AUTOMATIC SHORTLISTING OF CVS AND LINKING TO COMPANY PROFILE

Submitted as a project report for mini project

For the partial fulfillment of the degree of

Bachelor of Technology

in

Information Technology

By

PARTHIB CHOWDHURY 2021ITB040

TULIKA CHATTERJEE 2021ITB041

AMISHA SADHUKHAN 2021ITB048

Under the supervision of

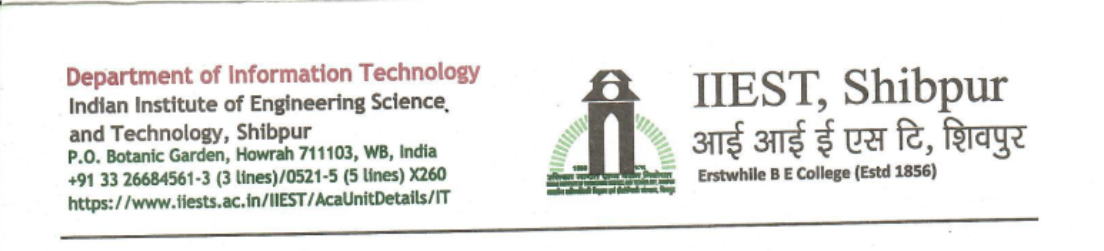
Prof. Shyamalendu Kandar

Assistant Professor

Department of Information Technology

IIEST, Shibpur

May, 2023



Date:

TO WHOM IT MAY CONCERN

This is to certify that Parthib Chowdhury -2021ITB040, Tulika Chatterjee-2021ITB041 and Amisha Sadhukhan-2021ITB048 have done their mini project on “**Automatic Shortlisting of CVs and Linking to the Company”** for the partial fulfillment of the degree of B.Tech. in Information Technology.

During this period they have completed the project. The report has fulfilled all the requirements as per the regulations of the institute and has reached the standard needed for submission.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Shyamalendu Kandar

Assistant Professor

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Dr.Prasun Ghosal

HOD,IT

**ACKNOWLEDGEMENT**

We would like to express our gratitude to all those who have contributed to the completion of this report. First and foremost, we would like to thank Prof. Shyamalendu Kandar, our project supervisor, for his invaluable guidance, support, and feedback throughout the entire project. We could not have completed this report without his expertise and guidance.

We would also like to express our appreciation to the team members who took the time to provide us with their valuable insights and feedback. Without their cooperation and willingness to share their experiences, this report would not have been possible.

In addition, we extend our thanks to our families and friends who have provided us with their unwavering support and encouragement throughout this project.

Finally, we would like to thank all those who have directly or indirectly contributed to the successful completion of this report.

**INDEX**

|  |  |  |
| --- | --- | --- |
| **Sl No** | **Topic** | **Page No** |
| 1 | Chapter 1  Introduction | 5 |
| 2 | Chapter 2  Literature Survey | 6 |
| 3 | Chapter 3  Designing Frontend | 7-8 |
| 4 | Chapter 4  Building Backend Server | 9-10 |
| 5 | Chapter 5  CV Screening Process | 11 |
| 6 | Chapter 6  Conclusion and Future scope | 12 |
| 7 | References | 13 |

Chapter 1

**Introduction**

**1.1 Problem statement**

Develop a website that facilitates automatic shortlisting of CVs and links to company profiles.

**1.2 Project Objective**

Automatic shortlisting of CVs and linking them to company profiles is a process where technology is used to streamline the recruitment process for companies. This process involves using HTML, CSS, and JavaScript for frontend , Node.Js, Express.Js for backend and string matching algorithm for screening, where a CV is given as input to match with the company’s description and hence shortlisting of the candidate based on the criteria set by the company.

The technology uses a set of predefined criteria and keywords to match the job requirements with the skills, qualifications, and experience of the candidates.

This project achieves the following:

* Portable
* User friendly
* Automated system
* Efficient utilization of time and resources

**1.3 Project Goals**

Once the software has identified the potential candidates, it can automatically link their CVs to the company profile to provide additional context and insights. This can include information about the company’s culture, values, mission, and vision. The linked profile can also include details about the company's history, products or services, and current job openings.

Furthermore, this process allows job seekers to have a better chance of getting noticed by employers, as their resumes are more likely to match the required qualifications and skills for the job. Additionally, this technology also links the shortlisted resumes to the company's profile, which can provide job seekers with a better understanding of the company and its culture.

Overall, automatic shortlisting of CVs and linking them to company profiles can save companies time and resources while also improving the quality of the recruitment process. By using this technology, companies can quickly identify the most suitable candidates and make informed decisions about whom to invite for an interview.

