Industry Project / IBM Project

Report

On

Artificial Intelligence Based Solutions for Corporate Employees

Developed By: -

Mayan Suthar (21162172005)

Mahendra Dod (2116210202)

Nawaz Bakali (20162121003)

Guided By:-

Prof. Neha Rajput (Internal)

Mr. Nirav Rajgor (External)

Submitted to

Faculty of Engineering and Technology

Institute of Computer Technology

Ganpat University

Year - 2024

CERTIFICATE

This is to certify that the **IBM** Project work entitled **"Artificial Intelligence Based Solutions for Corporate Employees"** by Mayan Suthar(Enrolment No.21162172005), Mahendra Dod(Enrolment No.21162102002) and Nawaz Bakali(EnrolmentNo.20162121003) of Ganpat University, towards the partial fulfillment of requirements of the degree of Bachelor of Technology – Computer Science and Engineering, carried out by them in the CSE(CBA/BDA/CS) Department. The results/findings contained in this Project have not been submitted in part or full to any other University / Institute for award of any other Degree/Diploma.

Name & Signature of Internal Guide

Name & Signature of Head

Place: ICT - GUNI

Date:

ACKNOWLEDGEMENT

IBM project is a golden opportunity for learning and self-development. I consider myself very lucky and honored to have so many wonderful people lead me through in completion of this project. First and foremost, I would like to thank Dr. Rohit Patel, Principal, ICT, and Prof. Dharmesh Darji, Head, ICT who gave us an opportunity to undertake this project. My grateful thanks to Prof. Neha Rajput & Prof. Nirav Rajgor for their guidance in project work Artificial Intelligence Based Solutions for Corporate Employees, who despite being extraordinarily busy with academics, took time out to hear, guide and keep us on the correct path. We do not know where would have been without his/her help. CSE department monitored our progress and arranged all facilities to make life easier. We choose this moment to acknowledge their contribution gratefully.

ABSTRACT

The AI Resume Analyzer for corporate employee is an innovative tool designed to streamline corporate employee hiring processes by automating resume screening. In today's competitive job market, HR departments are inundated with a large volume of resumes for every job opening. Benefits include time and cost savings, improved candidate quality, and enhanced diversity and inclusion. Overall, it revolutionizes recruitment, offering a powerful solution for efficient and effective hiring.

INDEX

CHAPTER: 1 INTRODUCTION

CHAPTER 1 INTRODUCTION

The AI project is to streamline the hiring process for corporate employees by leveraging artificial intelligence (AI) and natural language processing (NLP) technologies. This project aims to automate and enhance the resume screening process, enabling recruiters to efficiently identify qualified candidates from a large pool of applicants.

A tool which parses information from a resume using natural language processing and finds the keywords, cluster them onto sectors based on their keywords. And lastly show recommendations, predictions, and analytics to the applicant / recruiter based on keyword matching

Manual process of resume screening and Recruitment is time consuming and less accuracy. It Reduce manual effort and time spent on resume screening. Improve the accuracy and consistency of resume evaluation.

- Programing Language(Python)
- HTML/CSS, Scripting language for front end
- Ai Frameworks(Spacy,PyResparser,pandas)
- Database(MySql)

CHAPTER: 2 PROJECT SCOPE

CHAPTER 2 PROJECT SCOPE

Key Focus Areas:

- Automated Resume Parsing: Develop algorithms to extract relevant information from resumes such as education, work experience, skills, and achievements using natural language processing (NLP) techniques.

- Skill and Experience Matching: Implement algorithms to match the skills and experience listed in resumes with the requirements of the job roles

- Semantic Analysis: Utilize NLP techniques to understand the contextual meaning of phrases and sentences within resumes, allowing for deeper comprehension of candidates' qualifications and potential fit for different positions.

- Applicant Ranking and Scoring: Develop algorithms to rank and score candidates based on their compatibility with job roles, enabling recruiters to prioritize candidates efficiently.

- Bias Mitigation: Implement measures to mitigate unconscious biases in the resume screening process, promoting fair and equitable candidate evaluation.

- User-Friendly Interface: Design an intuitive and user-friendly interface for recruiters and hiring managers to interact with the resume analysis system, allowing for easy navigation, customization, and decision-making.

CHAPTER: 3 SOFTWARE AND HARDWARE REQUIREMENTS

CHAPTER 3 SOFTWARE AND HARDWARE REQUIREMENTS

Minimum Hardware Requirements

Processor	2.0 GHz
RAM	4GB
HDD	40GB

Table 3.1 Minimum Hardware Requirements

Minimum Software Requirements

Operating System	Any operating system which can support an Python interpreter.
Programming language	Python
Other tools & tech	MySql server, pyresparser.

Table 3.2 Minimum Software Requirements

CHAPTER: 4 PROJECT PLAN

CHAPTER 4 PROJECT PLAN

4.1 List of Major Activities

1) Integration: Develop a user-friendly interface for corporate employees to upload resumes and view analysis results.

2) Data Collection: Gather a diverse set of resumes representing various job roles and industries.

3) Preprocessing: Clean and preprocess the resumes to remove irrelevant information, standardize formatting, and extract key features.

4) Feature Extraction: Utilize NLP techniques to extract relevant features such as skills, experience, education, and achievements from the resumes.

5) Model Development: Train machine learning models, such as classification algorithms or neural networks, to classify resumes based on job fit.

