

CV ANALYSIS AND SCREENING SOFTWARE

UCS503 Software Engineering Project Report

Mid-Semester Evaluation

Submitted by:

101903033 MEGHNA RAI Offline

101903035 BHAWNA KHOSLA Offline

101903038 YASH DHAWAN Offline

101903039 HIMANSHI KHAROTIA Offline

BE Third Year, CoE/CoE MBA

Group No: 2

Submitted to:

Ms.Arwinder Dhillon



Computer Science and Engineering Department, TIET, Patiala

October 2021

TABLE OF CONTENTS

S.No.	Assignment	Page No.
1.	Project Selection Phase	
1.1	Software Bid	3
1.2	Project Overview	4
2.	Analysis Phase	
2.1	Use Cases	4
2.1.1	Use-Case Diagrams	
2.2	Activity diagram and Swimlane Diagrams	5
2.3	Data Flow Diagrams (DFDs)	6
2.3.1	DFD Level 0	6
2.3.2	DFD Level 1	6
2.3.2	DFD Level 2	7
2.4	Software Requirement Specification in IEEE Format	8
3.	Design Phase	
3.1	Class Diagram	18
3.2	Sequence Diagram	19
3.3	Collaboration Diagram	20
3.4	Database Design: ER Diagram	20
4.	Prototyping-GUIs	
4.1	Screen Shots	21

1. PROJECT SELECTION PHASE

1.1 Software Bid

Team Name: Team 404

Team ID (will be assigned by Instructor): 2

Please enter the names of your Preferred Team Members.

- You are required to form a **three to four person** teams
- Choose your team members wisely. You will not be allowed to change teams.

Name	Roll No	Project Experience	Programming Language used
Meghna Rai	101903033	Hotel Management and Room Allocation System	JAVA
Bhawna Khosla	101903035	Clipboard Application Landing Page	HTML/CSS
Yash Dhawan	101903038	Blog Website	Node JS, JavaScript, EJS
Himanshi Kharotia	101903039	Website made using HTML/CSS and JS	HTML/CSS, JavaScript

Programming Language / Environment Experience

List the languages you are most comfortable developing in, **as a team**, in your order of preference. Many of the projects involve Java or C/C++ programming.

1. Python
2. C++/JAVA
3. JavaScript

Choices of Projects:

Please select **4 projects** your team would like to work on, by order of preference: *[Write at-least one paragraph for each choice (motivation, reason for choice, feasibility analysis, etc.)]*

First Choice	CV Analysis Application: Through this project we aim to develop a system which will help a company efficiently shortlist and filter suitable candidates for their organization by analyzing their CVs on the basis of their requirements, by using ML algorithms.
Second Choice	E-Commerce Website: Through this website, made using HTML/CSS and JavaScript, we aim to improve the market presence of an organization by creating a portal which is not only efficient and easy to use but will also help the organization reach out to its target audience easily.
Third Choice	Credit Card Fraud Detection Application: The aim of this project is to build a classifier that can detect credit card fraudulent transactions. We will use a variety of machine learning algorithms that will be able to discern fraudulent from non-fraudulent one.
Fourth Choice	Airport Management System: This system will help in centralizing all the data of a particular airport (including passenger details, flight details etc.) and creating an interface which allows easy and efficient access of the data by using RDMS.

1.1 Project Overview

This CV Analysis and Screening Software is a Machine Learning and Natural Language Processing based application.

In today's day and age, selecting the right candidate for an organization through the recruitment process, from a vast pool of candidates, has become a fundamental issue. The traditional norm has been to conduct various personality and technical evaluation tests, which can be extremely time-consuming and can lead to selection of an unfit candidate.

Through this software, we aim to create a system which will provide an effective way to short list submitted candidate CVs from a large number of applicants and help in providing a consistent and fair CV ranking policy. It will rank the CVs according to the candidate's experience and key skills required for particular job position.

This system will ultimately help reduce the workload of the hiring department and select the right candidate for the desired job profile, which in turn provide the expert for the organization.

2. ANALYSIS PHASE

2.1 Use Cases

2.1.1 Use Case Diagrams

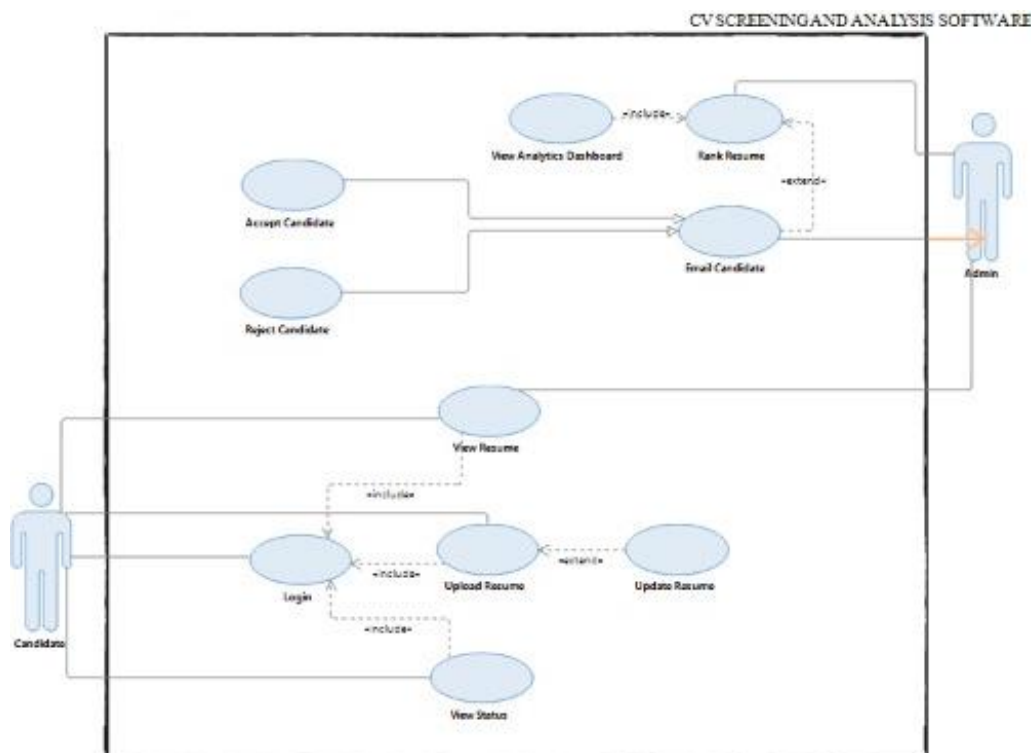


Fig 1: Use Case Diagram

Upload CV, View Status, Update CV, Rank Resume, Email Candidate, and View Analytics Dashboard along with the associations between these use cases.

