Developing Web Application Using Natural Processing Language with Flask for Employers: Smart Resume Analyzer

I. INTRODUCTION

The project is a web application system that utilizes natural language processing (NLP) to analyze resumes for a hiring company. The main aim of our project is to provide an efficient and objective solution for employers to evaluate candidate resumes based on job requirements, streamlining the hiring process and improving the candidate experience. For this reason, we developed a web application called- '**Smart Resume Analyzer'.** This system allows employers to upload their job requirements and candidates to upload their resumes. This analyzer extracts all information such as contact information, skills, work experience, and academic qualifications from the candidate's resume and compares it to the job requirements, providing a matching percentage and ranking the resume accordingly. In today's fast-paced business environment, hiring the right candidate for a job can be a challenging and time-consuming task for employers. Analyzing and evaluating numerous resumes to find the perfect match can take up a significant amount of time and resources.

II. BENEFITS OF OUR PROJECT

The **'Smart Resume Analyzer'** system developed for the employee hiring company offers several benefits to both employers and job seekers.

- **Time Saving:** Our smart resume analyzer system can quickly analyze resumes, screen candidates, and shortlist the most relevant applicants. This reduces the time and effort required for manual resume screening.
- **Improve Accuracy:** This system can accurately analyze resumes and identify relevant skills and qualifications. Using a smart resume analyzer system can help eliminate human errors in the resume screening process, reducing the risk of overlooking qualified candidates or unintentional biases.
- **Better Matching:** Our system can match job seekers with job openings based on their skills, qualifications, and experience. This improves the chances of employers finding the right candidate.
- **Cost- Effective:** This system can help reduce recruitment costs by automating the screening process, eliminating the need for manual screening, and reducing the need for external recruiters.
- **Improved Hiring Quality: -** a smart resume analyzer system can help employers identify the most qualified candidates, resulting in better hiring decisions and improved hiring quality.

III. IMPORTANT RESOURCES

In below, we provide some important resources of our project-

- Extract text form DOCX file, we used "**docx2text"** Python package and also we used "**PDFMiner"** package to extract text from PDF file.
- For check the file type, we used "**mimetypes"** package.
- For regular expression we used "**re"** package.
- For some Text-Preprocessing we used **Natural Language Toolkit (NLTK).**
- To create web applications. Here we used, **Flask Framework**.
- Here we used **Flask-MySQLdb** to connect our Flask Applications to **MySQL** databases. **Flask-MySQLdb** can provide a streamlined and efficient way to integrate Flask web applications with **MySQL** databases.

IV. SYSTEM DESIGN AND ARCHITECTURE

‘Smart Resume Analyzer’ is an online web application. It will usually work as a personal web application for any company. Initially, the hiring manager or the recruiter responsible for filling the position will be the one who receives and reviews the candidate's resume. This person will typically be looking for relevant skills and experience, as well as any other qualifications or requirements specified in the job posting. If the candidate's resume meets the initial screening criteria, the hiring manager or recruiter may then forward the resume to other individuals involved in the hiring process.

But using **‘Smart Resume Analyzer’** that can help streamline the resume review process and identify the most qualified candidates for a position. Here, employers upload their requirement in this system also candidate submits their resume to the company using this application system. Then **‘Smart Resume Analyzer’** software scans the document, looking for specific keywords and phrases related to the job requirements and qualifications.

