, extensive libraries, and user-friendly syntax contribute to thesuccessful

implementation of the LinC. Python enables
seamless integration of computer vision, machine learning, web development, and datahandling functionalities, making it a suitable choice for developing such applications.

5.2.2 Google Gemini Pro Vision

1. **ImageRecognition:**
   Google Gemini Pro Vision is employed to enhance LinC's image processing capabilities, particularly for analyzing and extracting information from uploaded documents. This feature is useful for users who upload scanned copies of resumes or other job-related documents.
2. **Natural Language Processing (NLP):**
   The platform uses Google Gemini Pro Vision's NLP capabilities to improve the AI chat feature. By understanding user queries more accurately, the system can provide more relevant and context-aware responses, enhancing the overall user interaction.
3. **Optical Character Recognition (OCR):** Google Gemini Pro Vision's OCR technology is integrated into LinC to automatically extract text from images. This is particularly helpful for parsing data from physical documents or images, converting them into editable and searchable text within the platform.
4. **Real-Time Data Analysis:** With the help of Google Gemini Pro Vision, LinC performs real-time analysis of user interactions and data inputs. This technology aids in providing instant feedback and suggestions, making the platform more responsive and interactive.

5.2.3 FLASK

Flask plays a crucial role in the above code as it serves as the web framework for developing the LinC. Here are the key aspects of Flask's importance in the code:

Uses Of Flask

- **Web Application Framework**: Flask acts as the core framework for your face recognition-based attendance system. It provides the necessary tools and structures to handle HTTP requests, manage routing, and render HTML templates. With Flask, you can define routes and bind them to specific functions, allowing you to create a web-based interface for your attendance system.
- **Request Handling**: Flask handles incoming HTTP requests and allows you to define specific functions, known as view functions, that are executed when a particular URL is

accessed. In your project, Flask is used to define routes for different functionalities, such as capturing attendance, adding new users, listing users, and deleting users.

- **Template Rendering**: Flask integrates with templating engines, such as Jinja2, to dynamically generate HTML pages. Templates allow you to define the structure and layout of your web pages, making it easier to display information, handle form submissions, and present the results to users. In your project, Flask uses templates to render the home page, user listing page, and other pages with relevant attendance information.
- **Data Presentation**: Flask enables you to pass data from your Python code to HTML templates, allowing you to display information dynamically. For example, you can pass attendance records, user lists, and other data retrieved from the backend to the HTML templates, which can then be rendered and presented to the user. Flask provides mechanisms to pass data as variables to templates, making it straightforward to present information in a user-friendly format.
- **User Input Processing**: Flask handles user input by providing convenient methods for accessing form data, query parameters, and other request data. In your project, Flask is used to retrieve user input from forms when adding new users or deleting users. It facilitates capturing user-provided information and passing it to the relevant function
- **Backend Integration**: Flask allows you to connect your web application to the backend components of your project.

Overall, Flask's importance lies in its ability to handle routing, template rendering, request handling, and facilitate the development and deployment of the app

5.2.4 HTML

1. User Interface Design:

HTML is used as the foundational markup language for designing the front-end of LinC. It defines the structure of web pages, ensuring that the platform's interface is intuitive, user-friendly, and accessible across various devices.

2. Semantic Markup:

The use of semantic HTML tags in LinC ensures that the content is well-structured and easily understandable by both users and search engines. This improves the platform's SEO performance and makes it more accessible to users with disabilities.

3. Integration with CSS and JavaScript:

HTML works seamlessly with CSS for styling and JavaScript for dynamic interactions, creating a cohesive and visually appealing user interface. This integration allows for smooth animations, responsive designs, and interactive elements that enhance the user experience.

4. Form Handling:

HTML forms are employed in LinC to facilitate user input for various features such as job application submissions, resume uploads, and account registration. These forms are designed to be secure and efficient, ensuring that user data is captured accurately and transmitted securely to the backend.

5.2.5 JavaScript

1. Frontend Interactivity:

JavaScript is essential for adding dynamic behavior to the LinC platform. It enables features like form validation, interactive elements, and responsive content, ensuring that the website feels lively and responsive to user interactions**.**

2. DOM Manipulation:

JavaScript is used extensively for manipulating the Document Object Model (DOM) in LinC. This allows the platform to dynamically update content, handle user inputs, and create a seamless user experience without requiring full page reloads.

3. AJAX for Asynchronous Requests:

AJAX (Asynchronous JavaScript and XML) enables LinC to send and retrieve data from the server asynchronously. This technology is used to update parts of the webpage with new data without refreshing the entire page, enhancing the user experience by making it faster and more interactive.

4. Integration with Third-Party Libraries:

JavaScript integrates with various third-party libraries such as jQuery and Bootstrap to simplify complex tasks and enhance the visual appeal of the platform. These libraries provide pre-built components and functionalities that save development time and ensure cross-browser compatibility.

5.2.6 SQLite3

1. Lightweight Database:

SQLite3 is used as the database for LinC, providing a lightweight yet powerful solution for managing user data. Its serverless nature makes it ideal for applications where a full-fledged database server is unnecessary, reducing overhead and simplifying deployment.

2. Data Persistence:

SQLite3 ensures that all user data, including job applications, resumes, and profiles, is stored persistently. This allows users to access their information anytime they return to the platform, ensuring a consistent and reliable experience.

3. Easy Integration with Python:

SQLite3 is seamlessly integrated with Python, allowing the backend of LinC to perform complex queries and manage data efficiently. This integration enables rapid development and easy maintenance, as both the database and application logic are tightly coupled.

4. Scalability:

While SQLite3 is designed for smaller-scale applications, it is scalable enough to handle LinC's expected user base. The database is optimized for read-heavy operations, making it an excellent choice for a job portal where data retrieval is frequent.

5.2.7 CSS

1. Styling and Layout:

CSS (Cascading Style Sheets) is used to define the visual presentation of LinC's web pages. It allows the platform to maintain a consistent look and feel across all pages, ensuring that users have a visually cohesive experience.

2. Responsive Design:

With CSS, LinC is designed to be fully responsive, meaning it adapts to different screen sizes and devices. This ensures that users can access the platform from desktops, tablets, and smartphones without compromising the user experience.

3. Animation and Transitions:

CSS is utilized to create smooth animations and transitions on the platform. These subtle effects enhance the user experience by providing visual feedback during interactions, making the platform feel more engaging and polished.

4. Cross-Browser Compatibility:

CSS is written with cross-browser compatibility in mind, ensuring that LinC displays correctly on all major web browsers. This approach prevents layout issues and ensures a consistent user experience regardless of the browser being used.

5.2.8 HTML, CSS AND BOOTSTRAP

- **HTML (Hypertext Markup Language):** HTML is the standard markup language for creating the structure and content of web pages. In your project, HTML is used to define the overall structure of the web pages, including headers, navigation menus, tables, forms, and other elements.
- **CSS (Cascading Style Sheets):** CSS is a style sheet language used for describing the look and formatting of a document written in HTML. It provides the ability to control the visual appearance of elements, such as fonts, colors, margins, and layouts. In your project, CSS is used to customize and style the HTML elements, ensuring a visually appealing and consistent design.
- **Bootstrap**: Bootstrap is a popular front-end framework that provides pre-designed templates, components, and CSS classes for building responsive and mobile-first websites. It simplifies the process of creating a professional-looking and responsive user interface. In your project, Bootstrap is utilized to enhance the styling and responsiveness of the web pages, making them adapt well to different screen sizes and devices.
- **Responsive Design**: With the use of CSS and Bootstrap, your project's web pages are designed to be responsive. This means that the layout and elements adjust dynamically based on the screen size of the device used to access the application. It ensures an optimal viewing experience across various devices, including desktops, tablets, and smartphones.
- **Templates and Components**: Bootstrap provides a wide range of pre-designed templates and ready-to-use components, such as navigation bars, buttons, forms, modals, and grids. These templates and components can be easily integrated into your

project, saving time and effort in designing and coding from scratch.