2.2 Activity diagram and Swimlane Diagrams

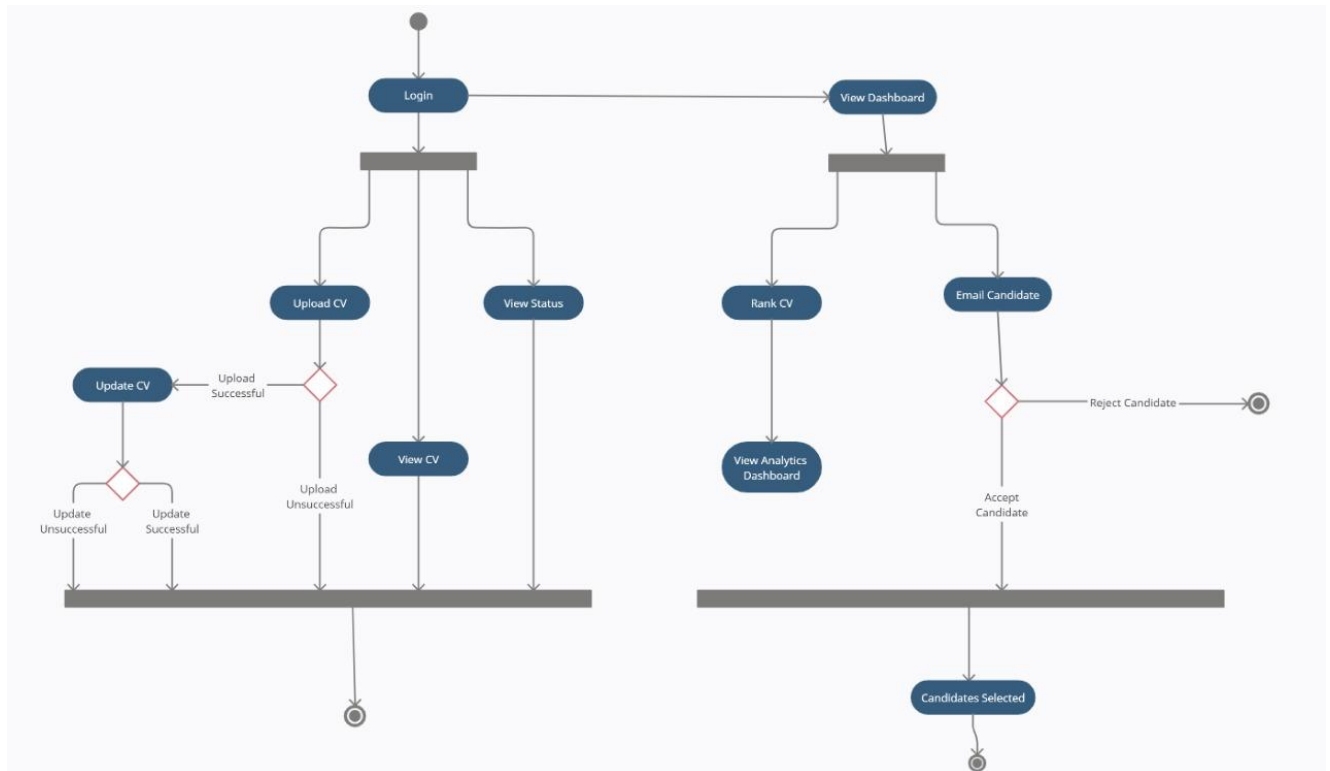


Fig 2: Activity Diagram

This diagram captures the dynamic behavior of the system. It is used to show message flow from one activity to another. The user logs in and can either upload or view his CV. He also has the option to View his status at this stage. If the upload of CV is successful, he can either update it or end the session. For the recruiter, he has the option to rank the CVs using the ML algorithm. Once he has ranked them, he can also email the candidates that he would like to select or reject. The entire session ends with the recruiter selecting the required number of candidates. Following is the swimlane diagram for the same.

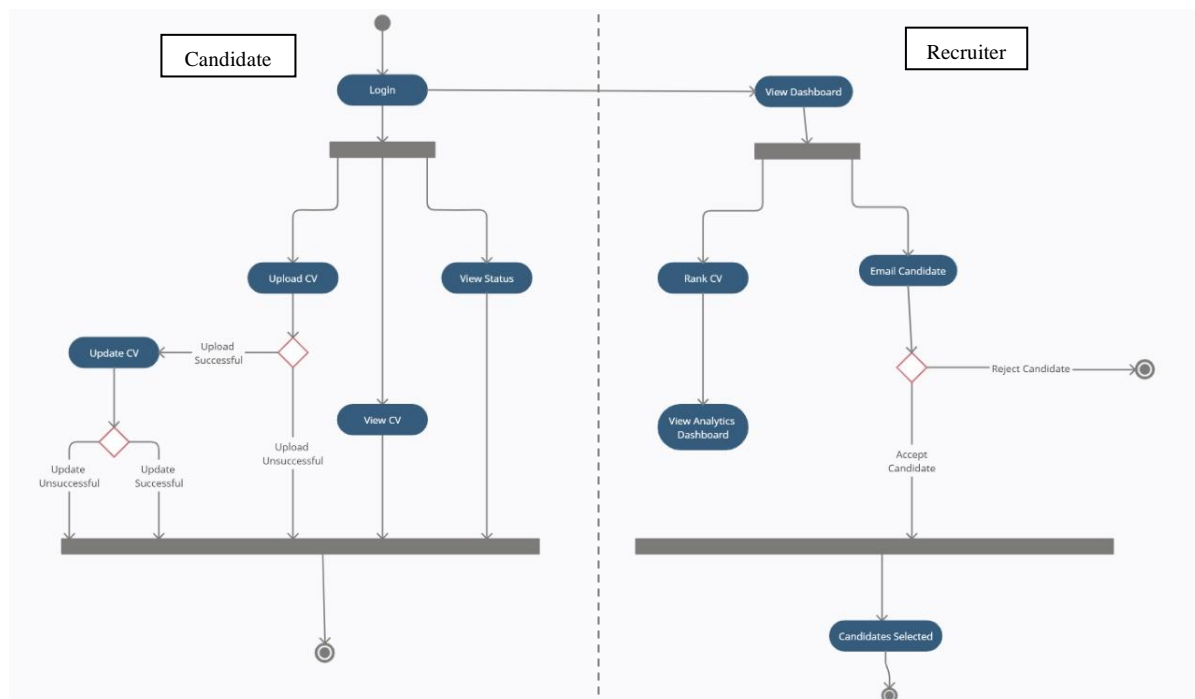


Fig 3: Swimlane Diagram

2.3 Data Flow Diagrams (DFDs)

2.3.1 DFD Level 0

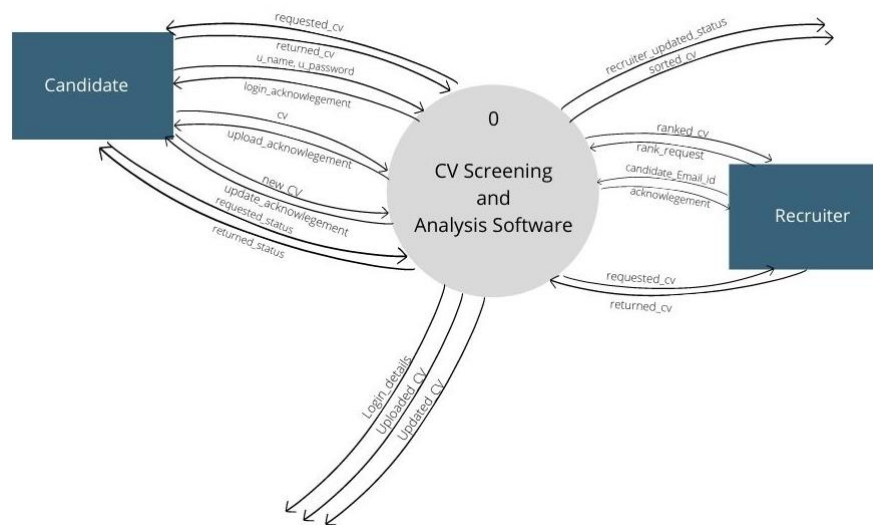


Fig 4: Level 0 Data Flow Diagram

This is the Level 0 Data Flow Diagram which basically includes all the data which will be sent to the system from the candidate's and recruiter's side and the corresponding information which will be sent to them. It also depicts the data which will be sent to the database.

