Software Requirements Specification and Design Description

for

Automatic Recruitment System

Prepared by

Ata Kaleli – 2385474

Doğukan Baysal – 2453694

Mehmet Fatih Çelik – 2385268

Melisa Çağılgan – 2328755

<university>

Middle East Technical University NCC

Computer Engineering Department

Supervised by Assist. Prof. Dr. Mariem Hmila

13/01/2023

Contents

List of Tables

List of Figures

1 Introduction

1.1 Purpose

There is a saying that every nice piece of work needs the right person in the right place at the right time. Nowadays, for every job advertisement, hundreds if not thousands of resumes are sent to the advertiser w. This represents the beginning of a long, tedious process of resume analyzing, ranking, and interviewing; nevertheless, the process itself may involve various stages. This everlasting hiring process might be destructive for a company with a vacancy that must be filled immediately by the most suitable candidate. In addition, this remains a significant objective even in the presence of limitless search time. Furthermore, this process is entirely based on human recruiters who may become cognitive-biased, and their judgments lose their reliability and validity as they spend more time analyzing resumes [3]. To overcome the problems mentioned above, getting help from time-independent and unbiased artificial intelligence processes in that field is inevitable.

Besides the problems discussed above, buying an online hiring application is more cost-efficient than paying a monthly salary to a human recruiter. According to the survey results in [9], 96% of recruiters agree that using technology makes the recruiting process cheaper, and 77% believe it will lower the hiring cost. If we consider the recruitment cost increases by nearly 400.000 dollars for a company hiring around 100 people per year [2], reducing the money spent in this area dramatically contributes to the companies' revenue. According to the compound annual growth rate of recruiting, automation reached 1.9% from 2015 to 2021 and is predicted to reach 2.7% by the end of 2032 [17].

Evidently, automatic recruitment systems play a vital role in recruitment, and many companies desperately need them. Even though various companies are trying to develop an automated solution to handle the time-consuming, unobjective, and expensive processes as much as possible, the application has yet to lead the market, and it is still open for improvement. Therefore, our primary focus is to provide companies with a platform where they can automate all their hiring stages and show them a perfect match for a job position within applied candidates.

1.2 Scope

Our main goal is to automatically score each applicant who applied to a specific company position and suggest the most suitable ones to the recruiters. The company employees would select the number of applicants that will be suggested. To achieve our goal, we have three objectives equivalently phases:

- First, collect applicants' resumes and cover letters with the help of pre-determined external job-searching platforms. Then, according to the company's pre-selected keywords, a content-based feature extraction method is applied to locate and extract the relevant keywords from these two documents and reduce the number of applicants as pre-evaluation.
- Second, collect the video interviews of the applicants and extract specific personal characteristics from the video, audio part, and answers from the audio part of the interviews using A.I. algorithms.
- Third, determine the candidates' scores to be delivered to the job advertiser.

The outcome of each phase is as follows:

Phase 1: Applicants' scores resulted from matching their resume keywords with the position description and requirements. After which, all candidates will be listed in order, and the ones with a score below the threshold value, decided by the recruiter, will be eliminated. All candidates will be notified via an automatic e-mail sent by our application. The accepted candidates in this preliminary phase will receive a video interview link if they pass the first stage.

Phase 2: Applicants will be scored based on their answers, facial expressions, and voices from the asynchronous video interview questions. Both video and audio will be used during emotion extraction from voice and facial expression. Still, scoring question answers will be only based on audio of an asynchronous video interview utilizing SoftMax Probability Distribution, the most suitable reply provided by AI Interactive Interview with scoring the response for the part corresponding to the job qualification. All question answers will be separate videos which will be scored separately. After this, scores of answers and derived propensity information (Diverse, Realistic, Rational, Planning) will be multiplied by their weights of importance which the company employees will decide [12]. Propensity information is essential while recruiting because media richness, gestures and facial expressions, tone of voice, and testing quick thinking ability provide information about appearance, mannerisms, and speech patterns [20]. This process will result in a single score for an applicant regarding an asynchronous video interview. We will allow the company to conduct a multiple-choice test for the technical part of the scoring process. The questions can be selected from question pools which should be created by the company or can be manually adjusted. Furthermore, we will score the applicants' resumes from phase 1 according to their education, experience, and language features [15].

Phase 3: Finally, the final score will be calculated using the scores from the technical, interview, and resume scoring parts. The company will be the one who determines the weights of the parts. After that, the system will create the final score of the applicants based on these weights given by the companies. The candidates who scored above a threshold, which the company employee decides, are shortlisted. The result is shown to the recruiter indicating the highest-scoring applicant as the best match for the corresponding position. After they are convinced about which applicant they must hire, our application will send an e-mail to all concerned applicants.

Besides the logical parts behind our application, we decided to keep the application's user interface as simple as possible for companies' usage. Despite this simplicity, we intend to cover most of the features and functionalities needed for the recruitment process of companies by using our application. The overall process scheme is presented in Figure 1 below.

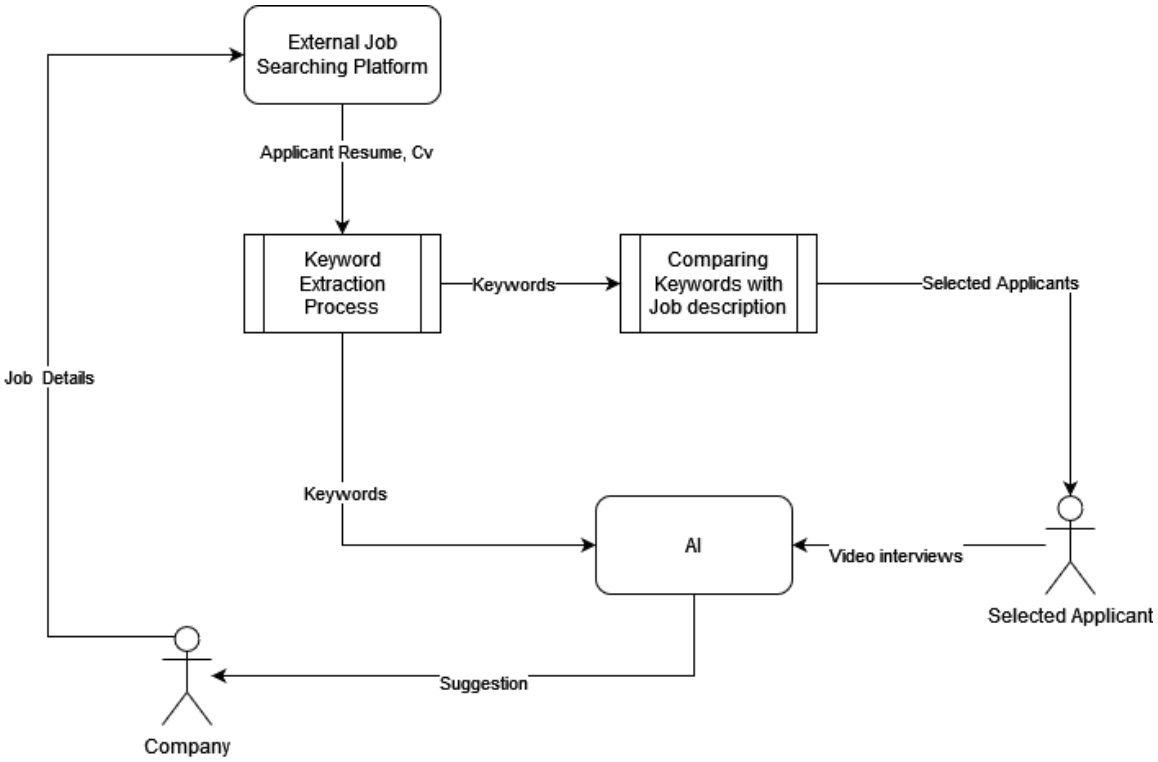

Figure 1: Process Scheme

1.3 Related Work

1.3.1 Similar Applications

According to our research, many automated recruitment systems are on the market. Considering the functionalities that we want our application to offer to the recruiters, each team member went through the literature to find at least two similar applications in the market. We selected three of them, which separately contain the key features. The functionality comparison between the three of them can be seen in Table 1, which lists the core functionalities we intend to develop in our application. The available applications with the closest concept and functionalities to our proposal were thoroughly investigated, and a comparison was constructed based on them. Our main goal is to prepare a platform where a company can handle most, if not all, functions mentioned in Table 1 through a single application. In the Table 1,

Features/ App Names	Similar Applications		
App Names	**GrabJobs[1]**	**LinkedIn Recruiter[2]**	**HireEZ[3]**
Post a job	✓	✓	✗
Apply for a Job without a CV	✓	✗	✗
Candidate Matching	✓	✓	✓
Candidate Keyword Filtering	✓	✓	✓
Candidate Notifications	✗	✓	✗
Reach-Out Candidates[4] via E-mail	✓	✓	✓
Video Interviews	✓	✗	✗
Customizable Hiring Stages	✓	✓	✓
Collaborate with Teammates	✓	✓	✗
Asynchronous Video Interviews	✗	✗	✗

Table 1: Similar Application Comparison

Moreover, as seen in Table 1, even the applications we consider most similar to those we want to develop don't offer asynchronous video interviews for their evaluation process. In traditional interviews, applicants may be nervous as they enter the interview in an environment they have never been to, so they may have difficulty reflecting their true personalities and knowledge to the recruiter. This problem can be solved by using an online interview, but the problem is that when more than one person applies for a job position and considering that most of the applicants meet the requirements of the position, it would be a complete waste of time to interview all of them online and process them as soon as possible and explain the results. Therefore, adding a second part to our recruitment process will make our application stand out by making a difference among other applications.

[1] *Find the Best Jobs and Grow your Career - GrabJobs*. (n.d.). https://grabjobs.co/

[2] *LinkedIn Recruiter | LinkedIn*. (n.d.). https://www.linkedin.com/products/linkedin-recruiter/

[3] Outbound Recruiting AI-Powered Platform, hireEZ. (2022, August 30). *Why hireEZ | AI-Powered Outbound Recruiting Platform | hireEZ*. https://hireez.com/why-hireez/

Methods	Usage
ACF	Personalized query-less job recommendations [16]
SSP	Analyzing human communication [5] [6]
BOW	Extracting features from documents such as CVs [7] [12]
BOVW	Content-based image retrieval, object detection, and image classification [5]
LIWC	Extracting lexical features related to interview content [4][5]
FACET	Classifying emotions from facial expressions [4]
openSMILE	An open-source audio feature extractor [4]
SensitiveNets	A discrimination-aware learning algorithm for the elimination of sensitive information [15]
Motion Tracker	Face and upper body motion estimator [4]
K-Means Clustering	An unsupervised learning algorithm grouping the unlabelled dataset into different clusters [5][6]
Word2vec	An NLP technique computing a feature vector for every word in the corpus to detect synonymous words or suggest additional words for a partial sentence [5]
Doc2vec	A subset of the word2vec technique computing a feature vector for every document in the corpus [6]
FACS	Dividing facial expressions into individual components of muscle movement [1]

Table 2: AI Methods with Usages

While searching the literature, we found algorithms to help us implement many of the features we showed in Table 1. Among them, we chose the ones with the best success percentages, which we think are parallel to the functionalities we want to implement. We listed the most useful ones in Table 2 with their usage purposes.

1.3.2 Logical Background Information

During our A.I. part implementations, we decided to use pre-trained models with transfer learning applications. Therefore, we also searched for some informative datasets. In Table 3, we listed some suitable datasets which will be helpful in the cv and character analysis processes by providing good employees' information and characteristics.

Datasets
MIT Interview [5]
IBM HR Analytics Employee Attrition & Performance [14]
Employee Turnover [22]
Hiring [13]
Synthetic E-Recruitment (User) Dataset [18]

Table 3: Datasets

In Table 4, we chose to list the A.I. models used in previous automatic recruitment processes and reported them to be relatively accurate in the literature. This was done so that we could make an informed decision on which model(s) to use while developing our application.

AI Models
Linear SVM Regression [5][12]

Ridge Regression [5]
Random Forest [6][12]
CNN [1]
DNN [1][12]

Table 4: AI Models

One of the things we learned during the literature research was that many of the automated hiring apps on the market use discriminatory and biased A.I. models. In our recruitment system, we will focus on creating a fair, competitive platform where every talented and suitable applicant to a specific position can shine among others. To achieve this, we aim to develop an application where sensitive information such as gender and ethnicity cannot be selected as keywords in any hiring stages by using the SensitiveNets algorithm [15].

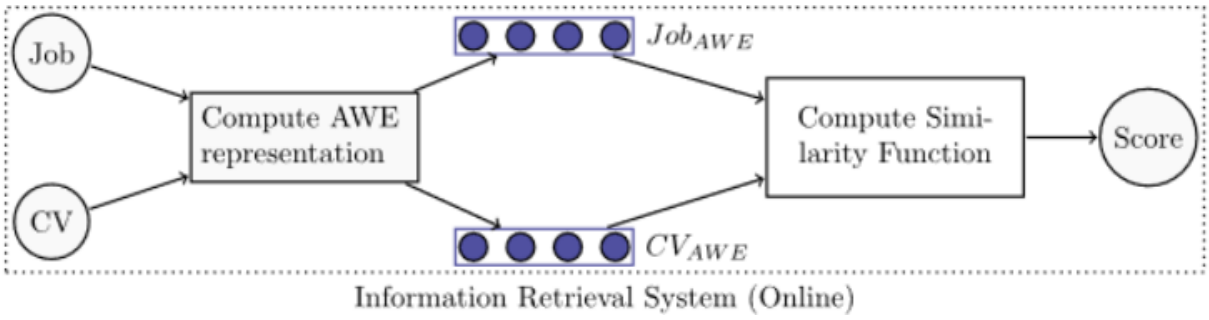

Figure 2: Job Description and CV Keyword Comparison Pipeline

For the keyword extraction mentioned in phase 1, we will use a simple algorithm that takes CV content and job description through Word2vec with configuration parameters as follows: Window size is five, minimum word count is three, and 200 dimensions. Then it will return to a score representing the match ratio between the job description and the CV content [8].