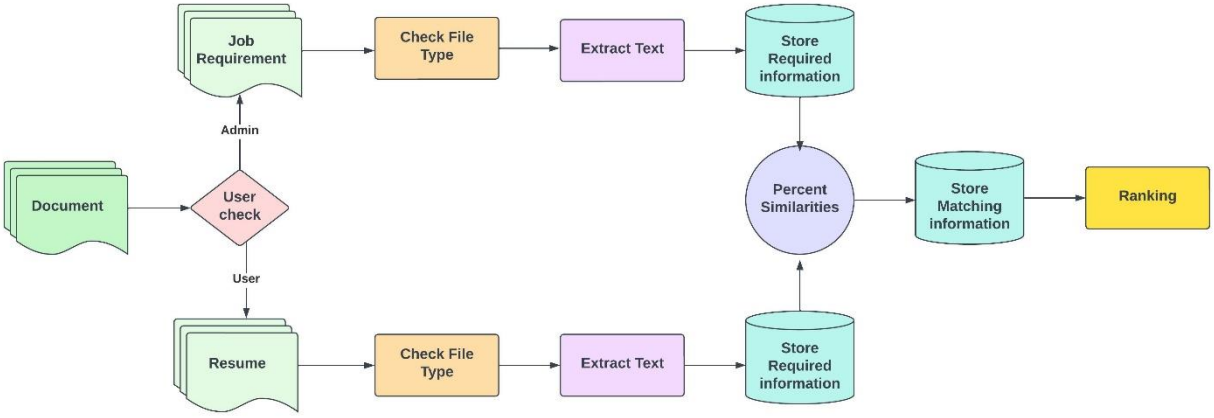

Fig. 1. Process system diagram of Back-End

Our application is a resume analyzer. This application only work for text based files that is Job Requirement and Resume. Now we explain the working process of this system-

- After upload the document at first it checks the file user. Our system check who upload this file Admin or Candidate/User. Job requirement file provide the Admin and Resume file provide the Candidate. Because extracting system of two file is difference.
- After check the user. Then it checks the file type. It checks the given file as docx or pdf. It works only for docx and pdf files.
- Then the next process is extract text from file. Job requirement text store in database table and Resume text store in another table.
    - From job requirement it extract Job Title, Company Name, Educational Qualification, Requirement Skills, Minimum Work Experience.
    - From resume it extract Candidate name, Email address, Phone number, Academic background, Total Experience.
- Then the next step is to find out the resume similarities with job requirement and store it.
- Last step is ranking the candidate based on their resume similarities.

V. IMPLEMENTATION AND RESULT

Our application has two types of features- **1)** Admin Site, **2)** User Site The purpose of having separate admin and user sites is to provide a clear and organized interface for each user type, with features and functions that are tailored to their specific needs and responsibilities within the system. This approach can help improve user experience and efficiency, as well as provide greater security and control over system access and data management.

A. Common Web-Based Features

1) **Home Activity:** This is the main landing page of our application. This is the common page from user and admin. User and Admin also can access Login and Registration page.

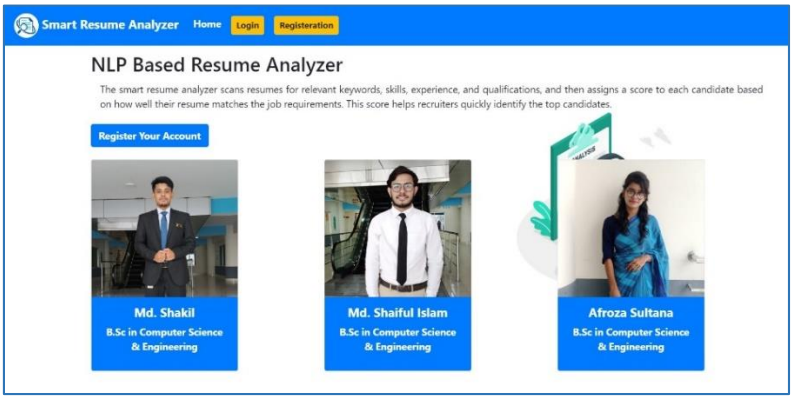

Fig. 2. Home Activity

2) **Registration Activity:** This is the registration page. At first user must create an account to apply the job. Their account will be created with their name, email and password. After create the account then goto the login page.

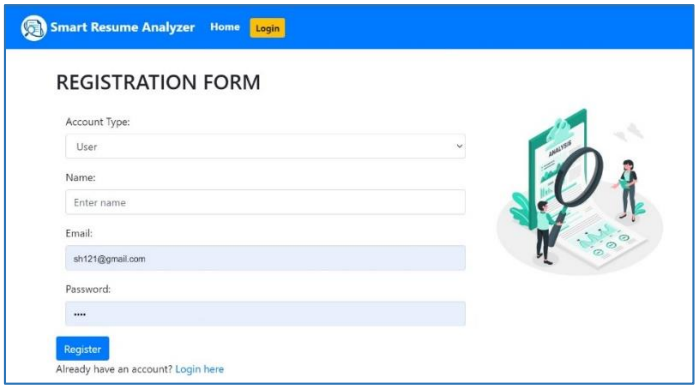

Fig. 3. Registration Activity

3) **Login Activity:** It requires users to enter their credentials to access their account. To access their account at first, select the account type (User and Admin) then type email and password. If account type is Admin then open the Admin Dashboard. If account type is User then open the Candidate/User Dashboard.