2.3.2 DFD Level 1

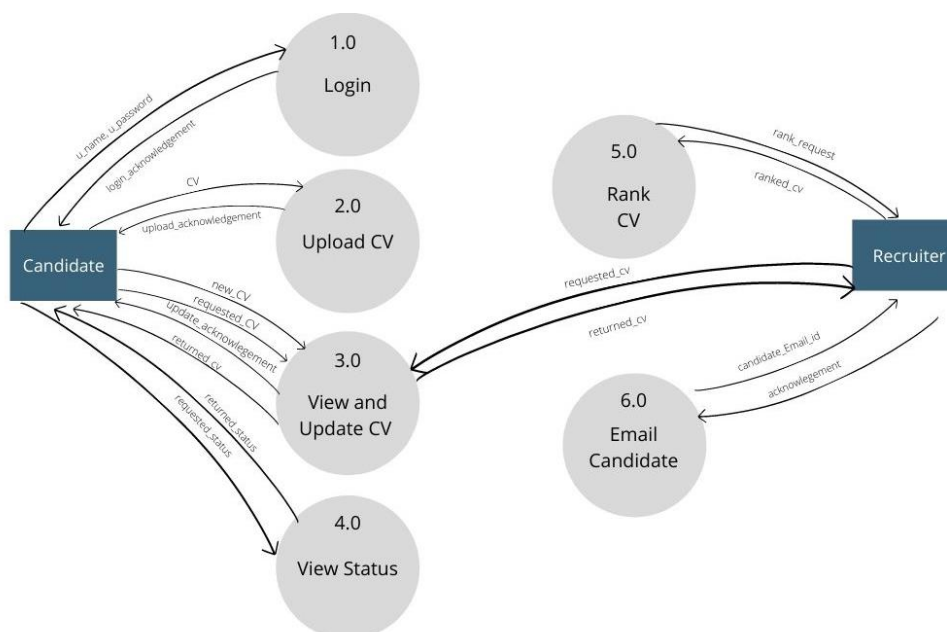


Fig 5: Level 1 Data Flow Diagram

This is an extension of the Level 1 DFD where we have specified the particular functions to which the data is sent and received from.

2.3.3 DFD Level 2

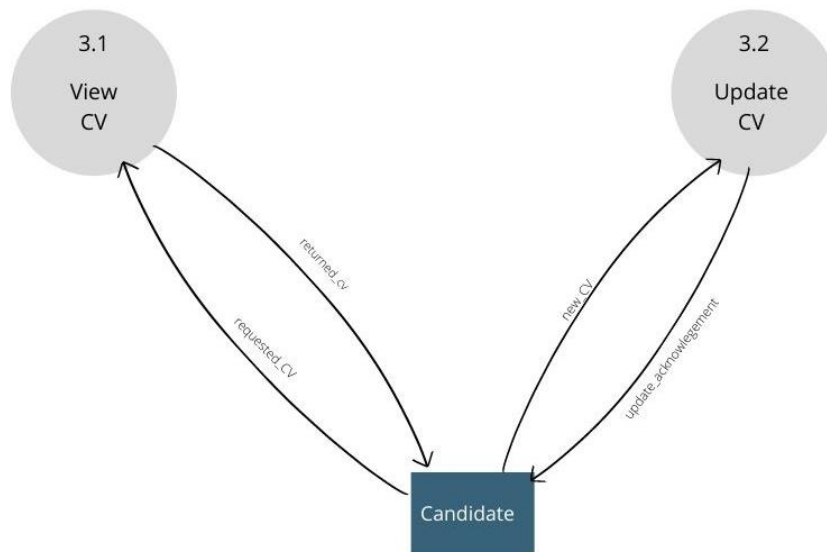


Fig 6: Level 1 Data Flow Diagram

This is a further extension of the Level 2 DFD in which we have extended the View and Update CV function and have specified the data which will be sent and received from these two functions.

2.4 Software Requirement Specification in IEEE Format

Software Requirements Specification for CV Analysis and Screening

Software

Version 1.0 approved

Prepared by

Meghna Rai	101903033
Bhawna Khosla	101903035
Yash Dhawan	101903038
Himanshi Kharotia	101903039

Team 404

28 September 2021

Table of Contents

Table of Contents	9
Revision History	9
1. Introduction	10
1.1 Purpose	10
1.2 Document Conventions	10
1.3 Intended Audience and Reading Suggestions	10
1.4 Project Scope	10
1.5 References.....	10
2. Overall Description	11
2.1 Product Perspective	11
2.2 Product Features.....	11
2.4 Design and Implementation Constraints.....	12
2.5 User Documentation.....	12
2.6 Assumptions and Dependencies.....	12
3. System Features	12
3.1 System Feature 1 for Login	12
3.2 System Feature 2 for Upload CV	13
3.3 System Feature for Update CV	13
3.4 System Feature for View CV	13
3.5 System Feature for View Status.....	13
3.6 System Feature for Rank CV	14
3.7 System Feature for Email Candidate	14
4. External Interface Requirements	14
4.1 User Interfaces	14
4.2 Software Interfaces.....	15
4.3 Communications Interfaces	15
5. Other Non-functional Requirements	15
5.1 Performance Requirements.....	15
5.2 Safety Requirements.....	15
5.3 Security Requirements.....	15
5.4 Software Quality Attributes	16
Appendix A: Glossary	16
Appendix B: Analysis Models	16
Appendix C: Issues List	Error! Bookmark not defined.

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

This CV Analysis and Screening Software is a Machine Learning and Natural Language Processing based application. In today's day and age, selecting the right candidate for an organization through the recruitment process, from a vast pool of candidates, has become a fundamental issue. The traditional norm has been to conduct various personality and technical evaluation tests, which can be extremely time-consuming and can lead to selection of an unfit candidate.

Through this software, we aim to create a system which will provide an effective way to short list submitted candidate CVs from a large number of applicants and help in providing a consistent and fair CV ranking policy. It will rank the CVs according to the candidate's experience and key skills required for particular job position. This system will help reduce the workload of the hiring department and select the right candidate for the desired job profile, which in turn provide the expert for the organization.

1.2 Document Conventions

The standards and typographical conventions that have been followed while creating the SRS are, the usage of the Times New Roman font with font size 14 for the headings and 12 for the rest of the text. For highlighting text having special significance and importance, the bold feature has been made use of.

1.3 Intended Audience and Reading Suggestions

This document is intended for all those people who are interested in availing and creating a software which can assist them in their hiring process and automating the same. It is specifically created for the HR and Hiring department of any organization and the applicants who would like to apply for those organizations. The suggested sequence for reading this document is the same as that mentioned in the Table of Contents.

1.4 Project Scope

The aim of this software is to create a system which improves the current recruitment process and makes it more efficient and less time consuming. With the help of this system, the recruiters will have all the typical recruitment facilities and features at their disposal. All the general staffing processes will be easily automated and can be closely monitored with great ease. This application is specifically designed for the hiring department of various organizations and the candidates who would like to work at these organizations and apply for them.