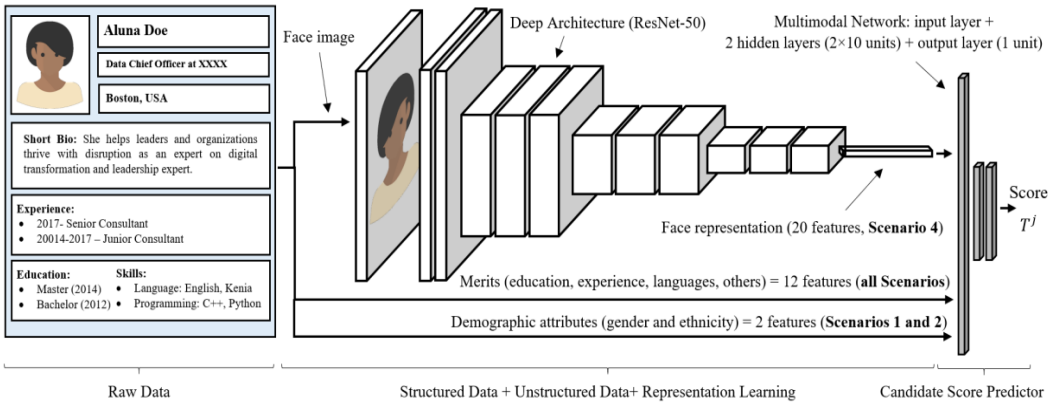

Figure 3: Job Description and CV Keyword Comparison Pipeline

CV scoring for phase 2 will be conducted using SensitiveNets algorithm, as seen in Figure 3, input for the SensitiveNets algorithm will be raw data containing text and image parts which will be obtained using Tika (http://tika.apache.org/). The content of the resume and image of the applicant will be

processed separately and then be scored in the candidate score predictor part according to the specific position the candidate applied to.

For asynchronous video interview scoring, we decided to use the model from [12];

As seen in Figure 4, a simple demonstration of the model used in the video interview process, both audio and video parts of the asynchronous video interview are used for predicting big-five personality trait values, represented as probabilities (between 0 and 1) [23]. Also, answers to the audio part questions will be analyzed using keyword derivation for job competency. Then, according to their importance weights, a score will be generated with this information. The company employees will again decide on weights in terms of importance. At last, a total score for the whole interview will be developed with techniques mentioned in phase 2 of the recruitment process.

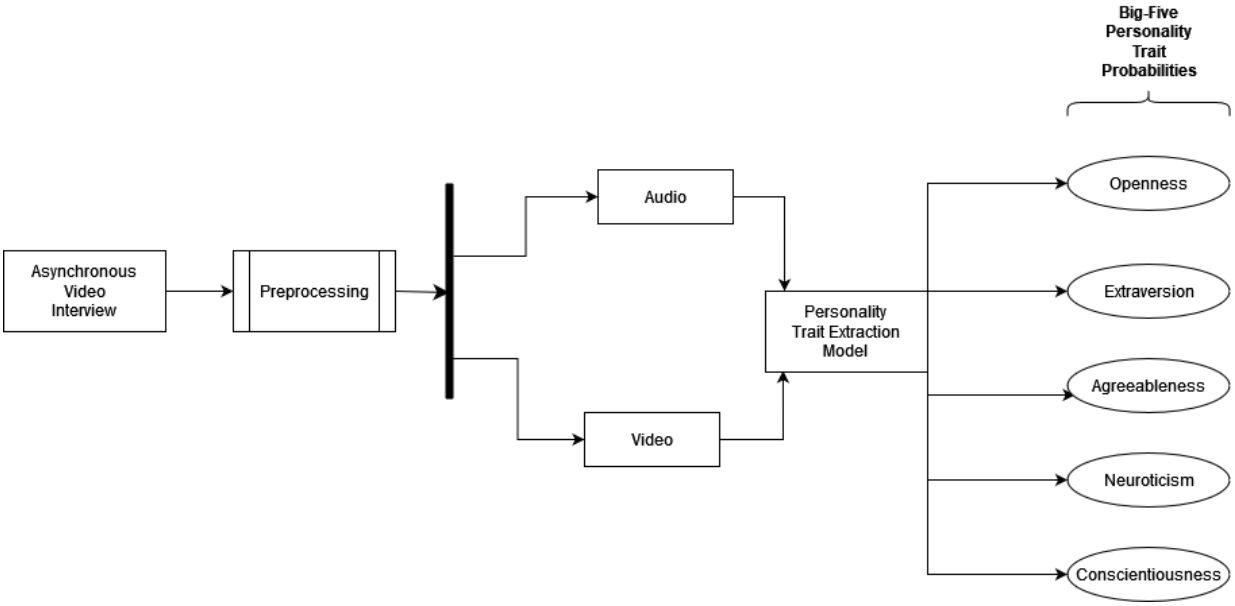

Figure 4: Big-Five Personality Trait Extraction Schema

1.4 Product Overview

1.4.1 Product perspective

The figure below shows our context model, which includes Job Search Engine APIs such as LinkedIn APIs. We will use these APIs to post jobs and receive cover letters and resumes from applicants.

Figure 5: Context Diagram

1.4.1.1 System Interfaces

- The system shall allow company employees to post a job advertisement to a job searching platform via APIs.

- The system shall get applicant information, resume, and cover letter from the job searching platform.

1.4.1.2 User interfaces

- **Login Screen:** In the Login Screen, the user will log in to the Automated Recruitment System by entering their e-mail and password. They also use continue with their LinkedIn, Facebook, or Google accounts. There is an ultimate guide about the recruiting automation system between the login panel so that users can search for information about it.

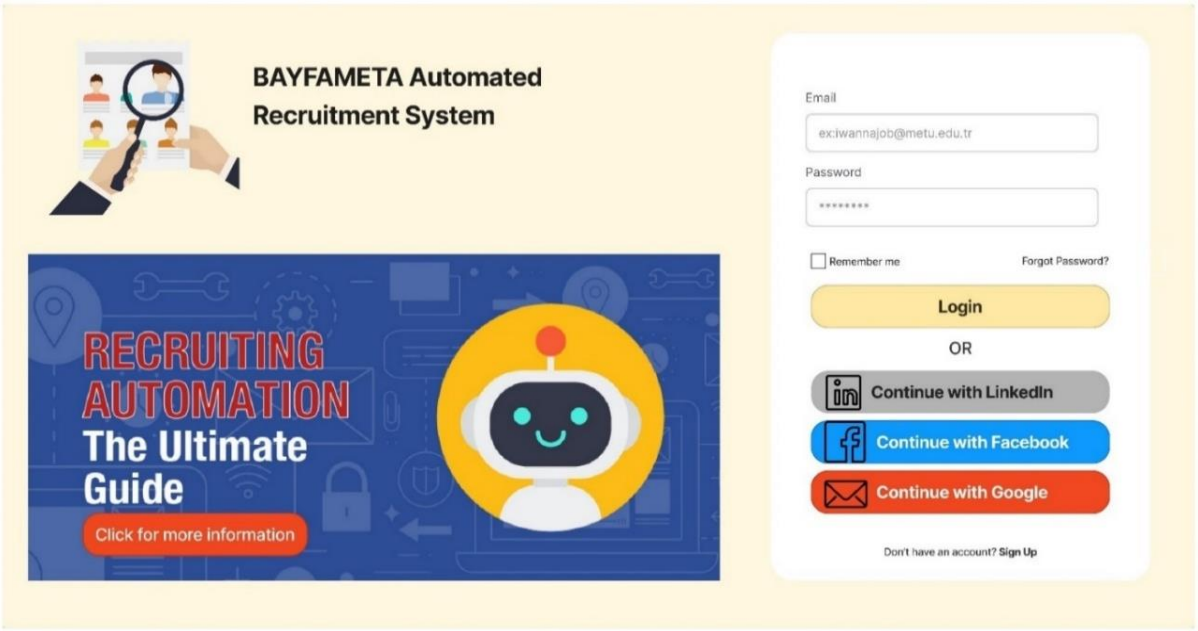

Figure 6: Login Screen

- **Position Search Screen:** Company employees can monitor their pending recruitment processes for open positions in the Position Search Screen.

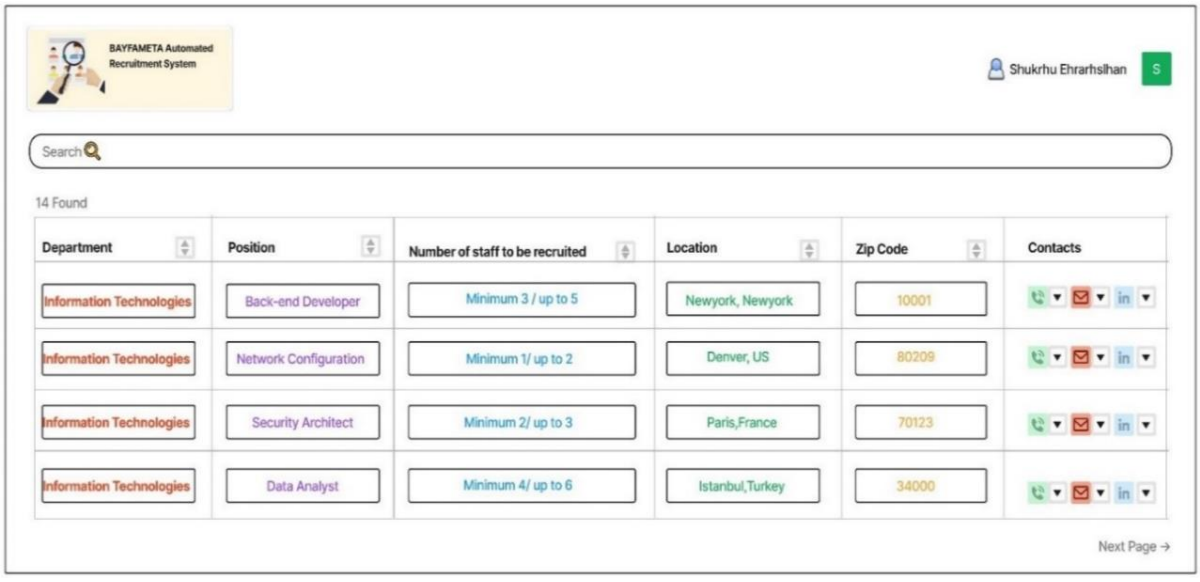

Figure 7: Position Search Screen

- **Applicant Analysis Screen:** In this screen, the company can search for candidates' scores after their video interview submission is finished. They will search candidates by typing their name, surname, or both. The company can also reach candidates' contact information on this screen. Suppose they want to seek more; in other words, see the candidate's overall result with percentages of propensity information. In that case, they will click the "Click Here for Details" button, then another screen will open and show the candidate's overall score.

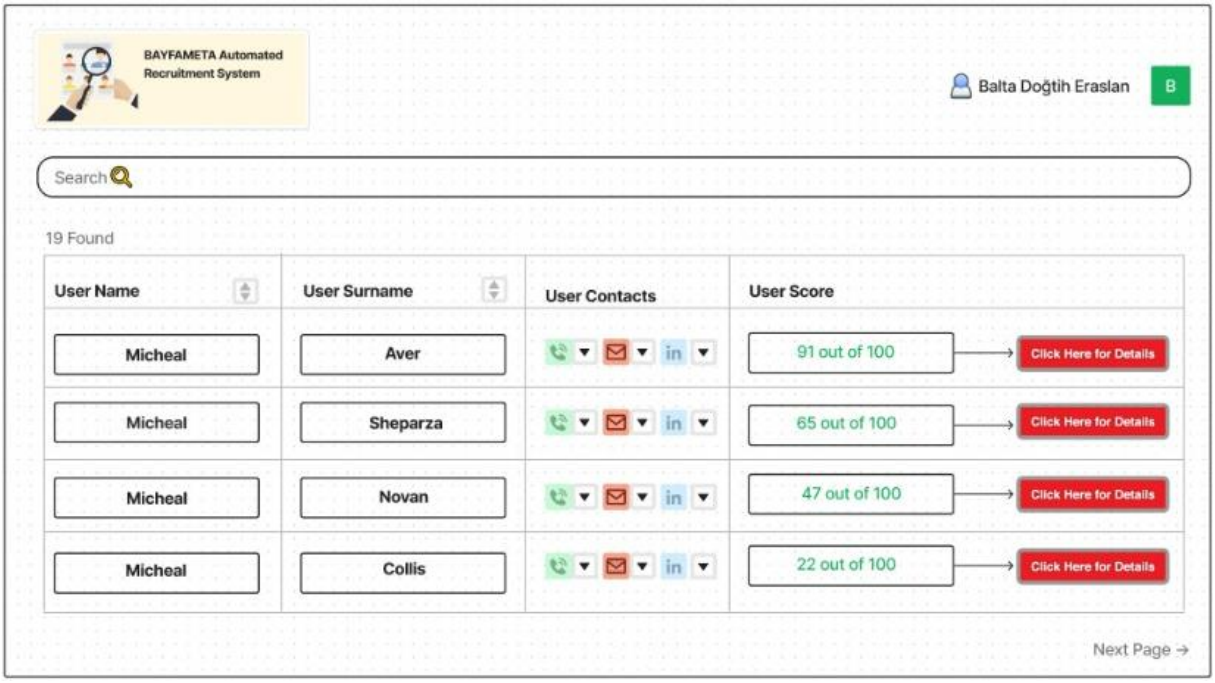

Figure 8: Applicant Analysis Screen

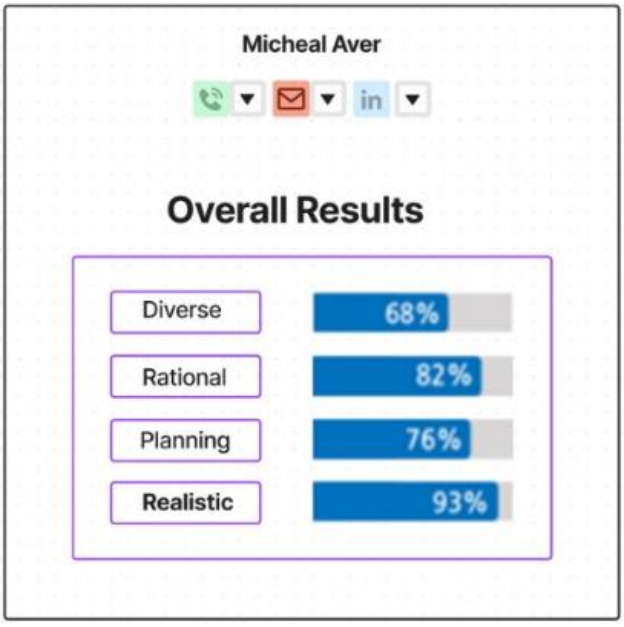

Figure 9: Applicant Analysis Details

1.4.2 Product functions

- The system shall allow company employees to view the position advertised from job-finding websites, such as LinkedIn and Xing.
- The system shall allow company employees to view the candidates that applied for positions.
- The system shall allow company employees to filter the candidates based on keyword searches from their cover letters and CVs.
- The system shall notify if any candidate's score is above the threshold, which the company employees decide for matching percentage.
- The system shall allow the company employees to send emails to participants with a template such as rejection, approval, and instructive template for specific applicants. Employees will construct these templates.
- The system shall allow company employees to adjust the weights of the scoring process (such as personality traits, technical test, and resume scoring weights) to make it more customizable for each position opening.
- The system shall allow applicants to upload their video interview, which then will be scored by the system automatically considering big-five personality traits.