Chapter 2

**Literature Survey**

Automated CV screening is a crucial aspect of modern recruitment processes, and there has been significant research in this field over the years. To implement this project, we have to study the **Full Stack Web Development** and the **JavaScript Programming language** in depth. In the literature survey, we will explore some of the latest and most relevant research papers and online resources:

1. **"The MEAN Stack Guide"** **by** **Brad Traversy**. This comprehensive guide covers the MEAN (MongoDB, Express, Angular, Node.js) stack, which is a popular full-stack development framework. The guide provides step-by-step instructions for building a full-stack web application, including backend and frontend development.
2. **"JavaScript: The Definitive Guide" by David Flanagan.** This book is considered a must-read for JavaScript developers and provides a comprehensive guide to the language, including its syntax, object-oriented features, and browser-specific quirks. It covers everything from basic programming concepts to advanced topics such as regular expressions, asynchronous programming, and security.
3. **MDN Web Docs -** [**https://developer.mozilla.org/en-US/docs/Web**](https://developer.mozilla.org/en-US/docs/Web)- This is a comprehensive online resource for web development documentation, including tutorials, guides, and references.
4. **W3Schools -** [**https://www.w3schools.com/**](https://www.w3schools.com/) This online tutorial provides a hands-on approach to learning web development, including HTML, CSS, JavaScript, and many other web technologies.
5. **Udemy -** [**https://www.udemy.com/**](https://www.udemy.com/)- Udemy offers a variety of online courses in web development, ranging from beginner to advanced levels.
6. **MongoDB-** [**https://www.mongodb.com/-**](https://www.mongodb.com/-) this is the official website for mongoDB which provides the documentation , step by step by guides to use this NO-SQL Database.

In conclusion, automated CV screening is an emerging technology that has the potential to revolutionize the recruitment process. With the help of natural language processing (NLP) and machine learning algorithms, we can make it more efficient in improving the accuracy and efficiency of candidate screening.By leveraging these resources, organizations can make informed decisions on implementing automated CV screening solutions to improve their recruitment processes and hire the right talent for their business needs.

Chapter 3

**Designing Frontend**

**4.1 Technologies used:**

* HTML
* CSS
* Javascript
* Bootstrap

**4.2 Designing Home Page:**

A home page is the primary web page that a visitor will view when they navigate to a website via a search engine, and it may also function as a landing page to attract visitors.

* Here we use HTML5 for the structural framework of the page, CSS3 for the style and design of the home page, and Javascript to make the website dynamic.
* It is the first page of the website where we link the CSS file using the “link” tag.
* In the body of the HTML page the following sections are included:

1)**Navigation bar:** It will contain the Home, About Us, and Contact Us , and Log Out buttons.

2)**Background Image and other details:** Background image and headings are included.

3)**Buttons:** A button “Click here” is created which will redirect to the user-form page is created. Javascript is implemented to redirect from the home page to the user form.

**4.3 Designing User Form Page:**

* A basic HTML form page that will take input from user-Name, Email Id, Phone Number, his/her resume and cover letter.
* Only .pdf,.doc and .docx resume files will be accepted as input.
* User can type the cover letter in the text area of the form provided.
* The form is submitted and stored in the database by clicking the button “Submit”.
  1. **Designing About Us page:**
* A basic HTML page providing information about the organization, company, the objectives of the company and the individuals behind the website.
* It gives an overview to the visitors, who we are, what we do and what sets us apart, before they use our website.
* It is a static HTML page designed using CSS and bootstrap.
  1. **Designing Contact Us page:**
* It is a basic HTML page providing visitors with the information on how to get in touch with the organization or company.
* The purpose of the page is to facilitate communication and make it easy for visitors to reach out with inquiries, feedback, or any other relevant information.
* Information (names, contact no. and email Ids) about the website owners are included in the Contact Us page.
* It is also a static HTML page designed with CSS and bootstrap.
  1. **Designing Login Page:**
* The login page is an essential component of the website as it serves the purpose of verifying the identity of users and granting them access to specific features, personalized content or secure areas of the page.
* It is a basic HTML form page that will take input from user- Email and Password.
* The form is submitted and stored in the database by clicking the ‘Login’ button.
* It also has a feature to SignUp directly in case we don’t have an account.
  1. **Designing SignUp Page**
* The signup page or the registration page is an important component of the website as it allows new users to create an account and gain access to the website’s features and functionalities.
* It is a basic HTML form page which asks Email and Password as inputs from the user. It also asks the user to Repeat Password.
* The form is submitted and stored in the database by clicking the ‘SignUp’ button.
* It also has a feature to Login directly in case we are already registered or have an account.