By utilizing HTML, CSS, and Bootstrap, your project achieves a well-structured, visually appealing, and responsive user interface. HTML defines the structure, CSS adds style and formatting, and Bootstrap enhances the design with pre-built templates and responsive components. This combination of technologies enables you to create an engaging and user-friendly interface for your face recognition-based attendance system.

5.3 SAMPLE CODE

```
import os
import base64
import io
import openai
import json
from flask import Flask, render_template, request, redirect, url_for, flash, session, jsonify,send_file
from dotenv import load_dotenv
from flask_sqlalchemy import SQLAlchemy
from werkzeug.security import generate_password_hash, check_password_hash
from werkzeug.utils import secure_filename
from pdf2image import convert_from_bytes
import google.generativeai as genai
from PIL import Image
from flask import render_template_string


# Load environment variables
load_dotenv()
genai.configure(api_key=os.getenv("GOOGLE_API_KEY"))

app = Flask(__name__)

app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///users.db'
app.config['SECRET_KEY'] = 'your_secret_key'
app.config['UPLOAD_FOLDER'] = 'uploads/'
```

```
os.makedirs(app.config['UPLOAD_FOLDER'], exist_ok=True)
db = SQLAlchemy(app)

class User(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    username = db.Column(db.String(80), unique=True, nullable=False)
    password = db.Column(db.String(80), nullable=False)
    # email = db.Column(db.String(120), unique=True, nullable=False)

    def __repr__(self):
        return f'<User {self.username}>'

def create_tables():
    with app.app_context():
        db.create_all()

def get_gemini_response(input, pdf_content, prompt):
    model = genai.GenerativeModel('gemini-pro-vision')
    response = model.generate_content([input, pdf_content[0], prompt])
    return response.text

def input_pdf_setup(uploaded_file):
    poppler_path = 'C:\\Program Files\\poppler-24.02.0\\Library\\bin'
    images = convert_from_bytes(uploaded_file.read(), poppler_path=poppler_path)
    first_page = images[0]

    # Convert to bytes
    img_byte_arr = io.BytesIO()
    first_page.save(img_byte_arr, format='JPEG')
    img_byte_arr = img_byte_arr.getvalue()

    pdf_parts = [
        {
            "mime_type": "image/jpeg",
            "data": base64.b64encode(img_byte_arr).decode()  # encode to base64
        }
    ]
    return pdf_parts

@app.route('/')
def home():
    is_authenticated = 'user_id' in session
    return render_template('home.html', is_authenticated=is_authenticated)

```

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

        user = User.query.filter_by(username=username).first()
        if user and check_password_hash(user.password, password):
            session['user_id'] = user.id
            flash('     Login successful!', 'success')
            return redirect(url_for('home'))
        else:
            flash('     Invalid username or password.', 'danger')

    return render_template('login.html')

@app.route('/logout')
def logout():
    session.pop('user_id', None)
    flash('       You have been logged out.', 'success')
    return redirect(url_for('home'))


@app.route('/register', methods=['GET', 'POST'])
def register():
    if request.method == 'POST':
            new_username = request.form['newUsername']
            new_password = generate_password_hash(request.form['newPassword'])
            # email = request.form['email']

            existing_user = User.query.filter_by(username=new_username).first()
            if existing_user:
                flash('Username already exists!', 'danger')
            else:
                new_user = User(username=new_username, password=new_password)
                db.session.add(new_user)
                db.session.commit()
                flash('Registration successful!', 'success')
                return redirect(url_for('login'))

        return render_template('register.html')

    @app.route('/streamlit')
    def streamlit_embed():
        return redirect("http://localhost:8501")

```

```
120.  @app.route('/part-time-jobs')
121.  def part_time_jobs():
122.      # Render the jobs page
123.      return render_template('job-list.html')
124.
125.  @app.route('/job-detail')
126.  def job_detail():
127.      return render_template('job-detail.html')
128.
129.  @app.route('/job-list')
130.  def job_list():
131.      return render_template('job-list.html')
132.
133.  @app.route('/freelancing')
134.  def freelancing():
135.      # Render the freelancing page
136.      return render_template('job-list.html')
137.
138.  @app.route('/career-counseling')
139.  def career_counseling():
140.      # Render the career counseling page
141.      return render_template('career-counseling.html')
142.
143.  @app.route('/ats')
144.  def ats():
145.      return render_template('ats.html')
146.
147.  @app.route('/resume')
148.  def resume():
149.      return render_template('resume.html')
150.
151.  @app.route('/dev')
152.  def dev():
153.      return render_template('dev.html')
154.
155.  @app.route('/evaluate', methods=['POST'])
156.  def evaluate():
157.      if 'file' not in request.files or request.files['file'].filename == '':
158.          return jsonify({"error": "No file uploaded"}), 400
159.
160.      uploaded_file = request.files['file']
161.      input_text = request.form['input_text']
162.      action = request.form['action']
163.
164.      if uploaded_file:
```

```
165.           pdf_content = input_pdf_setup(uploaded_file)
166.
167.           if action == "Tell Me About the Resume":
168.               input_prompt = """
169.               You are an experienced Technical Human Resource Manager. Your task
   is to review the provided resume of a student against the provided job description
   from an organization looking to fill a part-time or freelance position.
170.
171.           First, display the student's name.
172.           Second, list their skills.
173.           Third, provide their educational background.
174.
175.            Next, provide a detailed evaluation of whether the student's profile
   aligns with the role specified in the job description.
176.             Highlight the strengths and weaknesses of the student's resume in
   relation to the job requirements outlined by the organization.
177.               Focus on the student's relevant skills, experience, educational
   background, and any projects or internships that demonstrate their suitability for
   the role.
178.           Additionally, mention any notable achievements or certifications that
   enhance the student's profile.
179.              Provide a clear and concise summary of how well the student's
   qualifications match the job description.
180.               """
181.

182.           elif action == "Percentage match":
183.               input_prompt = """
184.                You are a skilled ATS (Applicant Tracking System) scanner with a
   deep understanding of part-time and freelance job requirements.
185.        Your task is to evaluate the student's resume against the provided job
   description from an organization. Calculate the percentage match of the resume to
   the job description.
186.
187.       First, provide the match percentage.
188.       Second, list any missing keywords or skills that are critical for the job.
189.        Third, offer your overall assessment of the student's fit for the role,
   highlighting their strengths and areas for improvement.
190.
191.       Your evaluation should help the student understand how closely their resume
   aligns with the job requirements and what they can improve.
192.               """
193.

194.
195.           response = get_gemini_response(input_text, pdf_content, input_prompt)
```

```
196.           return jsonify({"response": response})
197.
198.       return jsonify({"error": "Error processing the file"}), 500
199.
200.
201.   if __name__ == '__main__':
202.       create_tables()
203.       app.run(debug=True)
204.
205.
206.
207.   #Home.html
208.
209.   <!DOCTYPE html>
210.   <html lang="en">
211.   <head>
212.       <meta charset="UTF-8">
213.       <meta name="viewport" content="width=device-width, initial-scale=1.0">
214.       <title>LinC</title>
215.       <link rel="stylesheet" href="{{ url_for('static', filename='Hom.css') }}">
216.       <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
217.       <script defer src="{{ url_for('static', filename='script.js') }}"></script>
218.                   <script       src="https://kit.fontawesome.com/1572a930d7.js"
   crossorigin="anonymous"></script>
219.   </head>
220.   <body>
221.       <div id="header">
222.           <div class="container1">
223.               <nav>
224.                       <img  src="{{  url_for('static',  filename='logo.png')  }}"
   class="logo" alt="Logo">
225.                   <ul id="sidemenu">
226.                       <li><a href="#">Home</a></li>
227.                            <li><a  href="{{  url_for('streamlit_embed')  }}">LinC
   Bot</a></li>
228.                       <li><a href="#services">Explore</a></li>
229.                       <li><a href="#tools">Tools</a></li>
230.                       <li><a href="#contact">Contact</a></li>
231.                       <li id="auth-button">
232.                           {% if is_authenticated %}
233.                               <a href="{{ url_for('logout') }}">Logout</a>
234.                           {% else %}
235.                               <a href="{{ url_for('login') }}">Login/Register</a>
236.                           {% endif %}
```

