



*Mini Project Report On*

**Resumate: Resume Parser and Job Compatibility  
Checker**

*Submitted in partial fulfillment of the requirements for the  
award of the degree of*

**Bachelor of Technology**  
*in*  
**Computer Science & Engineering**

By

Ann Jacob (U2103037)

Aron Jude Maxwel (U2103049)

Bilna Bijoy (U2103064)

Cerin Saji (U2103065)

Under the guidance of

Ms. Seema Safar

Department of Computer Science & Engineering  
Rajagiri School of Engineering & Technology (Autonomous)  
(Affiliated to APJ Abdul Kalam Technological University)

Rajagiri Valley, Kakkanad, Kochi, 682039

May 2024



# CERTIFICATE

*This is to certify that the mini project report entitled "**Resumate**" is a bonafide record of the work done by **Ann Jacob (U2103037)**, **Aron Jude Maxwel (U2103049)**, **Bilna Bijoy (U2103064)**, **Cerin Saji (U2103065)** submitted to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2023-2024.*

Ms. Seema Safar  
Asst. Professor  
Dept. of CSE  
RSET

Mr. Harikrishnan M  
Asst. Professor  
Dept. of CSE  
RSET

Dr. Preetha K G  
Professor  
Dept. of CSE  
RSET



## **ACKNOWLEDGEMENTS**

I wish to express my sincere gratitude towards Dr P. S. Sreejith, Principal of RSET, and Dr. Preetha K.G., Head of the Department of Computer Science and Engineering for providing me with the opportunity to undertake my mini project, "Resumate".

I am highly indebted to my project coordinators, **Mr. Harikrishnan M.**, Asst. Professor, Department of Computer Science and Engineering and **Ms. Sherine Sebastian**, Asst. Professor, Department of Computer Science and Engineering for their valuable support.

It is indeed my pleasure and a moment of satisfaction for me to express my sincere gratitude to my project guide **Ms. Seema Safar** for her patience and all the priceless advice and wisdom she has shared with me.

Last but not the least, I would like to express my sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

**Ann Jacob**

**Aron Jude Maxwel**

**Bilna Bijoy**

**Cerin Saji**



## Abstract

Manually scanning through resumes to find candidates fit for a role can often be a time-consuming and tedious task for employers and managers. Recruiters may accidentally fail to notice well-qualified candidates and may face difficulties when trying to read through resumes that follow a different format compared to the usual one. To simplify this process, this project aims to build an automated tool to analyze resumes quickly and efficiently. The goal of ‘Resumate’ is to build an automated system to perform resume parsing and job compatibility checking. Resume parsing involves the extraction of appropriate information from a candidate’s resume and presenting it to hiring managers in an organized manner. The extracted information can be used to check the compatibility of the candidate’s skill set and qualifications with the role’s requirements. This aids in effectively screening resumes while saving time, energy, and money which in turn can be directed towards other important tasks. Natural language processing will be the foundation of this project and various tasks are expected to be supported by the SpaCy library and Python’s library for cosine similarity. Deep learning transformer-based models such as BERT will be used for natural entity recognition. Overall, the prominent features of the project will include resume parsing and comparing the specified job description with the entities retrieved from the resume to assess if a candidate is fit for the role.



# Contents

<b>Acknowledgements</b>	<b>i</b>
<b>Abstract</b>	<b>ii</b>
<b>List of Figures</b>	<b>vi</b>
<b>List of Tables</b>	<b>vii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Problem Definition . . . . .	1
1.3 Scope and Motivation . . . . .	1
1.4 Objectives . . . . .	2
1.5 Challenges . . . . .	2
1.6 Assumptions . . . . .	2
1.7 Societal / Industrial Relevance . . . . .	2
<b>2 Software Requirements Specification</b>	<b>3</b>
2.1 Introduction . . . . .	3
2.1.1 Purpose . . . . .	3
2.1.2 Product Scope . . . . .	3
2.2 Overall Description . . . . .	3
2.2.1 Product Perspective . . . . .	3
2.2.2 Product Functions . . . . .	4
2.2.3 Operating Environment . . . . .	5
2.2.4 Assumptions and Dependencies . . . . .	5
2.3 External Interface Requirements . . . . .	6
2.3.1 User Interfaces . . . . .	6
2.3.2 Hardware Interfaces . . . . .	6

2.3.3	Software Interfaces . . . . .	6
2.3.4	Communication Interfaces . . . . .	7
2.4	System Features . . . . .	7
2.4.1	Resume Parsing . . . . .	7
2.4.1.1	Description and Priority . . . . .	7
2.4.1.2	Stimulus/Response Sequences . . . . .	7
2.4.1.3	Functional Requirements . . . . .	7
2.4.2	Checking Job Compatibility . . . . .	8
2.4.2.1	Description and Priority . . . . .	8
2.4.2.2	Stimulus/Response Sequences . . . . .	8
2.4.2.3	Functional Requirements . . . . .	8
2.5	Other Nonfunctional Requirements . . . . .	8
2.5.1	Performance Requirements . . . . .	8
2.5.2	Safety Requirements . . . . .	9
2.5.3	Security Requirements . . . . .	9
2.5.4	Software Quality Attributes . . . . .	9
<b>3</b>	<b>System Architecture and Design</b>	<b>10</b>
3.1	System Overview . . . . .	10
3.2	Architectural Design . . . . .	11
3.3	Dataset identified . . . . .	12
3.4	Proposed Methodology/Algorithms . . . . .	12
3.4.1	Resume Parsing through NER: . . . . .	12
3.4.2	Job Compatibility Checking: . . . . .	12
3.5	User Interface Design . . . . .	14
3.6	Description of Implementation Strategies . . . . .	15
3.7	Module Division . . . . .	15
3.7.1	Front-end Modules: . . . . .	15
3.7.2	Back-end Modules: . . . . .	16
3.8	Work Schedule - Gantt Chart . . . . .	17
<b>4</b>	<b>Results and Discussions</b>	<b>18</b>
4.1	Overview . . . . .	18

4.2	Testing . . . . .	18
4.3	Quantitative Results . . . . .	20
4.4	Discussion . . . . .	22
<b>5</b>	<b>Conclusion</b>	<b>23</b>
5.1	Conclusion . . . . .	23
5.2	Future Scope . . . . .	23
	<b>References</b>	<b>24</b>
	<b>Appendix A: Presentation</b>	<b>25</b>
	<b>Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes</b>	<b>40</b>
	<b>Vision, Mission, POs, PSOs and COs</b>	<b>41</b>
	<b>Appendix C: CO-PO-PSO Mapping</b>	<b>45</b>



## **List of Figures**

2.1	Use Case Diagram of Resumate . . . . .	4
3.1	Sequence Diagram of Resumate . . . . .	11
3.2	UI-Login Page . . . . .	14
3.3	UI-Signup Page . . . . .	14
3.4	UI-Main Page . . . . .	15
3.5	Work Schedule detailed using a Gantt Chart . . . . .	17
4.1	Login Page . . . . .	18
4.2	Signup Page . . . . .	19
4.3	Main Page . . . . .	19
4.4	Confusion Matrix . . . . .	22



## **List of Tables**

4.1	Comparison of Overall Performance Metrics - Model 1 and Model 2 . . . . .	20
4.2	NER Performance Metrics (per entity) - Selected Model . . . . . . . . . . .	21



# **Chapter 1**

## **Introduction**

### **1.1 Background**

Employers are often faced with having to manually look through a large number of resumes to shortlist the candidates that may prove to be effective employees and valuable assets to the organization. This is a long and tedious task that leads to a waste of time, effort, and energy. Therefore, there arises a need to automate this process using modern software to make room for more creative work without wasting valuable resources.

### **1.2 Problem Definition**

Manually scanning through resumes to find candidates fit for a role can often be a time-consuming task for employers. Recruiters may fail to notice well-qualified candidates and may face difficulties when trying to read through resumes that follow different formats. Hence, an automated resume analyzer aids in effectively screening resumes while saving time, energy, and money.

### **1.3 Scope and Motivation**

Resumate is a software that is designed and developed for resume parsing to simplify and extract important information from resumes and determine the extent of compatibility of a candidate's skills compared to a specified set of skills or job description, mainly related to the IT field. A resume analysis software is extremely useful in managing recruitment processes easily and efficiently. Hiring managers can save time and effort by automating the process of determining candidates fit for a particular role and the level of compatibility for the same.

The motivation behind this project is mainly based on the fact that the hiring process

is growing hectic and complicated, day by day. Through automation, the work of the recruiters is made easy and less hectic.

#### **1.4 Objectives**

- Resume Parsing - Resume parsing is the process of extracting essential details from unstructured resumes and presenting them in an organized manner.
- Obtaining pertinent information and mapping to corresponding job titles
- Checking Job Compatibility - Job compatibility checking is the process of comparing the extracted details with the skills required for the job to determine how fit a particular candidate is for a given role.

#### **1.5 Challenges**

- Lack of resume datasets for a particular job title.

#### **1.6 Assumptions**

Certain assumptions have been made while planning the development of Resumate. These include:

- The input document will be written in English only.
- The structure of the input resume will contain 2-3 columns at most.
- Only the resumes of IT professionals will be given as input to the system.
- The resume format is restricted to PDF documents.

#### **1.7 Societal / Industrial Relevance**

The project applies to the recruitment industry, focusing on the recruitment process specific to IT professionals. Hiring managers and employers can use Resumate to improve the efficiency of the hiring process.

## **Chapter 2**

# **Software Requirements Specification**

### **2.1 Introduction**

#### **2.1.1 Purpose**

The purpose of this document is to provide a description of the web application ‘Resumate: Resume Parser and Job Compatibility Checker (version 1.0)’. It explains the purpose of the software, the interfaces of the software, what the software will do, and the constraints under which it must operate. This document is intended for both the software’s stakeholders and developers.

#### **2.1.2 Product Scope**

Resumate is a software that is designed and developed for resume parsing to simplify and extract important information from resumes and determine the extent of compatibility of a candidate’s skills compared to a specified set of skills or job description, related to the IT field. A resume analysis software is extremely useful in managing recruitment processes easily and efficiently. Hiring managers can save time and effort by automating the process of determining candidates fit for a particular role, and the level of compatibility for the same.

### **2.2 Overall Description**

#### **2.2.1 Product Perspective**

Resumate is a variation of standard document parsing technology that is intended to be used by employers to check the compatibility of a potential employee with a specified job description with the relevant information provided in their resume without having to manually look through the document. It is a derivative of existing resume parsing

software that is used to provide different insights to the applicant. Resumate can be used by departments within organizations that are responsible for hiring recruits.