1.4.3 Identified stakeholders and design concerns

Applicant: Their concern is to submit their resume and cover letter to an external job-searching system, record an asynchronous video interview, complete the technical questionnaire, which includes multiple choice and open-ended questions, and then submit it to the system.

Company Employee: The company employee determines the resume keyword extraction threshold value. Also, the employee can customize the technical test questions for the positions they want to recruit. They also choose the percentages for determining the score. The score includes resume keyword extraction, asynchronous video interview, and technical test points. Finally, the employee can score the threshold to eliminate the applicants as they want.

1.4.4 User characteristics

- **Users –**
  User actors are candidates for a position opening in a company. They are considered applicants who are qualified enough to attend video interviews based on the resume and cover letter keyword comparison with the position description and desired requirements. Functionalities for these actors will be as straightforward as possible. Their education level must satisfy position requirements. Users must be able to navigate the GUI interface of the software and have a camera, a microphone, and a web browser on their computer. Users will be English speakers to use the application.

- **Company –**
  Company actor is considered as recruiter teams working in human resource departments of related companies. Most of the functionalities are for the usage of this actor group. Actors are expected to have some recruitment information to design stages of the recruitment process on the application. Company employees must be able to navigate the GUI interface of the software, know how to locate a file in the computer, and have a web browser on their computer. Company employees must be English speakers to use the application.

1.4.5 Limitations

- A question in a video interview should finish within 3 minutes because, as stated in [9], giving enough time to applicants to respond (more than 90 seconds) may reduce anxiety.
- A video interview file must be at most 150 MB because of storage limitations.
- A user must upload his/her resume with a distinct file type, like Docx, doc, or pdf, such that Tika (http://tika.apache.org/) could process it to eliminate font colors, styles, table layouts and get the raw text which is needed for keyword extraction as stated in [10].
- A user must have a camera for video interviews.
- A user must have a microphone for video interviews.
- A user must have a reliable internet connection for uploading video interviews.

1.4.6 Assumptions and dependencies

We assumed that;

- Users will be English speakers as our application will be in English.
- Users consent to the company to record their video interview for ethical purposes.
- Users will have appropriate platforms, such as a web browser, for using automatic recruitment applications.

2 Specific requirements

2.1 External interfaces

We will use two specific APIs from Job Search Engine for our external interfaces. Relevant functionalities, inputs, and outputs can be seen below.

System Interface	Functionality	Input	Output
Applicant Information	Get user information, resume, and cover letter.	Request	Two files (Docx, doc, or pdf), User class
Recruitment	Post a position opening to a job searching platform	Position details, Company information	Confirmation

Table 5: External Interfaces

2.2 Functions

We included core functionalities for our use case diagram. We considered AI as an actor to represent the role separately from other actors and to underline its importance to our application.

Figure 10: Use Case Diagram

Use case	Register
Actors	User
Cross references	-
Typical Course of Events	
Actor Intentions	**System Responsibility**
1. User enters the registration page.	
	2. System asks the user to enter his/her name.

	3. System asks the user to enter his/her e-mail.
	4. System asks the user to enter his/her password.
4. User clicks the registration button.	
	5. System sends a confirmation message that the user has registered successfully.
6. User closes the registration page.	
Alternative Courses	
Step 3: User enters an existing e-mail, so the system gives a warning message and directs the user to the login page.	
Step 4: User enters a weak password, so the system re-asks user to enter a strong password.	

Table 6: Use Case – Register

Use case	Video Interview Submission
Actors	User
Cross references	-
Typical Course of Events	
Actor Intentions	**System Responsibility**
1. User enters to video interview submission page.	
	2. System asks a variety of interview questions to the user regarding the requirements of the position.
	3. System asks the user to upload his/her interview video, which takes a maximum of 15 minutes.
4. User clicks the interview submission button.	
	5. System shows a confirmation message that the video has been uploaded successfully.
6. User closes the video interview submission page and returns to the landing page.	
Alternative Courses	
Step 4: User tries to upload an interview video that takes more than 15 minutes, so the system warns the user about the length of the video and returns to step 2.	

Table 7: User Case - Video Interview Submission

Use case	Transfer Resume
Actors	Job Search Engine
Cross references	-
Typical Course of Events	

Actor Intentions	System Responsibility
	1. System takes the resumes automatically from the job search engine.
	2. System sends a daily report about how many applicants they take from the job search engine.
Alternative Courses	
N/A	

Table 8: Use Case - Transfer Resume

Use case	Transfer Cover Letter
Actors	Job Search Engine
Cross references	-
Typical Course of Events	
Actor Intentions	**System Responsibility**
	1. System takes the cover letters automatically from the job search engine.
	2. System sends a daily report about how many applicants they take from the job search engine.
Alternative Courses	
N/A	

Table 9: Use Case - Transfer Cover Letter

Use case	Transfer User Data
Actors	Job Search Engine
Cross references	-
Typical Course of Events	
Actor Intentions	**System Responsibility**
	1. System automatically takes the relevant applicant data such as name, surname, and email from the job search engine when users apply for a position.
	2. System sends a daily report about how many applicants they take from the job search engine.
Alternative Courses	
N/A	

Table 10: Use Case - Transfer User Data

Use case	Notify User
Actors	User, Company
Cross references	-
Typical Course of Events	
Actor Intentions	**System Responsibility**

	1. System sends a notification message to applicants about their stage progress
Alternative Courses	
N/A	

Table 11: Use Case - Notify User

Use case	Notify Company
Actors	User, Company
Cross references	-
Typical Course of Events	
Actor Intentions	**System Responsibility**
	1. System sends a notification message to the company about applicants' stage result information
Alternative Courses	
N/A	

Table 12: Use Case - Notify Company

Use case	Post Job Positions
Actors	Company
Cross references	-
Typical Course of Events	
Actor Intentions	**System Responsibility**
1. Company enters to post position screen.	
2. Company enters the details of the position they require.	
3. Company clicks the submit button.	
	4. System sends a confirmation message that the position has been added successfully.
5. Company closes the post position screen.	
Alternative Courses	
Step 3: Company fails to enter the necessary details of the position, so returns to step 2.	

Table 13: Use Case - Post Job Positions

Use case	Send Email with Templates
Actors	Company
Cross references	-
Typical Course of Events	
Actor Intentions	**System Responsibility**
	1. System creates template emails about whether applicants are selected or not.
2. Company selects the template.	

3. Company sends the template email to applicants.	
	4. System sends a confirmation message that the email has been sent successfully.
Alternative Courses	
Step 3: Company fails to send the template email to the applicant, so returns step 2.	

Table 14: Use Case - Send Email with Templates

Use case	Customize Hiring Stages
Actors	Company
Cross references	-
Typical Course of Events	
Actor Intentions	**System Responsibility**
1. Company enters Customize Hiring Stages screen.	
2. Company specifies the interview questions.	
3. Company specifies the number of interview questions for a specific position.	
4. Company specifies the keywords that filter applicants and clicks customize button.	
	5. System sends a confirmation message that the hiring stage has been customized successfully.
6. Company closes the Customize Hiring Stages screen.	
Alternative Courses	
N/A	

Table 15: Use Case - Customize Hiring Stages

Use case	View Positions
Actors	Company
Cross references	Filtering
Typical Course of Events	
Actor Intentions	**System Responsibility**
1. Company enters to "View Data" screen.	
2. Company selects "Positions" from the toolbox.	
3. Company clicks to view button.	
	4. System shows the list of positions they posted.
5. Company closes the "View Data" screen.	
Alternative Courses	
Step2: Company decides to filter the Positions (See Use Case "Filter").	

Table 16: Use Case - View Positions

Use case	View Candidates
Actors	Company
Cross references	Filtering
Typical Course of Events	
Actor Intentions	**System Responsibility**
1. Company enters to "View Data" screen.	
2. Company selects "Candidates" from the toolbox.	
3. Company clicks to view button.	
	4. System shows the list of candidates.
5. Company closes the "View Data" screen.	
Alternative Courses	
Step2: Company decides to filter the candidates (See Use Case "Filter").	

Table 17: Use Case - View Candidates

Table 18: Use Case - Filtering

Use case	Filtering
Actors	Company
Cross references	View Positions, View Candidates
Typical Course of Events	
Actor Intentions	**System Responsibility**
1. Company selects the filtering type such as filtering by name, score range, and clicks Apply Filter button.	
	2. System applies the filtering operation.
Alternative Courses	
N/A	
Use case	Extract Audio Features
Actors	AI
Cross references	Score Applicants
Typical Course of Events	
Actor Intentions	**System Responsibility**
1. AI creates an audio folder.	
2. AI extracts the audio features into the audio folder.	
3. AI scores the audio features (See Use Case "Score Applicants").	

	4. System shows a notification message that the applicant's score has been saved successfully.
Alternative Courses	
N/A	

Table 19: Use Case - Extract Audio Features

Use case	Extract Video Features
Actors	AI
Cross references	Score Applicants
Typical Course of Events	
Actor Intentions	**System Responsibility**
1. AI creates a video folder	
2. AI extracts the video features into the video folder.	
3. AI scores the video features (See Use Case "Score Applicants").	
	4. System shows a notification message that the applicant's score has been saved successfully.
Alternative Courses	
N/A	

Table 20: Use Case - Extract Video Features

Use case	Score Applicants
Actors	AI
Cross references	Extract Video Features, Extract Audio Features
Typical Course of Events	
Actor Intentions	**System Responsibility**
1. AI calculates the score of the applicants based on the extracted video and audio features.	
2. AI sends scores to the system.	
	3. System saves the scores of the applicants.
Alternative Courses	
N/A	

Table 21: Use Case - Score Applicants

2.3 Usability Requirements

- The system shall be designed to allow users to perform any operation efficiently by placing the most used functionalities in easy-to-reach places such as listing positions and users.
- The system shall be designed to allow a user to use the system without requiring technical support.
- The system shall have properly comprehensible icons to visualize the operations.
- The system shall allow users to use the system without requiring any prior knowledge.
- The functions in the system shall be well integrated.

2.4 Performance requirements

- The software shall process a video within 3 seconds, so there will not be long-lasting process queues in the background.
- The software's speed of response shall be below 1 second so that users can navigate the system without facing any delay.
- The software shall access its database below 1 second so that the core functionalities, including information retrieval such as recommending and filtering, can be done fast.
- The system shall send the data to the database provider in under 1 second to have a more secure data migration process.
- The system server shall work without problems while at least 3000 users are using the software so that multiple users can benefit from our system simultaneously.
- The system shall navigate between two tabs within 30 milliseconds to provide users with a smooth usage process.

2.5 Logical database requirements

In the database, we will store tables for users, files, score of files, positions, and stages for positions. Also, we have a table for the company in order to keep track of all positions, users, and stages. Scores are weak entities as scores can not exist without a file. Score_type attribute is a foreign key for Score entity.