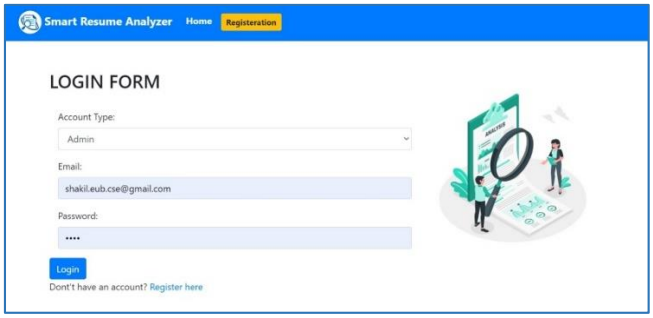

Fig. 4. Login Activity

B. Admin Site Web-Based Features

The admin site is a section of the application that is designed for administrative users who have special permissions to manage the system. This site typically includes a dashboard that provides an overview of the system's activity, as well as tools for managing user accounts, settings, and data. Features of Admin Site-

- Admin Dashboard Activity
- Ranking Activity
- Candidates Details Activity

1) **Admin Dashboard Activity:** This is the Admin Dashboard activity. From this page admin can post the job. They also show the candidate details and candidate ranking. At first, Admin post the job requirement. Uploaded job requirement in a specific format. File name must be 'Requirement' and file will be docx file. All the requirement file format available in **'Download File'** button.

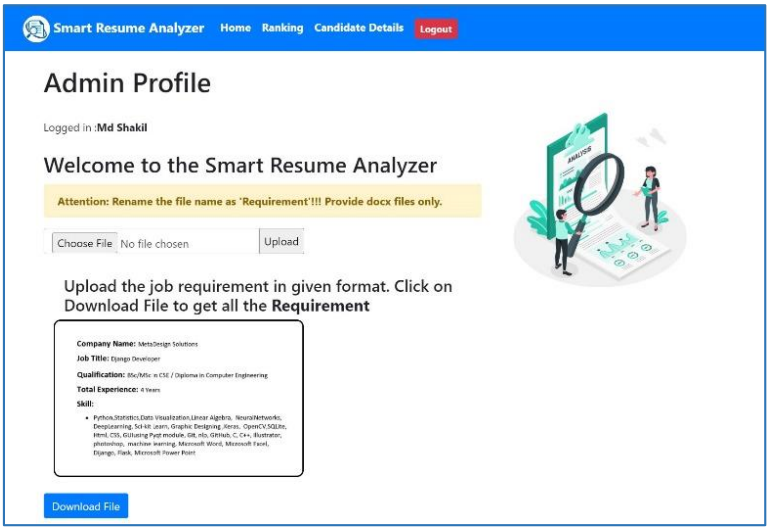

Fig. 5. Admin Dashboard Activity

2) **Ranking Activity:** Ranking activity can be a useful for selecting the best candidate for the job. When candidate upload their resume after upload the resume all necessary information store in database compare with job requirement. Ranking the candidate based on their resume similarities with job requirement.

Fig. 6. Ranking Activity

3) **Candidate Details:** If Admin wants to see the Candidate all information from **Candidate Details** activity they see the all information like- Candidate Name, Email Address, Phone Number, Academic Background, Skills, Total Experience and Matching percentage.