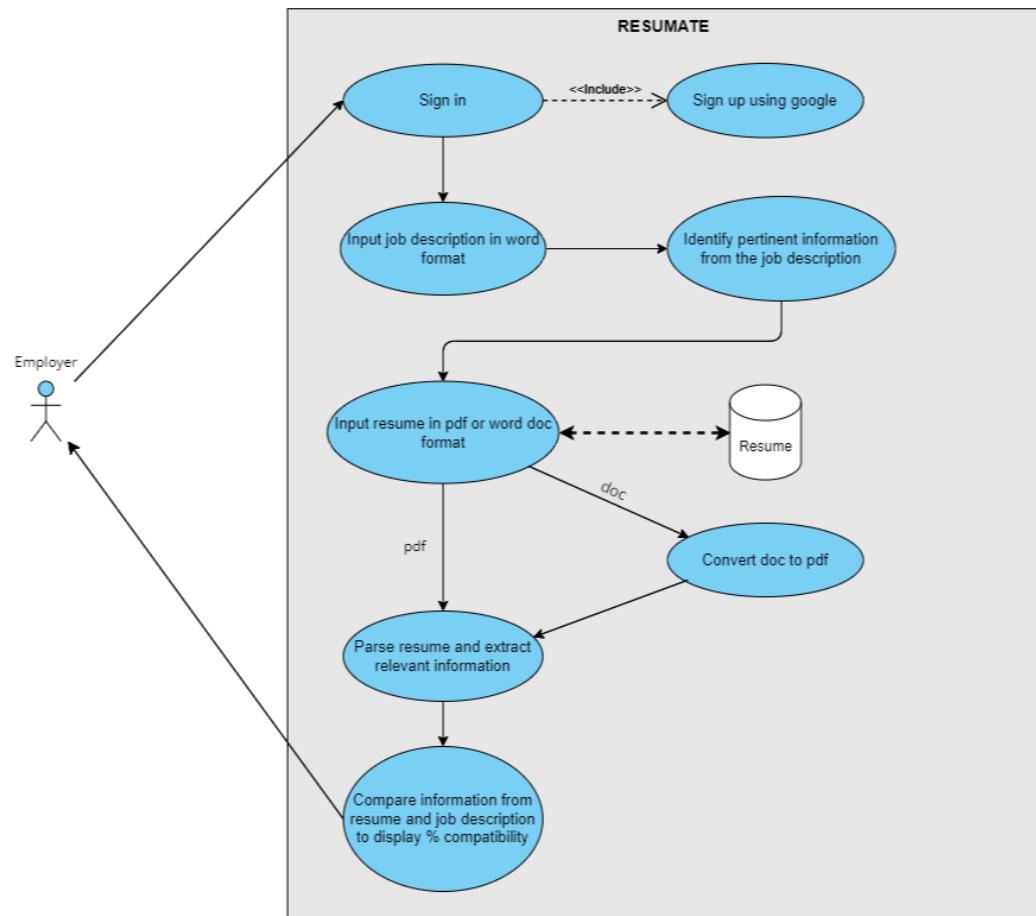


Figure 2.1: Use Case Diagram of Resumate

### 2.2.2 Product Functions

Resume Parsing (Priority: High)

- Extracts pertinent information from a resume (Fig 2.1) and presents it in an organized manner.
- Users upload a candidate's resume in PDF format by clicking on the 'Upload' button.

ton and selecting the file. The uploaded file undergoes parsing, and the output is displayed on the screen when the 'check compatibility' button is clicked.

#### Checking Job Compatibility (Priority: High)

- Compares the parsed resume with the skills required for the position mentioned in the job description to determine a candidate's suitability for a role.
- The output specifies the percentage compatibility rating of the candidate for the role.

#### 2.2.3 Operating Environment

As Resumate is intended to be a web application, there are no stringent requirements concerning the operating environment. A computer running Windows 8 or above with 8 gigabytes of RAM is being used to develop the application. Similar versions of Mac or Ubuntu can be used for the same.

#### 2.2.4 Assumptions and Dependencies

Certain assumptions have been made while planning the development of Resumate. These include:-

- The input document will be written in English only.
- The structure of the input resume will contain 2-3 columns at most.
- Only resumes of IT professionals will be given as input to the system.
- The resume format is restricted to PDF documents.
- The software will be used on desktops or laptops only.

## **2.3 External Interface Requirements**

### **2.3.1 User Interfaces**

The user interfaces for the software shall be compatible with major web browsers - Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari, by which a user can access the system. The user interfaces for the software are designed as a web-based application using Flask, a Python framework for web app development. Users submit the resumes in the specified format for classification and an analysis-triggering button is used. Results are displayed on the same interface. The interface is responsive, ensuring compatibility with various browsers and devices, and features error handling for invalid inputs or analysis failures. Overall, the interface aims to provide an intuitive and efficient user experience for resume analysis.

### **2.3.2 Hardware Interfaces**

Resumate is designed to be lightweight and does not impose stringent hardware requirements. The system can operate on standard computing hardware commonly found in modern workplaces. Minimum recommended specifications include a computer with a dual-core processor, 8 GB of RAM, and sufficient disk space to accommodate the application and its dependencies. Additionally, the system is web-based, and users can access it through standard web browsers such as Chrome, Firefox, or Safari. Internet connectivity is required for the users. The goal is to ensure widespread accessibility and ease of integration into standard computing environments.

### **2.3.3 Software Interfaces**

Resumate relies on several software components to achieve its functionalities. The core application is built using the latest version of Python to date, Python 3.12, and Flask, facilitating web-based interactions. Natural Language Processing (NLP) capabilities are harnessed through the integration of SpaCy for entity extraction and BERT for advanced text analysis. The system utilizes the Transformers library for managing BERT models. The database used is MongoDB, to store relevant data. Compatibility with the Python environment (version 3.6 and above) is crucial for seamless execution. These software interfaces collectively contribute to the efficient analysis of resumes within the system.

#### **2.3.4 Communication Interfaces**

This project supports major web browsers, which include Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari. Resumate engages in communication with external components to enhance its functionality. User interactions with the web-based interface are facilitated through standard HTTP and HTTPS protocols. Additionally, communication with the SpaCy NLP library and BERT models is established internally within the application. To enhance security, user authentication and authorization mechanisms are implemented through secure HTTPS connections. The system ensures reliable and efficient communication with external entities, supporting the seamless exchange of data for the analysis and classification of submitted resumes.

### **2.4 System Features**

This section describes the features provided by Resumate and explains how they can be used, and the results they will give back to the user.

#### **2.4.1 Resume Parsing**

##### **2.4.1.1 Description and Priority**

For a particular resume, the system will extract important information such as skills, experience, etc., and present it in an organized manner. This system functionality is of the highest priority as it obtains necessary details that are essential in checking job compatibility.

##### **2.4.1.2 Stimulus/Response Sequences**

Users can input a candidate's resume as a singular file in PDF format by clicking on the 'Upload Resume' button and selecting the required file from their system. Once uploaded, the selected file will undergo parsing and the output will be displayed on the screen.

##### **2.4.1.3 Functional Requirements**

- 2.4.1.3.1 The uploaded resume must be in PDF format (as seen in fig 2.1). Since the input allows only files of these formats to be selected, no other file format will

be accepted. Hence, related format-specific errors will be avoided.

REQ-1: Doc-format-1

- 2.4.1.3.2 The uploaded resume must be in English. Resumes in other languages will be flagged and will not be accepted by the system.

REQ-2: Eng-lang

## **2.4.2 Checking Job Compatibility**

### **2.4.2.1 Description and Priority**

Based on a specified set of desirable skills or job descriptions, the output of the parsing process is compared with it to determine how compatible the candidate is for the given role. This system functionality is also of the highest priority as it gives the result of the analysis, which is used to determine if the candidate is fit for the job or not.

### **2.4.2.2 Stimulus/Response Sequences**

Users can input the job description by copying and pasting the text to the given text box (or manually typing it in). By clicking on the ‘Check Compatibility’ button, the output of the parsed resume is compared with the contents of the job description. The output of this process is text displayed on the screen that specifies the compatibility rating of the candidate for the role in %.

### **2.4.2.3 Functional Requirements**

- 2.4.2.3.1 The uploaded description file must be in PDF format. Since the input allows only files of these formats to be selected, no other file format will be accepted and hence related format-specific errors will be avoided.

REQ-1: Doc-format-2

## **2.5 Other Nonfunctional Requirements**

### **2.5.1 Performance Requirements**

The application should be able to respond to the user without much delay. It should be able to display accurate results with minimal room for error.

### **2.5.2 Safety Requirements**

Measures are to be implemented to protect user privacy and ensure that personally identifiable information is securely handled and stored.

### **2.5.3 Security Requirements**

Users must have an authorized email-linked account to register and log in before performing any operations on the website. This ensures that user data is stored securely and handled in compliance with relevant privacy regulations and unauthorized access is prevented.

### **2.5.4 Software Quality Attributes**

- Reliable: The application will not crash under any circumstance such as when a user enters invalid values during login. The input files will not be tampered with or corrupted.
- Scalable: The system will be capable of supporting a large number of clients and servers.
- Portable: Since this project is implemented as a web application, it is portable and is supported by all major web browsers.



## **Chapter 3**

# **System Architecture and Design**

### **3.1 System Overview**

Resumate is a comprehensive automated resume analysis system that revolutionizes the recruitment process by leveraging advanced parsing technology and natural language processing (NLP) techniques. Employers and hiring managers can efficiently evaluate candidate suitability against job descriptions, enhancing decision-making and streamlining the hiring process. The system's architecture is depicted below, outlining the entire process from input processing to visualization and reporting.

#### **Detailed Architecture:**

##### **User Interface**

Users access the Resumate interface to input job descriptions and upload resumes for analysis. The interface facilitates intuitive interaction and provides feedback on the analysis process.

##### **Input Processing:**

Job descriptions are inputted by users, specifying the requirements, qualifications, and skills for a particular position. Resumes from job applicants are uploaded in various formats and undergo pre-processing for further analysis.

##### **Document Parsing:**

Resumate employs advanced parsing technology to extract relevant information from both the job description and resumes. NLP techniques are applied to identify key skills, experiences, qualifications, and other pertinent details.

##### **Comparison Algorithm:**

A robust matching algorithm compares the extracted information from resumes with the requirements outlined in the job description. It evaluates the compatibility of each applicant's qualifications and experiences, generating insights for decision-making.

### **Analysis and Insights:**

Resumate generates a rating that highlights the degree of compatibility between each applicant's resume and the job description. Insights are presented as a percentage compatibility rating.

### **Visualization and Reporting:**

Results are presented in percentage format, aiding in quick understanding and decision-making.

## **3.2 Architectural Design**

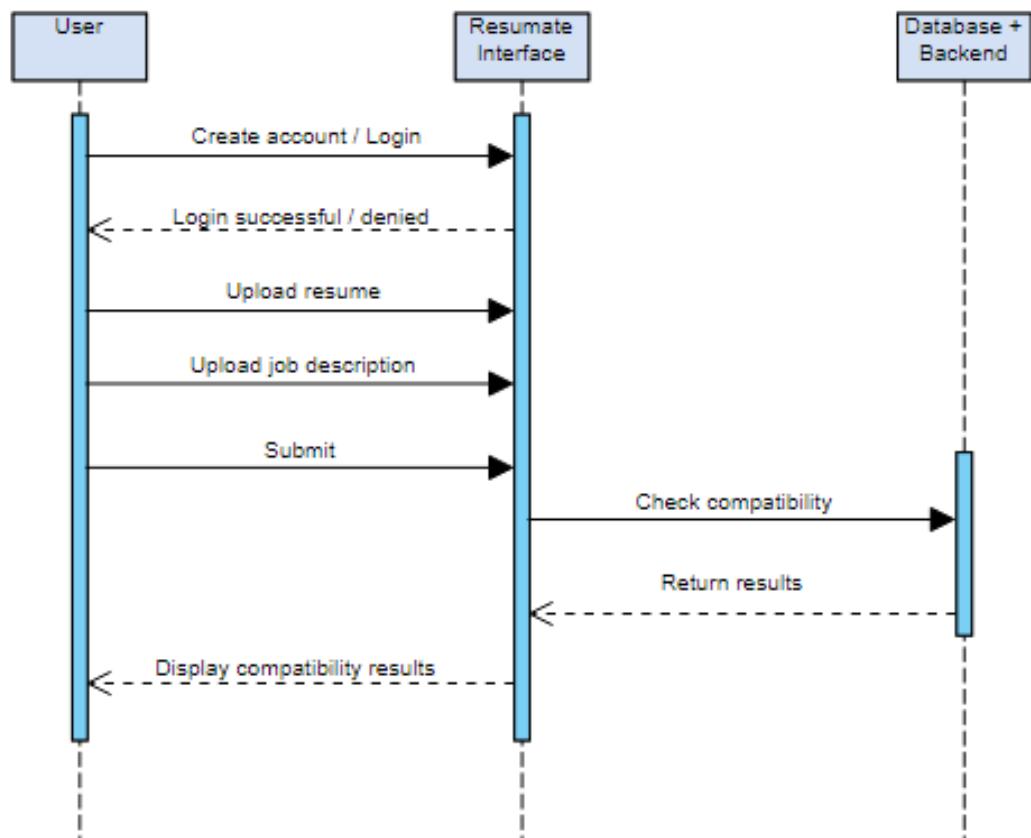


Figure 3.1: Sequence Diagram of Resumate

### **3.3 Dataset identified**

- Resume dataset from Huggingface - The dataset contains over 30,000 resumes stored in string format. One of the challenges faced while using this dataset is sorting and extracting resumes of job-seekers specifically in the IT field.
- Resume dataset from Kaggle - This is a collection of Resume Examples taken from livecareer.com for categorizing a given resume into any of the labels defined in the dataset. An obstacle faced was that the number of resumes were not enough to properly train the model required for the software.