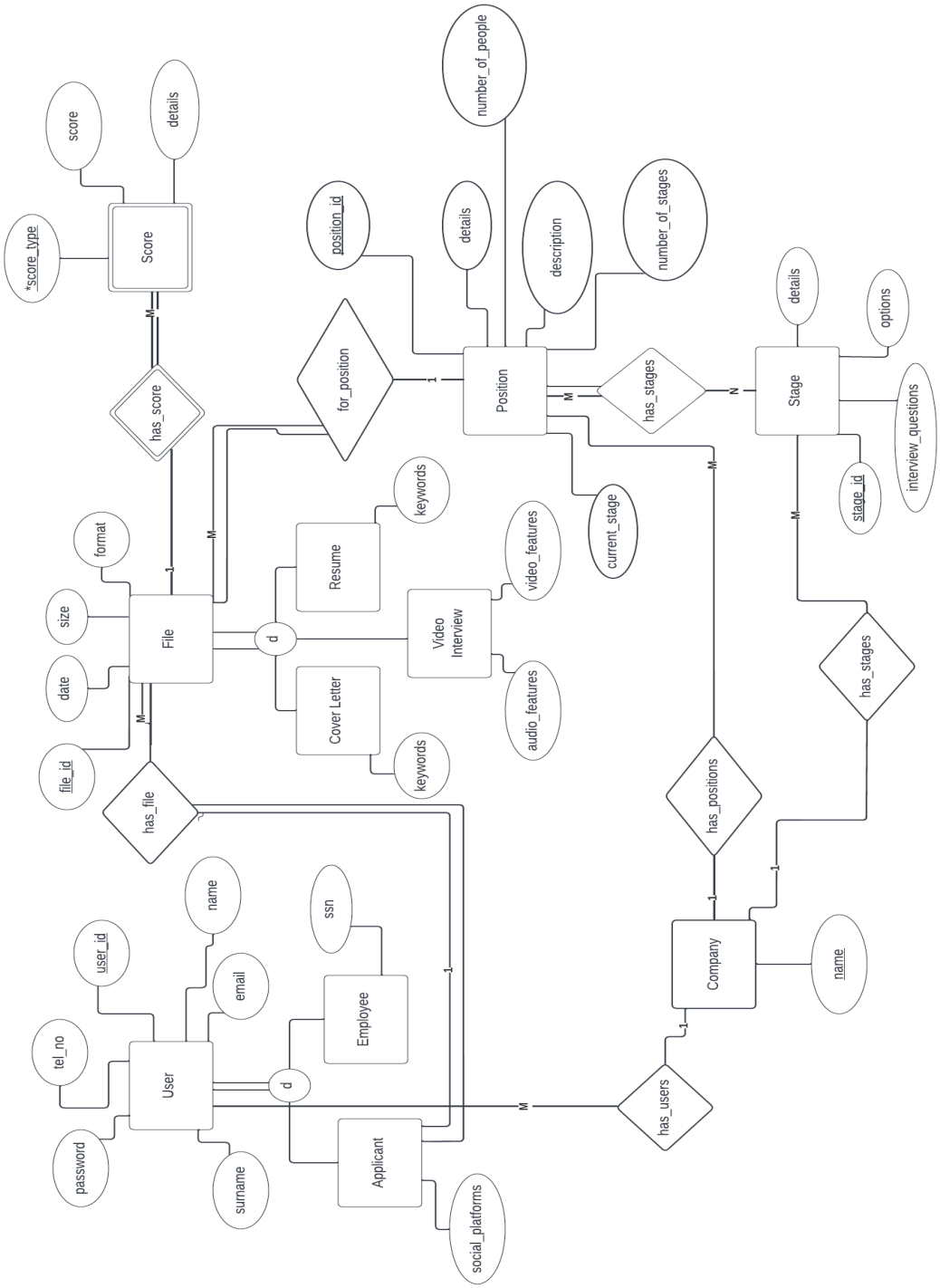

Figure 11: Entity Relationship Diagram

2.6 Software system attributes

- ***Reliability***
  - *The database should be kept up to date regularly.*
  - *The system should have a backup system so that the necessary information, such as ongoing recruitment process files and account information, is not lost if the system fails.*
- ***Availability***
  - *A user may submit his/her video interview at all hours of the day and night if it is within the given time slot.*
- ***Security***
  - *While a user is entering his/her account details, the password should be kept in a hidden format in the interface.*
  - *The user password must contain at least eight characters, including one numeric value, one lower, and one upper letter.*
  - *User password data shall be stored in the database encrypted using the SHA256 hashing technique.*
- ***Maintainability***
  - *Developers shall follow coding standards to minimize the source code as much as possible, avoiding redundancy. Also, proper and precise commenting and coding are necessary.*
  - *Modular development approach shall be used such that if a filtering operation request for the position, user lists or a functionality request from a company is received, it can be implemented easily and efficiently.*
- ***Portability***
  - *All linked files shall be stored in a single folder, allowing for one-command copying so the software can be translated from one computer environment to another.*
  - *The system shall be developed in a high-level language deployed on many platforms, which helps the system run on all the operating systems.*

2.7 Supporting information

- More details on how asynchronous video interviews can be used in the recruitment process [11].

3 Software Estimation

Inputs:

- Position search screen - low
- Applicant search screen - low
- Post positions screen – medium
- Login screen - low
- Video interview submission screen - high

Outputs:

- Confirmation for registration - low
- Confirmation for login - low
- Confirmation for position posting - low
- Position search result – medium (filtering)
- Applicant search result – medium (filtering)

- Matching screen - high
- Notification screen - low

Inquiries:

- Registration - low
- Login - low
- User resume keyword posting - low
- Video interview posting - high
- Video interview feature posting - medium
- Rating and Comment posting - low
- Position search - medium
- Applicant search - medium
- Update stage – low

Logical Data Files:

- User Table (superclass) - medium
- Applicant Table (subclass) - high
- Employee Table (subclass) – low
- File Table - medium
- Video Interview Table - low
- Resume Table - low
- Cover Letter Table - low
- Stage Table - medium
- Position Table - high
- Score Table – high
- Company Table - high

External Interface Files:

- Resume - medium
- Cover Letter - medium
- Applicant Information – low

General System Characteristics (GPC)	Degree of Influence (DI) – 0:least, 5: most	Explanation
Data Communications	4	How many communication facilities are there to aid in the transfer or exchange of information with the application or system?
Distributed Processing	5	How are distributed data and processing functions handled?
Performance	5	Did the user require response time or throughput?
Heavily used configuration	4	How heavily used is the current hardware platform where the application will be executed?
Transaction Rates	5	How frequently are transactions executed daily, weekly, monthly, etc.?
On line data entry	5	What percentage of the information is entered On-Line?
Design for end-user efficiency	5	Was the application designed for end-user efficiency?
Online updates	2	How many internal logical files are updated by online transaction?
Complex processing	5	Does the application have extensive logical or mathematical processing?
Usable in other applications	3	Was the application developed to meet one or many user's needs?
Installation ease	0	How difficult is conversion and installation?
Operational ease	2	How effective and/or automated are start-up, back up, and recovery procedures?
Multiple sites	5	Was the application specifically designed, developed, and supported to be installed at multiple sites for multiple organizations?
Facilitate change	3	Was the application specifically designed, developed, and supported to facilitate change?
Total of Degree of Influence	53	
Value adjustment factor (VAF)	1,18	
		VAF = Total Degree of Influence * 0.01 + 0.65

Table 22: Degree of Influence

Program Characteristics	Low Complexity	Medium Complexity	High Complexity	Total	
Inputs	3	1	1	19	3*Low + 4*Medium + 6*High
Outputs	4	2	1	33	4*Low + 5*Medium + 7*High
Inquiries	5	3	1	33	3*Low + 4*Medium + 6*High
Logical Internal Files	5	3	4	125	7*Low + 10*Medium + 15*High
External Interface Files	1	2	0	19	5*Low + 7*Medium + 10*High
			Unadjusted Total of Function Points (UTFP)	229	
			Adjusted Total of Function Points (ATFP)	270,22	
			ATFP = UTFP * VAF		

Table 23: Total of Function Points

Programming Language	Percentage in Project	Language Unit Size	Language Unit Size in Project	
HTML	0,15	15	2,25	
JavaScript	0,15	80	12	
C#	0,3	54	16,2	
Python	0,4	20	8	
		Language Unit Size	38	
		Lines of Code	10268,36	LOC = LUS * ATFP
		Kilo Delivered Source Instruction (KDSI)	10,26836	KDSI = LOC / 1000

Table 24: Language Unit Size

Development Type	Organic	We have a small team of 4 students. Our application will have little innovation to the industry and we have a stable development environment so our mode is organic
Estimated Effort in Man-Months	27,68773198	**2.4 * KDSI^1.05**
Estimated Development Time in Month	8,830986359	**2.5 * Man Months^0.38**
Estimated Team Size	3,135293256	**~4 = Man Months / Development Time**

Table 25: COCOMO

Adjusted Total of Function Points (ATFP)	270,22	
Kind of Software	Business	Our software will perform specific tasks for metting user needs.
Software Organization's Skills/Abilities	Average	We have some experince on the technologies and programming languages that we will be using
Estimated Effort in Man-Months	50,76352801	**(ATFP^(3 * 0.43))/27**
Estimated Development Time in Months	11,10806777	**3 * MM^(1/3)**
Estimated Team Size	4,569969241	**~5 = MM / Development Time**

Table 26: Jones's First-Order Effort Estimation

Schedule Tables Estimate	Nominal Schedules Table as we are not professional at our job	
Lines Of Code	10268,36	~15000
Estimated Effort in Man-Months	15	
Estimated Development Time in Months	7	

Table 27: Schedule Tables Estimate

Estimated development time and team sizes are close to each other in each estimation technique. Only values calculated from Jones's First-Order Effort Estimation are little higher than expected. But calculation ranges show that we have enough members and time to complete our application.

4 Architectural Views

4.1 Logical View

4.1.1 Class Diagram

Figure 12 shows our class diagram. We represented our database tables as classes that will be used in our application. Company objects can access main functionalities by storing positions, users, and stages. In our project, we have two user types where some functionalities and attributes are common, so we decided to create a superclass called User which will contain these common parts. We used the same approach for file types as well.

If a user is deleted from the system, his/her related files should be deleted as well, and if a file is deleted, scores of these files are no longer needed.

Stage class contains process steps for positions. Each position can have multiple stages; these stages are not position-dependent so that a stage can be re-used for another position. Therefore, there is a many to many relationships between them.

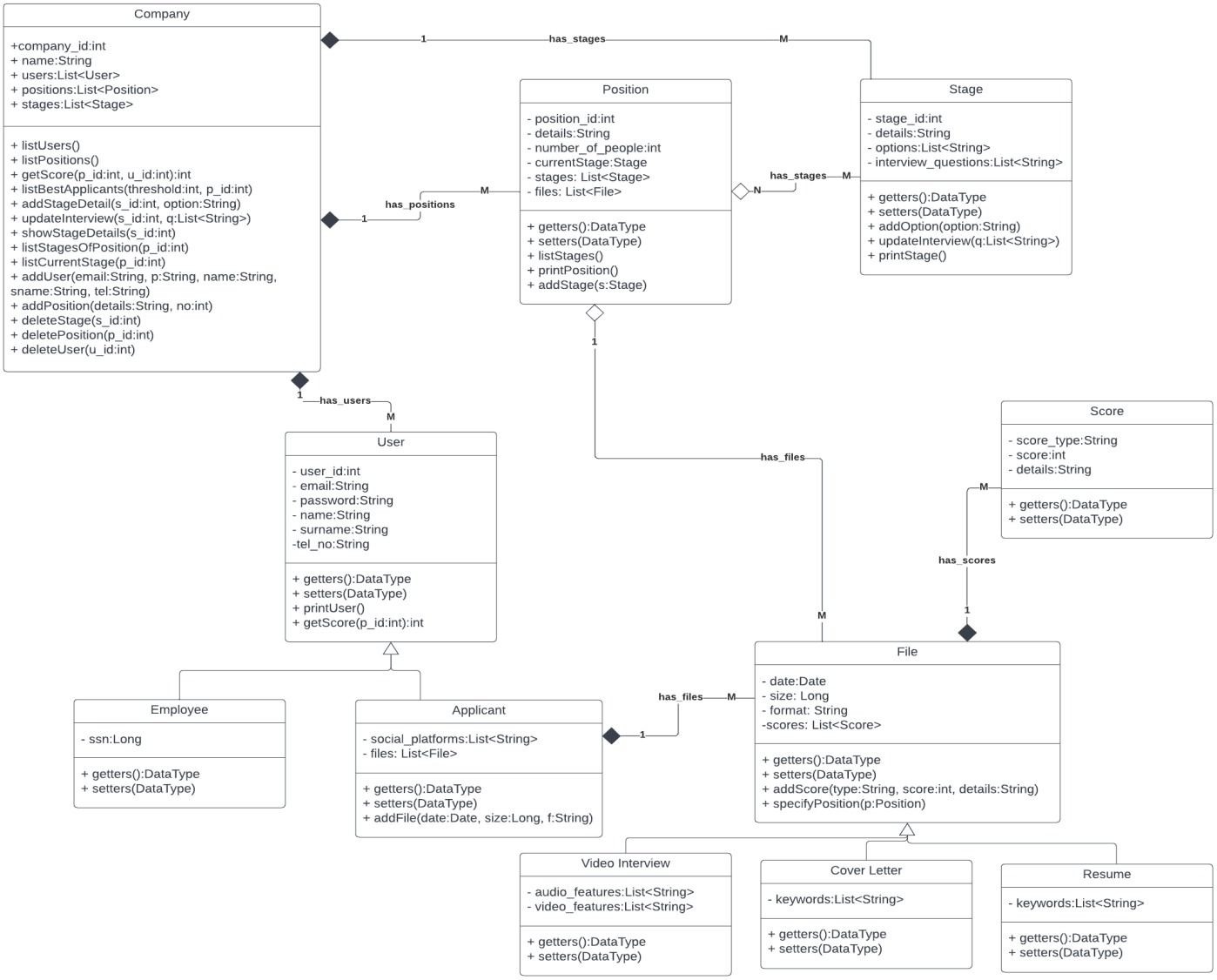

Figure 12: Class Diagram

4.2 Process View

The process view shows how the system is composed of interacting processes.

4.2.1 Activity Diagram

Figure 2 shows the activity diagram which describes dynamic aspects of the system. Once the applicant is accepted to the first stage which means s/he got above the threshold of the matching resume and cover letter keyword extraction, there are three parts to do parallelly. The system extracts the emotion from the applicant's asynchronous video interview, CV and Cover Letter part is for evaluating contents of them, the technical test for measuring the applicant's level. The company can customize the technical test questions as they wish.