Chapter 4

**Building Backend Server**

**4.1 Technologies used:**

* Node.js
* Express.js
* MongoDB
* Mongoose.
* Passport.js

**4.2 Building a Node.js server using Express.js:**

* Here we use Node.js as the runtime JavaScript environment and express.js as its web application framework.
* We create a browser URL “localhost://3000” where our whole application will run.
* Then we created different routes

a)**”/home”:** Here we have hosted our “index.ejs” web page which is our home page.

b)**”/register”:** In this route we have hosted “registration.ejs” where the client can register on our website before accessing our service.

c)**”/login”:** It renders the “login.ejs” file displaying the login form with the user so that the authenticated user can access our web service.

d)**”/form”:** After clicking the “click here” button in our “/home” route the job applicants are directed to this route which renders the “login.ejs” where they can fill up the form and upload their CV so they can get their best-suited job.

e) **Company Routes:** After submitting the job application, the CVs are screened and the best-matched CVs are directed to these routes along with the job applicant details. These routes are only accessed by the respective companies. In our project, we have taken three companies who are recruiting for Web development, App development, and Data analyst role. Their respective routes are “/web”, ”/app”, ”/data”.

**4.3 Connecting to the database:**

A database is the organized collection of data or information that has been stored inside the computer. Throughout the project, We have used MongoDB as our Database where we store all the data in JavaScript object format.

At first, we connect our whole server with the **MongoDB** database using the **MongoDB shell**. We are using Mongoose as the **Object Data Modelling Library(ODM)** which supports Node.js and Express.js.

We create a database named **“form database”.** Then we have created mongoose models which treat as a collection of JavaScript objects where the data will be stored. The following are the models:

* **Users:** In this Mongoose model, we have stored all the user’s data who have registered on our website.
* **Web\_devs:** In this model, all the form data of the best-suited job applicants who are screened by the dataset given by the web development company, are stored.
* **App\_devs:** The same for the app development company. We just save the details of the job aspirants whose CV is most suited for the job profile.
* **Data\_analysts:** This model stores the data analyst job applicants’ all details.
  1. **Functioning of our website:**
* When job applicants visit our website, they will be asked first to log in or signup to access our service.
* After logging in they can view the home page where they can click the “click here” button which will redirect them to the user-form page.
* In the user-form page the client can fill up the form and upload their CV to apply for a suitable job on our website. Now, based on their CV they will be screened from the rest of the applicants whose CV is not suitable for that job.
* Now, the respective company can visit their specific routes to see the Job Applicants’ list who are shortlisted by our CV screening process.

Chapter 5

**CV Screening Process**

**Objective:** Here comes the CV Screening part. Our resume screening system extracts the relevant skills, experience, and education from the CVs and shortlisted them in accordance with the job description. We have datasets for each company where the respective company gives their requirements on which the CVs will be matched and the best matched (more than 60%) CVs are screened out and uploaded to specific routes on our website with the job applicant details.

**Implementation:** To implement the shortlisting of CV, we have used JAVASCRIPT String Matching Algorithm.

* At first, the texts are extracted from both the dataset and resume file and stored in two strings.
* We have replaced all the special characters in the strings with blank spaces.
* We have created an array of words from both strings not having any blank space.
* Now, words in the array of the dataset have been traversed and finding it in the CV array of words. We want to find how many words are present in the CV which are also present in the dataset. After which we find the matching percentage and if the matching percentage is more than 60% then that CV has been uploaded to the website along with the job seeker’s details.

This whole logic has been implemented for the other two company datasets. In this way, the whole CV screening is done.

Chapter 6

**Conclusion and Future Scope**

In conclusion, building this automated CV screening project has been a challenging yet rewarding experience. We have learned a great deal about the importance of efficient recruitment processes and the role technology can play in enhancing them.

Through this project, we have gained practical experience in designing and developing a system for automating CV screening. We have also learned valuable skills in programming, data analysis, and project management, which will be beneficial for our future careers in the field of computer science.

Overall, we are proud of the project we have developed and the results it has achieved. In the future, we also implement machine-learning algorithms to make it more precise and accurate which improves the hiring process significantly.

**References:**

1. W3Schools. (n.d.). HTML Tutorial. Retrieved from <https://www.w3schools.com/html/>
2. "JavaScript: The Definitive Guide"(August 1996) by David Flanagan.
3. "The MEAN Stack Guide" by Brad Traversy(n.d.). Retrieved from [https://youtu.be/webDevGuide](https://youtu.be/EqzUcMzfV1w).
4. Mozilla Developer Network. (n.d.). CSS. Retrieved from <https://developer.mozilla.org/en-US/docs/Web/CSS>
5. JavaScript. (n.d.). Retrieved from <https://www.javascript.com/>
6. Node.js. (n.d.). Retrieved from <https://nodejs.org/en/>
7. Express.js. (n.d.). Retrieved from <https://expressjs.com/>
8. MongoDB. (n.d.). Retrieved from <https://www.mongodb.com/>
9. Bootstrap. (n.d.). Retrieved from <https://getbootstrap.com/>
10. Udemy(n.d.). Retrieved from <https://www.udemy.com/>
11. MDN Web Docs. (n.d.). Retrieved from <https://developer.mozilla.org/en-US/>