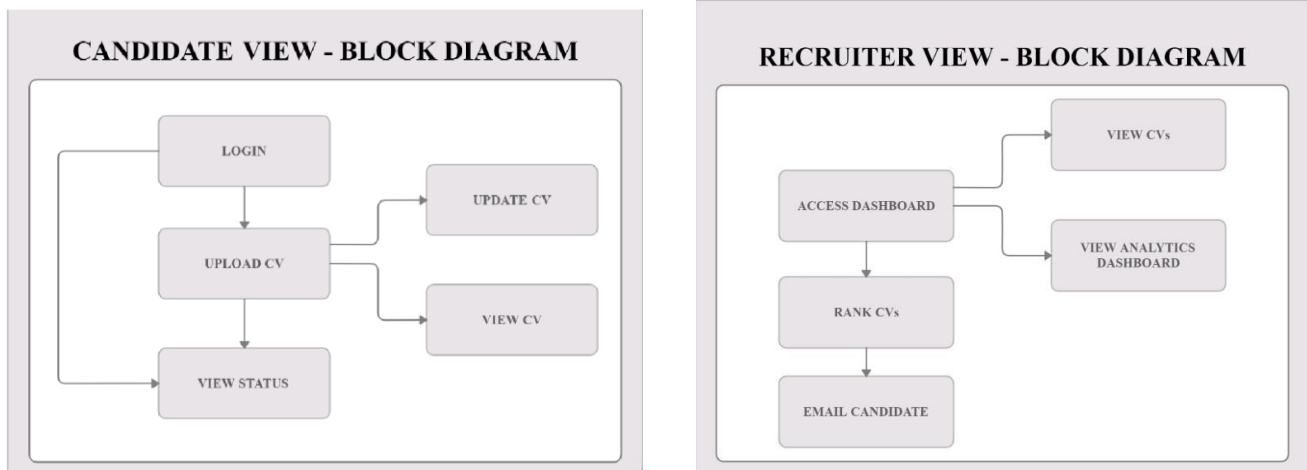
1.5 References

- [1] A. Robey, K. Shukla and K. Agarwal, "Personality Prediction System through CV Analysis," *International Research Journal of Engineering and Technology (IRJET)*, p. 4, 2019.
- [2] B. T. J. A. M. Abdulazeez, "Classification Based on Decision Tree Algorithm for Machine Learning," *Journal of Applied Science and Technology Trends (JASTT)*, p. 10, 2021.

- [3] A. S. Khan, H. Ahmad and M. Z. Asghar, "Personality Classification from Online Text using Machine Learning Approach," *International Journal of Advanced Computer Science and Applications (IJACSA)*, p. 17, 2020.

2. Overall Description

2.1 Product Perspective



2.2 Product Features

The features and functions that are provided by this software or product are as follows:

1. Login
2. Upload CV
3. Update CV
4. View CV
5. View Status
6. Rank Resume
7. View Analytics Dashboard
8. Email Candidate

2.3 Operating Environment

The environment in which this software will operate and function, including the hardware platform, operating system and versions are as follows:

1. Operating System Version:

- Windows: Microsoft® Windows® 7/8/10/11 (32- or 64-bit)
- Mac: Mac® OS X® 10.10 (Yosemite) or higher, up to 10.14 (macOS Mojave)
- Linux: GNOME or KDE desktop

2. Random Access Memory (RAM):

- 4 GB RAM minimum; 8 GB RAM recommended.

3. Free digital storage:

- 2 GB of available digital storage minimum, 4 GB Recommended

2.4 Design and Implementation Constraints

The items or issues that will limit the options available to developers/users and the related design and implementation constraints:

1. Incompatibility of the hardware being used by the developers/users with the hardware that is required for the functioning of the software
2. Insufficient memory of the system
3. Unavailability of the required number of the CVs in the software for the proper functioning of the ML algorithm to rank the CVs appropriately
4. Unavailability of the communication protocols (SMTP, POP etc.) within the system
5. Not having the appropriate software for the connectivity of the backend and frontend of the system

2.5 User Documentation

The different user documentation components that will be delivered along with the software in order to assist the user to operate the software easily and without any hurdles will be the Software Requirements Specification, which contains all the in-depth details of the features which will be provided by the system as well as a GUI Interface List to show a pictorial representation of how the system has to be used.

2.6 Assumptions and Dependencies

The assumption or dependencies that could affect the requirements which have been stated in the SRS and could negatively affect the system if these assumptions are incorrect, are not shared, or change as follows:

1. The user will have a pre-assigned ID and Password which will be shared with them through their email address during registration.
2. The recruiter will have all rights to access the Recruiter Dashboard without any authentication or need of ID and password.
3. We will have a set number of CVs already present in the database so that the system functions smoothly and the functioning can be explained easily.
4. This system will be used for the ranking of candidates in field of Software Development. Hence, the ranking will be done according to SDE specific requirements.

3. System Features

3.1 System Feature 1 for Login

Purpose	This screen provides the user interface to register new users into the system.
Inputs	A candidate needs to enter his personal details to register into the system on the registration page given on welcome screen.
Processing	The details will get stored on MYSQL DB by SQL query on the backend side.
Outputs	If required fields are not filled or the store id is not valid then it will give an error message else the user will be registered successfully.

3.2 System Feature 2 for Upload CV

Purpose	This screen provides a user interface where we want candidate to upload their CV.
Inputs	The user needs to Uploads their CV.
Processing	The CVs will get stored on MYSQL DB by SQL query on the backend side.
Outputs	If login credentials are valid then Candidate can upload their CV.

3.3 System Feature for Update CV

Purpose	This screen provides a user interface to update their CV
Inputs	The user needs to Uploads their Updated CV.
Processing	The Updated CVs will get stored on MYSQL DB by SQL query on the backend side.
Outputs	If login credentials are valid then Candidate can update their CV.

3.4 System Feature for View CV

Purpose	This screen provides a user interface to view their CV.
Inputs	The user needs to click View Button to view their CV.
Processing	The Uploaded CV retrieve from the data base and showed on the screen
Outputs	If user then Candidate can update their CV.

3.5 System Feature for View Status

Purpose	This screen provides a candidate to view their status either applicant received a mail or not.
Inputs	The user needs to click on View Status option to view their current status.

Processing	The candidate will get viewed status if recruiter approved him using the email candidate option. Then by communication protocols, user will get the mail.
Outputs	A dialogue box will appear where candidate can see either his CV get approved or not.

3.6 System Feature for Rank CV

Purpose	This screen gives recruiter an option to view candidate's Ranking on basis of algorithm.
Inputs	The recruiter needs to click on Rank Resume Option .
Processing	The Candidates CV will get ranked according to our algorithm i.e. CV analysis and Screening Software.
Outputs	Ranked CVs will be displayed to recruiter.

3.7 System Feature for Email Candidate

Purpose	This screen gives recruiter an option to email candidate based on Ranking.
Inputs	The recruiter needs to click the Email Candidate Option.
Processing	Recruiter sends the mail using communication protocol i.e. SMTP, POP.
Outputs	When Recruiter sends the email, the view status of candidate will be updated.

4. External Interface Requirements

4.1 User Interfaces

The User interface offered by our application is very easy to access, understandable and user-friendly. It features a login screen where the candidate has to enter his/her Email address and password. Once the candidate logs in, he/she would redirect to the page where they can upload their CV. Options of update CV and view CV appears after uploading the CV, clicking on these options candidate can edit and view their previously uploaded CV. Additionally, there is an option of view status which would help the candidate to view his/her CV's status. Thus, even a candidate is using our application for first time can comfortably explore all its features easily.

4.2 Software Interfaces

The various software components, tools, libraries, and integrated commercial components required for this system to function smoothly over any device are as follows:

1. Textract version 1.6.3
2. Requests version 2.22.0
3. Flask version 1.1.1
4. Gensim version 3.8.0
5. Sklearn version 0.0
6. PyPDF2 version version.26.0
7. Autocorrect version 0.4.4
8. Nltk version 3.4.5
9. Contractions version 0.0.21
10. Textsearch version 0.0.17
11. Inflect version 2.1.0
12. Numpy version 1.17.2
13. Pandas version 1.3.3
14. Pdfminer.six version 20181108
15. Python 3.6.0 |Anaconda 4.3.0 (64-bit)

4.3 Communications Interfaces

For the purpose of communication and exchange of information between the users of the product we will require certain e-mail communication protocols like, the three common protocols used to deliver email over the Internet: Simple Mail Transfer Protocol (**SMTP**), the Post Office Protocol (**POP**), and the Internet Message Access Protocol (**IMAP**).