### **3.4 Proposed Methodology/Algorithms**

#### **3.4.1 Resume Parsing through NER:**

**Description:** Utilize spaCy's NER capabilities to identify and extract key entities such as skills, education, work experience, certifications, etc., from the resume text.

**Algorithm:**

```
entities = []
for token in resume_text:
    if token.ent_type_ != "":
        entities.append((token.text, token.ent_type_))
```

#### **3.4.2 Job Compatibility Checking:**

**Description:**

Cosine similarity is a metric used to measure the similarity between two non-zero vectors. It is widely used in text mining and information retrieval to measure the similarity between documents or pieces of text. The cosine similarity is calculated as the cosine of the angle between two vectors, which is equivalent to the dot product of the vectors divided by the product of their magnitudes.

### **Algorithm:**

1. Input Vectors: Given two vectors, A and B, each of dimension n.
2. Initialization: Initialize three variables: dot\_product to 0, magnitude\_A to 0, and magnitude\_B to 0.
3. Compute Dot Product and Magnitudes:

For each element i from 1 to n:

- Compute the dot product:  $\text{dot\_product} += \text{A}[i] * \text{B}[i]$
  - Compute the magnitude of vector AA:  $\text{magnitude\_A} += (\text{A}[i])^2$
  - Compute the magnitude of vector BB:  $\text{magnitude\_B} += (\text{B}[i])^2$
4. Finalize Magnitudes: Take the square root of magnitude\_A and magnitude\_B to get the final magnitudes of vectors A and B:
    - $\text{magnitude\_A} = \sqrt{\text{magnitude\_A}}$
    - $\text{magnitude\_B} = \sqrt{\text{magnitude\_B}}$
  5. Calculate Cosine Similarity: Divide the dot product by the product of the magnitudes of vectors A and B:
    - $\text{cosine\_similarity} = \text{dot\_product} / (\text{magnitude\_A} * \text{magnitude\_B})$

### 3.5 User Interface Design

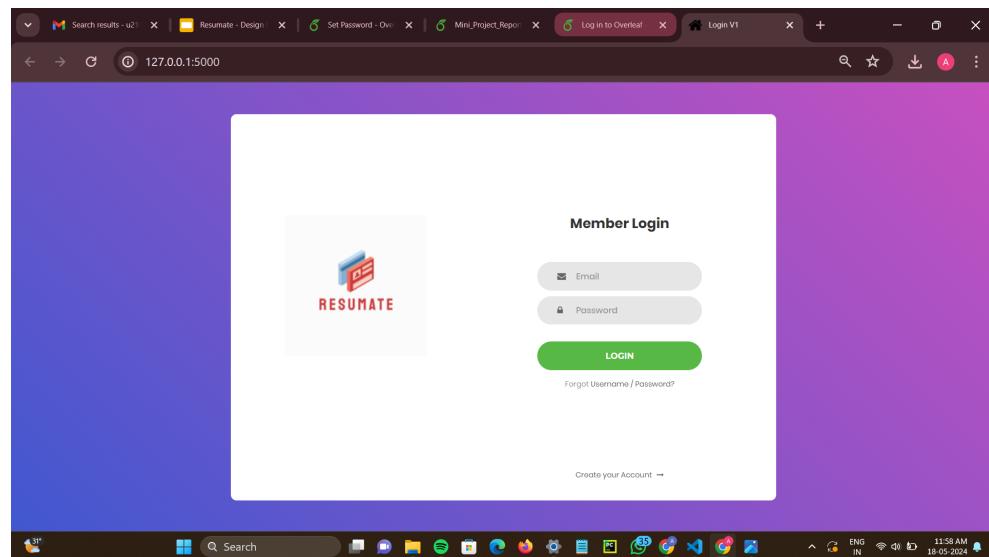


Figure 3.2: UI-Login Page

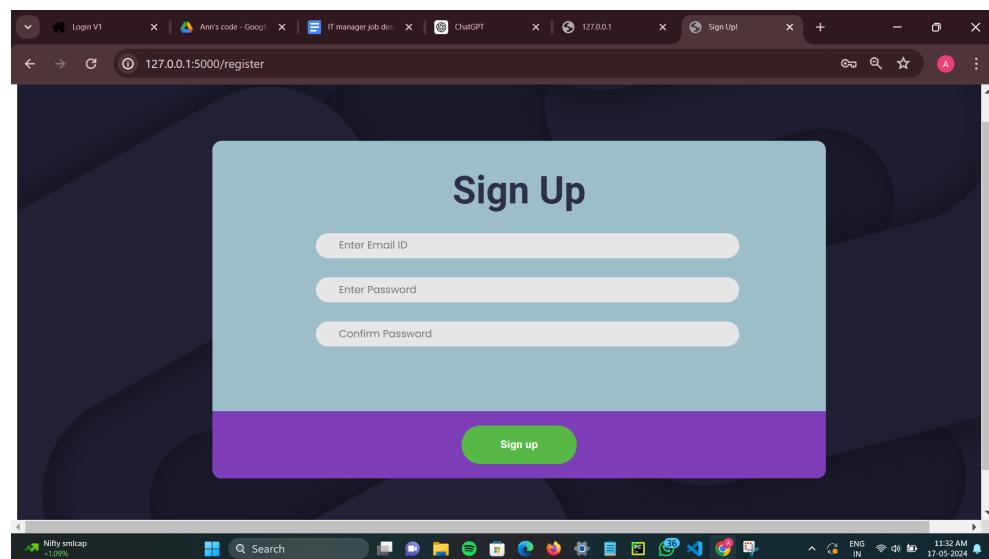


Figure 3.3: UI-Signup Page

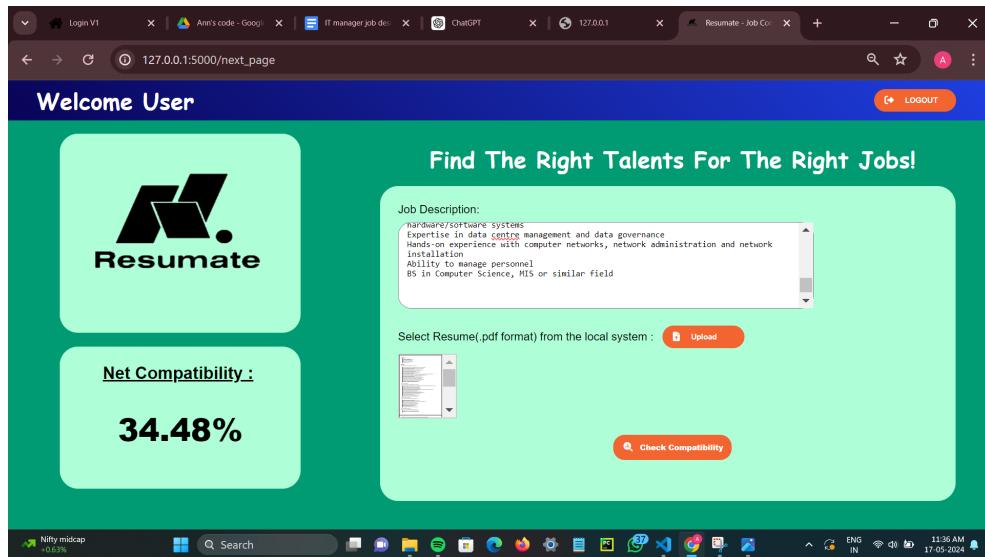


Figure 3.4: UI-Main Page

### 3.6 Description of Implementation Strategies

The user interface is implemented using the python framework Flask. Hybrid approach of spaCy and Bert are used for named entity recognition and text analysis.

**Flask Integration:** Flask is used to develop the web application for Resumate. It provides a lightweight and flexible framework for building web services in Python.

**SpaCy for NER:** We can leverage spaCy to perform named entity recognition (NER) on resume texts, which helps in identifying entities such as names, skills, etc.

**BERT for Contextual Embeddings:** BERT can be utilized to generate contextual word embeddings for resume texts, which capture the semantic meaning of words based on their context in the sentence.

### 3.7 Module Division

#### 3.7.1 Front-end Modules:

- User Interface Module:** This module will be responsible for creating the user interface for the application, including features like user login, uploading resumes and job description and generating compatibility ratings.

### **3.7.2 Back-end Modules:**

1. **Resume Input Module:** This module will handle the back-end logic for processing resume uploads and extracting text data.
2. **Text Processing Module:** This back-end module will focus on preprocessing the extracted text data from resumes. It involves tasks including identifying key sections like education, experience, skills, etc.
3. **Parsing Module:** This module is responsible for parsing the preprocessed text to extract specific information such as the candidate's name, contact details, educational background, work experience, skills, etc.
4. **Skill Matching Module:** This back-end module will compare the skills extracted from resumes against a predefined set of skills or a job description that is taken from the database. It determines the level of compatibility of a candidate's skills with the required skills for a particular role.
5. **Database Module:** Manages the storage and retrieval of parsed resume data and associated information as well as the skills and educational qualifications based on the job description.
6. **Model Training Module:** This module involves training machine learning models to enhance the parsing and skill-matching capabilities. It includes tasks like training, and evaluation.
7. **Reporting Module:** Generates reports summarizing the compatibility of candidates with job requirements. It includes a percentage rating showing the compatibility to aid decision-making in the hiring process.