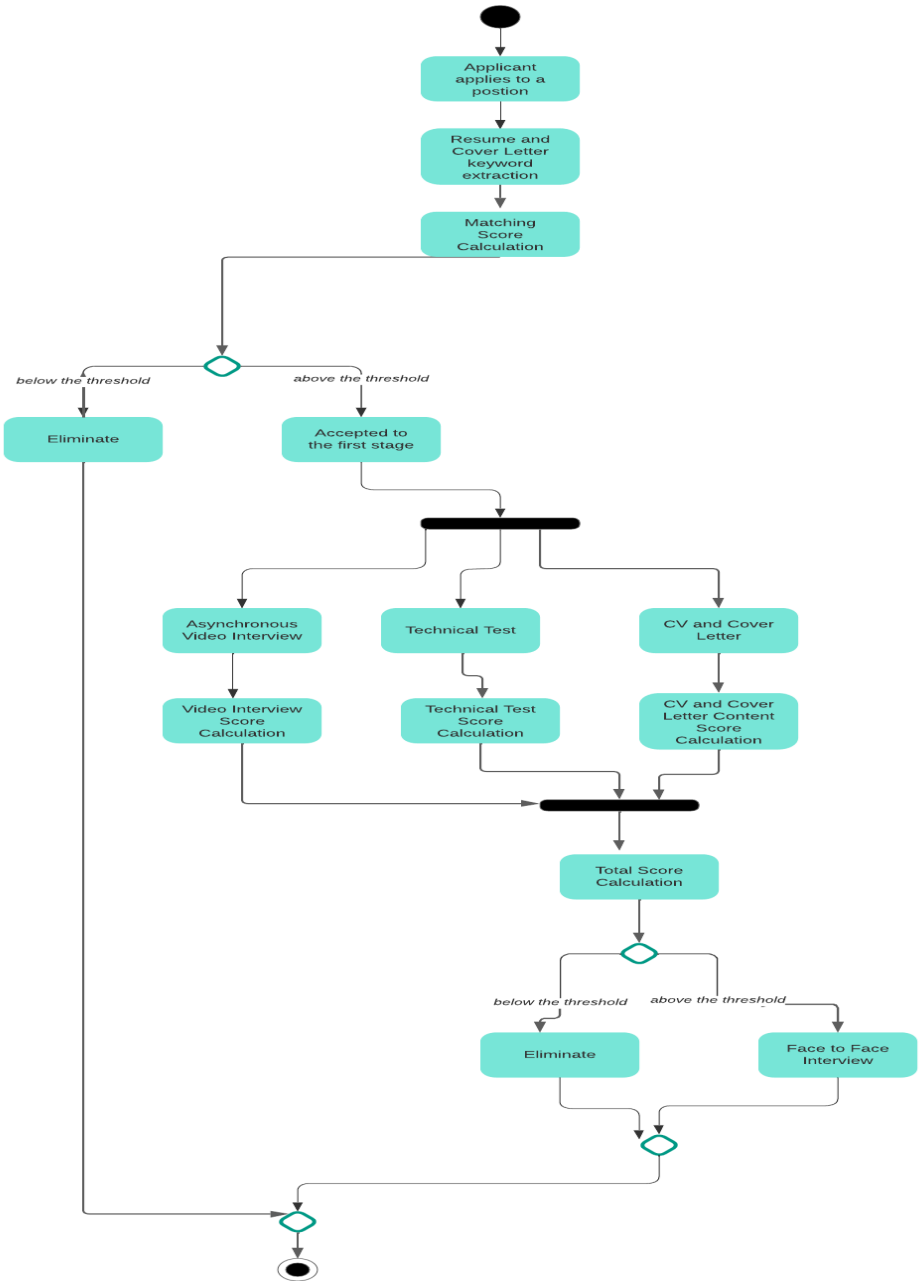

Figure 13: Activity Diagram

4.2.2 Sequence Diagrams

The following figures display the sequence diagrams, which illustrate the user interactions with forms and the way in which the information is processed.

Registration: The user will register to the system by entering their name, password, and e-mail address. Suppose the user enters an existing e-mail or weak password (not using a mix of alphabetical and numeric characters). In that case, the system will send an error message and ask the user to enter their information again until they enter it successfully. After that, the user will be added to the user table, and the system will send a confirmation message to the user.

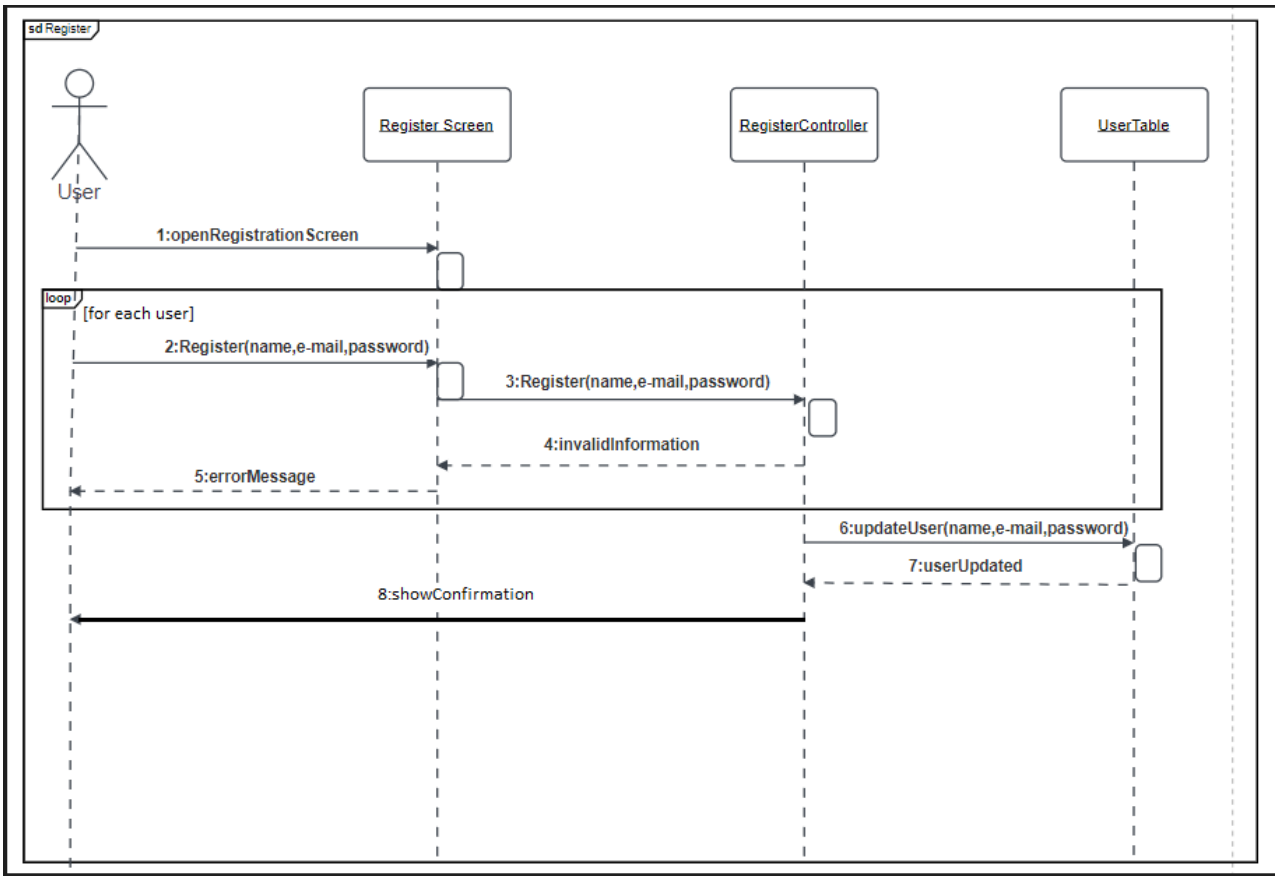

Figure 14: Registration Sequence Diagram

Video Interview Submission: The user will enter the submission screen and upload their video interview file, including the date, file size, and file format. If they exceed the video limit of 15 minutes, the system will send an error message and ask the user to fix the video length. After the user uploads the file with no problem, the system will add the file to Video Interview Table and send a confirmation message to the user.

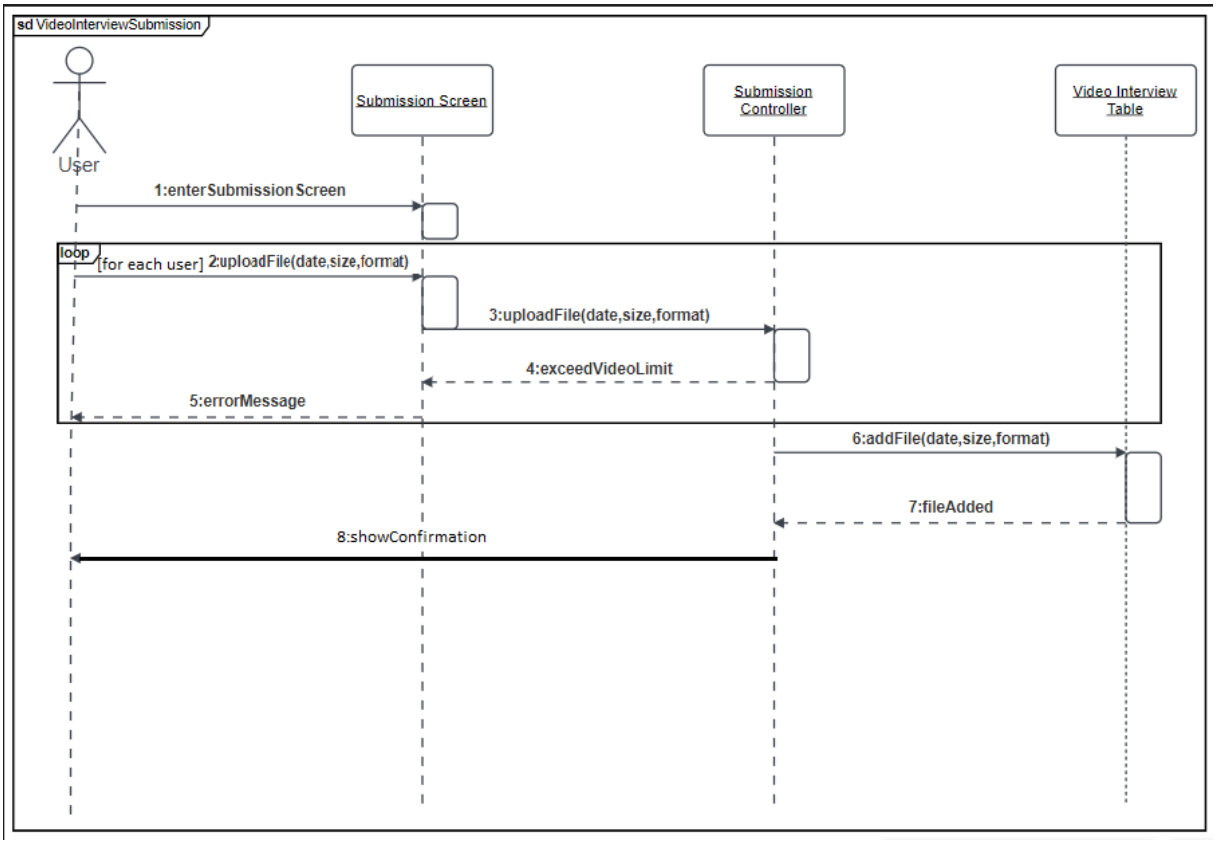

Figure 15: Video Interview Submission Sequence Diagram

Technical Question Submission: The company can create a random technical test, which means the questions will be selected randomly from the question pool, or they can create this technical test by specifying the exact questions from the question table. After that, a technical test will be created, and the user will be notified. Then, they will solve questions and submit their answers. The system will calculate the user's score and store it in the score table. When all the process is done, the user will get a confirmation message that their technical test has been submitted successfully.

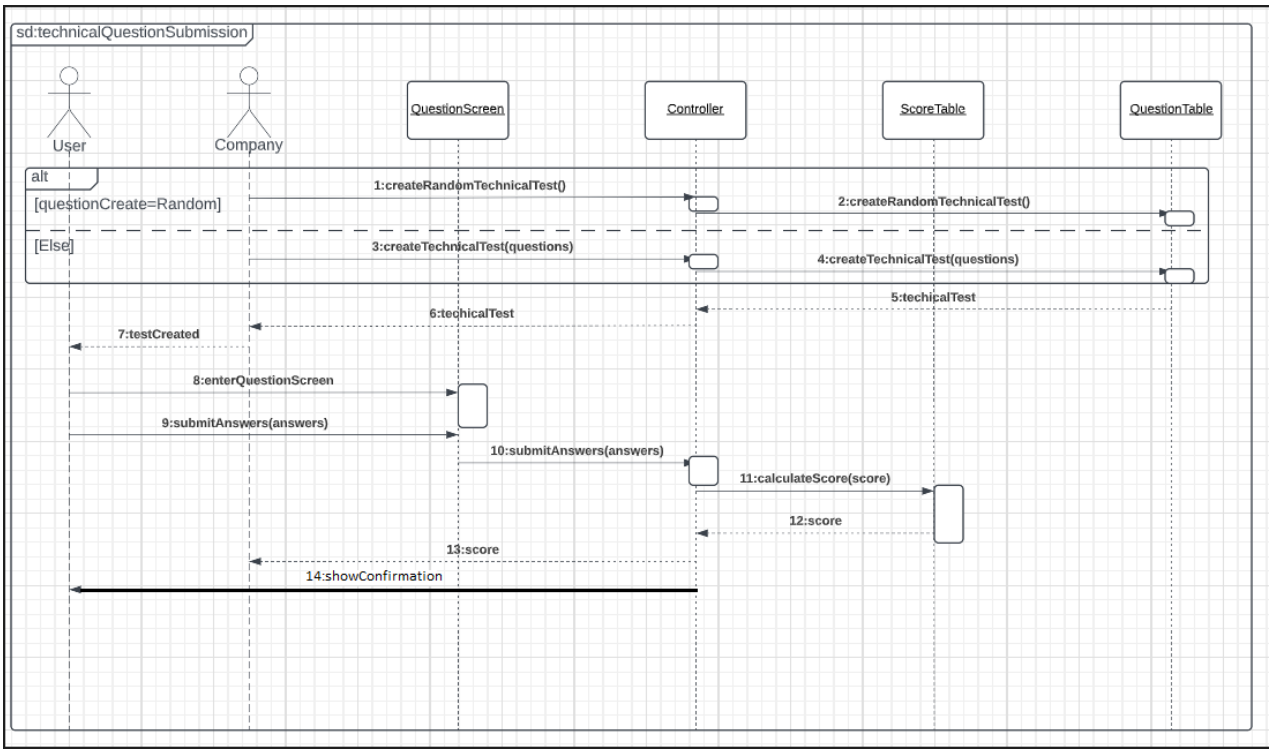

Figure 16: Technical Question Submission Sequence Diagram

Extract Video Features: AI creates a video folder. It takes interview videos of the users from the video interview table. After that, it will extract the video features of the users. Then, it will calculate the user's score and save it in the score table. When all the process is done, the system shows a confirmation message that the user's score is saved successfully.