5. Other Non-functional Requirements

5.1 Performance Requirements

The performance requirements for the product under various circumstances are as follows:

1. The software can run on any browser and on any operating system.
2. Multiple users can access the website at same time.
3. Only textual information will be handled by the website.
4. SQL queries takes less time in execution; therefore, website can perform very well for multiple user access modes

5.2 Safety Requirements

To ensure the safety of the data stored in the database of our system, it is vital to ensure that the authentication of the users is carried out in a proper manner. The data must be end-to-end encrypted and should be resistant and immune to various database attacks.

5.3 Security Requirements

All the system data should be secured and be encrypted with minimum needs so that it's protected from outside environment also from internal attack.

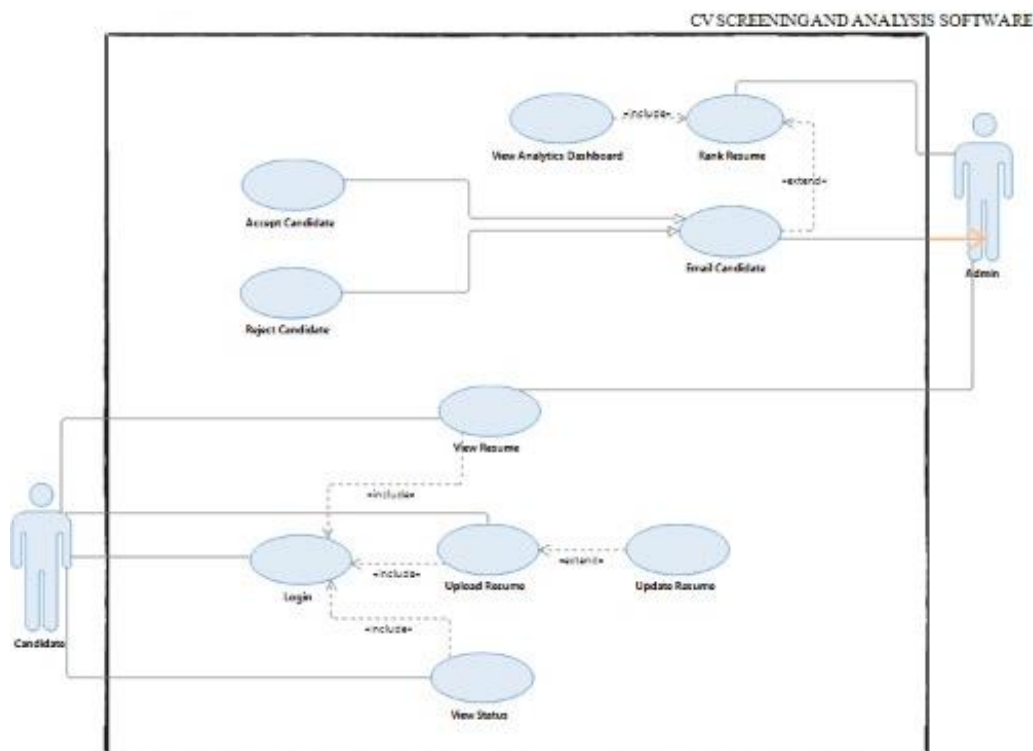
5.4 Software Quality Attributes

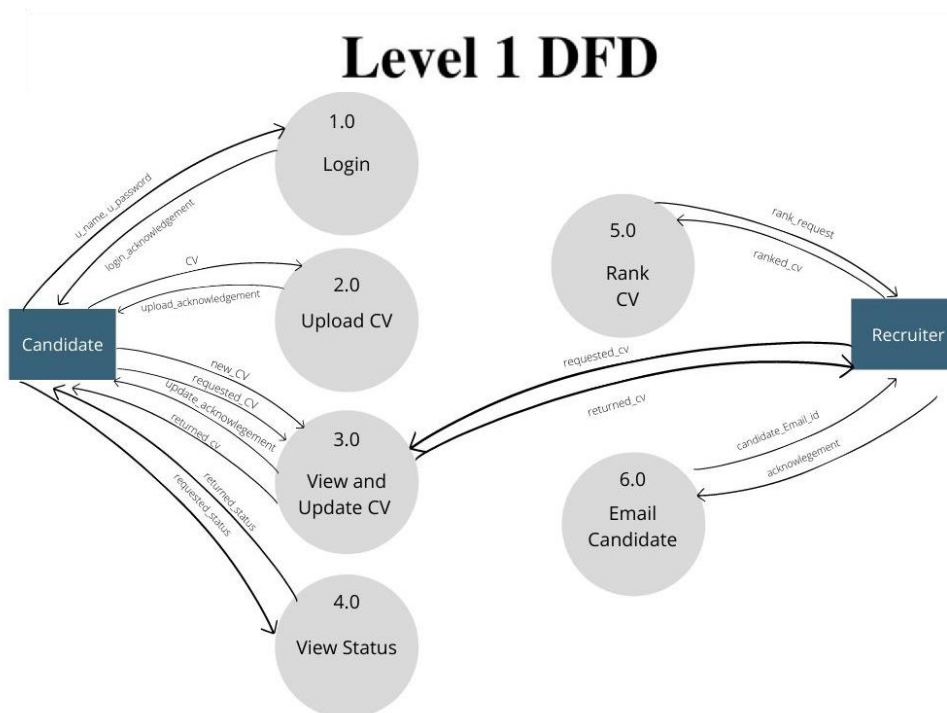
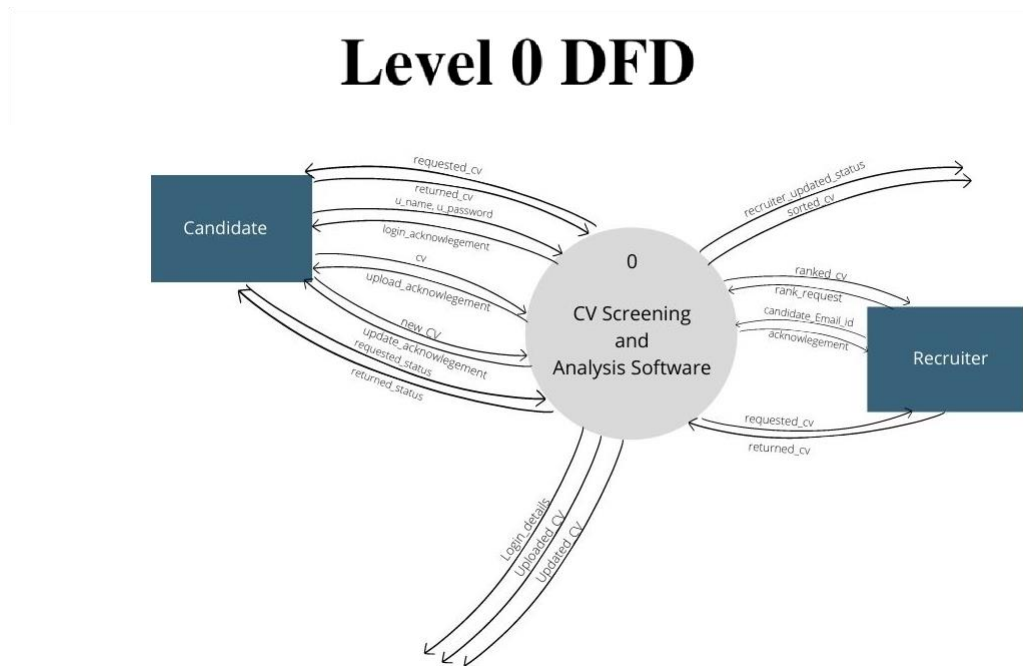
1. **Adaptability:** The software must be adaptable and should function smoothly in any operating environment that is supported by the system.
2. **Scalability:** The system should be able to adopt itself to increased usage or able to handle more data as time progress.
3. **Reliability:** The system should be reliable to perform the business that is, when user perform some important action, it should be acknowledged with confirmation.
4. **Usability:** The software must be easy to use and the user must be able to understand the flow of system easily that is, users should able to use software without any guidelines or help from experts/manuals.
5. **Responsiveness:** System should be responsive to the user input or to any external interrupt which is of higher priority and return back to same state. For example: When the system gets interrupted by call, then it should be able to save state and return to same state/page which was there before it got interrupted.