6) Deployment: Deploy the Resume Analyzer as a website

CHAPTER: 5 IMPLEMENTATION DETAILS

CHAPTER 5 IMPLEMENTATION DETAIL

5.1 Er Diagram of Implementation

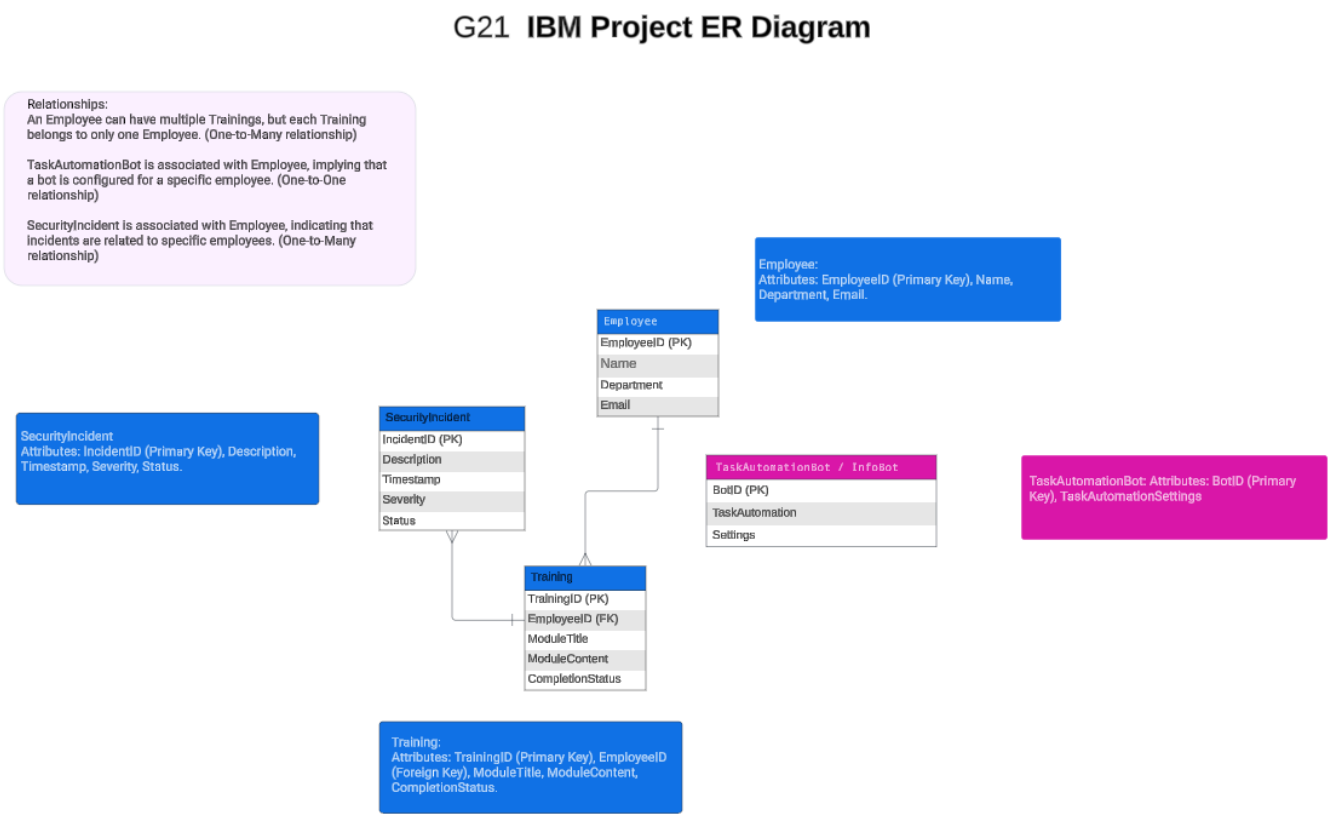


Figure 5.1 Project Implementation ER Diagram

5.2Project Site Map

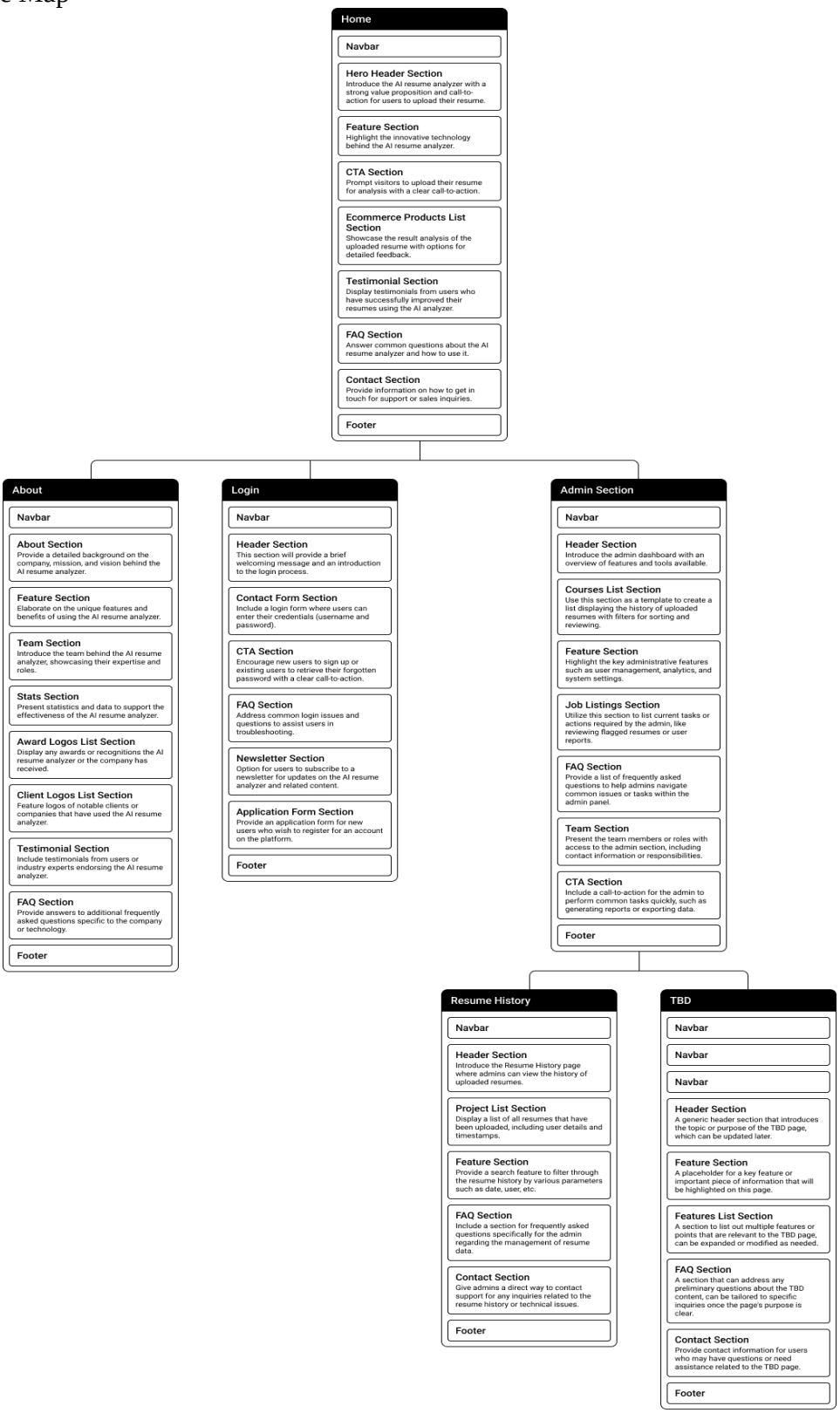


Figure 5.2 Project Sitemap

5.3 Data Dictionary

Quary Used

```
CREATE DATABASE g21_ibm_project;
USE g21_ibm_project;
CREATE TABLE Employee (
  EmployeeID INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  Name VARCHAR(50) NOT NULL,
  Department VARCHAR(50) NOT NULL,
  Email VARCHAR(100) NOT NULL UNIQUE
);
CREATE TABLE SecurityIncident (
  IncidentID INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  Description VARCHAR(255) NOT NULL,
  Timestamp DATETIME NOT NULL,
  Severity VARCHAR(20) NOT NULL,
  Status VARCHAR(20) NOT NULL,
  EmployeeID INT NOT NULL,
  FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID)
);
CREATE TABLE Training (
  TrainingID INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  EmployeeID INT NOT NULL,
  ModuleTitle VARCHAR(100) NOT NULL,
  ModuleContent TEXT NOT NULL,
  CompletionStatus VARCHAR(20) NOT NULL,
  FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID)
);
CREATE TABLE TaskAutomationBot (
  BotID INT NOT NULL AUTO_INCREMENT PRIMARY KEY,
  TaskAutomationSettings TEXT NOT NULL,
  EmployeeID INT NOT NULL,
  FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID)
);
```

Figure 5.3 Sample SQL code for creating database tables

5.4 Project Details

A Natural Language Processing tool is utilized to extract information from resumes, identifying keywords and clustering them into sectors based on their relevance. The tool then provides recommendations, predictions, and analytics to both applicants and recruiters by matching keywords.

Tech Stack:

Frontend

- Streamlit
- HTML
- CSS
- JavaScript

Backend

- Streamlit
- Python
- Database
- MySQL

Modules

- pandas
- pyresparser
- pdfminer3
- Plotly
- NLTK

Library and Tools Used in Project:

Pandas: pandas is a Python package that provides fast, flexible, and expressive data structures designed to make working with "relational" or "labeled" data both easy and intuitive. It aims to be the fundamental high-level building block for doing practical, real world data analysis in Python.

pip install pandas

Streamlit-aggrid: used aggrid library to visualize Data in Data Tables.

pip install streamlit-aggrid

Streamlit: Streamlit lets you transform Python scripts into interactive web apps in minutes, instead of weeks. Build dashboards, generate reports, or create chat apps. Once you've created an app, you can use our Community Cloud platform to deploy, manage, and share your app

pip install streamlit

Pyresparser: Pyrespasrser is a simple resume parser used for extracting information from resumes

pip install pyresparser

Plotly: Plotly is Open Source Graphing Library for Python

pip install plotly

NLTK: The Natural Language Toolkit (NLTK) is a Python package for natural language processing(NLP).

pip install nltk

Cryptography: Python Cryptography Library For Encryption Of chat. The cryptography library in Python offers robust support for symmetric encryption algorithms such as AES

pip install cryptography

Each Module Info:

Client: -

- Fetching Location and Miscellaneous Data Using Parsing Techniques to fetch
- Basic Info
- Skills
- Keywords

Admin: -

- Get all applicant's data into tabular format
- Download user's data into csv file
- View all saved uploaded pdf in Uploaded Resume folder
- Get user feedback and ratings

Pie Charts for: -

- Ratings
- Predicted field / roles
- Experience level
- Resume score
- User count
- City
- State
- Country

Feedback: -

- Form filling
- Rating from 1 – 5
- Show overall ratings pie chart
- Past user comments history

Requirements:

Have these things installed to make your process smooth

- Python https://www.python.org/downloads/
- MySQL https://www.mysql.com/downloads/
- Visual Studio Code (Prefered Code Editor) https://code.visualstudio.com/Download
- Visual Studio build tools for C++ https://aka.ms/vs/17/release/vs_BuildTools.exe

5.5 Implementation

Implement Website on Python Streamlit.

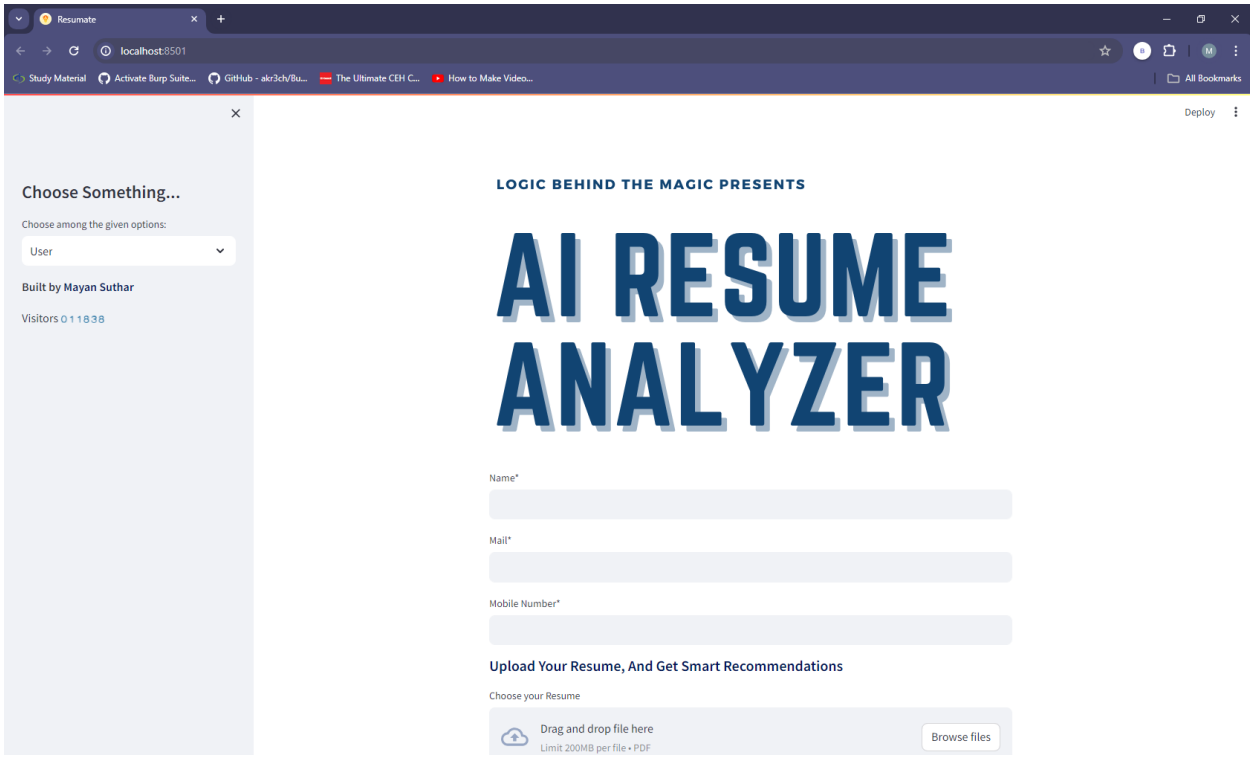


Figure 5.5.1 Home Page.

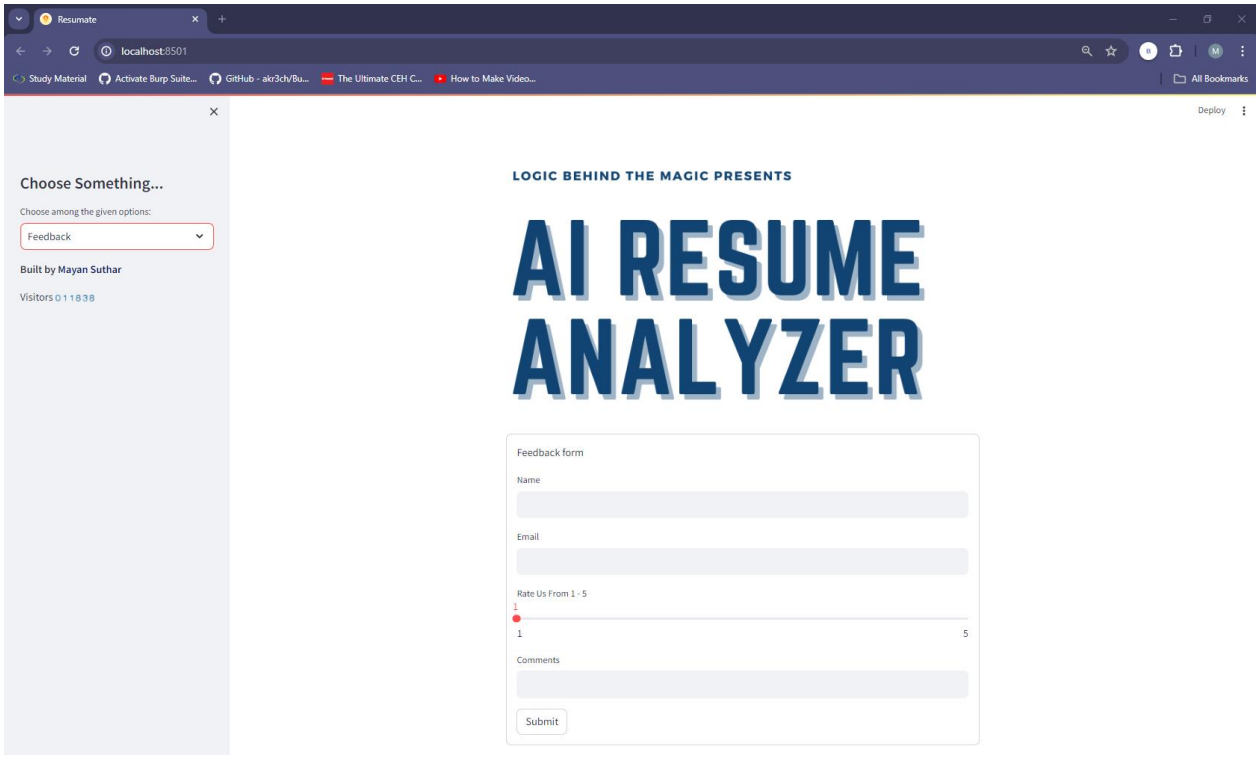


Figure 5.5.2 Feedback Page.