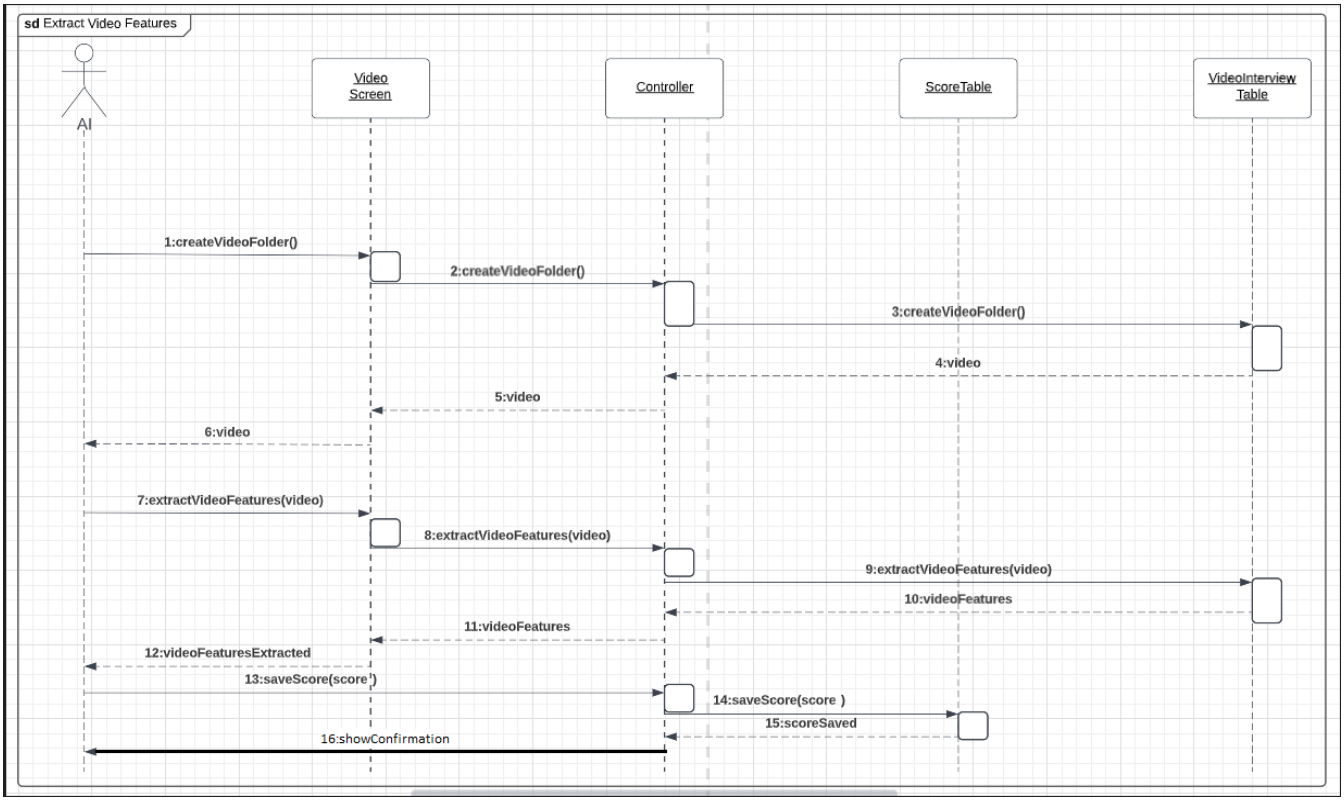

Figure 17: Extract Video Features Sequence Diagram

4.2.3 Data Flow Diagrams

Figure 18 illustrates our automatic recruitment system at the context level. It shows the system as a single process with its relationship to external entities. The system has two external entities: applicants and company employees. The system's input and output data are indicated by incoming/outgoing arrows, which help to display how data flows between those entities and the system during the main functions.

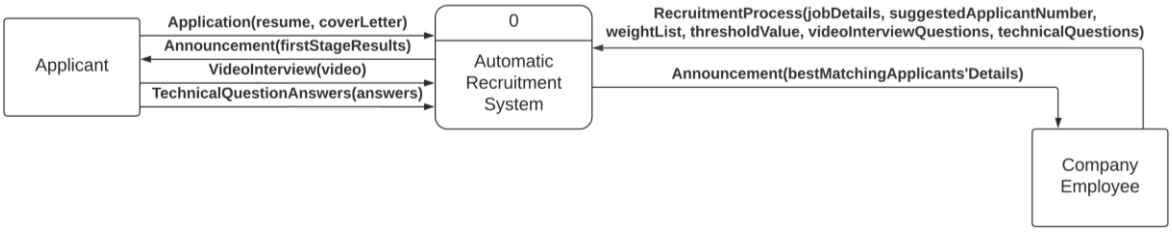

Figure 18:DFD Context Level

Figure 19 illustrates the level-0 of our automatic recruitment system. It is an expanded version of the context-level diagram that has been decomposed into multiple processes to highlight the main functions of the system. The diagram includes 8 processes in total, and illustrates data saving and retrieval through the use of incoming/outgoing arrows with the database entity.

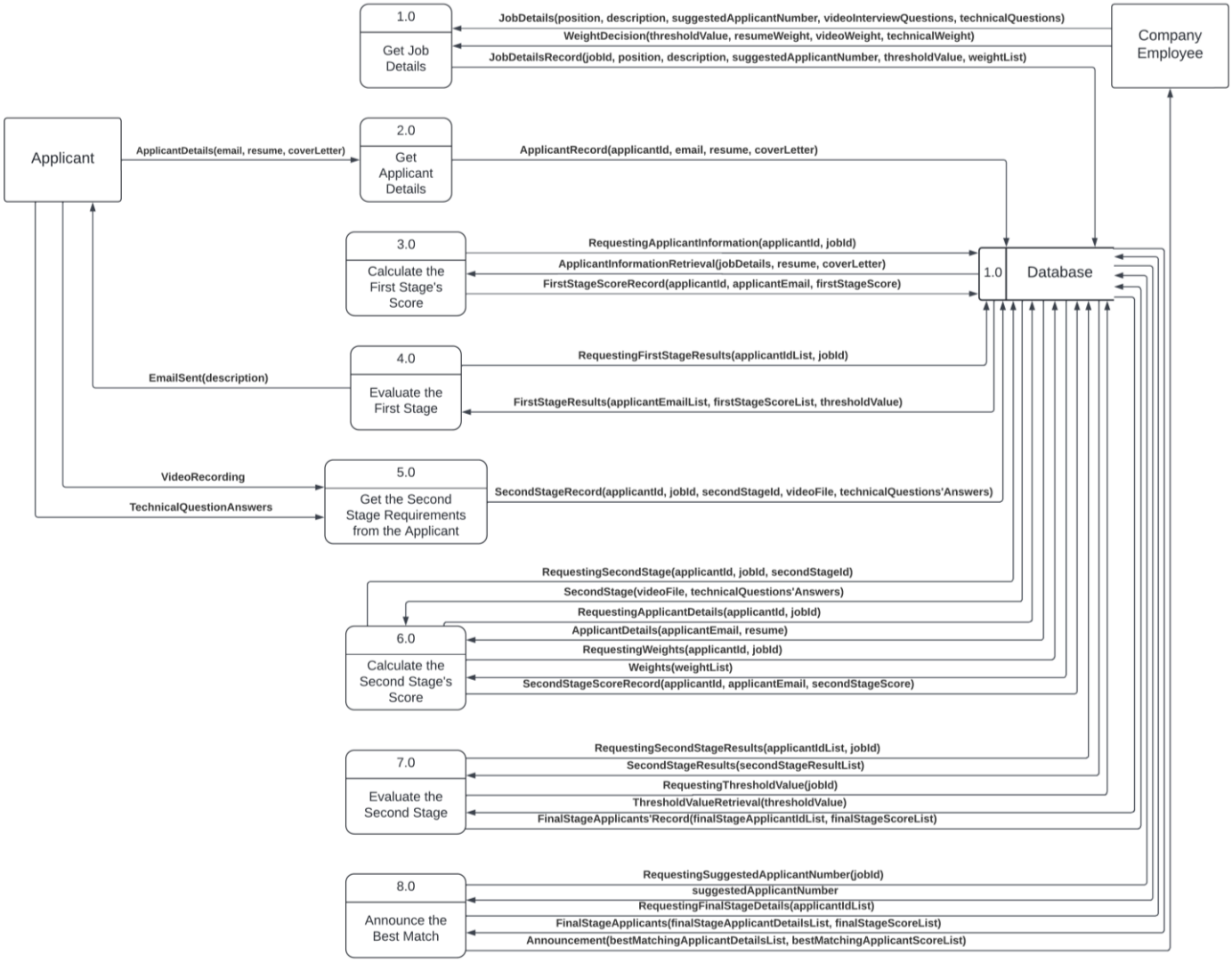

Figure 19: DFD Level 0

The Level-1 data flow diagram provides a detailed view of the processes and data flows in our automatic recruitment system. Figures 20 through 27 show the Level-1 data flow diagrams as a breaking down version of the eight high-level functions shown in the Level-0 diagram into smaller, more specific subprocesses. This provides a more in-depth understanding of how the system handles data. The collective arrows in the diagram show the joint functions, and the arrows between the two subprocesses indicate the data flow between them. The subprocesses are invoked by starting from x.1 and continuing through the left, which is similar for each data flow. Data flows starting with the keyword "Requesting" are for database queries.