Appendix A: Glossary

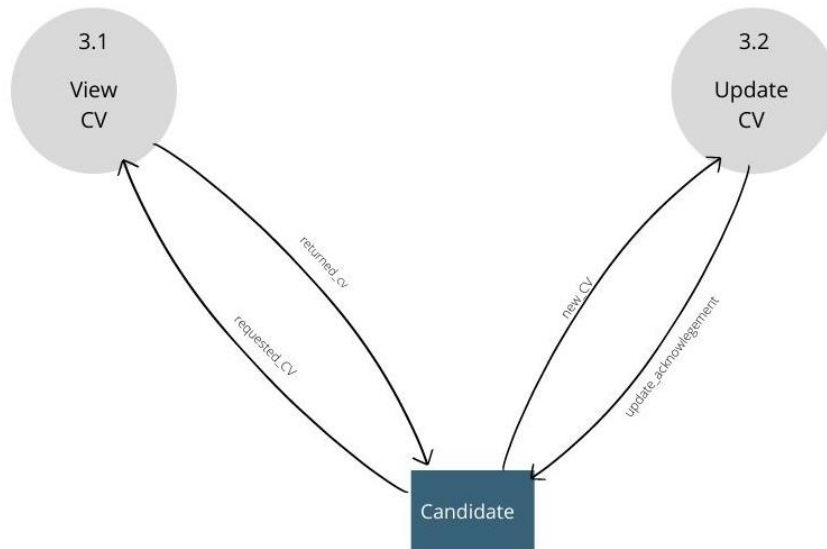
Appendix B: Analysis Models

USE CASE DIAGRAM:



DATA FLOW DIAGRAM:

Level 2 DFD



3. DESIGN PHASE

3.1 Class Diagram

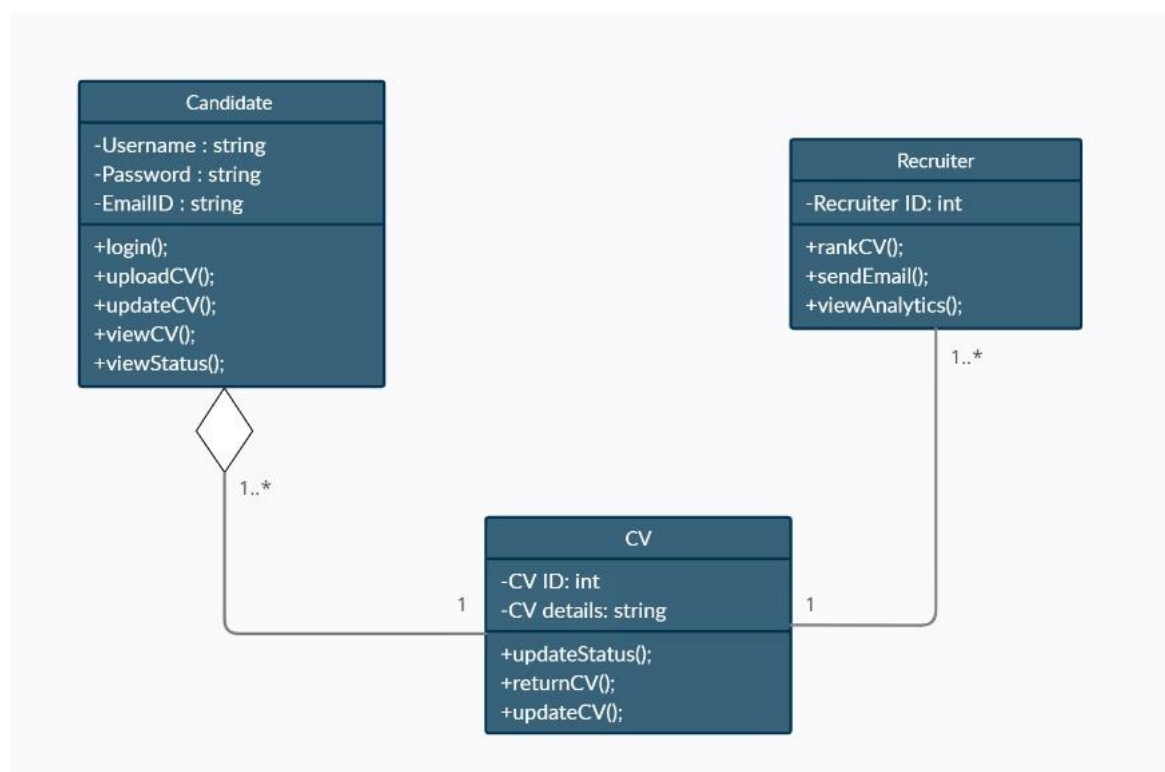


Fig 7: Class Diagram

This is a static structure diagram that describes the structure of a system by showing the system's classes, attributes, methods and the relationships among objects. We have 3 classes in our system Candidate, Recruiter and CV along with their corresponding attributes and methods. The relationships between these classes have also been demonstrated in the diagram.