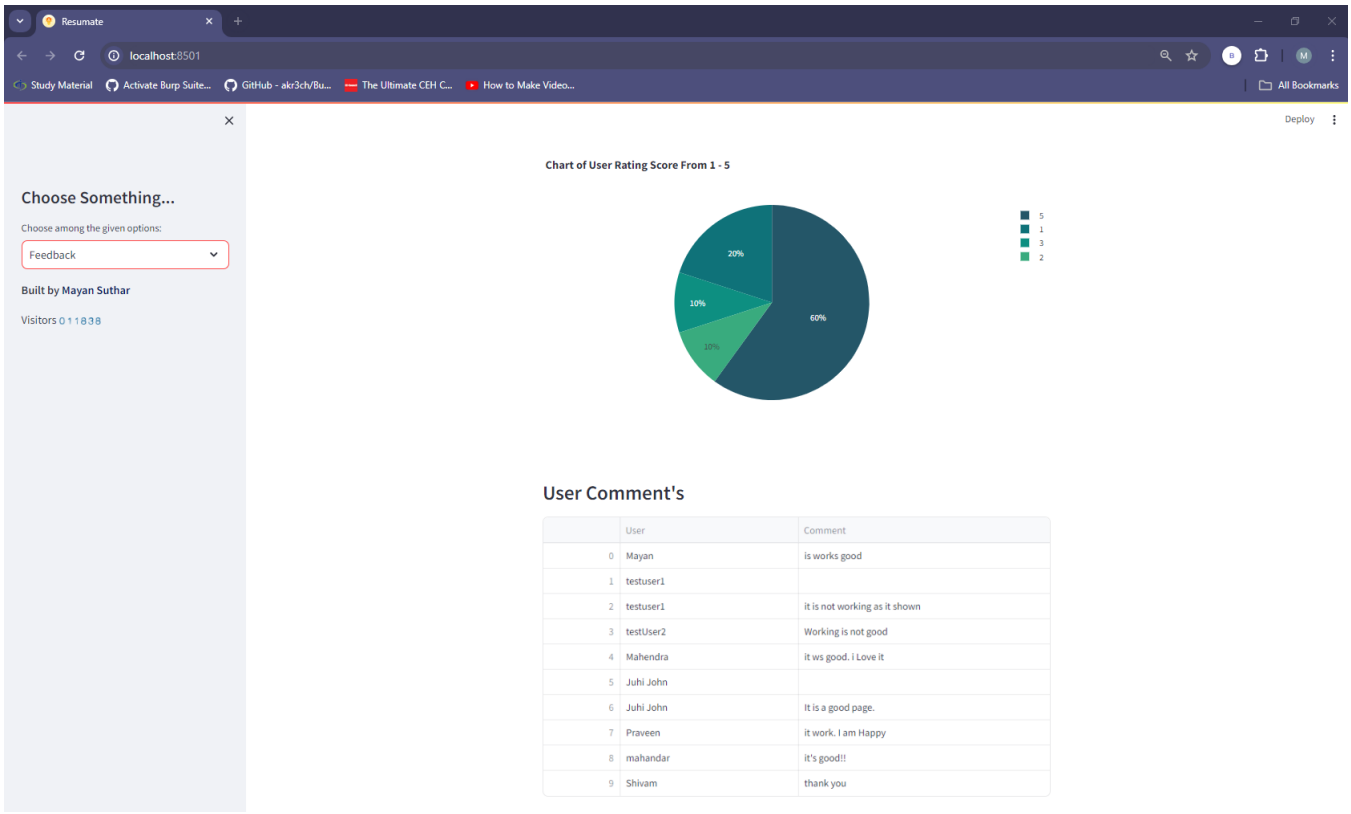


Figure 5.5.3 Feedback Page - Pie Chart Summarizes of Other Users' Comments.

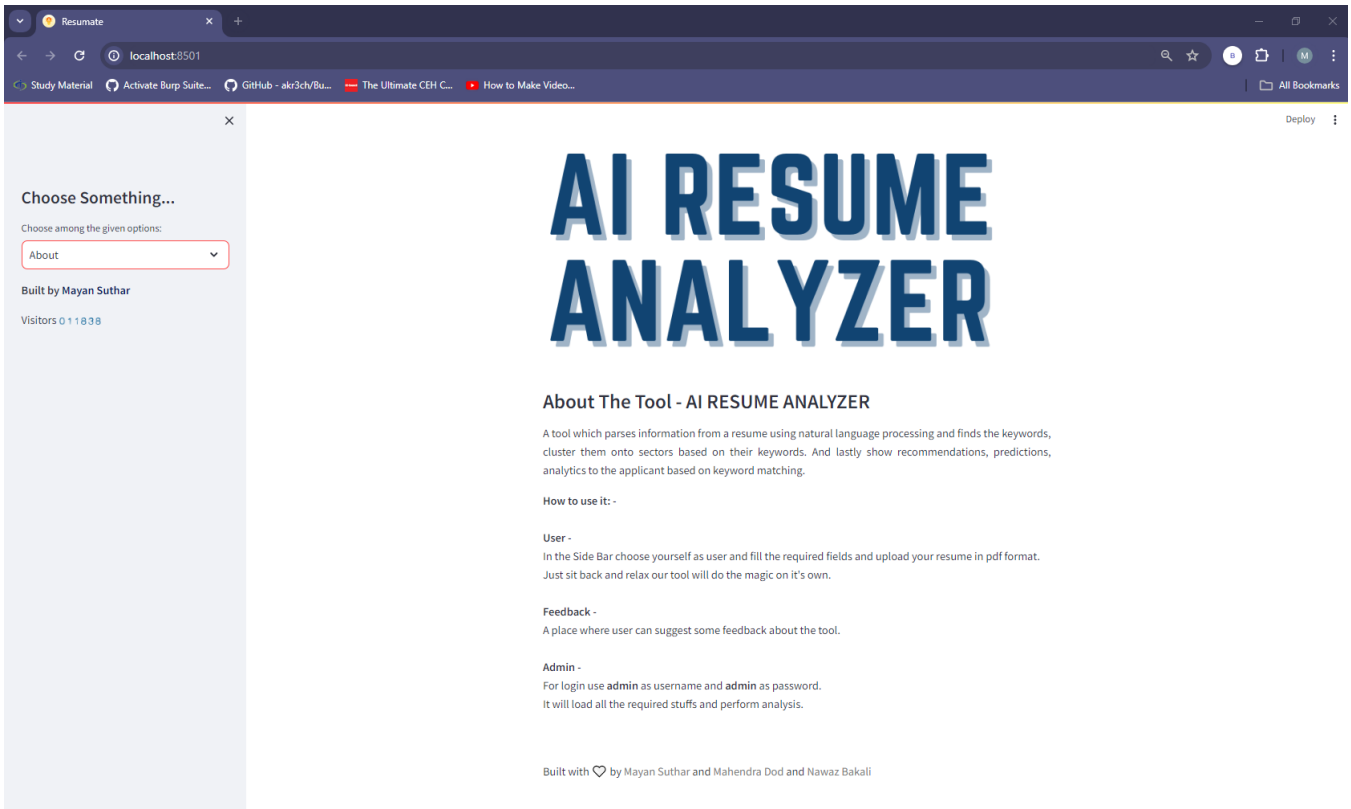


Figure 5.5.4 About Us Page.

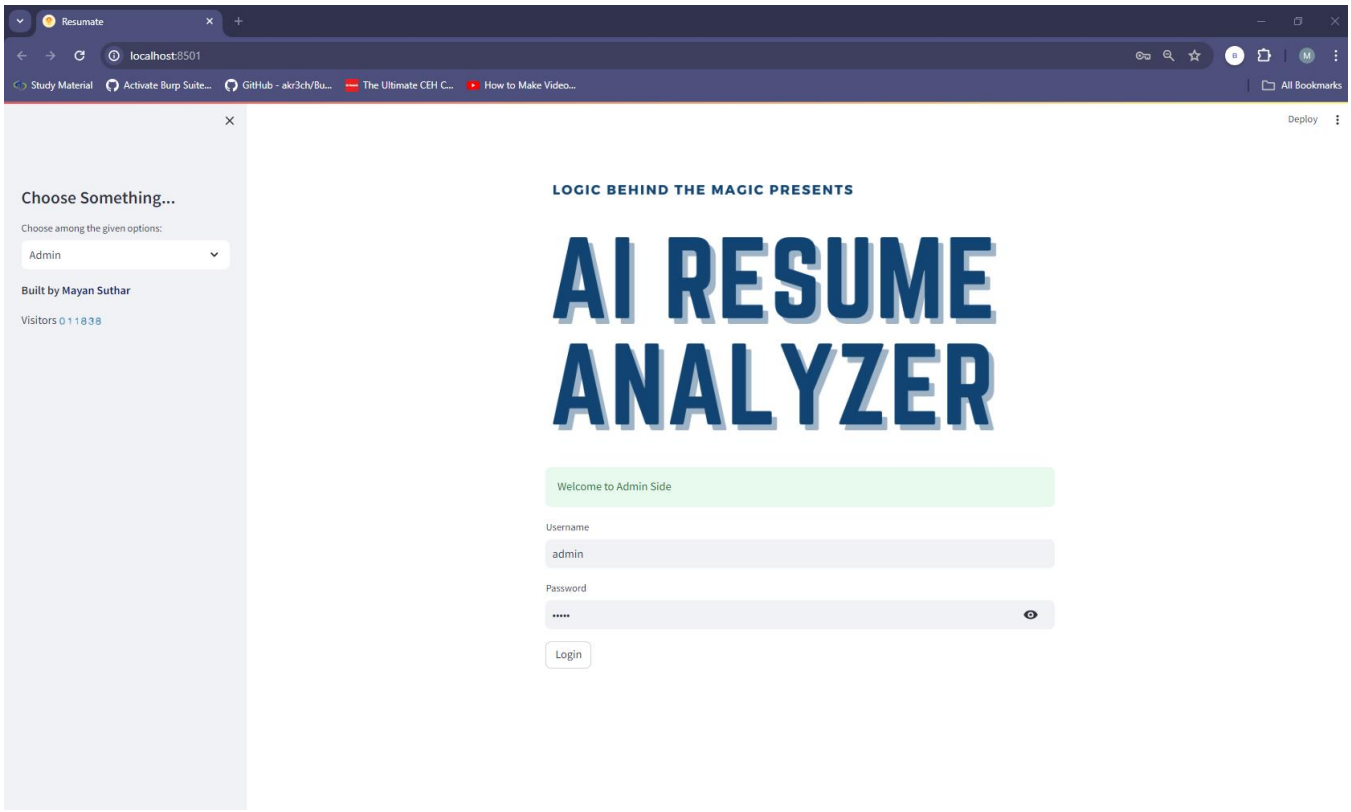


Figure 5.5.5 Admin Login.