### 3.8 Work Schedule - Gantt Chart

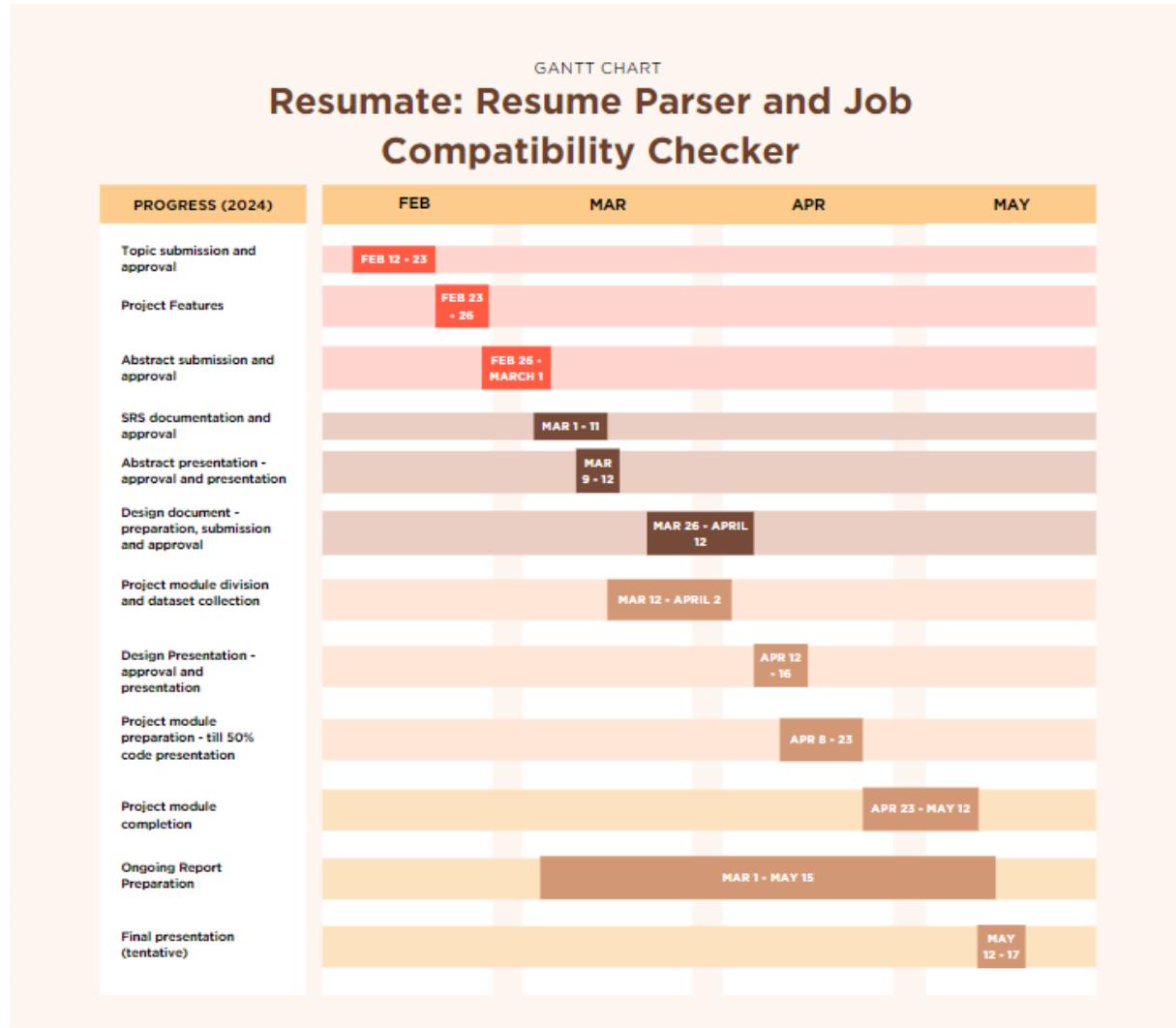


Figure 3.5: Work Schedule detailed using a Gantt Chart

# Chapter 4

## Results and Discussions

### 4.1 Overview

Resumate is a web application designed to streamline the job application process by allowing users to upload their resumes and job descriptions, and then providing a compatibility percentage based on a detailed analysis of the content. Utilizing Natural Language Processing and sophisticated matching algorithms, Resumate extracts key information from both documents, compares them and return results.

### 4.2 Testing

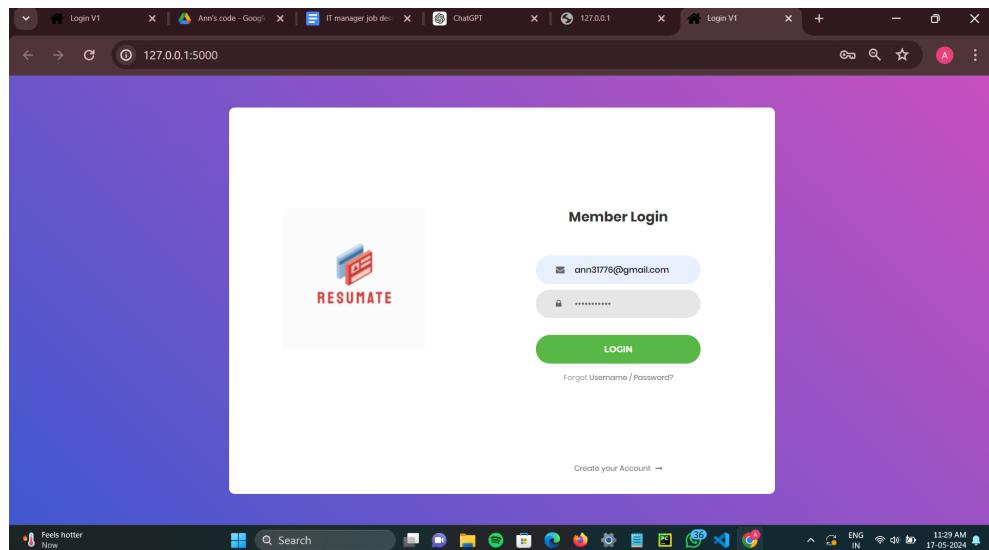


Figure 4.1: Login Page

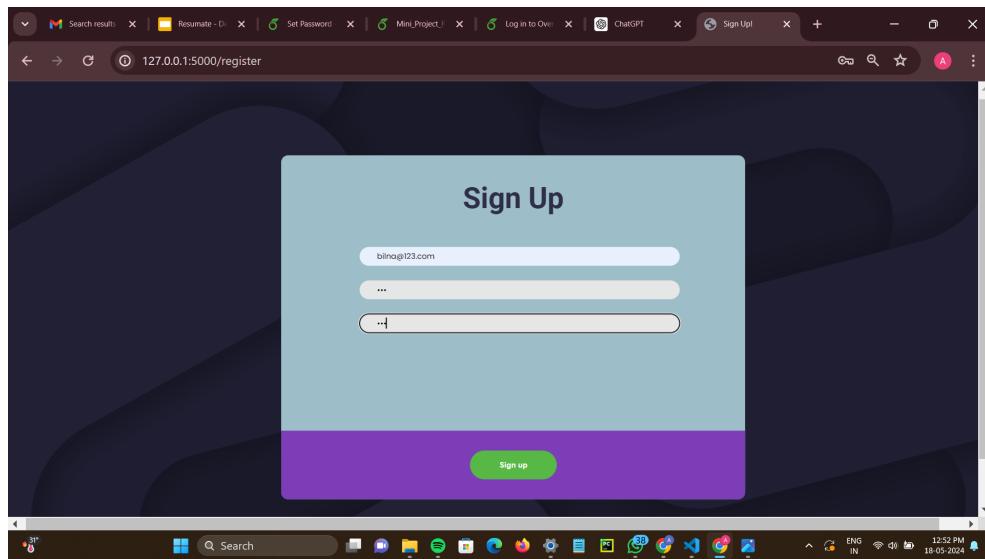


Figure 4.2: Signup Page

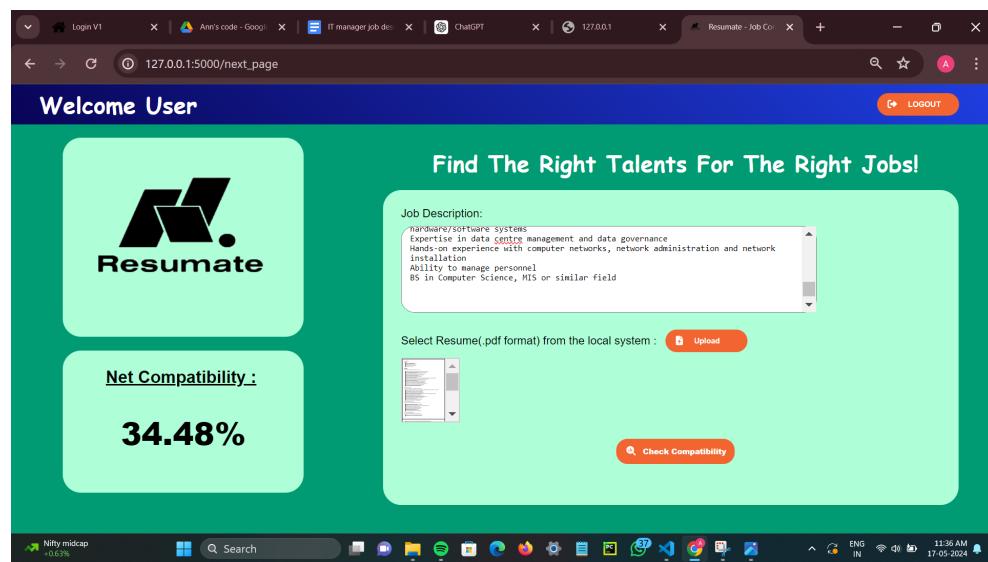


Figure 4.3: Main Page

### 4.3 Quantitative Results

The table 4.1 gives an overview of the results obtained from the evaluation of both the trained models on the test dataset. Considering the model with better performance (model 2), the overall NER precision is 85.59, NER Recall is 87.21 and NER F1-Score is 86.39. These metrics indicate that the selected model has performed well in the tokenization and NER tasks.

High Precision (85.59%): The model is good at correctly identifying entities without making too many mistakes by labeling non-entities as entities.

High Recall (87.21%): The model successfully identifies the majority of the actual entities present in the text.

Balanced F1-Score (86.39%): The model maintains a good balance between precision and recall, indicating robust performance in identifying entities correctly and comprehensively.

Metric	Model 1	Model 2
Tokenization Accuracy	100.00	100.00
NER Precision	69.35	85.59
NER Recall	52.84	87.21
NER F1-Score	59.98	86.39
SPEED	110	119

Table 4.1: Comparison of Overall Performance Metrics - Model 1 and Model 2

The table 4.2 gives an overview about the Precision, Recall and F1-scores of the selected model (model 2) for each recognized entity.

Category	Precision	Recall	F1-Score
LINKEDIN LINK	99.01	99.50	99.26
SKILLS	82.08	76.18	79.02
CERTIFICATION	81.44	83.76	82.58
NAME	93.00	93.47	93.23
LOCATION	79.47	83.43	81.40
COMPANIES WORKED AT	90.07	94.09	92.04
WORKED AS	87.41	90.82	89.08
YEARS OF EXPERIENCE	88.96	97.89	93.21
COLLEGE NAME	62.60	70.69	66.40
DEGREE	82.82	84.11	83.46
LANGUAGE	83.03	86.16	84.57
UNIVERSITY	79.14	73.83	76.39
YEAR OF GRADUATION	48.15	24.53	32.50
AWARDS	84.00	89.36	86.60
EMAIL ADDRESS	62.50	83.33	71.43
Unlabelled	0.00	0.00	0.00

Table 4.2: NER Performance Metrics (per entity) - Selected Model

Fig 4.4 describes the confusion matrix based on the test results obtained from the selected model. The provided metrics give the precision, recall, and F1 score for a Named Entity Recognition (NER) task, but since there is a lack of direct scores, TP is assumed to be 1000 for simplicity.

Calculating FP and FN on the basis of the following relationship, we get:

$$P = TP / (TP + FP), R = TP / (TP + FN)$$

Therefore, FP=117, FN=147

Given that the true negatives (TN) are not specified, we'll use a placeholder value to complete the matrix.