```
                    </li>
                </ul>
            </nav>
        </div>
    </div>

    <!-- -------------services------------ -->

    <div id="services">
        <div class="container1">
            <h1 class="sub-title">Explore Opportunities</h1>
            <div class="services-list">
                <div>
                    <i class="fa-solid fa-briefcase"></i>
                    <h2>Part-Time Jobs</h2>
                     <p>Explore various part-time job opportunities that fit your schedule and help you gain valuable work experience while you study.</p>
                    <a href="/part-time-jobs">Apply now</a>
                </div>
                <div>
                    <i class="fa-solid fa-laptop-code"></i>
                    <h2>Freelancing</h2>
                   <p>Find freelance projects that match your skills and interests, allowing you to work on your terms and build your portfolio.</p>
                    <a href="/freelancing">Start freelancing</a>
                </div>
                <div>
                    <i class="fa-solid fa-user-tie"></i>
                    <h2>Career Counseling</h2>
                      <p>Get personalized career advice and guidance to help you navigate your career path, improve your resume, and prepare for interviews.</p>
                    <a href="/career-counseling">Get guidance</a>
                </div>
            </div>
        </div>
    </div>

    <!-- --------------------- Tools ---------------------------- -->

    <div id="tools">
        <div class="container1">
            <h1 class="sub-title">Tools & Resources</h1>
            <div class="work-list">
                <!-- ATS System Tool -->
                <div class="work">
                    <img src="/static/Images/ats.jpeg" alt="ATS System">
```

```
280.
281.                     <div class="layer">
282.                         <h3>ATS System</h3>
283.                           <p>Automate the resume screening process and match job
    descriptions with candidate profiles efficiently.</p>
284.                         <a href="{{ url_for('ats') }}"><i class="fa-solid fa-arrow-
    up-right-from-square"></i></a>
285.                     </div>
286.                 </div>
287.
288.                 <div class="work">
289.                          <img  src="/static/Images/resumebuilder.jpeg"  alt="Resume
    Builder">
290.                     <div class="layer">
291.                         <h3>Resume Builder</h3>
292.                        <p>Create professional resumes with ease using our intuitive
    resume builder tool.</p>
293.                          <a href="{{ url_for('resume') }}"><i class="fa-solid fa-
    arrow-up-right-from-square"></i></a>
294.                     </div>
295.                 </div>
296.
297.                 <div class="work">
298.                     <img src="/static/Images/jobtracking.jpeg" alt="Job Tracker">
299.                     <div class="layer">
300.                         <h3>Job Tracker</h3>
301.                        <p>Keep track of your job applications and follow up on them
    with our job tracker tool.</p>
302.                         <a href="{{ url_for('ats') }}"><i class="fa-solid fa-arrow-
    up-right-from-square"></i></a>
303.                     </div>
304.                 </div>
305.             </div>
306.             <a href="{{ url_for('dev') }}" class="btn">Developers</a>
307.         </div>
308.     </div>
309.

310.     <!-- -------------contact------------ -->
311.     <div id="contact">
312.         <div class="container1">
313.             <div class="row">
314.                 <div class="contact-left">
315.                     <h1 class="sub-title">Contact Us</h1>
316.                                                 <p><i    class="fa-solid    fa-paper-
    plane"></i>22l31a5484@vignaniit.edu.in</p>
```

```
317.                 <p><i class="fa-solid fa-square-phone"></i>7702853524</p>
318.                 <div class="social-icons">
319.                         <a  href="https://www.linkedin.com/in/vishnu-vardhan-
   36848a275/"><i class="fa-brands fa-linkedin-in"></i></a>
320.                       <a href="https://github.com/hello-mr-vishu"><i class="fa-
   brands fa-github"></i></a>
321.                      <a href="https://twitter.com/hello_mr_vishu"><i class="fa-
   brands fa-square-x-twitter"></i></a>
322.                         <a  href="https://www.instagram.com/hello_mr.vishu/"><i
   class="fa-brands fa-square-instagram"></i></a>
323.                        <a  href="https://www.kaggle.com/vishnu80"><i class="fa-
   brands fa-kaggle"></i></a>
324.                 </div>
325.
326.             </div>
327.             <div class="contact-right">
328.                 <form  method="POST" name="submit-to-google-sheet" id="contact-
   form">
329.                      <input type="hidden" name="access_key" value="11711a0b-
   bc53-47d7-bca9-225fd6f27f56">
330.                       <input type="text" name="Name" placeholder="Your Name"
   required>
331.                     <input type="email" name="Email" placeholder="Your Email"
   required>
332.                         <textarea  name="Message"  rows="6"  placeholder="Your
   Message"></textarea>
333.                     <button type="submit" class="btn btn2">Submit</button>
334.                     <div id="result"></div>
335.                 </form>
336.                 <span id="msg"></span>
337.             </div>
338.         </div>
339.     </div>
340.
341.       <script>
342.         document.addEventListener('DOMContentLoaded', function() {
343.        const  googleSheetURL  =  'https://script.google.com/macros/s/AKfycbw-
   GUGjpFuzSZiMCTAUcmpx5X90zvVDzy_NaZd19pEqP8Iyi06VNdL8rs72nQyIufOuEA/exec';
344.     const web3FormsURL = 'https://api.web3forms.com/submit';
345.     const form = document.getElementById('contact-form');
346.     const result = document.getElementById('result');
347.     const msg = document.getElementById("msg");
348.
349.     form.addEventListener('submit', function(e) {
350.         e.preventDefault();
351.         const formData = new FormData(form);
```

```
352.            const object = Object.fromEntries(formData);
353.            const json = JSON.stringify(object);
354.            result.innerHTML = "Please wait...";
355.            msg.innerHTML = "Sending message...";
356.
357.            // Submit to Google Sheets
358.            fetch(googleSheetURL, { method: 'POST', body: new FormData(form) })
359.                .then(response => {
360.                    if (response.ok) {
361.                        msg.innerHTML = "Message sent to LinC successfully";
362.                    } else {
363.                        msg.innerHTML = "Error sending message to LinC";
364.                    }
365.                    setTimeout(function() {
366.                        msg.innerHTML = "";
367.                    }, 5000);
368.                })
369.                .catch(error => console.error('LinC Error!', error.message));
370.
371.            // Submit to Web3Forms
372.            fetch(web3FormsURL, {
373.                method: 'POST',
374.                headers: {
375.                    'Content-Type': 'application/json',
376.                    'Accept': 'application/json'
377.                },
378.                body: json
379.            })
380.            .then(async (response) => {
381.                let json = await response.json();
382.                if (response.status == 200) {
383.                    result.innerHTML = "Form submitted successfully....! We'll get
    back to you soon";
384.                } else {
385.                    console.log(response);
386.                    result.innerHTML = json.message;
387.                }
388.            })
389.            .catch(error => {
390.                console.log(error);
391.                result.innerHTML = "Something went wrong!";
392.            })
393.            .then(function() {
394.                form.reset();
395.                setTimeout(() => {
396.                    result.style.display = "none";
```

```
397.                 }, 3000);
398.             });
399.         });
400.     });
401.
402.           </script>
403.
404.     </body>
405.     </html>
406.
407.
408.