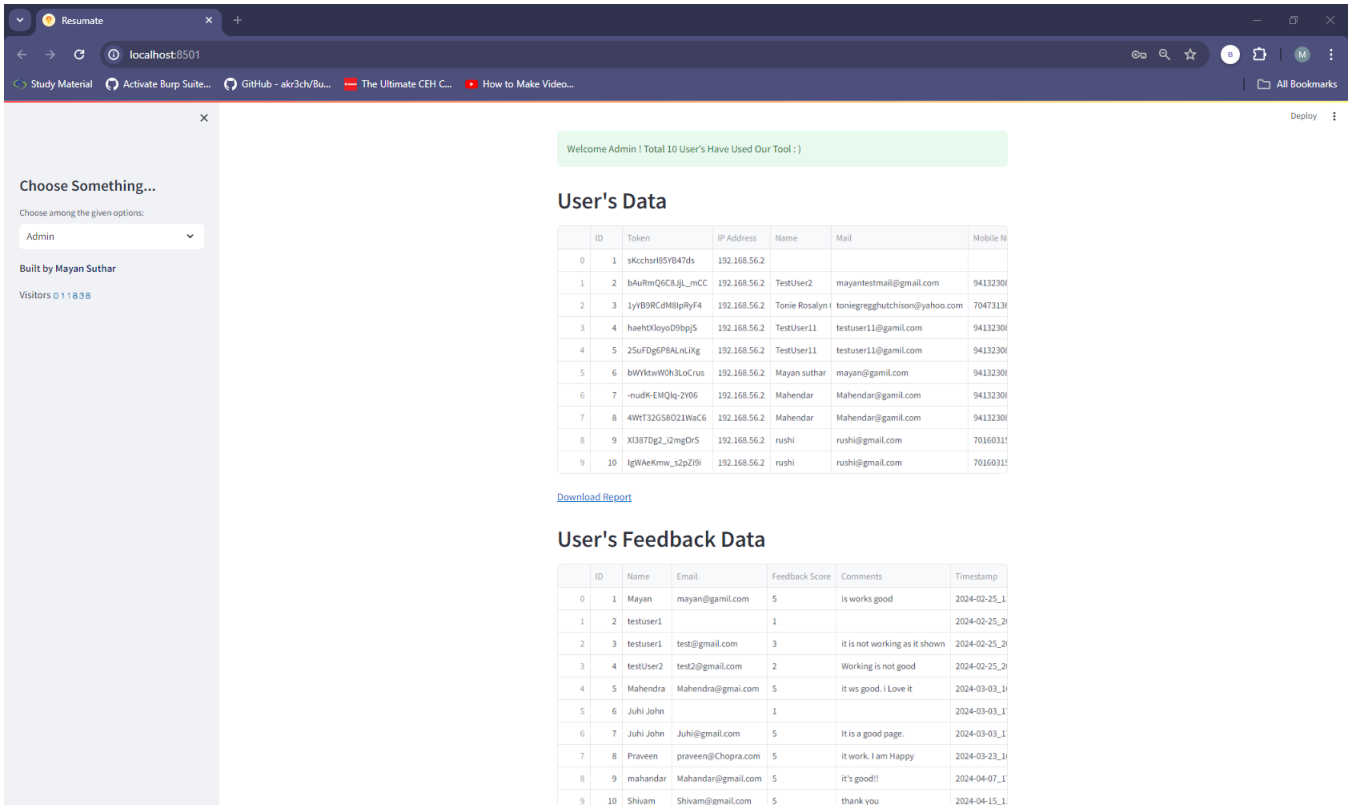


Figure 5.5.6 Admin Page - User Resume Data and Feedback Data. It also has the Download Report option on the page.

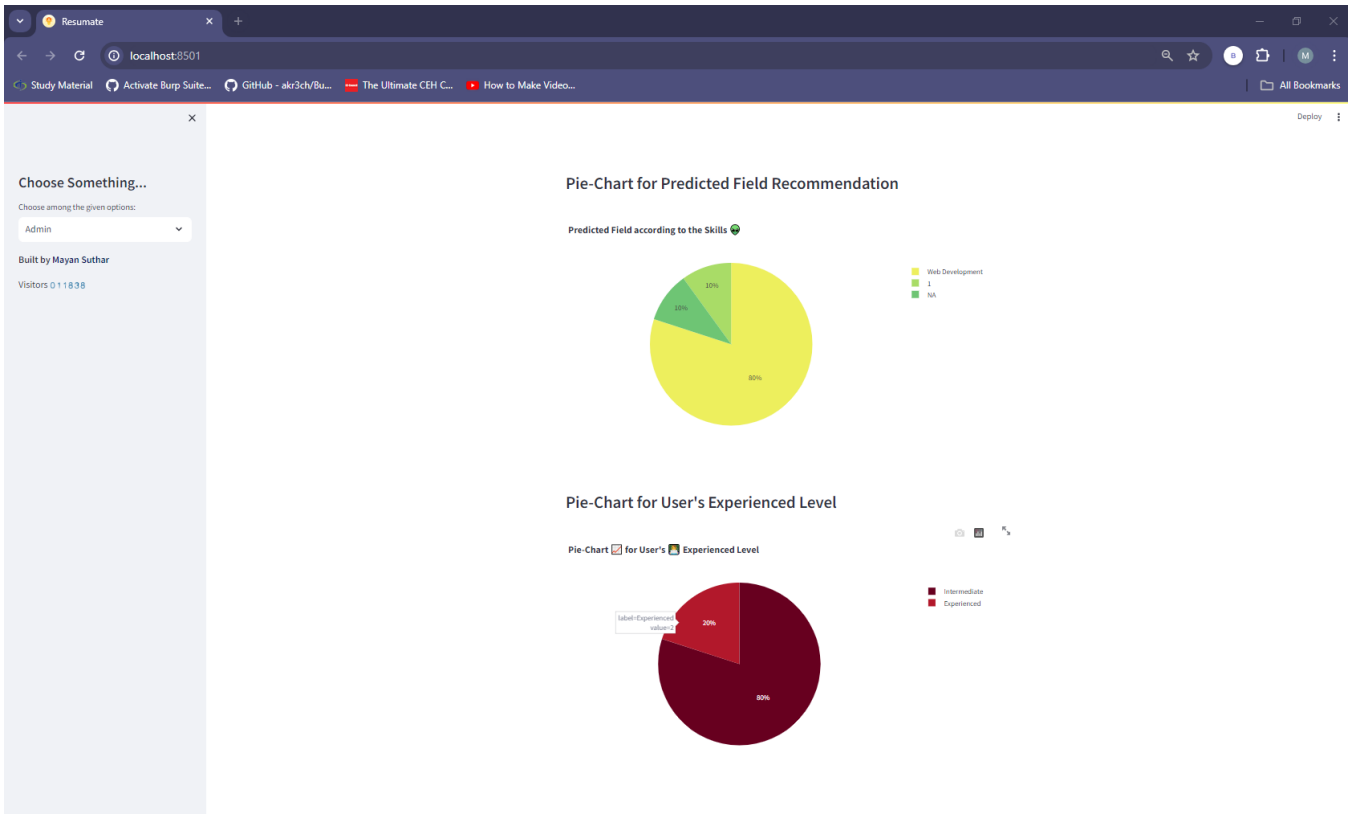


Figure 5.5.7 Admin Page - Pie-Chart for User's Experienced Level and for Predicted Field Recommendation.