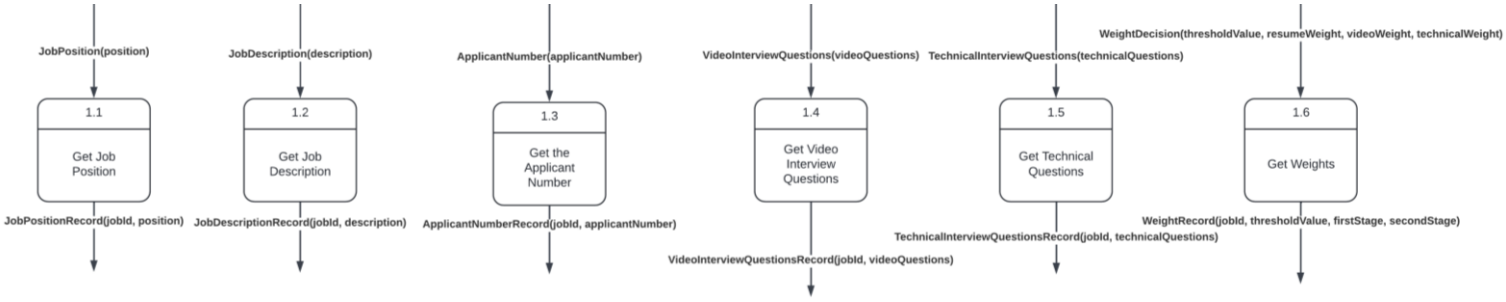

Figure 20: DFD Level-1 Subprocesses of Process 1

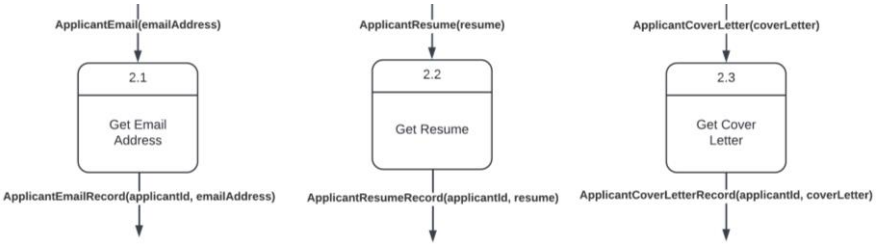

Figure 21: DFD Level-1 Subprocesses of Process 2

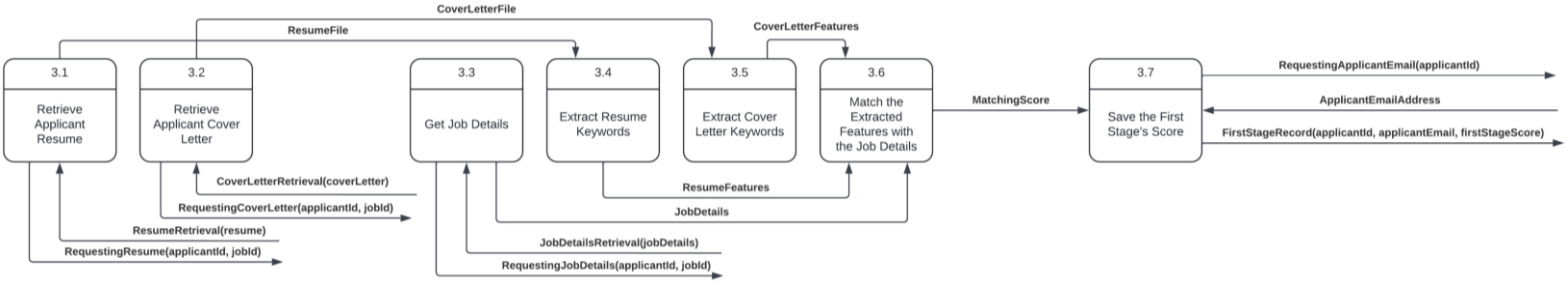

Figure 22: DFD Level-1 Subprocesses of Process 3

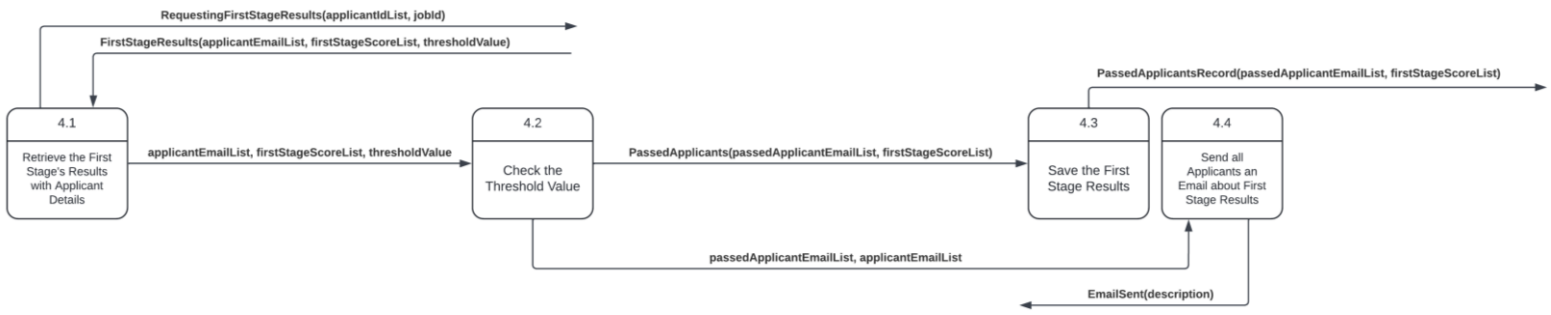

Figure 23: DFD Level-1 Subprocesses of Process 4

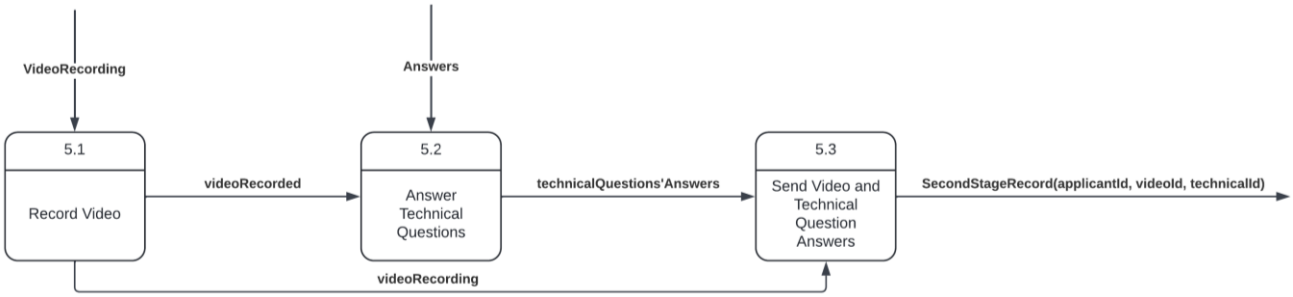

Figure 27: DFD Level-1 Subprocesses of Process 5

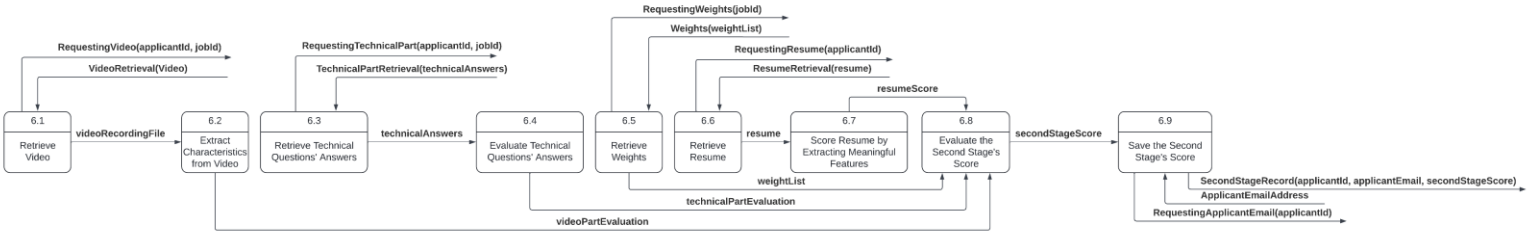

Figure 26: DFD Level-1 Subprocesses of Process 6

Figure 25: DFD Level-1 Subprocesses of Process 7

Figure 24: DFD Level-1 Subprocesses of Process 8

4.3 Development View

The development view shows how the software is decomposed for development.

4.3.1 Component Diagram

The diagram shown as Figure 28 describes system's components. This component diagram is created considering the classes of the class diagram of the system. File component has Video Interview, Cover Letter, and Resume components, and the User component has Employee and Applicant components. Company is a required interface for Position and User. Similarly, File is a required interface for Score, Position is a required interface for File. The stage is a provided interface for Position. Finally, the file is a provided interface for Applicant.

The score will be computed based on scoring the resume, cover letter, and video interview's answers based on the weights set by the company employee, so the overall score of that applicant will be calculated. Technical tests are a part of stage component. Therefore, technical tests are conducted in the Stage.

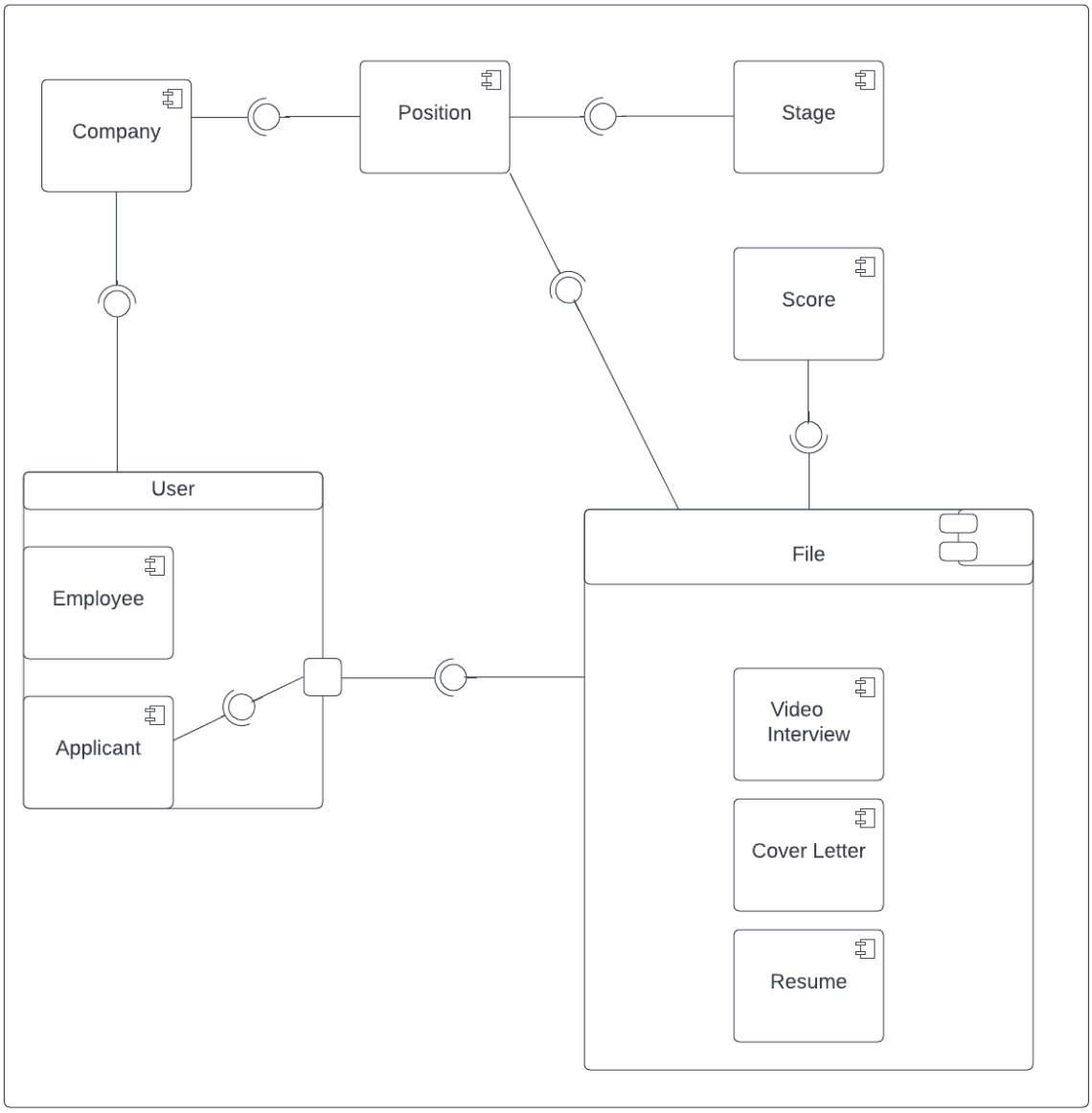

Figure 28: Component Diagram

4.4 Physical View

The physical view shows the mapping of software onto hardware.

4.4.1 Deployment Diagram

Figure 29 shows our deployment diagram. To design it, we used our component diagram and connected the nodes accordingly. All our components in the same node as our system will be hosted together. Moreover, our system uses an external job searching engine.

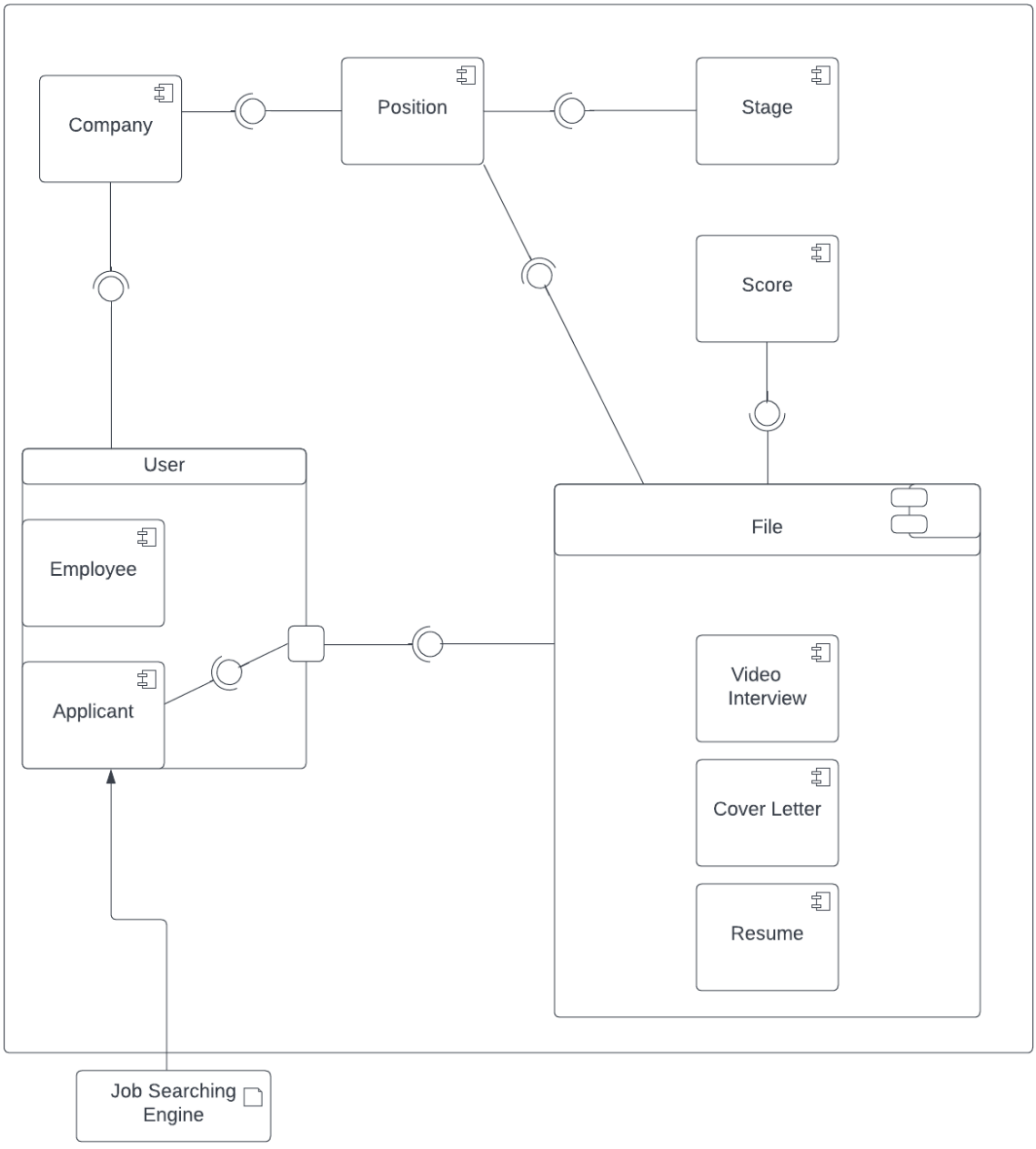

Figure 29: Component Diagram

5 Project Scheduling

5.1 Milestones and Tasks

ID	Task	Duration	Task Dependency	Responsible Member
T1	App research and analysis	1		All
T2	Use case description for SRS	1	T4	Fatih & Ata
T3	Database modelling (ERD)	0.5		Doğukan
T4	Use case diagram design	0.5		Doğukan
T5	Software estimation	0.5	T1, T3, T4	All
T6	Algorithm research and analysis	1.5	T1	Doğukan & Melisa
T7	User interface prototype design	1		Ata
T8	SRS report writing	2		All
T9	Class diagram design for SDD	0.5	T3	Doğukan
T10	Activity diagram design for SDD	0.5	T8(M1)	Fatih
T11	Sequence diagram designs for SDD	0.5	T2, T4	Ata
T12	Data flow diagram designs for SDD	0.5	T8(M1)	Melisa
T13	Component and deployment diagram design for SDD	0.5		Fatih
T14	Project scheduling for SDD	1.5	T8(M1)	Doğukan
T15	SDD report writing	2		All
T16	Create database with C#	1	T2, T4	Fatih
T17	Connections with database and backend through C#	1.5	T16	Fatih
T18	Connection between database and python AI	1.5	T16	Doğukan
T19	Emotion Detection From audio of video Interview	2		Doğukan & Melisa
T20	Answer correctness detection from audio of video interview	2		Doğukan & Melisa
T21	Personality trait, emotion detection from video of video interview	3		Doğukan & Melisa
T22	Applicant analysis screen for video interview scores	3		Ata
T23	Video interview submission screen	2		Ata
T24	Connection between C# backend Python AI	0.5	T25, T26	Fatih & Doğukan
T25	API for AI	1		Doğukan
T26	API for C# backend	1.5		Fatih
T27	Filtering Options and implementations for search screens(Query)	1	T16, T22, T26	Fatih & Ata
T28	Frontend connections with C# backend	1	T26	Ata
T29	Prepare the platforms for prototype deployment	1	T8(M1), T15(M2), T16(M3), T21(M4), T23(M5), T28(M6)	All