3.2 Sequence Diagram

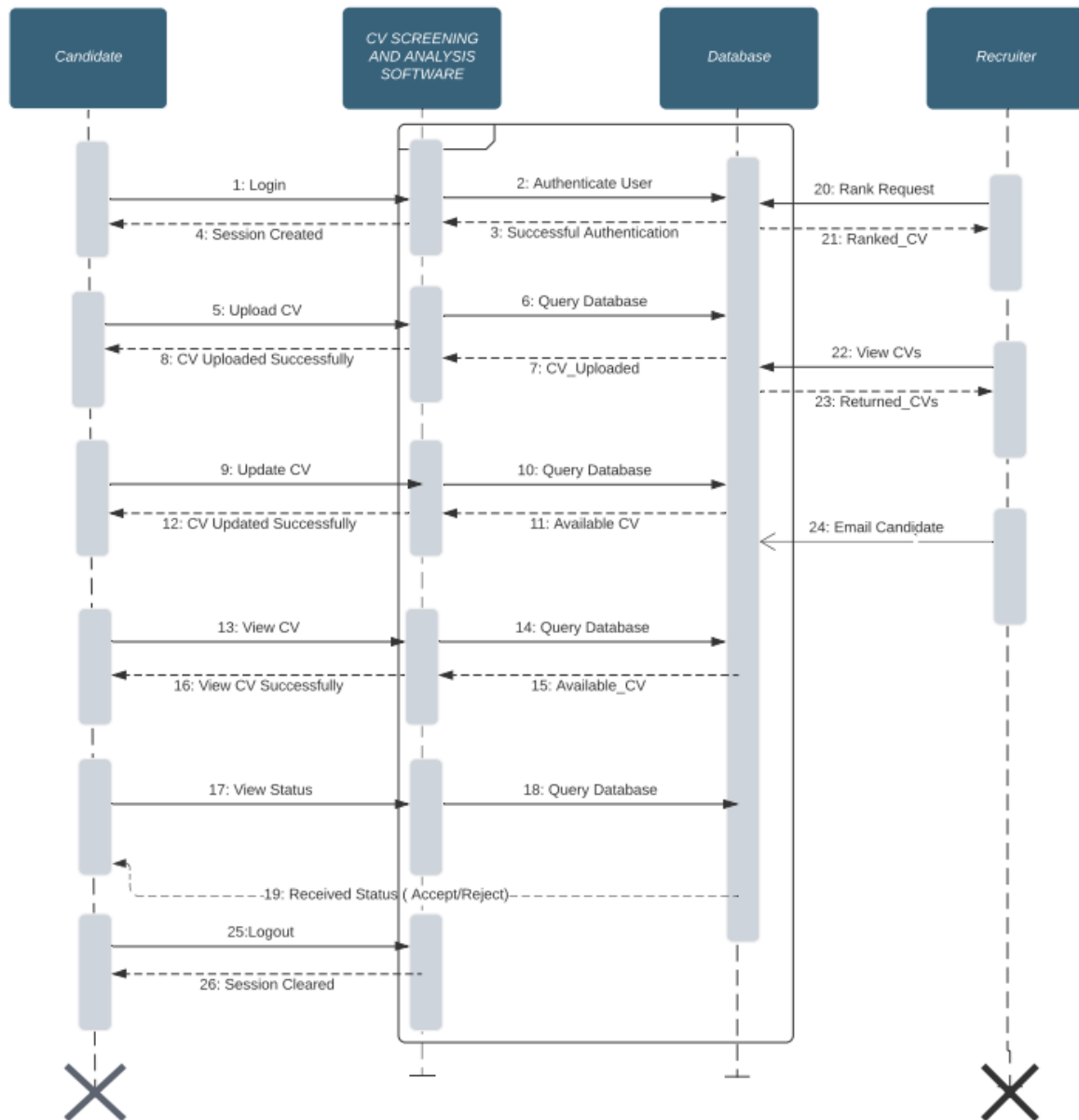


Fig 8: Sequence Diagram

This diagram specifically focuses on the processes and objects that live simultaneously, and the messages that are exchanged between them to perform functions like Login, Upload CV, Update CV etc. before the lifeline ends. It also shows the interaction between these functions and activities.

3.3 Collaboration Diagram

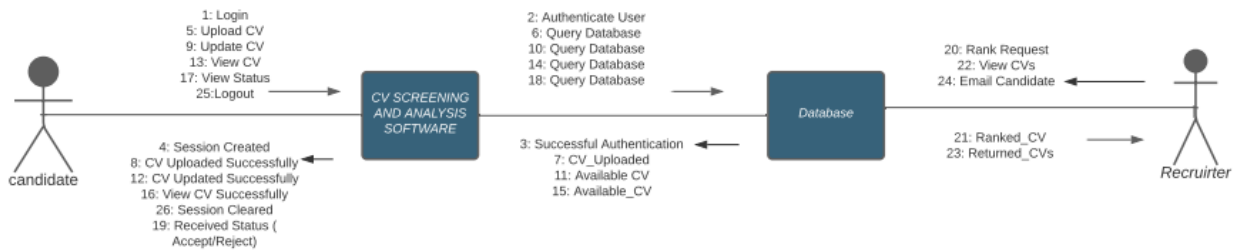


Fig 9: Collaboration Diagram

This diagram further shows the relationship between the two objects in the system, Candidate and Recruiter. It depicts the architecture of the object residing in the system as it is based on object-oriented programming.

3.4 Database Diagram: ER Diagram

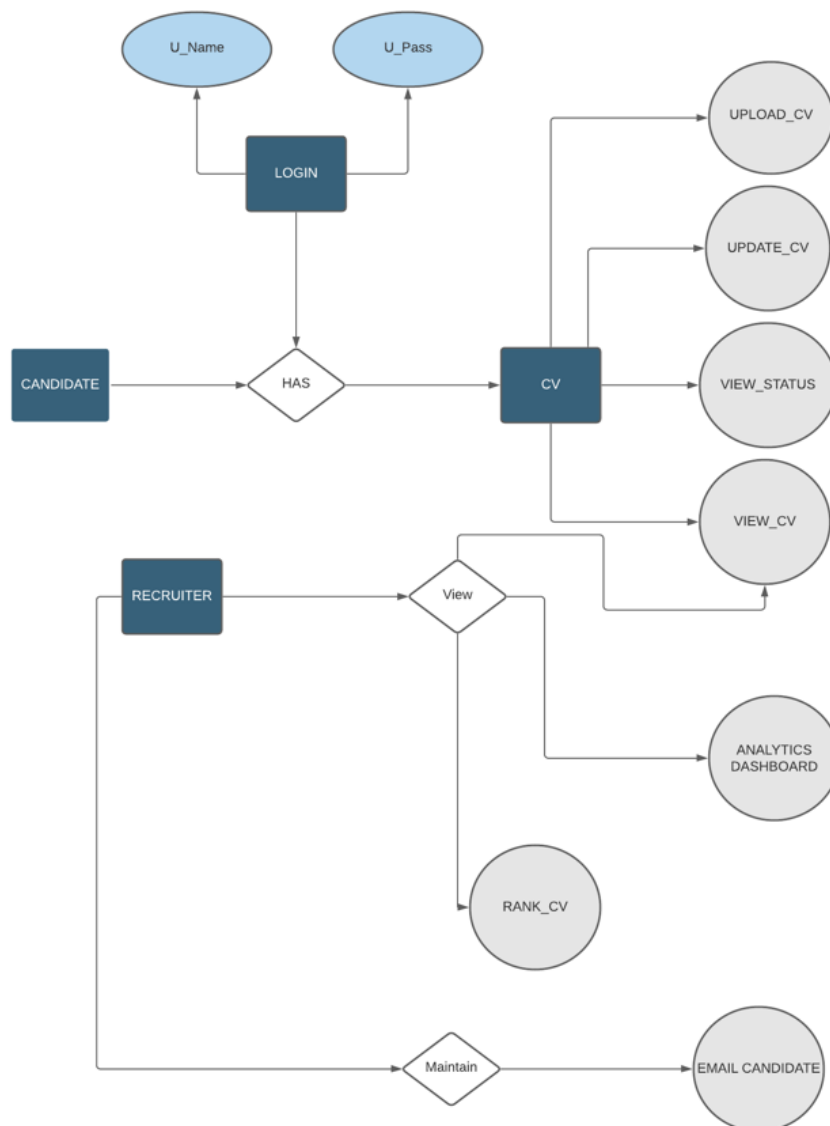


Fig 10: Entity-Relationship Diagram

This diagram depicts the various entities in our system along with the different relationships between them. We have also specified the various active processes in the system.