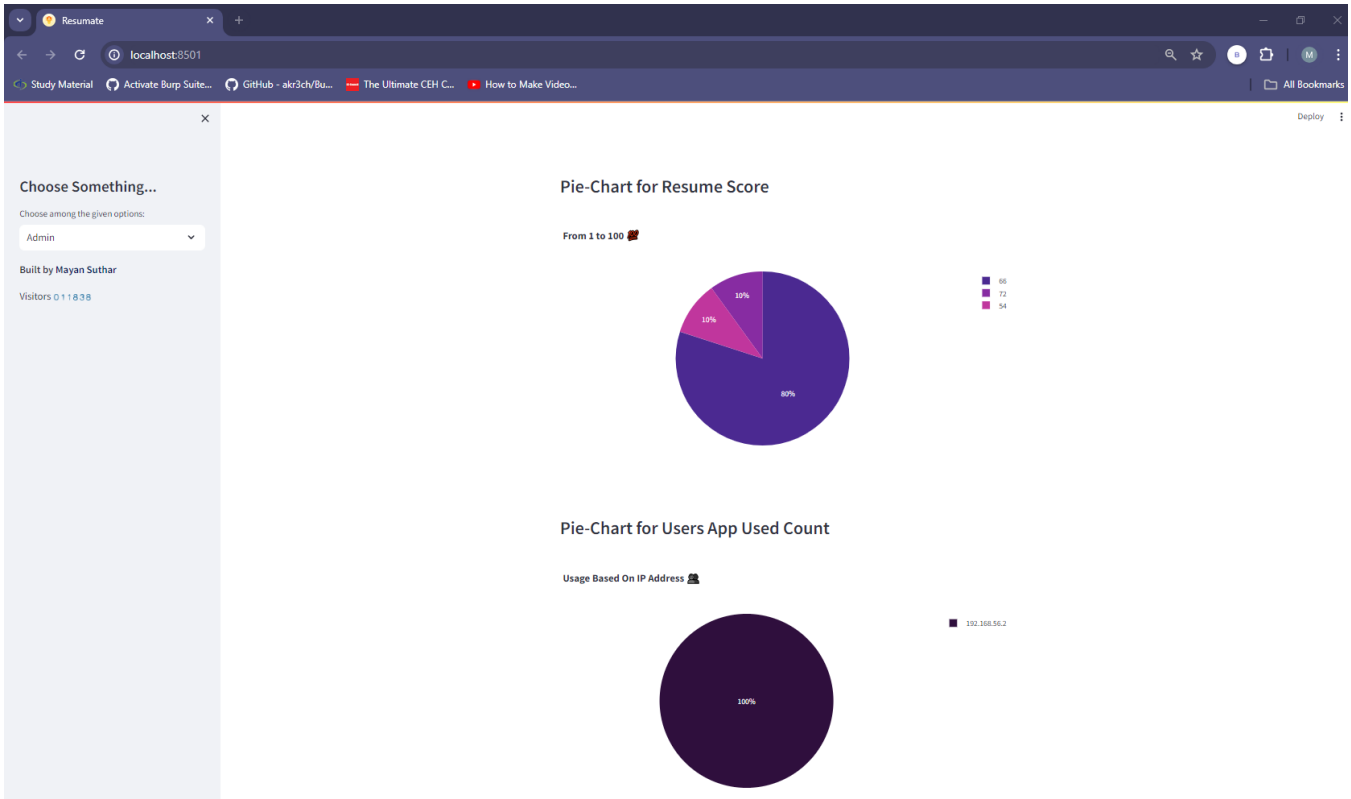


Figure 5.5.8 Admin Page - Pie-Chart for Users App Used Count and for Users App Used Count.

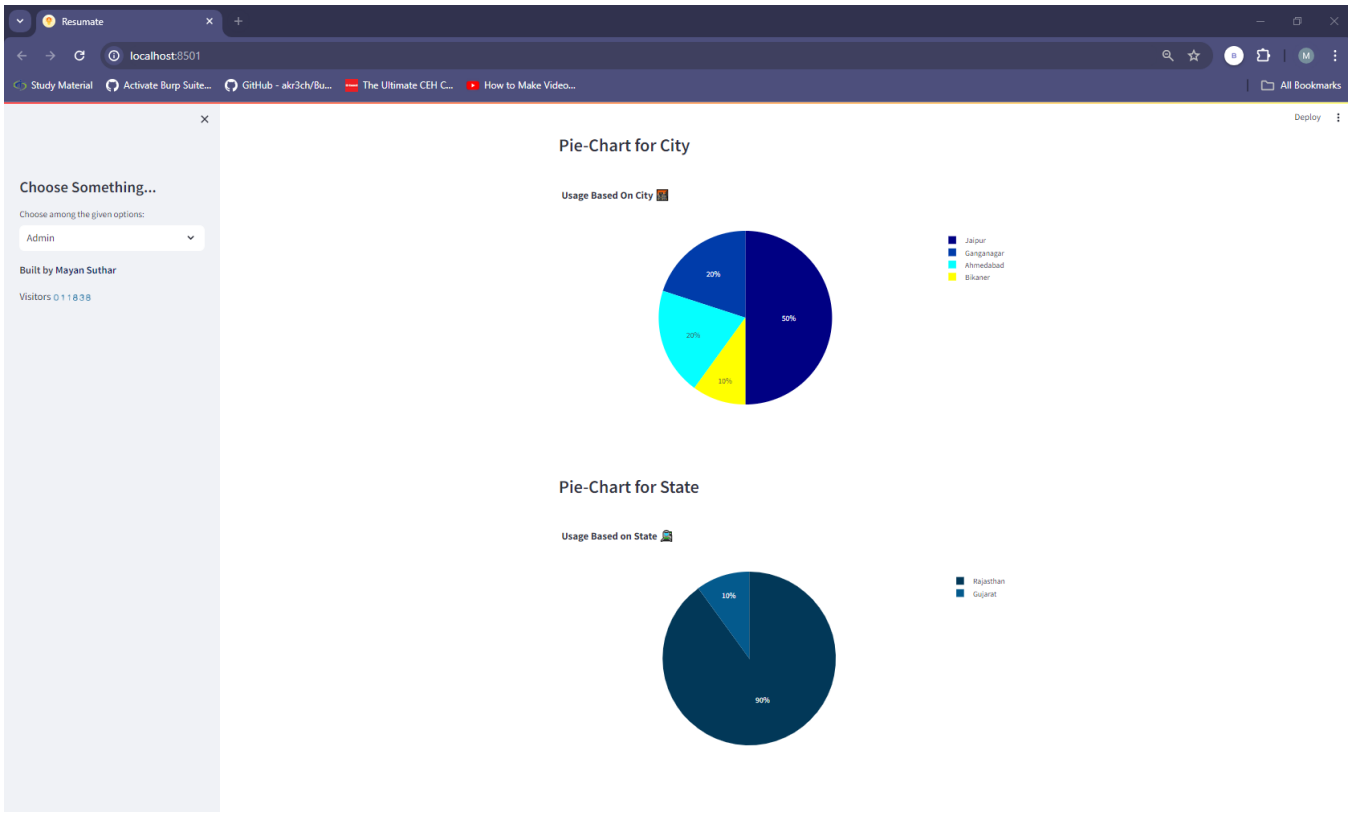


Figure 5.5.8 Admin Page - Pie-Chart for State and City.

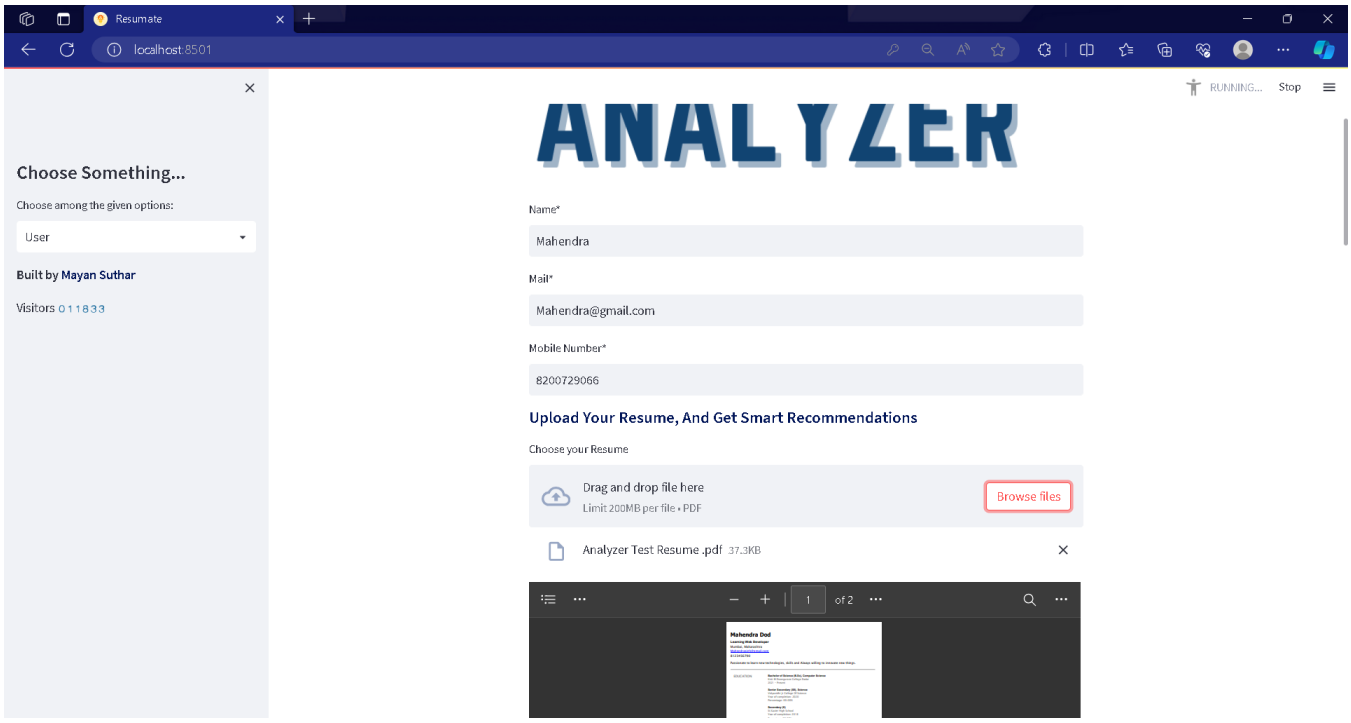


Figure 5.5.9 Upload Resume and Enter Data.

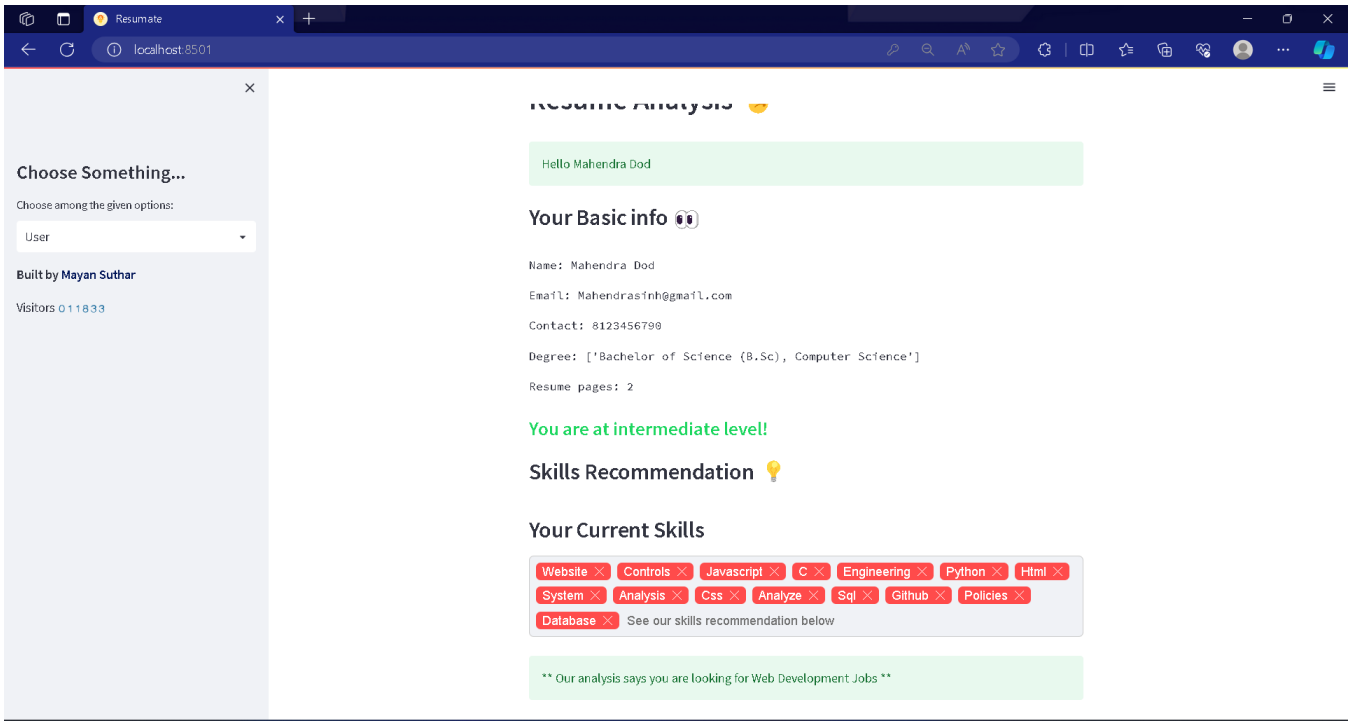


Figure 5.5.10 Parser Resume Result.