409.     <!DOCTYPE html>
410.     <html lang="en">
411.     <head>
412.         <meta charset="UTF-8">
413.         <meta name="viewport" content="width=device-width, initial-scale=1.0">
414.         <title>Login</title>
415.         <link rel="stylesheet" href="/static/log.css">
416.              <link    href='https://unpkg.com/boxicons@2.1.4/css/boxicons.min.css'
    rel='stylesheet'>
417.     </head>
418.
419.     <body>
420.
421.         <div class="wrapper">
422.             <form action="{{ url_for('login') }}" method="POST">
423.                 <h1>Login</h1>
424.                 <div class="input-box">
425.                                 <input  type="text"  id="Username"  name="username"
    placeholder="Username" required>
426.                     <i class='bx bxs-user'></i>
427.                 </div>
428.                 <div class="input-box">
429.                             <input  type="password"  id="password"  name="password"
    placeholder="Password" required>
430.                     <i class='bx bxs-lock-alt'></i>
431.                 </div>
432.                 <button type="submit" value="Login" class="btn">Sign In</button>
433.                 <div class="register-link">
434.                     <p>Don't have an account? <a href="{{ url_for('register')
    }}">Register</a></p>
435.                 </div>
436.             </form>
437.             {% with messages = get_flashed_messages(with_categories=true) %}
```