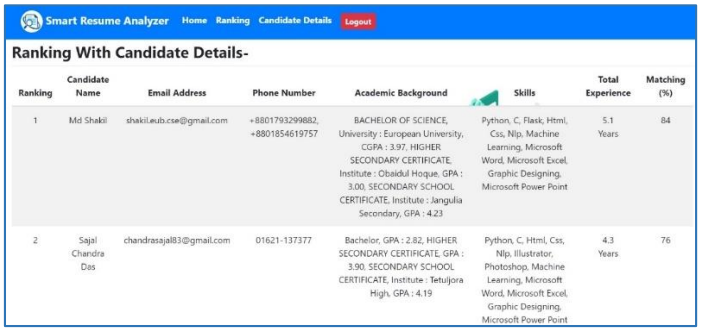

Fig. 7. Candidate Details Activity

C. User Site Web-Based Features

The user site is the another section of the application that is designed for regular users who interact with the system to perform specific tasks or access information. Features of Admin Site-

- User Dashboard Activity
- Job Post

1. **User Dashboard Activity:** This is the User Dashboard activity. From this page User can upload their resume for applying the job. They also show the Job details from '**Job Post'** activity. At first, Admin post the job requirement. Based on job requirement if User is able to apply for this job then the User can apply. User can upload their resume in DOCX or PDF format. File name must be **'User Original Name'**. Also create candidate resume in specific format. Resume format available in **'Download File'** button.

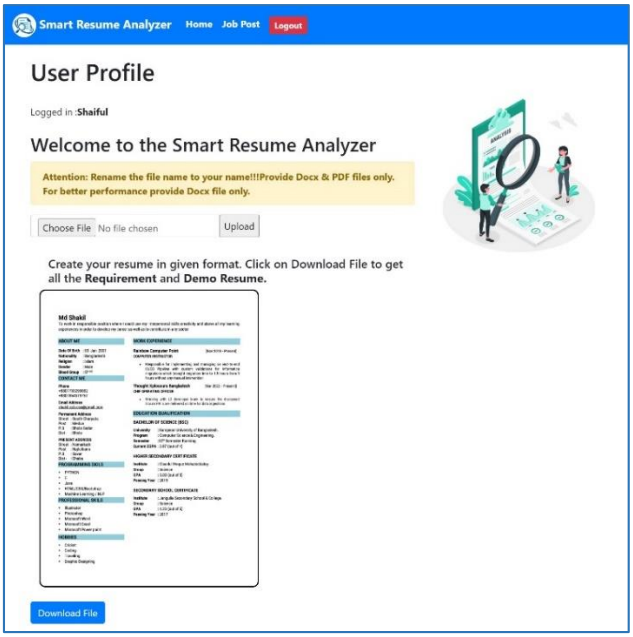

Fig. 8. User Dashboard Activity

2. **Job Post Activity:** From this activity Candidate show the job requirement.

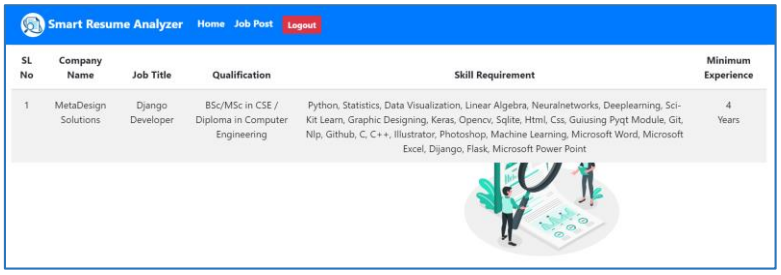

Fig. 9. Job Post Activity

VI. PROBLEM FACED

1. One simple problem of our project is it don't work perfectly with PDF file. In PDF file, sometime regular expression doesn't extract the candidate Institute name. It's not the major problem. Otherwise it extracts all the information perfectly.
2. We faced **Library version problem**. Some library version does not work well with another library version. Like- when we extract text from PDF using PyPDF2 package then we faced version problem.
3. Some time we faced **Data type** problem. Like- MySQL database don't store all data from list. It stores single index data.

VII. CONCLUSION AND FUTURE WORK

The resume analyzing web application system has the potential for further development and enhancement to improve its functionality and usability. Additional features can be added to the system, such as automated screening of candidate social media profiles, automated scheduling of interviews, and analysis of candidate responses to interview questions. A mobile application can be developed for the system to allow job seekers to submit resumes and receive job recommendations on the go. And also solve this project problem to update it. The final testing process is completed without any error.