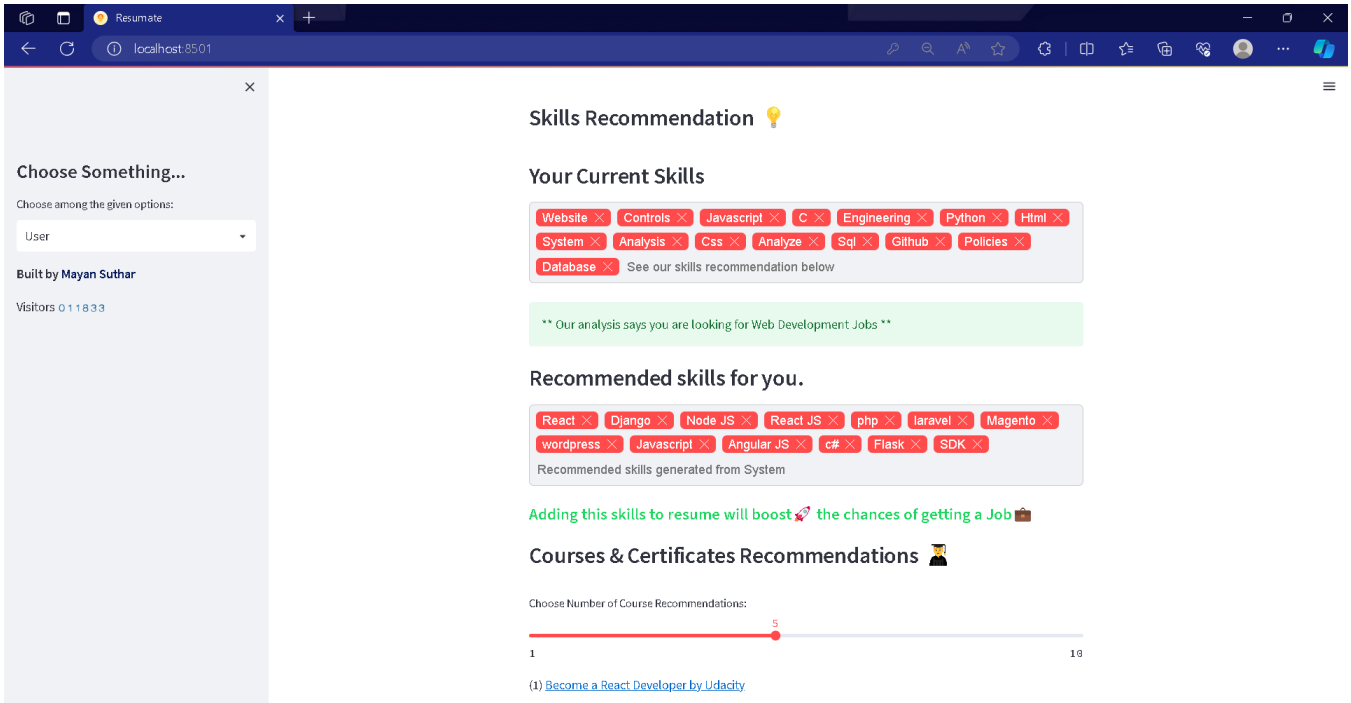


Figure 5.5.11 Parser Resume Result.

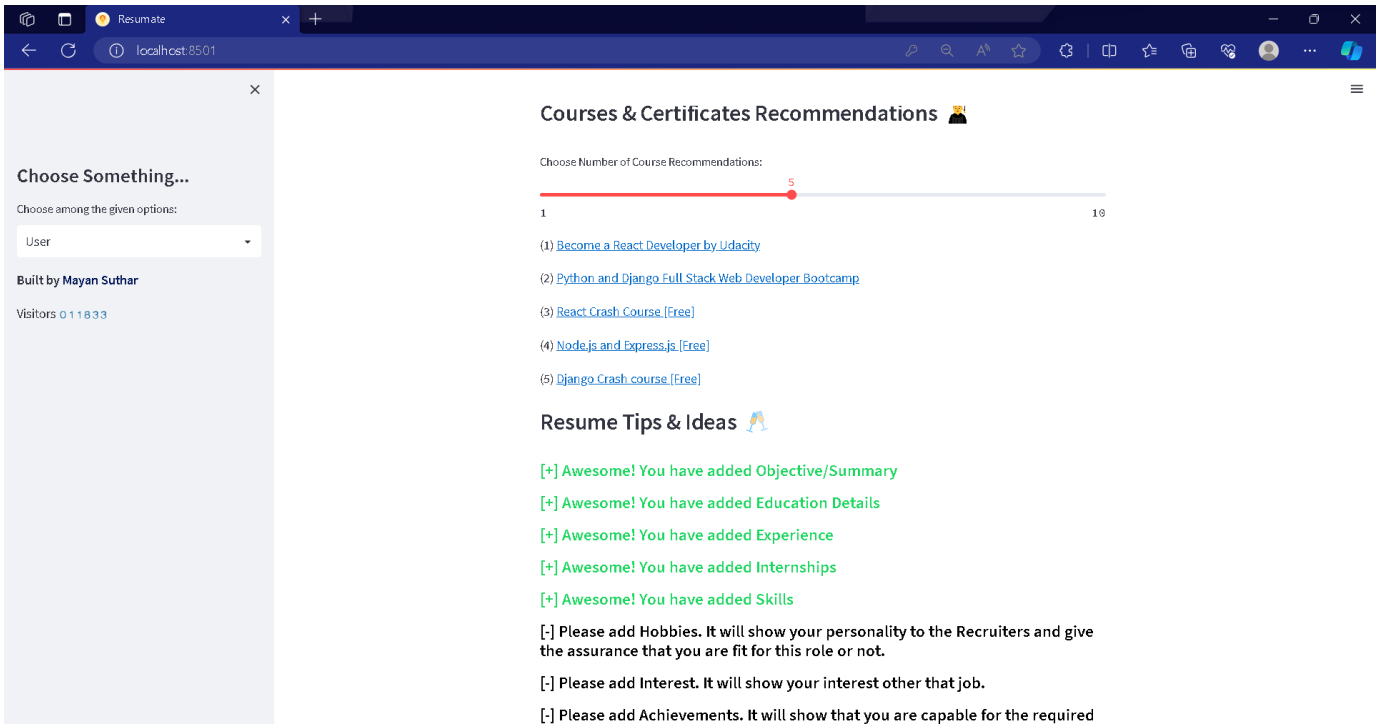


Figure 5.5.12 Parser Resume Result.

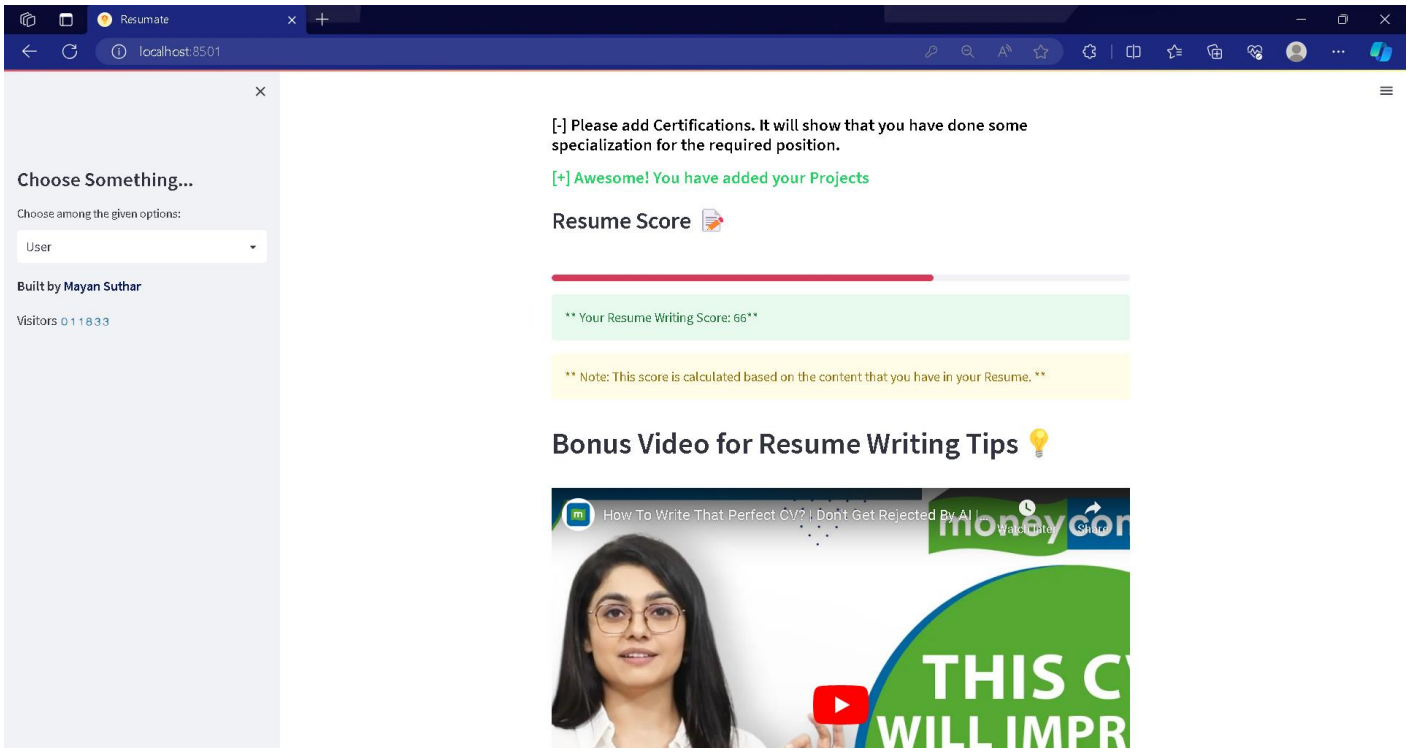


Figure 5.5.13 Parser Resume Score.