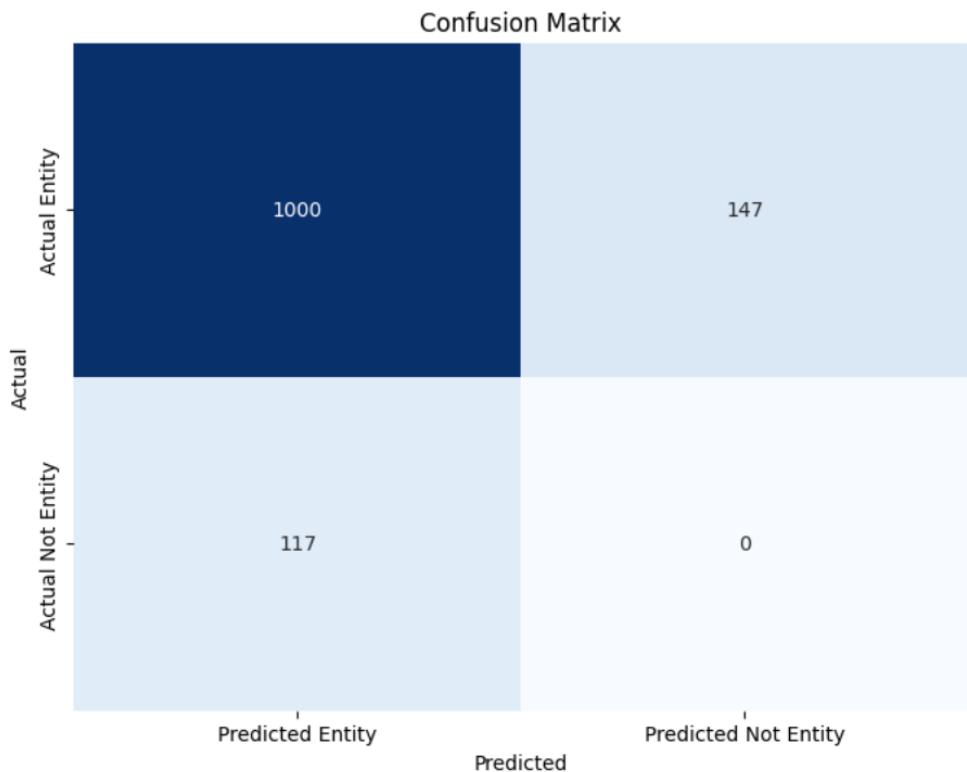


Figure 4.4: Confusion Matrix

#### 4.4 Discussion

Comparing the results of the two trained models, it is observed that the second model performs better compared to the first one and hence it is chosen as the parser model for the development of the resume parsing module of Resumate. This is because the first model was trained on a dataset with only 200 annotated resumes whereas the second model was trained on a dataset with 1000 annotated resumes. Hence, the second model was able to train better and produce more accurate results.



# **Chapter 5**

## **Conclusion**

### **5.1 Conclusion**

Resumate is a tool that can impact and improve the hiring process effectively. Its capability to parse resumes and extract pertinent data, as well as rate the skills and capabilities of a potential employee is a very simple and straightforward process that helps automate the recruitment to a certain degree.

### **5.2 Future Scope**

The future scope of Resumate includes:

- Expanding its capabilities to accept resumes in a wider range of formats, offering more detailed compatibility results
- Providing personalized recommendations to improve resumes.
- Integration with job portals and Applicant Tracking Systems (ATS) will streamline applications, while advanced AI and machine learning models will enhance accuracy.



## References

- [1] Nirmiti Bhoir, Mrunmayee Jakate, Snehal Lavangare, et al. "Resume Parser using a hybrid approach to enhance the efficiency of Automated Recruitment Processes". Authorea. April 17, 2023.
- [2] Alkeshwar Jivtode, Kisan Jadhav, and Dipali Kandhare. "RESUME ANALYSIS USING MACHINE LEARNING AND NATURAL LANGUAGE PROCESSING." IRJETS. May 5, 2023.
- [3] D. Wang, J. Su and H. Yu, "Feature Extraction and Analysis of Natural Language Processing for Deep Learning English Language," in IEEE Access, vol. 8.
- [4] M. Soni, S. Gomathi and Y. Bhupendra Kumar Adhyaru, "Natural Language Processing for the Job Portal Enhancement," 2020 7th International Conference on Smart Structures and Systems (ICSSS), Chennai, India, 2020.
- [5] Abdul Wahab, Dr. M N. Nachappa. "Resume Parser with Natural Language Processing." International Research Journal of Engineering and Technology (IRJET) Volume: 09 Issue: 03, March 2022.
- [6] Nirali Bhaliya, Jay Gandhi, Dheeraj Kumar Singh. "NLP-based Extraction of Relevant Resume using Machine Learning." International Journal of Innovative Technology and Exploring Engineering (IJITEE), Volume-9 Issue-7, May 2020.



## **Appendix A: Presentation**





# Resumate

Final Presentation

Ms. Seema Safar  
Asst. Professor  
Department of CSE, RSET

Ann Jacob - U2103037  
Aron Jude Maxwel - U2103049  
Bilna Bijoy - U2103064  
Cerin Saji - U2103065  
Department of CSE, RSET



## Table of contents

1. Introduction
2. Problem Definition
3. Objectives
4. System Design
5. Datasets
6. Work Division –Gantt Chart
7. Software/Hardware Requirements
8. Results
9. Conclusion
10. Future Enhancements
11. References



# Introduction

- Natural language processing (NLP), the domain of our project, is a machine learning technology that gives computers the ability to interpret, manipulate and comprehend human language.
- The goal of *Resumate* is to build an automated system to perform resume parsing and job compatibility checking.
- Manually scanning through resumes to find candidates fit for a role can often be a time-consuming task for employers. Recruiters may fail to notice well-qualified candidates and may face difficulties when trying to read through resumes that follow different formats.
- Hence an automated resume analyzer aids in effectively screening resumes while saving time, energy and money.

3



# Problem Definition

- To develop a software to perform resume parsing and job compatibility checking using NLP techniques, which can help recruiters save a lot of time and effort in selecting candidates.

4

# Objectives

1

Data Collection

4

UI Development

2

Preprocessing

5

Compatibility Analysis

3

NLP Model Development

# System Design - System Overview

## User Interface

Via the UI interface, users can login, input job descriptions and upload resumes for analysis.

## Input Processing

The resume to be analysed and the specific job descriptions are inputted, specifying the requirements, qualifications and skills for a particular position.

## Document Parsing

Resume parsing yields pertinent information from the document.

## Comparison Algorithm

A robust matching algorithm compares the extracted information from resumes with the requirements outlined in the job description.

## Analysis and Insights

The level of compatibility is specified through a simplified rating in percentage format.

## System Design - Architectural Design

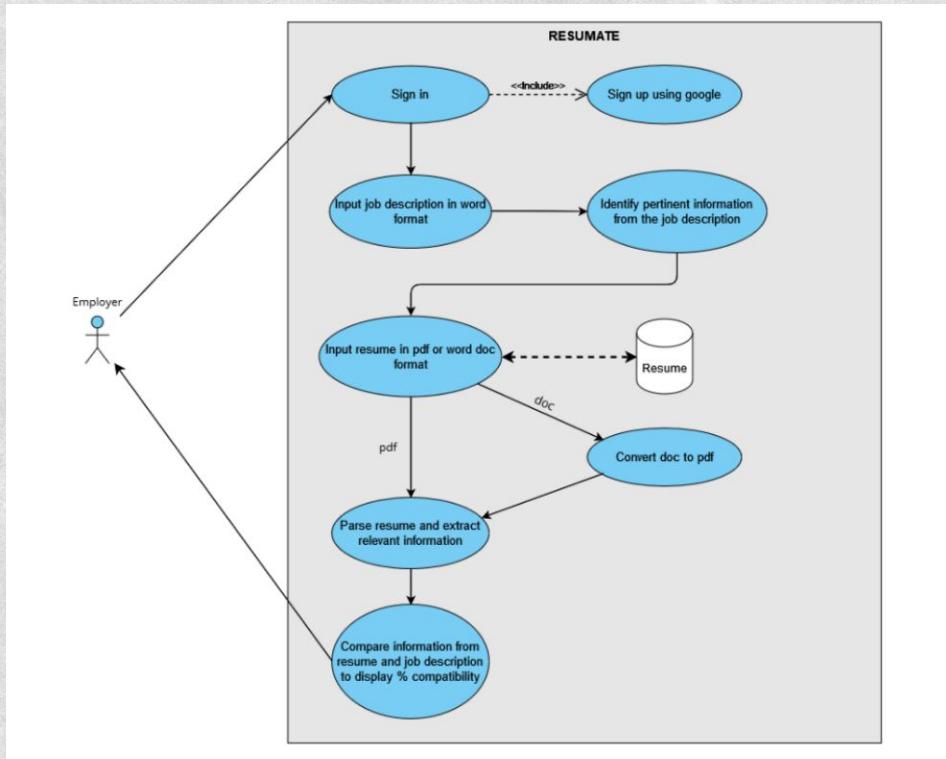


Fig 1: Use Case diagram of Resumate

# System Design - Module Division

## Front - end Modules:

**1. User Interface Module:** This module will be responsible for creating the user interface for the application, including features like user login, uploading resumes and job description viewing the compatibility ratings.

## Back - end Modules:

**1. Resume Input Module:** This module will handle the back-end logic for processing resume uploads and extracting text data.

**2. Text Processing Module:** This backend module will focus on preprocessing the extracted text data from resumes.

**3. Parsing Module:** This module is responsible for parsing the preprocessed text to extract required information.

**4. Skill Matching Module:** This backend module will compare the skills extracted from resumes against a predefined set of skills or a job description. It determines the level of compatibility of a candidate's skills with the required skills for a particular role.

**5. Database Module:** Used to store login and authentication details in the database.

**6. Model Training Module:** This module involves training a machine learning model to enhance the parsing and skill-matching capabilities. It includes tasks like training and evaluation.

**7. Reporting Module:** Generates reports summarizing the compatibility of candidates with job requirements. It includes a percentage rating showing the compatibility to aid decision-making in the hiring process.

## System Design - Algorithms

- 1. Resume Parsing through NER:** Utilize spaCy's NER capabilities to identify and extract key entities such as skills, education, work experience, certifications, etc., from the resume text.

```
entities = []
for token in resume text:
    if token.ent type != "":
        entities.append((token.text, token.ent type))
```

## 2. Job Compatibility Checking - Cosine Similarity Algorithm

1. Input Vectors: Given two vectors, A and B, each of dimension n.
2. Initialization: Initialize three variables: `dot_product` to 0, `magnitude_A` to 0, and `magnitude_B` to 0.
3. Compute Dot Product and Magnitudes:  
For each element i from 1 to n:
  - Compute the dot product: `dot_product += A[i] * B[i]`
  - Compute the magnitude of vector AA: `magnitude_A += (A[i])^2`
  - Compute the magnitude of vector BB: `magnitude_B += (B[i])^2`
4. Finalize Magnitudes: Take the square root of `magnitude_A` and `magnitude_B` to get the final magnitudes of vectors A and B:
  - `magnitude_A = sqrt(magnitude_A)`
  - `magnitude_B = sqrt(magnitude_B)`
5. Calculate Cosine Similarity: Divide the dot product by the product of the magnitudes of vectors A and B:
  - `cosine_similarity = dot_product / (magnitude_A * magnitude_B)`