```
438.            {% if messages %}
439.                <ul>
440.                    {% for category, message in messages %}
441.                        <li class="{{ category }}">     {{ message }}</li>
442.                    {% endfor %}
443.                </ul>
444.            {% endif %}
445.        {% endwith %}
446.        </div>
447.    </body>
448.
449.      </html>
450.
451.
452.
453.
454.
455.
456.
457.
458.
459.
460.
```

CHAPTER VI

SCREENSHOTS

6.1 HOME PAGE

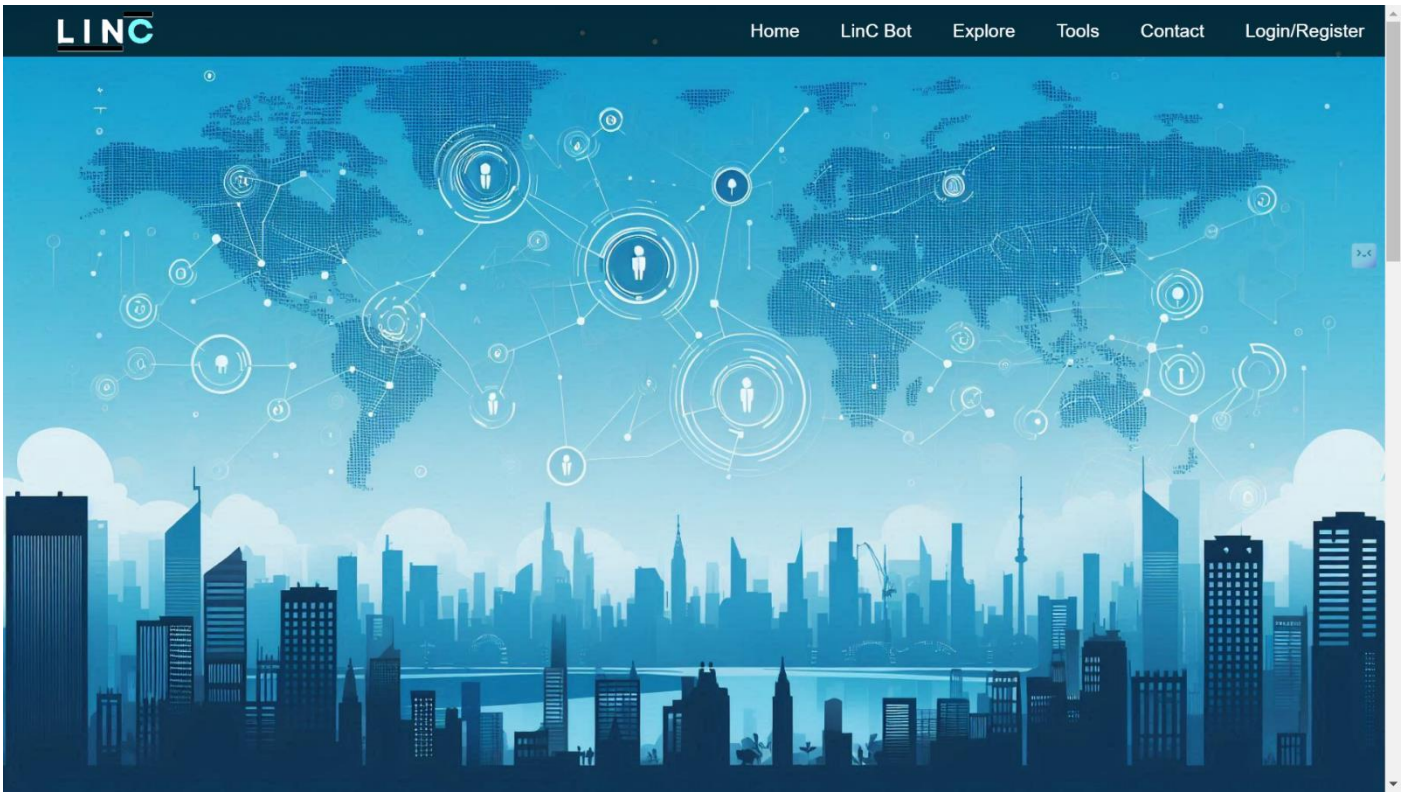
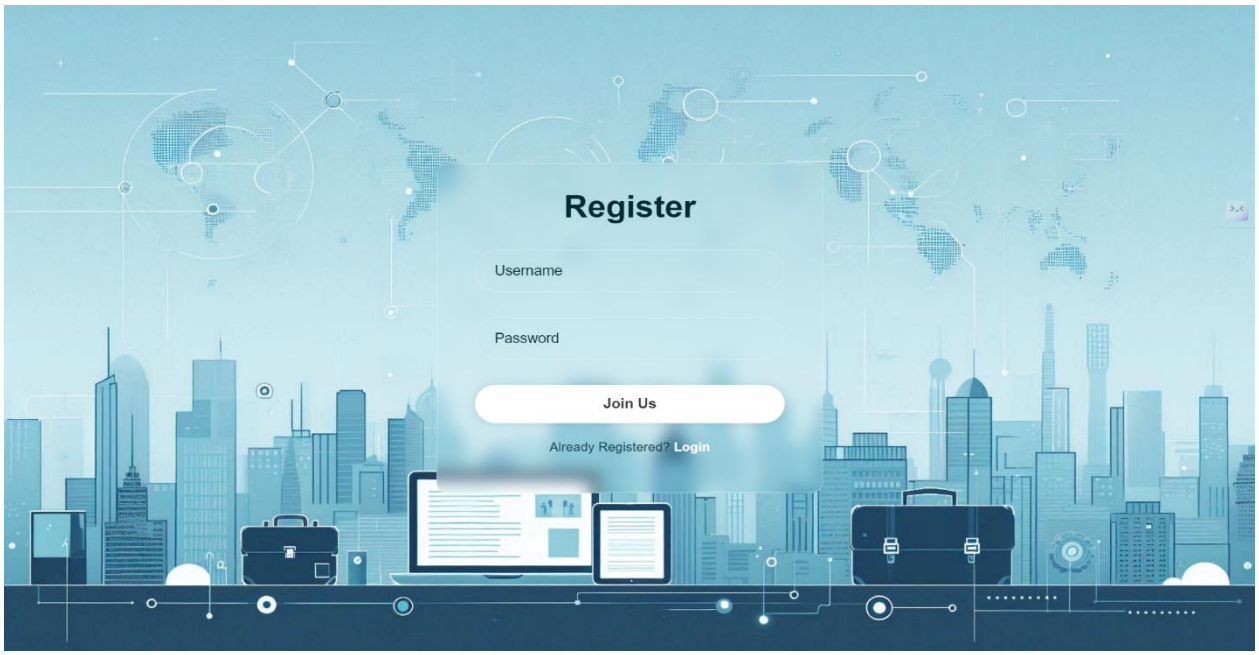


Fig - 6.1 : Home page

6.2 LOGIN AND REGISTER PAGE



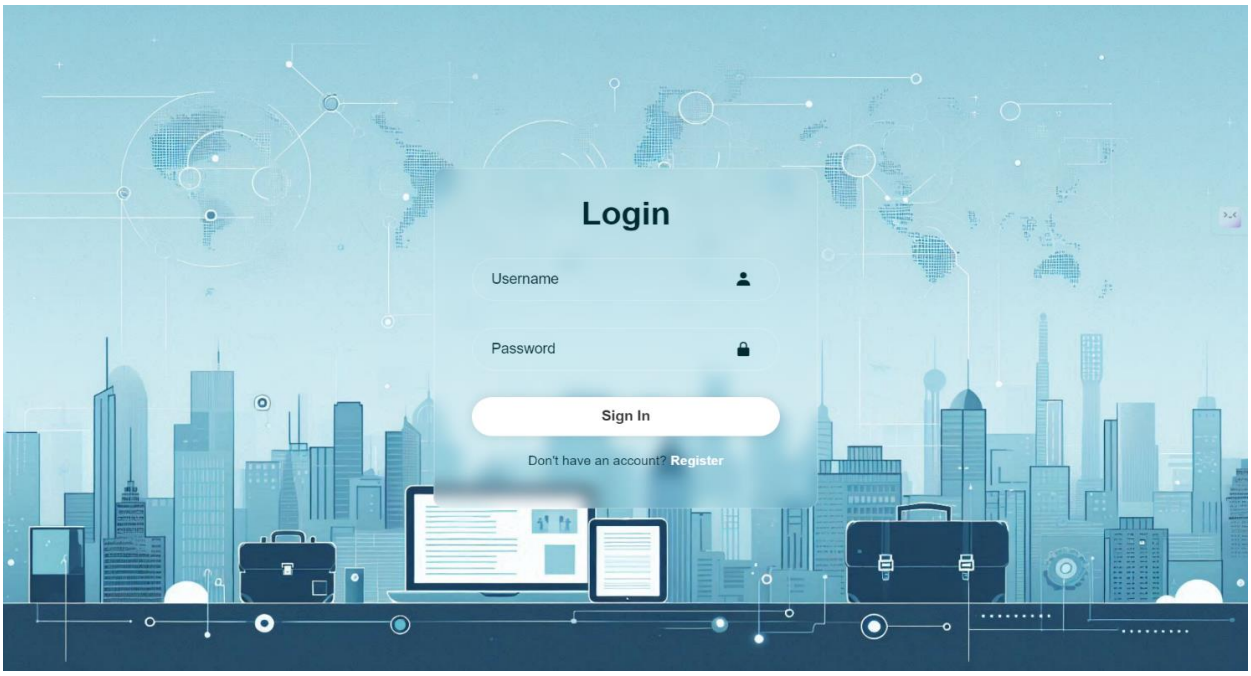
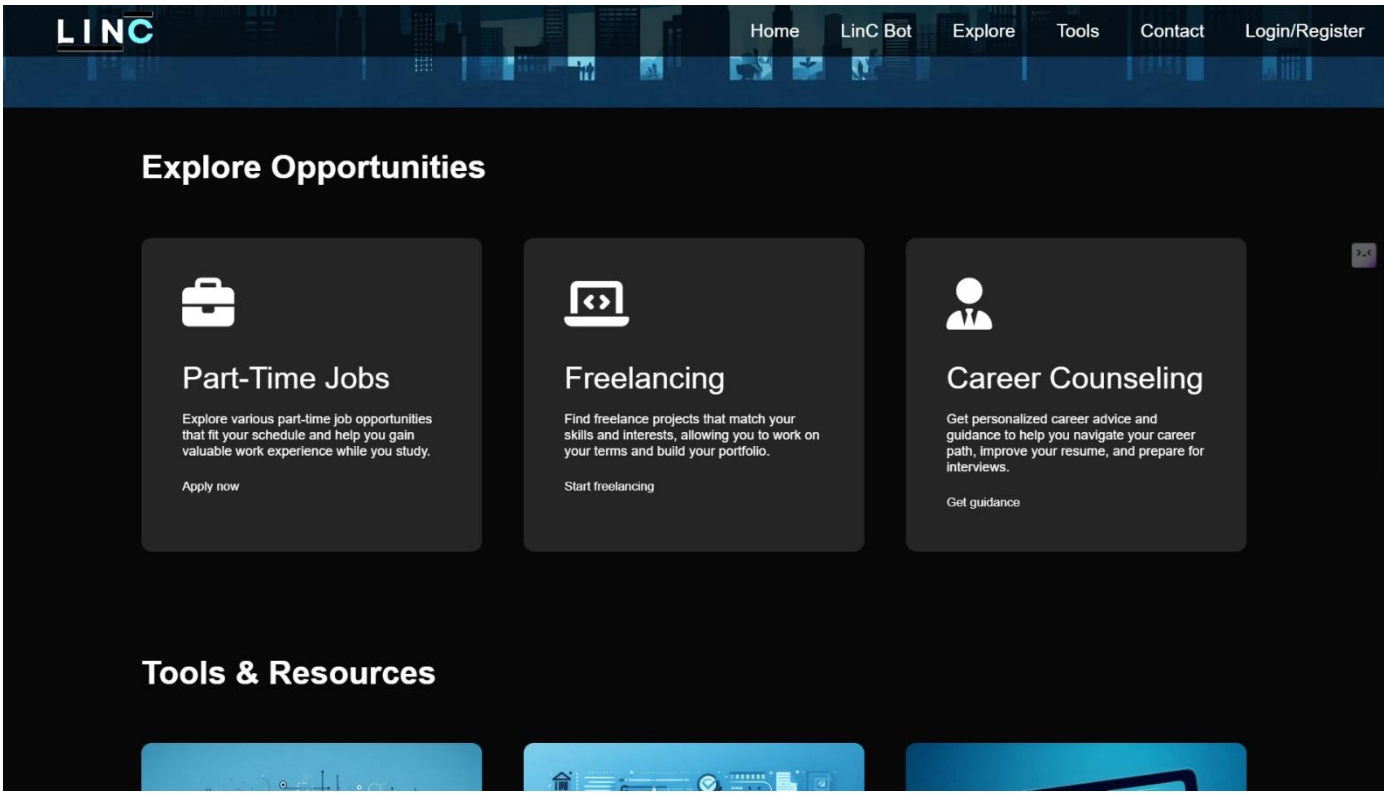


Fig – 6.2 : Login Page

6.3 EXPLORE AND TOOLS



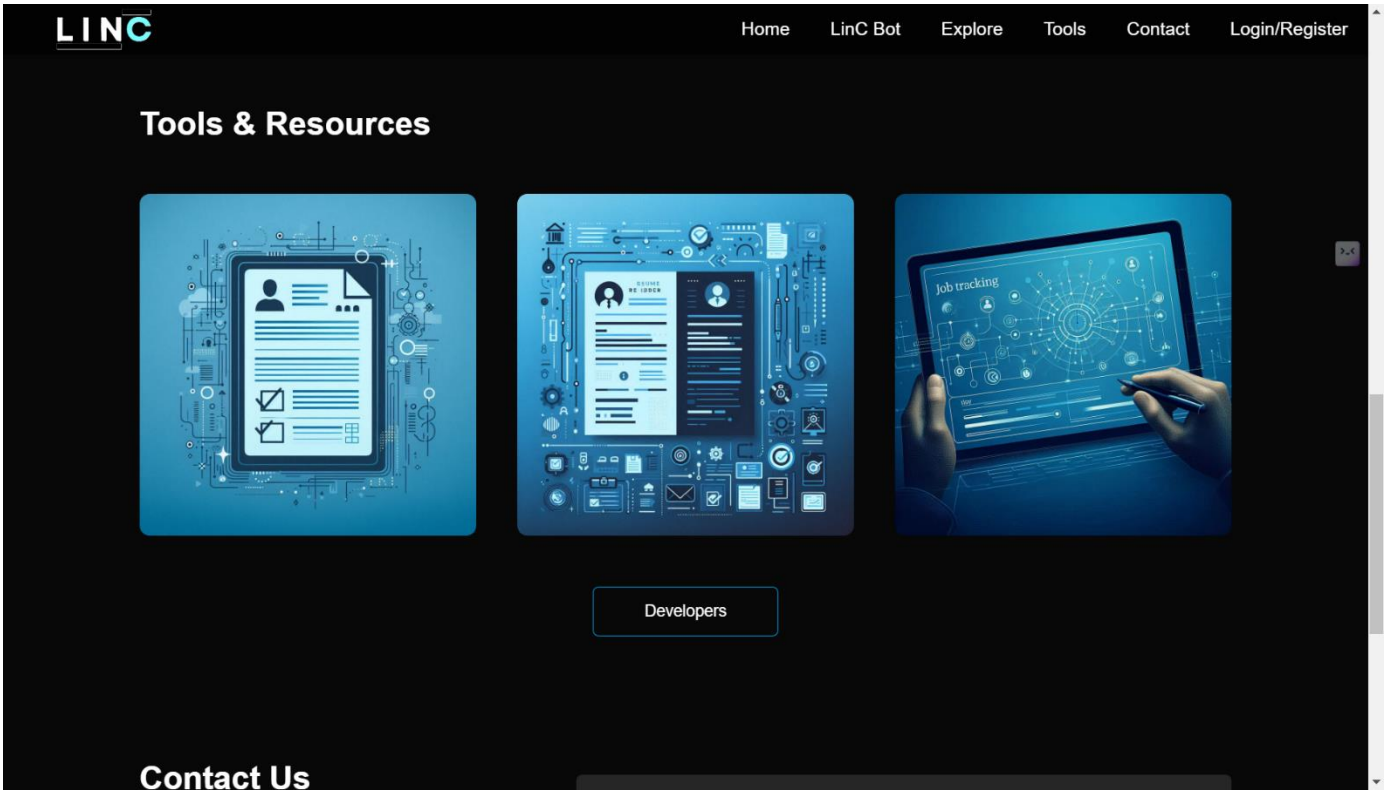


Fig – 6.3 : Explore and Tools

6.4 CONTACT

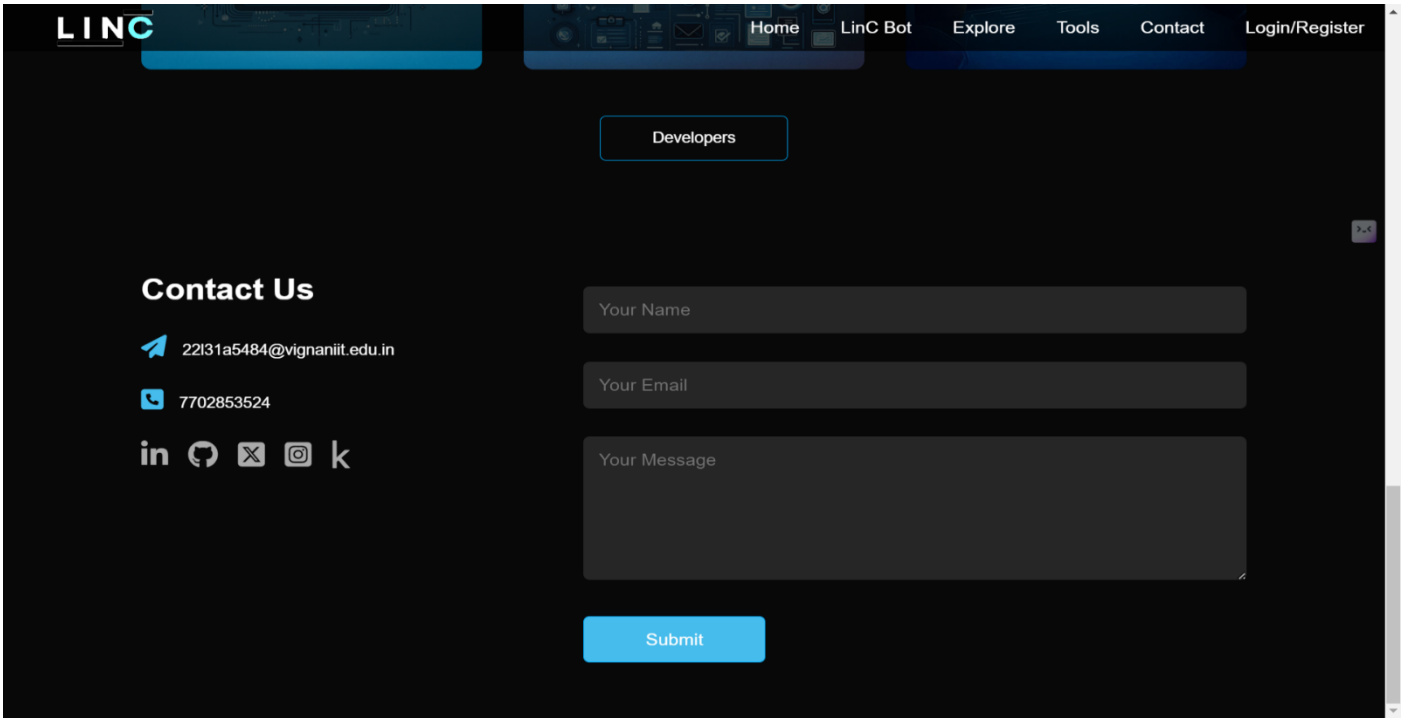


Fig – 6.4 : Contact Form

6.5 ATS

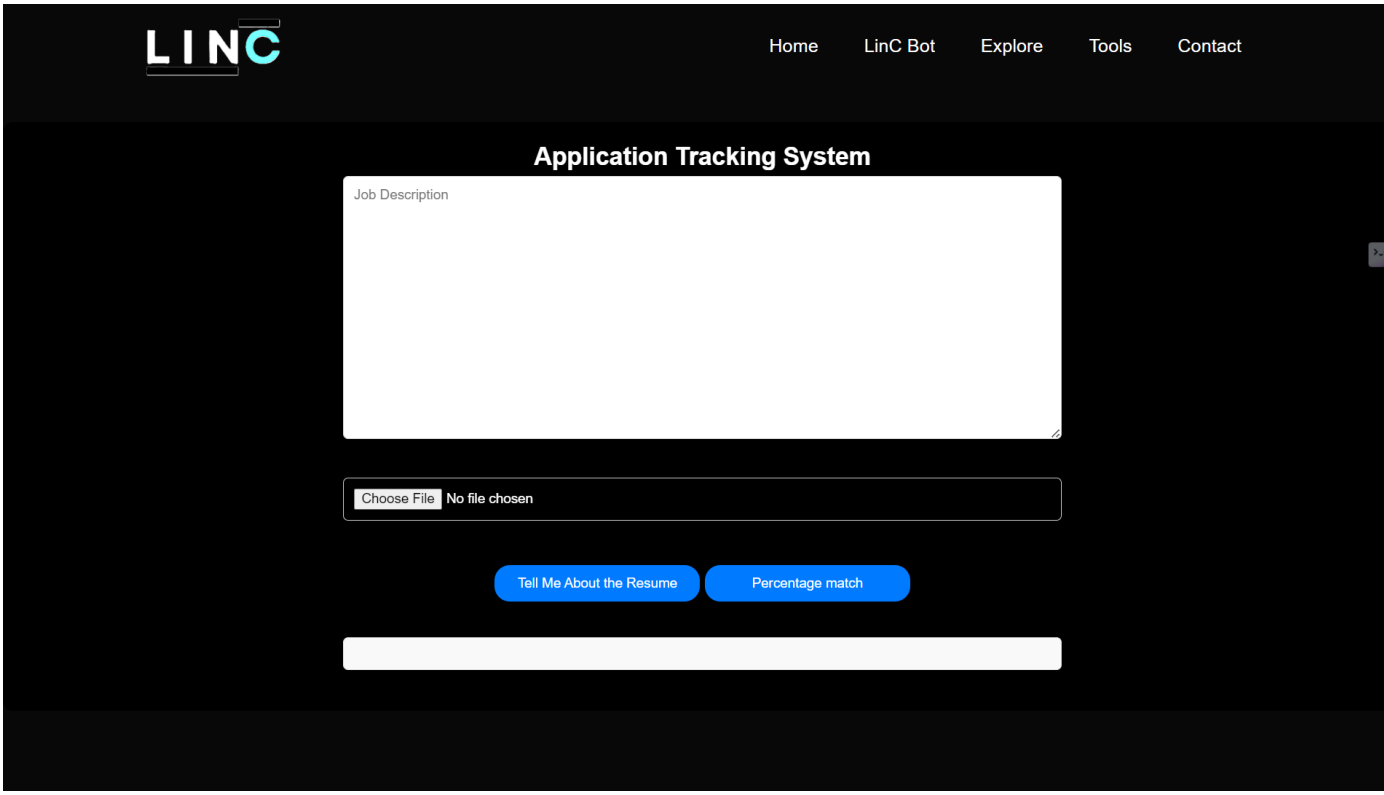


Fig – 6.5 : Application Tracking System

6.6 RESUME BUILDER

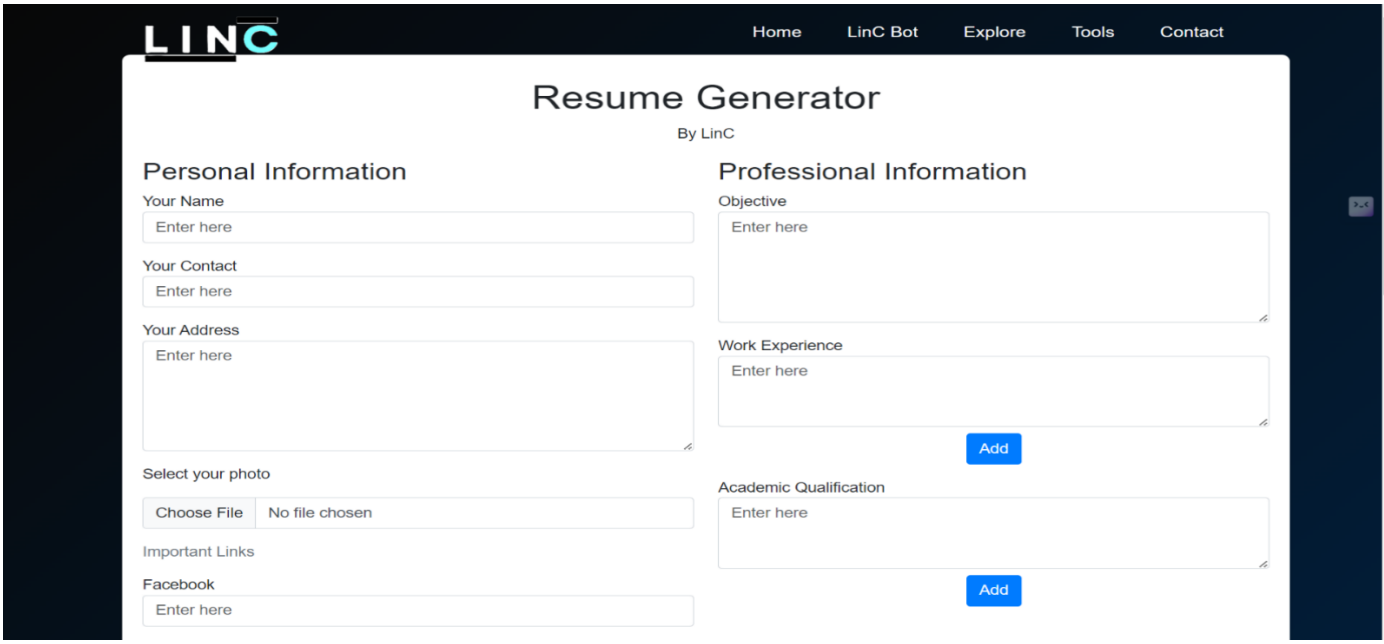


Fig – 6.6: Resume Builder

6.7 CHATBOT STREAMLIT

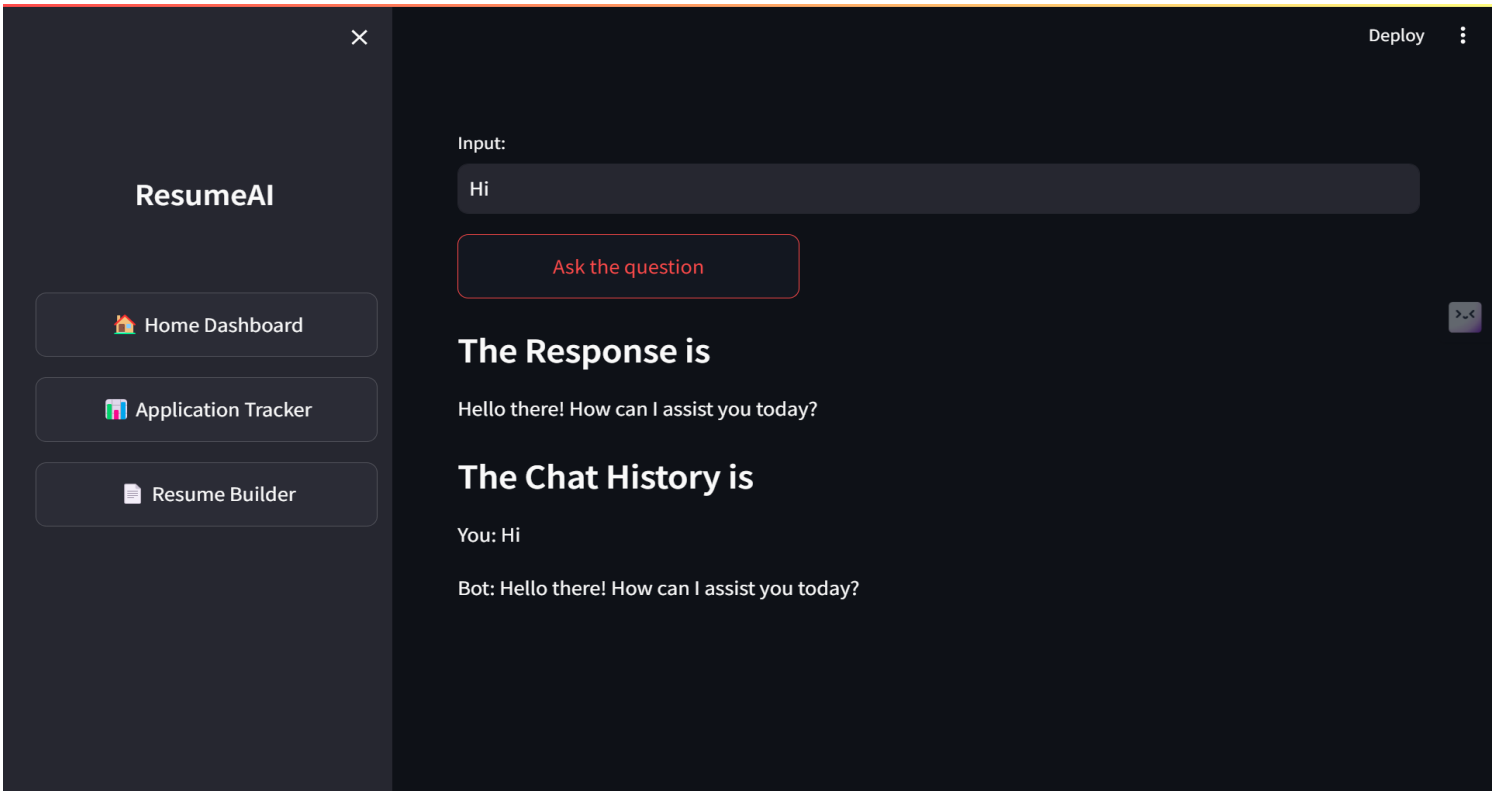


Fig – 6.7 : ChatBot

CHAPTER VII

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 CONCLUSION

The LinC platform represents a significant advancement in how job seekers connect with opportunities and manage their career progress. By integrating cutting-edge technologies like Python for backend development, Google Gemini Pro Vision for enhanced data analysis, and various frontend tools such as JavaScript and CSS, LinC offers a comprehensive, user-friendly, and efficient solution to the challenges faced in the job search process.

LinC addresses critical gaps in existing job portals by introducing features like a customizable resume builder, an accessible Applicant Tracking System (ATS), and real-time support through AI-powered chat. These innovations not only streamline the job application process but also empower users with the tools and knowledge needed to succeed in a competitive job market.

Throughout the development of LinC, the focus has remained on delivering a platform that is both powerful and intuitive, ensuring that users of all technical backgrounds can navigate it with ease. The combination of a robust backend, an interactive and responsive frontend, and the seamless integration of various technologies makes LinC a reliable and forward-thinking platform.