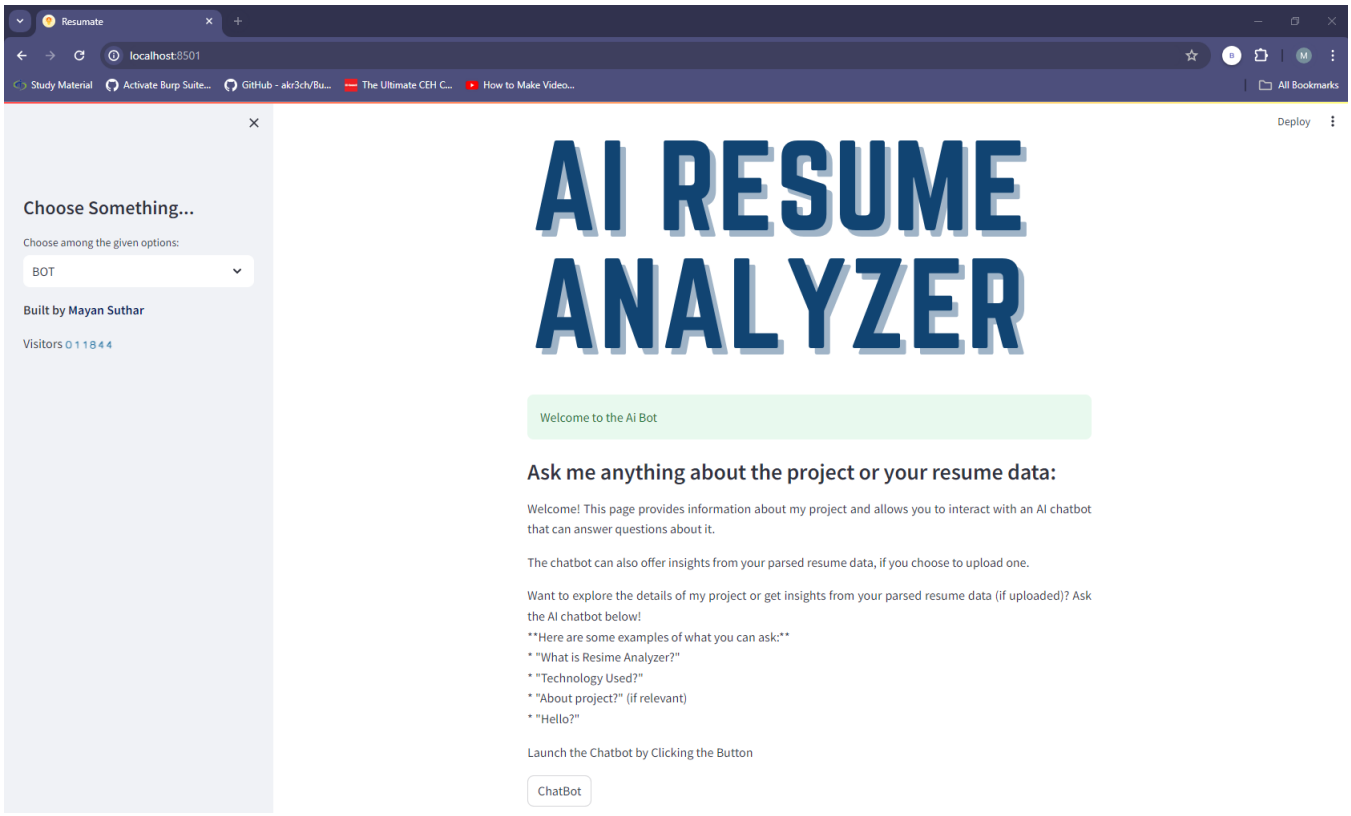


Figure 5.5.14 Chatbot Page.

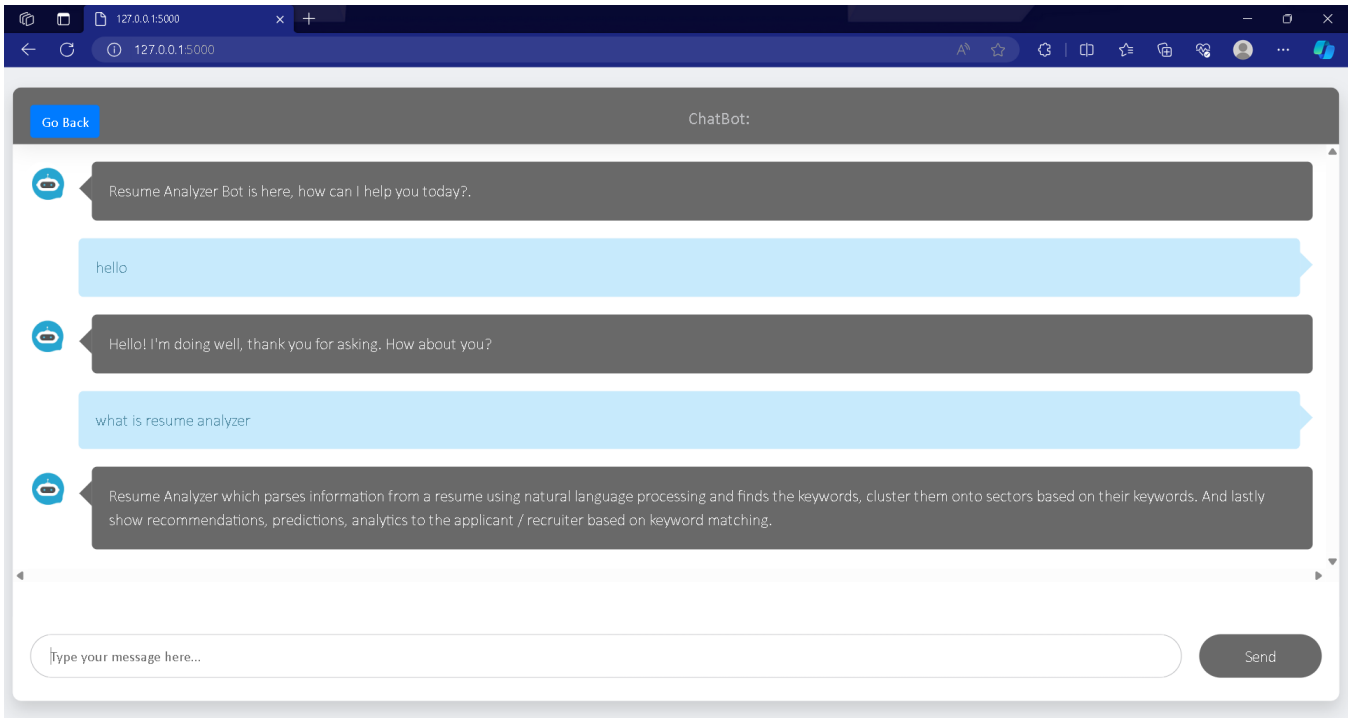
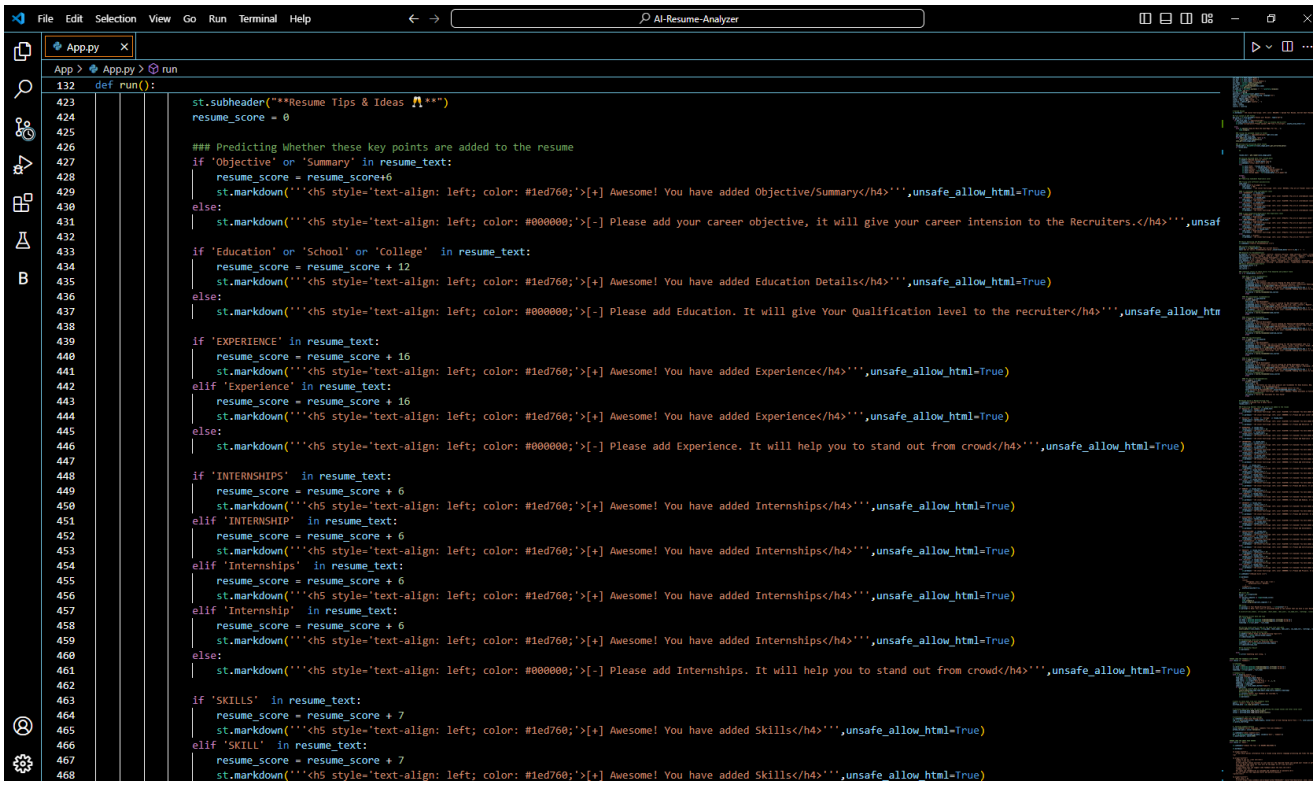