13

## System Design - Design Model

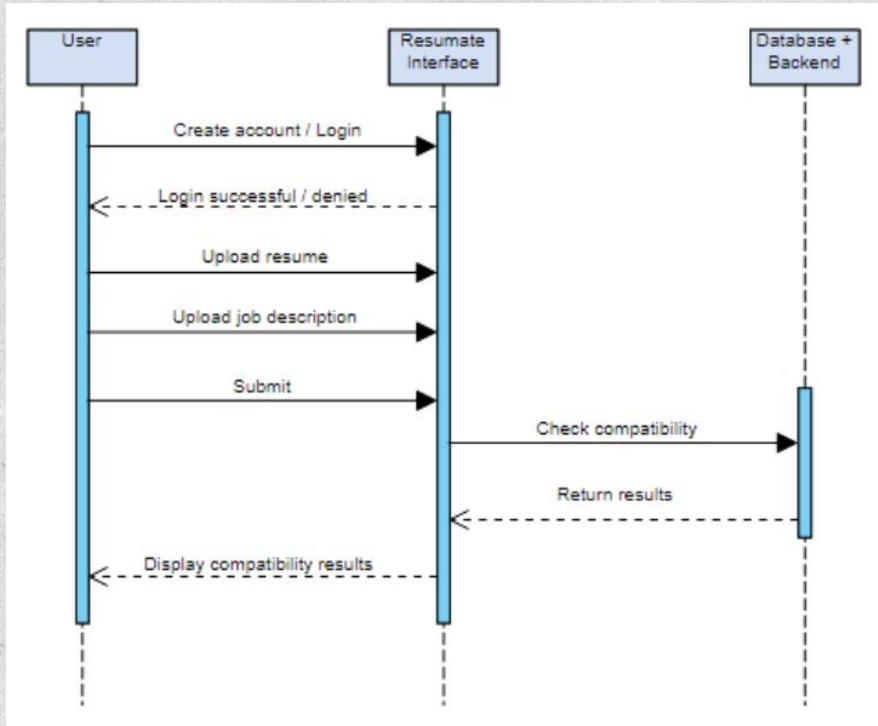


Fig 2: Sequence diagram of Resumate

14

# Datasets

Resume dataset from Huggingface - The dataset contains over 30,000 resumes stored in string format. One of the challenges faced while using this dataset is sorting and extracting resumes of job-seekers specifically in the IT field (Fig 3: Dataset in string format)

	input float64	Resume_test string
Data Science Job	null	Education Details MCA YMCAUST Faridabad Haryana Data Science internship Skill Details Data Structure Experience Less than 1 year months C Experience Less than 1 year months Data Analysis...
Project manager Job	null	Cyber Security Analyst Cyber span 1Securityspan span 1Analystspan San Diego CA Experienced analyst with strong attention to detail adept in problem solving and possesses intermediate...
Project manager Job	null	Quality Engineer Quality Engineer Alpharetta GA Work Experience Quality Engineer Ingenico Inc Alpharetta GA August 2013 to Present Responsibilities Create automation test case and plan...
Network Administrator Job	null	Technical Support Administrator Technical Support span 1Administratorspan New Egypt NJ Work Experience Technical Support Administrator Toll Brothers Inc Horsham PA April 2017 to Present...
Network Administrator Job	null	Associate Technology Risk Analyst Associate Technology Risk span 1Analystspan Associate Technology Risk Analyst Fidelity Investments Durham NC Work Experience Associate Technology...
Project manager Job	null	IT Security Analyst span 1ITspan span 1Securityspan span 1Analystspan Sr IT Security Analyst US EPA Office OCFO Alexandria VA Authorized to work in the US for any employer Work Experience IT...
Database Administrator Job	null	System Network Administrator Systemspan 1Networkspan span 1Administratorspan System Network Administrator San Jose CA A passionate technologist with over 3 years of experience seeking to...
Network Administrator Job	null	Network Research Engineer Network Research Engineer Washington DC Authorized to work in the US for any employer Work Experience Network Research Engineer Virginia Tech Blacksburg VA...

15

## Resumate: Resume Parser and Job Compatibility Checker

Resume dataset from Kaggle - This is a collection of Resume Examples taken from livecareer.com for categorizing a given resume into any of the labels defined in the dataset.

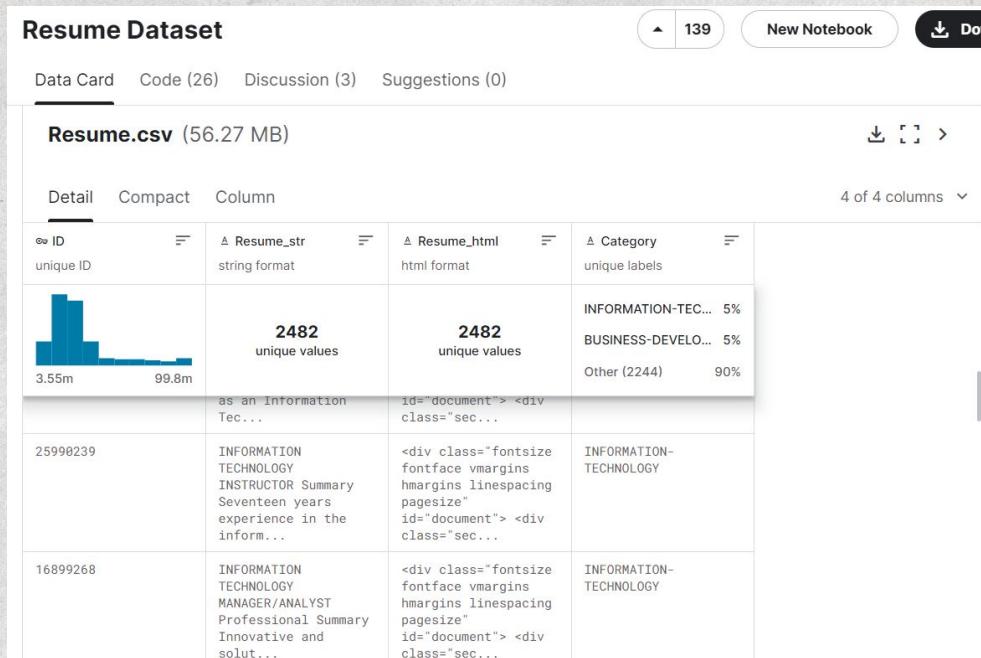


Fig 4: Resume dataset in csv file

16

# Work Division

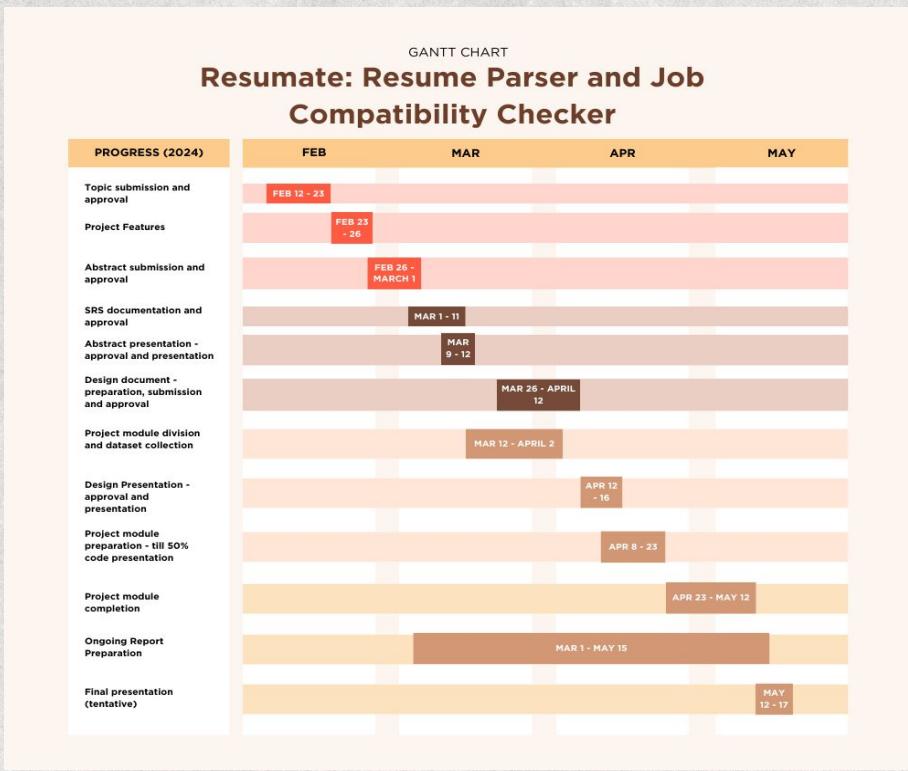


Fig 5: Gantt Chart detailing work division

17

*Resumate: Resume Parser and Job Compatibility Checker*

# Software Requirements

1

Flask (python framework) for web-app development

4

Python library for cosine similarity

2

SpaCy - entity extraction

5

MongoDB - database

3

BERT - advanced text analysis

6

Frontend - HTML, CSS and JavaScript

18

# Hardware Requirements

## Processor

A modern multi-core processor (e.g., Intel Core i5) for efficient code compilation and development tasks.

## RAM

A minimum 8GB of RAM to handle multiple development tools and applications simultaneously without slowdowns. However, 16 to 32GB would be recommended.

## Storage

A solid-state drive (SSD) with sufficient storage capacity (e.g., 256GB or higher) for storing project files, libraries, and development tools.