4. PROTYPING – GUIs

4.1 Screenshots

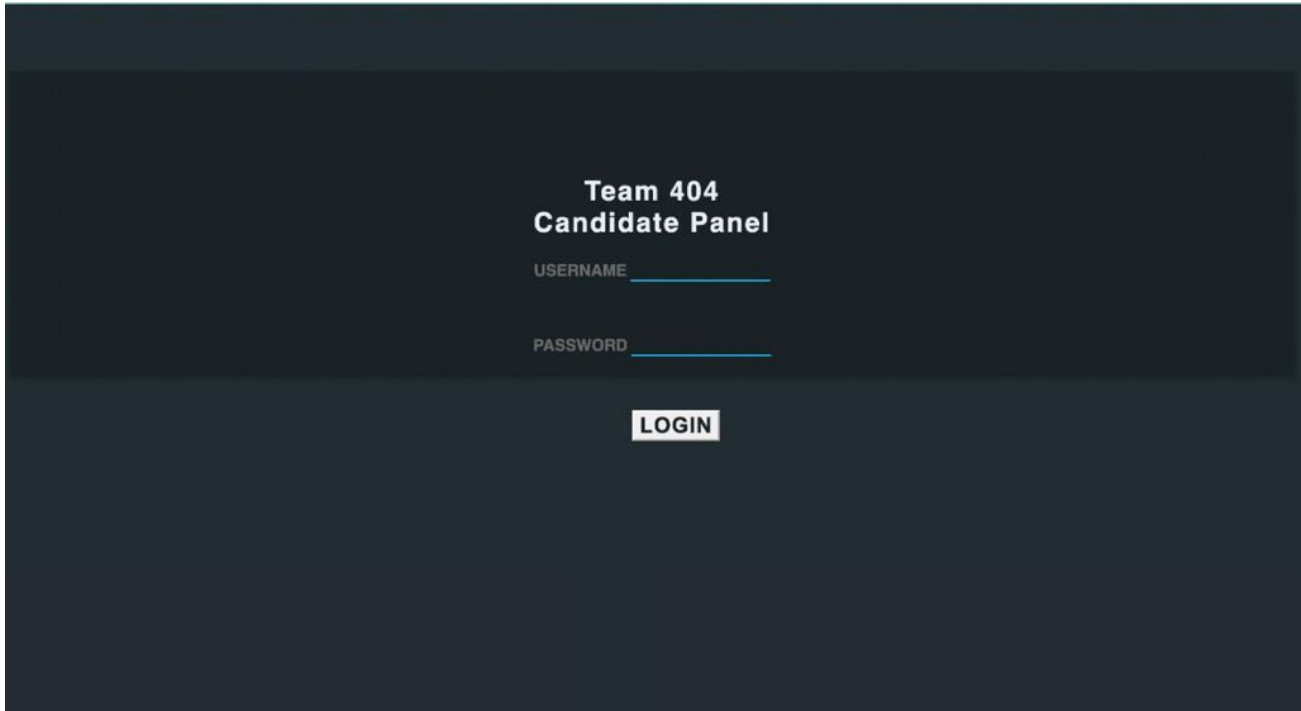


Fig 11: Candidate Login Page

This is the login page for the candidate. It is the first page which the candidate sees before accessing the system and is used for authentication and security purposes.

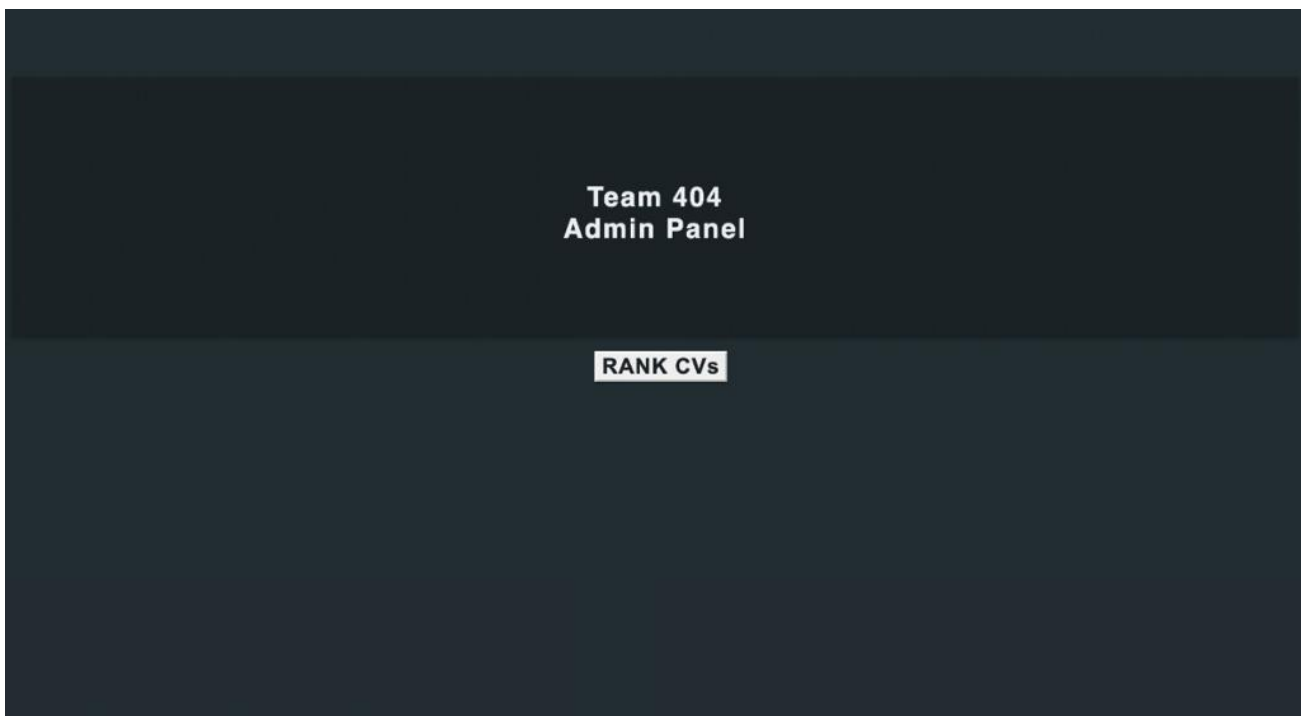


Fig 12: Recruiter Rank CV Page

This is the main page where the ML algorithm is being used. Once the Recruiter presses on the

Rank CVs button, the algorithm gets to work and arranges all of the CVs in order according to requirements set in the algorithm. What we get as a result is the following page.

Team 404 Admin Panel		
Candidate's Name	University/College	Status
Candidate 1	University 1	Send Email
Candidate 2	University 2	Send Email
Candidate 3	University 3	Send Email
Candidate 4	University 4	Send Email
Candidate 5	University 5	Send Email
Candidate 6	University 6	Send Email

Fig 13: Recruiter Ranked CVs Page

This is the page which depicts the ranking of the CVs according to the ML algorithm

```

1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <meta http-equiv="X-UA-Compatible" content="IE=edge">
7   <meta name="viewport" content="width=device-width, initial-scale=1">
8   <link rel="stylesheet" href="style.css">
9   <title>Document</title>
10 </head>
11
12 <body>
13   <div class="container">
14     <div class="row">
15       <div class="col-lg-3 col-md-2"></div>
16       <div class="col-lg-6 col-md-8 login-t">
17         <div class="col-lg-12 login-key">
18           <i class="fa fa-key" aria-hidden="true"></i>
19         </div>
20         <div class="col-lg-12 login-title">
21           Team 404 Admin Panel
22         </div>
23         <div class="col-lg-12 login-button">
24           <i class="fa fa-key" aria-hidden="true"></i>
25         </div>
26       </div>
27     </div>
28   </div>
29
30   <div class="col-lg-6 login-btm lc">
31     <button type="submit" class="btn btn-primary">Rank CVs</button>
32   </div>
33 </body>
34 </html>

```

```

1 label {
2   margin-bottom: 0px;
3 }
4
5 .form-control-label {
6   margin-top: 2px;
7   font-size: 18px;
8   color: #000000;
9   font-weight: bold;
10  letter-spacing: 0px;
11 }
12
13 .btn-outline-primary {
14   border-color: #000000;
15   color: #000000;
16   font-size: 24px;
17   border-radius: 0px;
18   font-weight: bold;
19   letter-spacing: 1px;
20   box-shadow: 0 1px 3px #000000;
21 }
22
23 .btn-outline-primary:hover {
24   background-color: #000000;
25   color: #000000;
26   right: 0px;
27 }
28
29 .login-btm {
30   float: left;
31 }
32
33 .login-button {
34   padding-right: 0px;
35   text-align: center;
36   margin-left: 690px;
37   margin-bottom: 25px;
38 }
39
40 .login-text {
41   text-align: center;
42   padding-left: 400px;
43   color: #000000;
44 }
45
46 .login-btm {

```

Fig 14: HTML and CSS Code of frontend of the software