Figure 5.5.15 Chabot Page.



```
def run():
    st.subheader("**Resume Tips & Ideas **")
    resume_score = 0

    ### Predicting Whether these key points are added to the resume
    if 'Objective' or 'Summary' in resume_text:
        resume_score = resume_score+6
        st.markdown('''<h5 style='text-align: left; color: #1ed760;'>[+] Awesome! You have added Objective/Summary</h4>''',unsafe_allow_html=True)
    else:
        st.markdown('''<h5 style='text-align: left; color: #000000;'>[-] Please add your career objective, it will give your career intension to the Recruiters.</h4>''',unsaf

    if 'Education' or 'School' or 'College'  in resume_text:
        resume_score = resume_score + 12
        st.markdown('''<h5 style='text-align: left; color: #1ed760;'>[+] Awesome! You have added Education Details</h4>''',unsafe_allow_html=True)
    else:
        st.markdown('''<h5 style='text-align: left; color: #000000;'>[-] Please add Education. It will give Your Qualification level to the recruiter</h4>''',unsafe_allow_htm

    if 'EXPERIENCE' in resume_text:
        resume_score = resume_score + 16
        st.markdown('''<h5 style='text-align: left; color: #1ed760;'>[+] Awesome! You have added Experience</h4>''',unsafe_allow_html=True)
    elif 'Experience' in resume_text:
        resume_score = resume_score + 16
        st.markdown('''<h5 style='text-align: left; color: #1ed760;'>[+] Awesome! You have added Experience</h4>''',unsafe_allow_html=True)
    else:
        st.markdown('''<h5 style='text-align: left; color: #000000;'>[-] Please add Experience. It will help you to stand out from crowd</h4>''',unsafe_allow_html=True)

    if 'INTERNSHIPS'  in resume_text:
        resume_score = resume_score + 6
        st.markdown('''<h5 style='text-align: left; color: #1ed760;'>[+] Awesome! You have added Internships</h4>''',unsafe_allow_html=True)
    elif 'INTERNSHIP'  in resume_text:
        resume_score = resume_score + 6
        st.markdown('''<h5 style='text-align: left; color: #1ed760;'>[+] Awesome! You have added Internships</h4>''',unsafe_allow_html=True)
    elif 'Internships'  in resume_text:
        resume_score = resume_score + 6
        st.markdown('''<h5 style='text-align: left; color: #1ed760;'>[+] Awesome! You have added Internships</h4>''',unsafe_allow_html=True)
    elif 'Internship'  in resume_text:
        resume_score = resume_score + 6
        st.markdown('''<h5 style='text-align: left; color: #1ed760;'>[+] Awesome! You have added Internships</h4>''',unsafe_allow_html=True)
    else:
        st.markdown('''<h5 style='text-align: left; color: #000000;'>[-] Please add Internships. It will help you to stand out from crowd</h4>''',unsafe_allow_html=True)

    if 'SKILLS'  in resume_text:
        resume_score = resume_score + 7
        st.markdown('''<h5 style='text-align: left; color: #1ed760;'>[+] Awesome! You have added Skills</h4>''',unsafe_allow_html=True)
    elif 'SKILL'  in resume_text:
        resume_score = resume_score + 7
        st.markdown('''<h5 style='text-align: left; color: #1ed760;'>[+] Awesome! You have added Skills</h4>''',unsafe_allow_html=True)
```

Figure 5.5.16 Code Snippet.

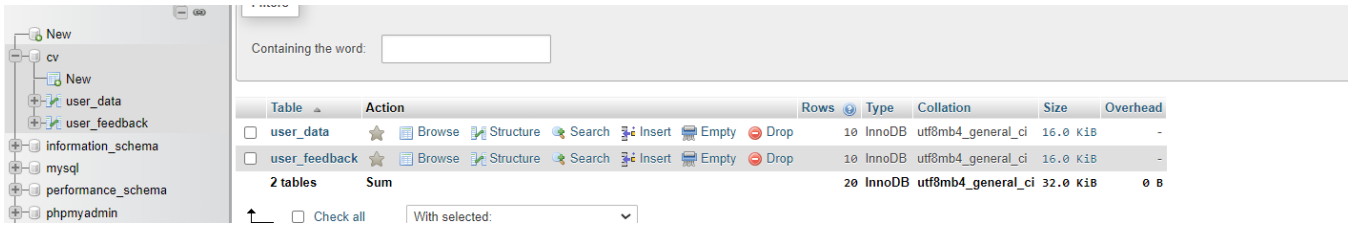


Figure 5.5.17 Database.

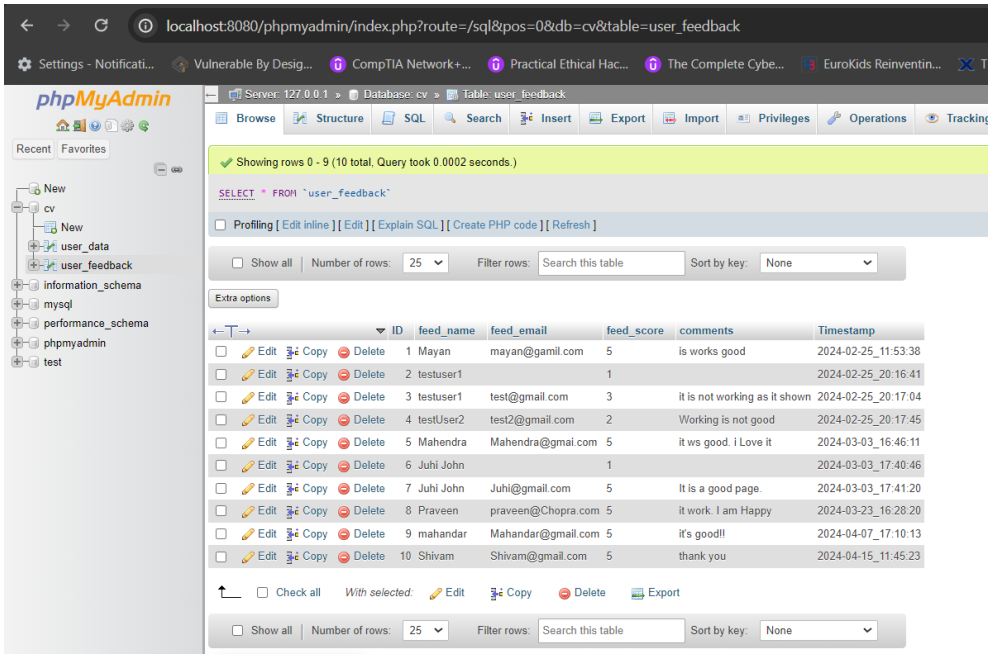


Figure 5.5.18 DataBase – user_data table (ID, feed_name, feed_email, feed_score, comments).

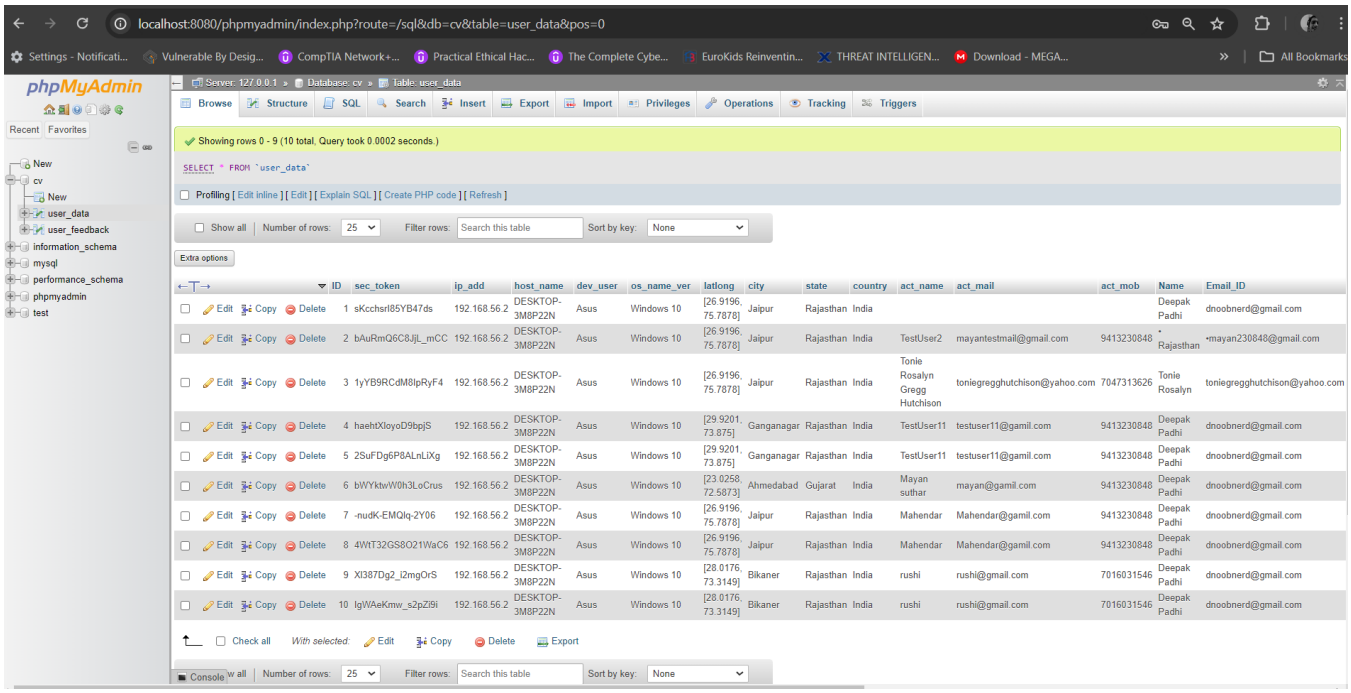


Figure 5.5.18 DataBase – user_feedback table (ID,sec_token,ip_add, host_name, dev_user, os_name_ver, latlong, city, state ,country, act_name, act_mail, act_mob, Name, Email_ID, resume_score, Timestamp, Page_no, Predicted_Field, User_level, Actual_skills, Recommended_skills, Recommended_courses, pdf_name).

CHAPTER: 6 CONCLUSION AND FUTURE WORK

CHAPTER 6 CONCLUSION

Conclusion

The AI resume checker helps corporate workers by making it easier to go through resumes when hiring. It saves time, cuts down on unfair treatment, and picks better candidates. With AI, companies can hire people more efficiently and choose the right person for the job with more confidence. Till now we have made algorithm in python for resume parsing. And prepare ChatBot

CHAPTER: 7 REFERENCES

CHAPTER 7 REFERENCES

1) https://www.geeksforgeeks.org/project-how-to-build-a-resume-parser-using-python/
2) https://pypi.org/project/pyresparser/
3) https://spacy.io/usage/spacy-101
4) https://pypi.org/project/cryptography/
5) NLTK :: Natural Language Toolkit
6) Site Map: https://www.figma.com/file/xQFjFq7g2f2Yg1IO6Pkt8y/IBM---AI-Resume-Analyzer?type=design&node-id=0-1&mode=design&t=IpOPSLQAz7FOdBjG-0
7) ER-Diagram:https://lucid.app/lucidchart/773b9ec9-7324-45eb-a1d6-070067ec19d4/edit?invitationId=inv_9ba6c9d4-8fe7-4aed-94fc-fbccfafc4ab3&page=0_0#