As the job market continues to evolve, LinC is well-positioned to adapt and grow, offering even more features and enhancements in the future. The project's success lies in its ability to meet user needs effectively while maintaining flexibility for ongoing improvements. LinC is not just a job portal; it's a comprehensive career management tool designed to support users in every step of their professional journey.

FUTURE ENHANCEMENTS

1. Integration with More Job Boards:

In the future, LinC can expand its job listings by integrating with more job boards and recruitment platforms worldwide. By aggregating job postings from various sources, LinC can offer users a more comprehensive range of opportunities, tailored to their specific skills and preferences. This will increase the platform's relevance and utility, especially for users seeking niche or international roles.

2. Advanced AI-Powered Career Guidance:

Building on the existing AI features, LinC could incorporate an advanced AI-powered career guidance system. This enhancement would offer personalized career advice based on user profiles, job market trends, and skill gaps. By analyzing user data and market demands, the AI could suggest career paths, educational resources, or certifications to help users achieve their career goals.

3. Gamification for Skill Development:

Introducing gamification elements could make the process of skill development and job application more engaging. LinC could offer users rewards, badges, or levels based on their activities, such as completing courses, applying for jobs, or updating their profiles. This gamified approach can motivate users to stay active on the platform, continually improving their skills and career prospects.

4. Enhanced Mobile Experience:

While LinC is currently responsive, future enhancements could focus on creating a more robust mobile experience through a dedicated mobile app. This app could offer offline capabilities, push notifications for job alerts, and a streamlined user interface tailored specifically for mobile users. Enhancing the mobile experience will make LinC more accessible and convenient for users on the go.

5. Multilingual Support:

To cater to a global audience, LinC could introduce multilingual support, allowing users to interact with the platform in their native language. This enhancement would involve translating the user interface, job listings, and support materials into multiple languages, ensuring that non-English speaking users can fully utilize the platform's features without language barriers.

6. Integration with Learning Platforms:

LinC can be further enhanced by integrating with online learning platforms such as Coursera, Udemy, or LinkedIn Learning. This would allow users to directly access relevant courses from within the platform, enabling them to acquire new skills or certifications that

align with their career goals. The integration could also include personalized course recommendations based on the user's career aspirations and skill gaps.

---

7. Enhanced Applicant Tracking System (ATS) for Employers:

To make LinC more attractive to employers, the platform could develop an enhanced ATS with advanced features such as AI-driven candidate matching, predictive hiring analytics, and customizable pipelines. These tools would help employers streamline their recruitment processes, making LinC a one-stop solution for both job seekers and recruiters.

---

8. Collaboration and Networking Features:

Adding collaboration and networking features could transform LinC into a more holistic career management platform. Users could join groups, participate in discussions, or connect with mentors and industry experts. These features would foster a sense of community, allowing users to build professional networks, share knowledge, and gain insights from peers and experts.

---

9. Blockchain-Based Credential Verification:

Incorporating blockchain technology for credential verification could add an extra layer of trust and security to the platform. This enhancement would allow users to securely verify and display their educational and professional credentials on their profiles. Employers could easily authenticate these credentials, reducing the risk of fraud and making the hiring process more efficient.

---

10. Job Market Analytics and Insights:

LinC could offer users detailed job market analytics and insights, such as industry trends, salary benchmarks, and in-demand skills. By providing this data, LinC would help users make informed decisions about their career paths, job applications, and skill development. This feature could also include real-time labor market updates, giving users a competitive edge in the job market.