# Results

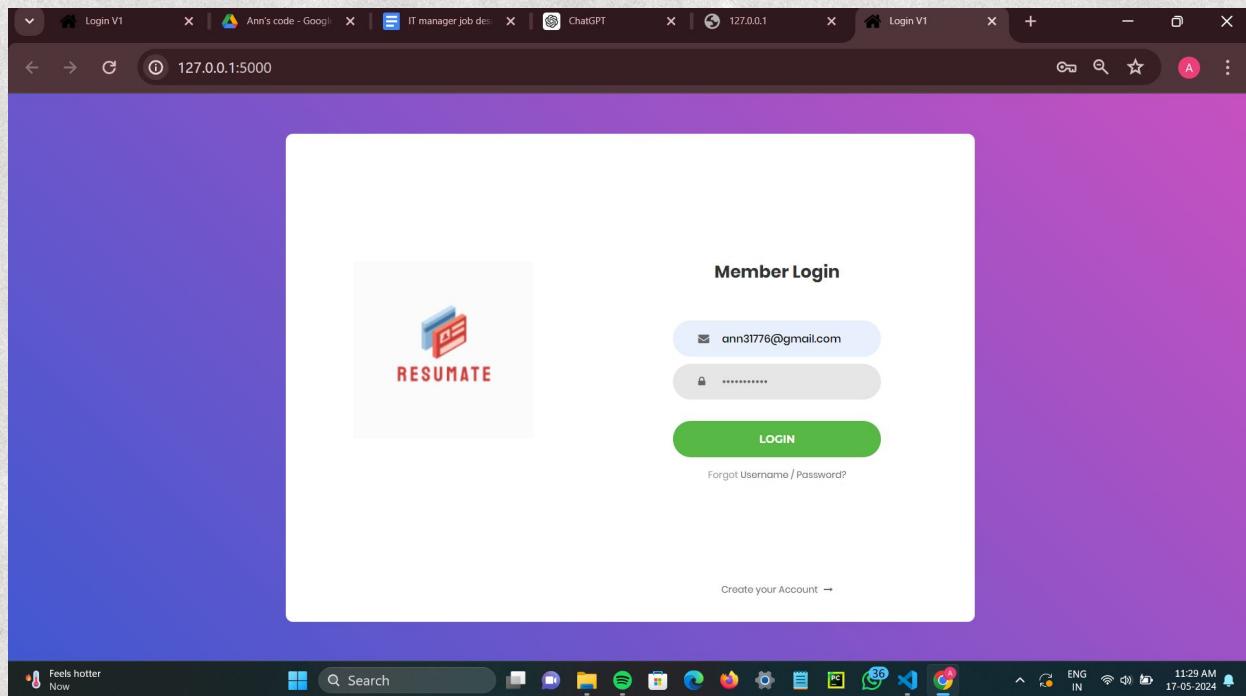


Fig 6: Login Page

## Resumate: Resume Parser and Job Compatibility Checker

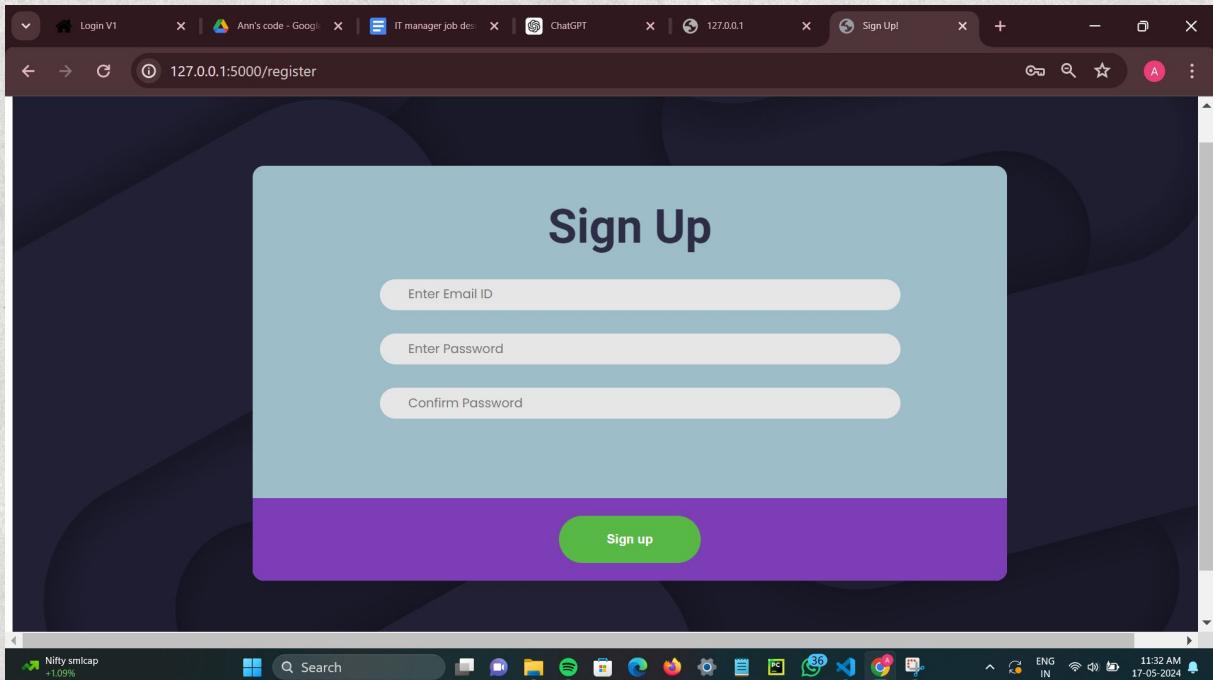


Fig 7: Sign up page

21

## Resumate: Resume Parser and Job Compatibility Checker

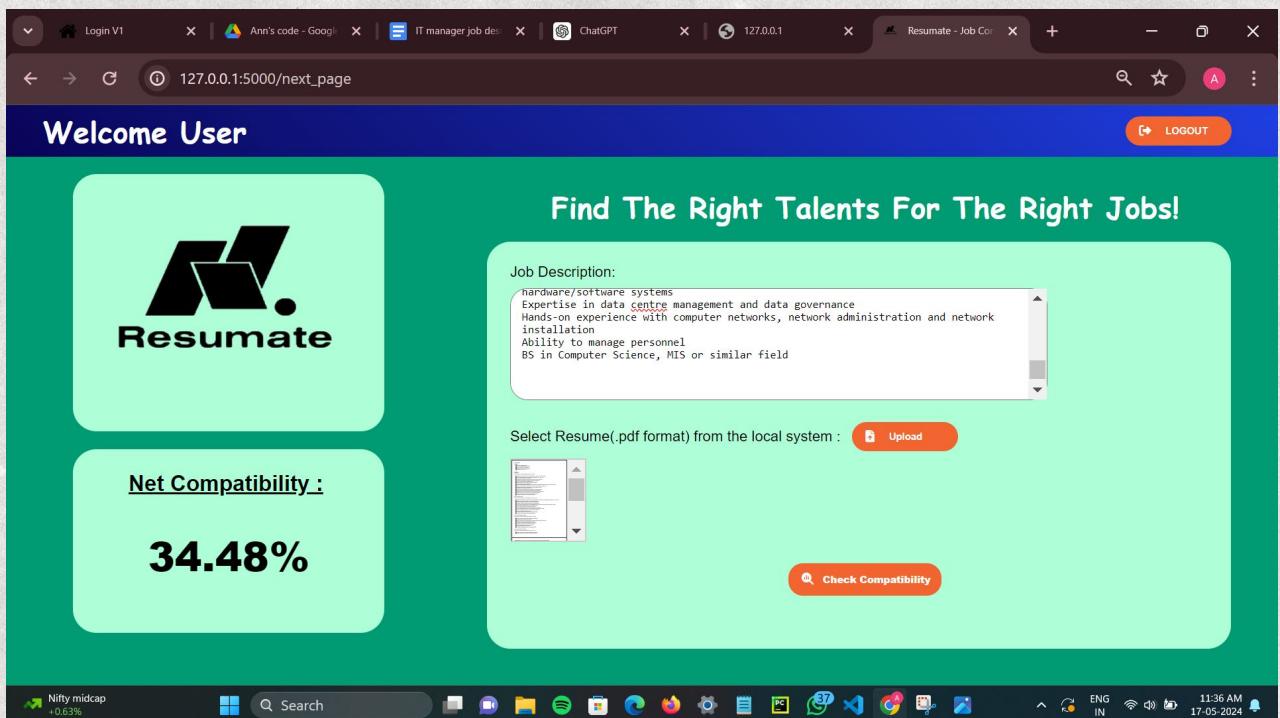


Fig 8: Main Page

22

## Resumate: Resume Parser and Job Compatibility Checker

MODEL 1

```
===== Results =====
TOK      100.00
NER P    69.35
NER R    52.84
NER F    59.98
SPEED    110

===== NER (per type) =
          P      R      F
Name      93.02  97.56  95.24
Designation 69.66  55.36  61.69
Location   76.32  68.24  72.05
Email Address 80.00  96.97  87.67
College Name 67.92  73.47  70.59
Degree     64.71  80.49  71.74
Skills      34.09  18.29  23.81
Companies worked at 71.76  36.75  48.61
Graduation Year 47.83  25.00  32.84
Years of Experience 100.00  33.33  50.00
```

MODEL 2

```
===== Results =====
TOK      100.00
NER P    85.59
NER R    87.21
NER F    86.39
SPEED    119

===== NER (per type) =
          P      R      F
LINKEDIN LINK  99.01  99.50  99.26
SKILLS        82.08  76.18  79.02
CERTIFICATION 81.44  83.76  82.58
NAME          93.00  93.47  93.23
LOCATION       79.47  83.43  81.40
COMPANIES WORKED AT 90.07  94.09  92.04
WORKED AS      87.41  90.82  89.08
YEARS OF EXPERIENCE 88.96  97.89  93.21
COLLEGE NAME   62.60  70.69  66.40
DEGREE         82.82  84.11  83.46
LANGUAGE        83.03  86.16  84.57
UNIVERSITY      79.14  73.83  76.39
YEAR OF GRADUATION 48.15  24.53  32.50
AWARDS          84.00  89.36  86.60
EMAIL ADDRESS   62.50  83.33  71.43
Unlabelled      0.00   0.00   0.00
```

Fig 9 and 10: Testing Results

23

## Resumate: Resume Parser and Job Compatibility Checker

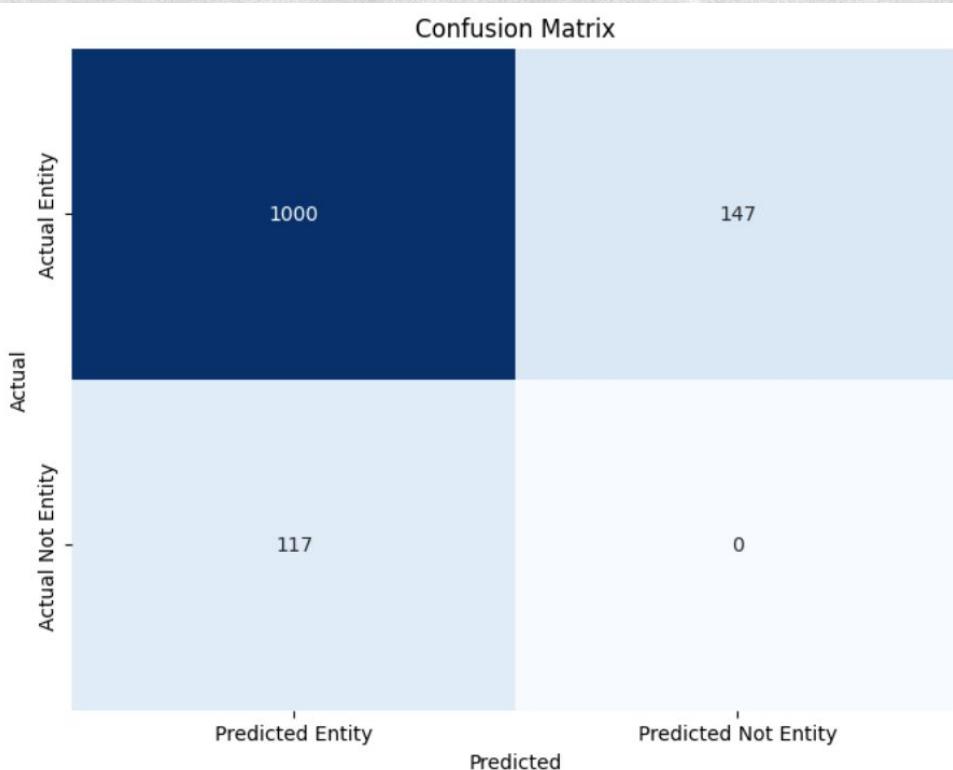


Fig 11:  
Confusion  
Matrix

24



## Conclusion

Resumate is a tool that can impact and improve the hiring process effectively. Its capability to parse resumes and extract pertinent data, as well as rate the skills and capabilities of a potential employee is a very simple and straightforward process that helps automate the recruitment to a certain degree.

25



## Future Enhancements

- Recommend skills to be acquired in order to increase job compatibility.
- Enable multiple resumes to be uploaded, compare them and show the best one among them.
- Resumes of various formats can be uploaded.

26



## References

- [1] Nirmiti Bhoir, Mrunmayee Jakate, Snehal Lavangare, et al. "Resume Parser using a hybrid approach to enhance the efficiency of Automated Recruitment Processes". Authorea. April 17, 2023.
- [2] Alkeshwar Jivtode, Kisan Jadhav, and Dipali Kandhare. "RESUME ANALYSIS USING MACHINE LEARNING AND NATURAL LANGUAGE PROCESSING." IRJETS. May 5, 2023.
- [3] D. Wang, J. Su and H. Yu, "Feature Extraction and Analysis of Natural Language Processing for Deep Learning English Language," in IEEE Access, vol. 8.
- [4] M. Soni, S. Gomathi and Y. Bhupendra Kumar Adhyaru, "Natural Language Processing for the Job Portal Enhancement," 2020 7th International Conference on Smart Structures and Systems (ICSSS), Chennai, India, 2020.
- [5] Abdul Wahab, Dr. M N. Nachappa. "Resume Parser with Natural Language Processing." International Research Journal of Engineering and Technology (IRJET) Volume: 09 Issue: 03, March 2022.
- [6] Nirali Bhaliya, Jay Gandhi, Dheeraj Kumar Singh. "NLP-based Extraction of Relevant Resume using Machine Learning." International Journal of Innovative Technology and Exploring Engineering (IJITEE), Volume-9 Issue-7, May 2020.