By considering these future enhancements, the LinC can continue to evolve and adapt to meet the evolving needs and expectations of educational institutions and workplaces, providing them with a robust and cutting-edge solution for Job Opportunities.

CHAPTER VIII

REFERENCES

REFERENCES

[1] **S. Smith and A. Jones,** "A Study on Applicant Tracking Systems: Their Role in Modern Recruitment," *International Journal of Human Resource Management*, vol. 12, no. 3, pp. 123-134, 2021.

[2] **J. Lee, M. Kim, and T. Nguyen,** "Advancements in Online Resume Builders: A Comparative Analysis," *Journal of Digital Career Development*, vol. 8, no. 2, pp. 45-59, 2022.

[3] **R. Patel and L. Brown,** "The Impact of AI in Job Application Processes: Enhancing Efficiency and User Experience," *Proceedings of the IEEE Conference on Artificial Intelligence Applications*, vol. 2023, pp. 78-85, 2023.

[4] **C. Martin and F. Garcia,** "Job Tracking Systems: Improving Candidate Experience and Recruitment Outcomes," *International Conference on Job Market Innovations*, vol. 2020, pp. 212-220, 2020.

[5] **K. Wright, D. Adams, and H. Thomas,** "Design and Implementation of a Comprehensive Job Portal: A Case Study," *Journal of Web Development and Design*, vol. 15, no. 4, pp. 340-355, 2019.

[6] **A. Roberts and M. Clark,** "Leveraging Machine Learning for Resume Screening: Innovations and Challenges," *Journal of Applied AI Research*, vol. 14, no. 1, pp. 55-68, 2022.

[7] **N. Gupta and S. Patel,** "User-Centric Design in Job Portals: Enhancing the Job Search Experience," *International Conference on Human-Computer Interaction*, vol. 2021, pp. 102-110, 2021.

[8] **L. Johnson and R. Lee,** "The Evolution of Online Job Portals: From Simple Listings to Comprehensive Career Management Systems," *Journal of Internet Technology and Services*, vol. 10, no. 3, pp. 91-103, 2020.

[9] **M. Singh and A. Kumar,** "AI-Powered Chatbots in Recruitment: A Study on Effectiveness and User Experience," *Proceedings of the International Symposium on AI in HR Tech*, vol. 2023, pp. 143-150, 2023.

[10] **P. Zhao and Y. Wang,** "Integration of AI in Job Application Systems: Enhancing Efficiency and Accuracy," *International Journal of Computer Applications*, vol. 28, no. 2, pp. 78-85, 2019.