## **Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes**



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING  
RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY (AUTONOMOUS)  
RAJAGIRI VALLEY, KAKKANAD, KOCHI, 682039  
(Affiliated to APJ Abdul Kalam Technological University)**



## **Vision, Mission, Programme Outcomes and Course Outcomes**

### **Institute Vision**

To evolve into a premier technological institution, moulding eminent professionals with creative minds, innovative ideas and sound practical skill, and to shape a future where technology works for the enrichment of mankind.

### **Institute Mission**

To impart state-of-the-art knowledge to individuals in various technological disciplines and to inculcate in them a high degree of social consciousness and human values, thereby enabling them to face the challenges of life with courage and conviction.

### **Department Vision**

To become a centre of excellence in Computer Science and Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

### **Department Mission**

To inspire and nurture students, with up-to-date knowledge in Computer Science and Engineering, ethics, team spirit, leadership abilities, innovation and creativity to come out with solutions meeting societal needs.

## **Programme Outcomes (PO)**

Engineering Graduates will be able to:

- 1. Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
- 2. Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
- 3. Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
- 4. Conduct investigations of complex problems:** Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- 5. Modern Tool Usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
- 6. The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
- 7. Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- 8. Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- 9. Individual and Team work:** Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

**10. Communication:** Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports documentation. Make effective presentations, and give and receive clear instructions.

**11. Project management and finance:** Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

**12. Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

### **Programme Specific Outcomes (PSO)**

A graduate of the Computer Science and Engineering Program will demonstrate:

#### **PSO1: Computer Science Specific Skills**

The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

#### **PSO2: Programming and Software Development Skills**

The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry.

#### **PSO3: Professional Skills**

The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

## **Course Outcomes**

After the completion of the course the student will be able to:

### **CO1:**

Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)

### **CO2:**

Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)

### **CO3:**

Perform requirement analysis, identify design methodologies and develop adaptable reusable solutions of minimal complexity by using modern tools advanced programming techniques (Cognitive Knowledge Level: Apply)

### **CO4:**

Prepare technical report and deliver a presentation (Cognitive Knowledge Level: Apply)

### **CO5:**

Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)

## **Appendix C: CO-PO-PSO Mapping**



## COURSE OUTCOMES:

After completion of the course the student will be able to

<b>SL. NO</b>	<b>DESCRIPTION</b>	<b>Blooms' Taxonomy Level</b>
CO1	Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO2	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO3	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO4	Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO5	Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)	Level 3: Apply

## CO-PO AND CO-PSO MAPPING

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PS O3
C O1	3	3	3	3		2	2	3	2	2	2	3	2	2	2
C O2	3	3	3	3	3	2		3	2	3	2	3	2	2	2
C O3	3	3	3	3	3	2	2	3	2	2	2	3			2
C O4	2	3	2	2	2			3	3	3	2	3	2	2	2
C O5	3	3	3	2	2	2	2	3	2		2	3	2	2	2

3/2/1: high/medium/low

## JUSTIFICATIONS FOR CO-PO MAPPING

MAPPING	LOW/ MEDIUM/ HIGH	JUSTIFICATION
101003/CS6 22T.1-PO1	<b>HIGH</b>	Identify technically and economically feasible problems by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS6 22T.1-PO2	<b>HIGH</b>	Identify technically and economically feasible problems by analysing complex engineering problems reaching substantiated conclusions using first principles of mathematics.
101003/CS6 22T.1-PO3	<b>HIGH</b>	Design solutions for complex engineering problems by identifying technically and economically feasible problems.
101003/CS6 22T.1-PO4	<b>HIGH</b>	Identify technically and economically feasible problems by analysis and interpretation of data.
101003/CS6 22T.1-PO6	<b>MEDIUM</b>	Responsibilities relevant to the professional engineering practice by identifying the problem.
101003/CS6 22T.1-PO7	<b>MEDIUM</b>	Identify technically and economically feasible problems by understanding the impact of the professional engineering solutions.
101003/CS6 22T.1-PO8	<b>HIGH</b>	Apply ethical principles and commit to professional ethics to identify technically and economically feasible problems.
101003/CS6 22T.1-PO9	<b>MEDIUM</b>	Identify technically and economically feasible problems by working as a team.
101003/CS6 22T.1-PO10	<b>MEDIUM</b>	Communicate effectively with the engineering community by identifying technically and economically feasible problems.
101003/CS6 22T.1-P011	<b>MEDIUM</b>	Demonstrate knowledge and understanding of engineering and management principles by selecting the technically and economically feasible problems.
101003/CS6 22T.1-PO12	<b>HIGH</b>	Identify technically and economically feasible problems for long term learning.
101003/CS6 22T.1-PSO1	<b>MEDIUM</b>	Ability to identify, analyze and design solutions to identify technically and economically feasible problems.
101003/CS6 22T.1-PSO2	<b>MEDIUM</b>	By designing algorithms and applying standard practices in software project development and Identifying technically and economically feasible problems.
101003/CS6 22T.1-PSO3	<b>MEDIUM</b>	Fundamentals of computer science in competitive research can be applied to Identify technically and economically feasible problems.
101003/CS6 22T.2-PO1	<b>HIGH</b>	Identify and survey the relevant by applying the knowledge of mathematics, science, engineering fundamentals.

101003/CS6 22T.2-PO2	<b>HIGH</b>	Identify, formulate, review research literature, and analyze complex engineering problems get familiarized with software development processes.
101003/CS6 22T.2-PO3	<b>HIGH</b>	Design solutions for complex engineering problems and design based on the relevant literature.
101003/CS6 22T.2-PO4	<b>HIGH</b>	Use research-based knowledge including design of experiments based on relevant literature.
101003/CS6 22T.2-PO5	<b>HIGH</b>	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes by using modern tools.
101003/CS6 22T.2-PO6	<b>MEDIUM</b>	Create, select, and apply appropriate techniques, resources, by identifying and surveying the relevant literature.
101003/CS6 22T.2-PO8	<b>HIGH</b>	Apply ethical principles and commit to professional ethics based on the relevant literature.
101003/CS6 22T.2-PO9	<b>MEDIUM</b>	Identify and survey the relevant literature as a team.
101003/CS6 22T.2-PO10	<b>HIGH</b>	Identify and survey the relevant literature for a good communication to the engineering fraternity.
101003/CS6 22T.2-PO11	<b>MEDIUM</b>	Identify and survey the relevant literature to demonstrate knowledge and understanding of engineering and management principles.
101003/CS6 22T.2-PO12	<b>HIGH</b>	Identify and survey the relevant literature for independent and lifelong learning.
101003/CS6 22T.2-PSO1	<b>MEDIUM</b>	Design solutions for complex engineering problems by Identifying and survey the relevant literature.
101003/CS6 22T.2-PSO2	<b>MEDIUM</b>	Identify and survey the relevant literature for acquiring programming efficiency by designing algorithms and applying standard practices.
101003/CS6 22T.2-PSO3	<b>MEDIUM</b>	Identify and survey the relevant literature to apply the fundamentals of computer science in competitive research.
101003/CS6 22T.3-PO1	<b>HIGH</b>	Perform requirement analysis, identify design methodologies by using modern tools & advanced programming techniques and by applying the knowledge of mathematics, science, engineering fundamentals.
101003/CS6 22T.3-PO2	<b>HIGH</b>	Identify, formulate, review research literature for requirement analysis, identify design methodologies and develop adaptable & reusable solutions.

101003/CS6 22T.3-PO3	<b>HIGH</b>	Design solutions for complex engineering problems and perform requirement analysis, identify design methodologies.
101003/CS6 22T.3-PO4	<b>HIGH</b>	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS6 22T.3-PO5	<b>HIGH</b>	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.
101003/CS6 22T.3-PO6	<b>MEDIUM</b>	Perform requirement analysis, identify design methodologies and assess societal, health, safety, legal, and cultural issues.
101003/CS6 22T.3-PO7	<b>MEDIUM</b>	Understand the impact of the professional engineering solutions in societal and environmental contexts and Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS6 22T.3-PO8	<b>HIGH</b>	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions by applying ethical principles and commit to professional ethics.
101003/CS6 22T.3-PO9	<b>MEDIUM</b>	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS6 22T.3-PO10	<b>MEDIUM</b>	Communicate effectively with the engineering community and with society at large to perform requirement analysis, identify design methodologies.
101003/CS6 22T.3-PO11	<b>MEDIUM</b>	Demonstrate knowledge and understanding of engineering requirement analysis by identifying design methodologies.
101003/CS6 22T.3-PO12	<b>HIGH</b>	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by analysis, identify design methodologies and develop adaptable & reusable solutions.
101003/CS6 22T.3-PSO3	<b>MEDIUM</b>	The ability to apply the fundamentals of computer science in competitive research and prior to that perform requirement analysis, identify design methodologies.
101003/CS6 22T.4-PO1	<b>MEDIUM</b>	Prepare technical report and deliver presentation by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS6 22T.4-PO2	<b>HIGH</b>	Identify, formulate, review research literature, and analyze complex engineering problems by preparing technical report and deliver presentation.

101003/CS6 22T.4-PO3	<b>MEDIUM</b>	Prepare Design solutions for complex engineering problems and create technical report and deliver presentation.
101003/CS6 22T.4-PO4	<b>MEDIUM</b>	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions and prepare technical report and deliver presentation.
101003/CS6 22T.4-PO5	<b>MEDIUM</b>	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and Prepare technical report and deliver presentation.
101003/CS6 22T.4-PO8	<b>HIGH</b>	Prepare technical report and deliver presentation by applying ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
101003/CS6 22T.4-PO9	<b>HIGH</b>	Prepare technical report and deliver presentation effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
101003/CS6 22T.4-PO10	<b>HIGH</b>	Communicate effectively with the engineering community and with society at large by prepare technical report and deliver presentation.
101003/CS6 22T.4-PO11	<b>MEDIUM</b>	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work by prepare technical report and deliver presentation.
101003/CS6 22T.4-PO12	<b>HIGH</b>	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by prepare technical report and deliver presentation.
101003/CS6 22T.4-PSO1	<b>MEDIUM</b>	Prepare a technical report and deliver presentation to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas.
101003/CS6 22T.4-PSO2	<b>MEDIUM</b>	To acquire programming efficiency by designing algorithms and applying standard practices in software project development and to prepare technical report and deliver presentation.
101003/CS6 22T.4-PSO3	<b>MEDIUM</b>	To apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs by preparing technical report and deliver presentation.
101003/CS6 22T.5-PO1	<b>HIGH</b>	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
101003/CS6 22T.5-PO2	<b>HIGH</b>	Identify, formulate, review research literature, and analyze complex engineering problems by applying engineering and management principles to achieve the goal of the project.

101003/CS6 22T.5-PO3	<b>HIGH</b>	Apply engineering and management principles to achieve the goal of the project and to design solutions for complex engineering problems and design system components or processes that meet the specified needs.
101003/CS6 22T.5-PO4	<b>MEDIUM</b>	Apply engineering and management principles to achieve the goal of the project and use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
101003/CS6 22T.5-PO5	<b>MEDIUM</b>	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO6	<b>MEDIUM</b>	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities by applying engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO7	<b>MEDIUM</b>	Understand the impact of the professional engineering solutions in societal and environmental contexts, and apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO8	<b>HIGH</b>	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice and to use the engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO9	<b>MEDIUM</b>	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO11	<b>MEDIUM</b>	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PO12	<b>HIGH</b>	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PSO1	<b>MEDIUM</b>	The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas. Apply engineering and management principles to achieve the goal of the project.

101003/CS6 22T.5-PSO2	<b>MEDIUM</b>	The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry and to apply engineering and management principles to achieve the goal of the project.
101003/CS6 22T.5-PSO3	<b>MEDIUM</b>	The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur and apply engineering and management principles to achieve the goal of the project.