Table 28: Tasks

ID	Milestone	Date
M1	SRS	Week 5
M2	SDD	Week 9
M3	Database creation	Week 12
M4	Video interview processing	Week 14
M5	User interface design	Week 14
M6	Standard software connections	Week 14
M7	Prototype deployment	Week 15

Table 29: Milestones

5.2 Gantt Chart

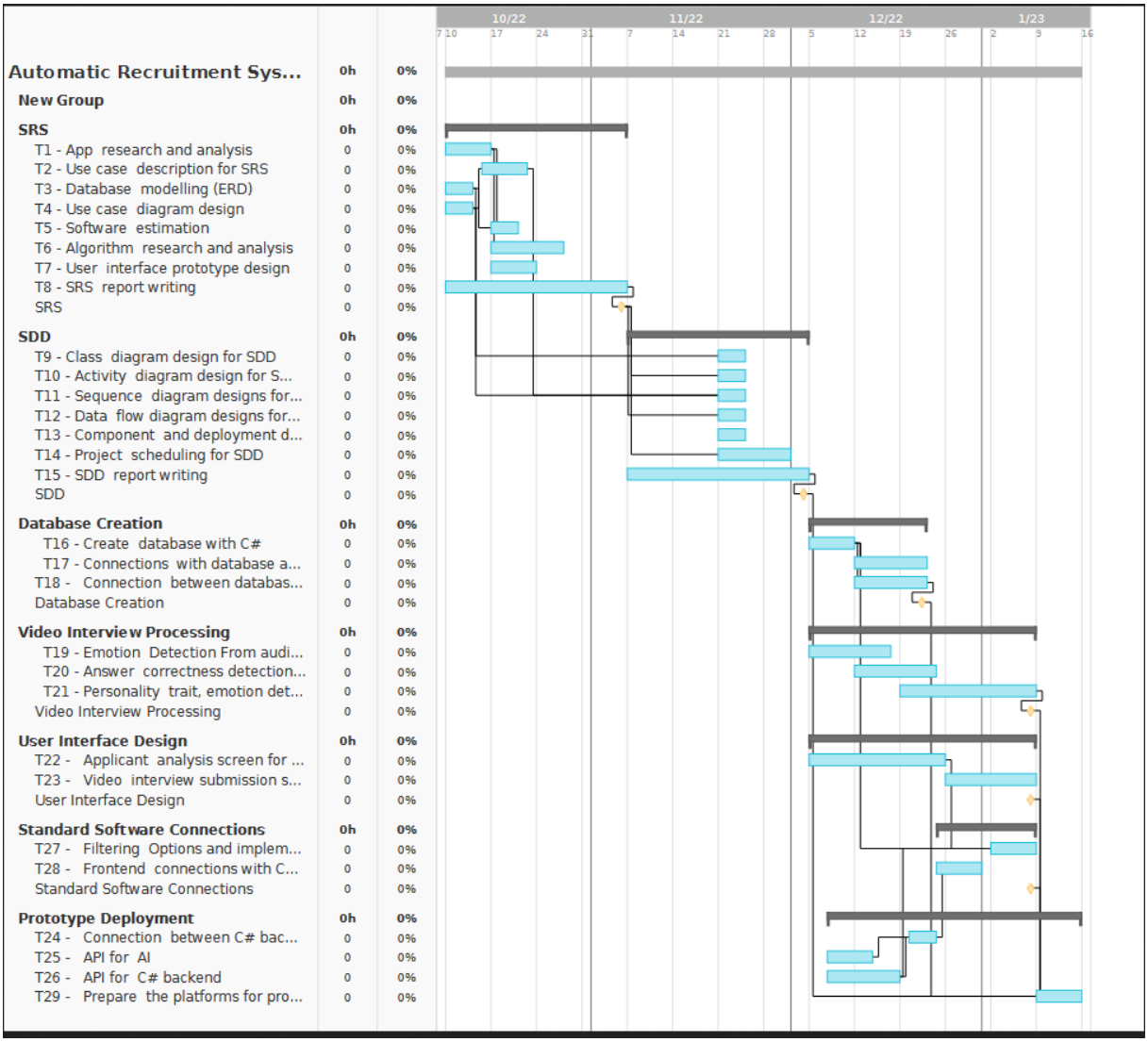

Figure 30: Gantt Chart

6 Conclusion

6.1 Critical Evaluation and Conclusions

Our team identified that the current job recruitment process is biased, consumes a lot of time and money, and poses a significant problem. As a solution, we proposed an application that addresses these issues by incorporating all of the necessary steps for an effective recruitment process. Before we began reviewing the literature, we identified and extracted the key features that should be present in a job recruitment application. During our research, we found many recruitment applications, but none of them offered all of the functionalities that we deemed necessary for our application. Our application offers recruiters an end-to-end, automated recruitment process that is unbiased, easy to follow, and efficient.

During the first semester, our main focus was on developing the core functionalities of our application. We were able to successfully complete several key features including the user interface for the applicant search screen and video interview submission screen. Additionally, we developed a system for extracting personality traits from audio and video portions of the applicants' interviews. We also completed the necessary backend connections and functionalities, as well as the database. As a result, we were able to release the first prototype of our application. The frontend of the application closely resembles what we outlined in our SRS report, the backend has been expanded with more endpoints and services, and our AI system is performing with a high level of accuracy in forecasting personality traits from video.

6.2 Retrospective

At the beginning, we organized the order of duties unsuccessfully, resulting in poor time management. Lack of communication leads to misunderstanding of some project concepts. Also, we did not divide work among group members successfully. To solve these problems, we conducted a meeting and decided to analyze our duties and responsibilities more carefully and put them in chronological order; accordingly, we planned to use the parallelism technique for the duties that can be conducted separately and increase our communication with each other to be a better team overall. As a result, we successfully divided our tasks and could work in parallel. On the other hand, we poorly estimated workload for the tasks and have some time management problems at the end of the semester. Unfortunately, workload for each member increased drastically. At the end, we supported each other and managed to complete tasks for this semester successfully.

6.3 Future Works

For the future works, we have multiple things to do. For the frontend part, we will complete the remaining screens with their relevant connections to the backend, such as the login screen, position screen, and post positions screen. In addition to this, the screens, which are already implemented, will be improved. For the backend part, resume and cover letter keyword extraction and comparison functionality will be added, login functionality will be implemented, and it will be integrated to the frontend, such that different users with different roles will see separate screens. For example, applicants will be directed to the video submission screen, but company employees will be directed to the position, and user management screens. Moreover, stage customization and technical test parts will also be added to the application. Finally, for the AI part, we will improve the model we built to receive better big-five trait estimations. Furthermore, we will score resumes and cover letters according to their content and try to score answers given to the asynchronous video interview questions by applicants.

7 References

1. Anand, S., Gupta, N., Mulay, M., Sherawat, A., & Gangarde, R. (2021). Personality Recognition & Video Interview Analysis. *International Journal of Engineering Research & Technology (IJERT)*, *10*(05).
2. Bika, N. (2022, July 5). *Recruiting costs FAQ: Budget and cost per hire*. Recruiting Resources: How to Recruit and Hire Better. Retrieved from https://resources.workable.com/tutorial/faq-recruitment-budget-metrics.
3. Black, J. S., & van Esch, P. (2020). Ai-enabled recruiting: What is it and how should a manager use it? Business Horizons, 63(2), 215–226. Retrieved from https://doi.org/10.1016/j.bushor.2019.12.001
4. Booth, B. M., Hickman, L., Subburaj, S. K., Tay, L., Woo, S. E., & D'Mello, S. K. (2021). Bias and fairness in multimodal machine learning: A case study of automated video interviews. *Proceedings of the 2021 International Conference on Multimodal Interaction*. Retrieved from https://doi.org/10.1145/3462244.3479897.
5. Chen, L., Feng, G., Leong, C. W., Lehman, B., Martin-Raugh, M., Kell, H., Lee, C. M., & Yoon, S.-Y. (2016). Automated scoring of interview videos using doc2vec multimodal feature extraction paradigm. *Proceedings of the 18th ACM International Conference on Multimodal Interaction*. Retrieved from https://doi.org/10.1145/2993148.2993203.
6. Chen, L., Zhao, R., Leong, C. W., Lehman, B., Feng, G., & Hoque, M. E. (2017). Automated video interview judgment on a large-sized corpus collected online. *2017 Seventh International Conference on Affective Computing and Intelligent Interaction (ACII)*. https://doi.org/10.1109/acii.2017.8273646
7. CV. Cambridge Dictionary. (n.d.). Retrieved from https://dictionary.cambridge.org/dictionary/english/cv.

8. Fernández-Reyes, F. C., & Shinde, S. (2019). CV Retrieval System based on job description matching using hybrid word embeddings. *Computer Speech &Amp; Language*, *56*, 73–79. https://doi.org/10.1016/j.csl.2019.01.003
9. Gupta, P., Fernandes, S. F., & Jain, M. (2018). Automation in recruitment: A new frontier. Journal of Information Technology Teaching Cases, 8(2), 118–125. Retrieved from https://doi.org/10.1057/s41266-018-0042-x.
10. Jie Chen, Chunxia Zhang, Zhendong Niu, "A Two-Step Resume Information Extraction Algorithm", Mathematical Problems in Engineering, vol. 2018, Article ID 5761287, 8 pages, 2018. Retrieved from https://doi.org/10.1155/2018/5761287.
11. Lukacik et al., 2022 E. Lukacik, J.S. Bourdage, N. Roulin Into the void: A conceptual model and research agenda for the design and use of asynchronous video interviews Human Resource Management Review, 32 (1) (2022), p. 100789, Retrieved from https://doi.org/10.1016/j.hrmr.2020.100789.
12. Park, J. Y., & Ko, C. B. (2022). Proposal for AI Video Interview Using Image Data Analysis. *International Journal of Internet, Broadcasting and Communication*, *14*(2). Retrieved from https://doi.org/10.7236/IJIBC.2022.14.2.212.
13. Patel, P. (2019, May 14). *Hiring*. Kaggle. Retrieved from https://www.kaggle.com/datasets/pankeshpatel/hiring.
14. Pavansubhash. (2017, March 31). *IBM HR Analytics Employee Attrition & Performance*. Kaggle. Retrieved from https://www.kaggle.com/datasets/pavansubhasht/ibm-hr-analytics-attrition-dataset
15. Pena, A., Serna, I., Morales, A., & Fierrez, J. (2020). Bias in multimodal AI: Testbed for fair automatic recruitment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops* (pp. 28-29).

16. Rafter, R., Bradley, K., & Smyth, B. (2000). Automated collaborative filtering applications for online recruitment services. *Lecture Notes in Computer Science*, 363–368. Retrieved from https://doi.org/10.1007/3-540-44595-1_48.
17. Recruiting automation software market. Future Market Insights. (n.d.). Retrieved from https://www.futuremarketinsights.com/reports/recruiting-automation-software-market.
18. Shaiq, D. (2021, October 6). *Synthetic e-recruitment (user) dataset*. Kaggle. Retrieved from https://www.kaggle.com/datasets/daniyalshaiq/synthetic-erecruitment-user-dataset
19. St-Jean, E. and Thibodeau, P. (2022) What is an applicant tracking system (ATS)?, SearchHRSoftware. TechTarget. Retrieved from https://www.techtarget.com/searchhrsoftware/definition/applicant-tracking-system-ATS.
20. Waung, Marie & Hymes, Robert & Beatty, Joy. (2014). The Effects of Video and Paper Resumes on Assessments of Personality, Applied Social Skills, Mental Capability, and Resume Outcomes. Basic and Applied Social Psychology. 36. 238-251. 10.1080/01973533.2014.894477.
21. What is an API? (n.d.) Amazon. Available at: https://aws.amazon.com/what-is/api/?nc1=h_ls.
22. Wijaya, D. (2020, September 5). *Employee turnover*. Kaggle. Retrieved from https://www.kaggle.com/datasets/davinwijaya/employee-turnover.
23. *Ponce-López, V., Chen, B., Oliu, M., Corneanu, C., Clapés, A., Guyon, I., Baró, X., Escalante, H. J., & Escalera, S. (2016). Chalearn lap 2016: First Round Challenge on first impressions - dataset and results. Lecture Notes in Computer Science, 400–418. https://doi.org/10.1007/978-3-319-49409-8_32*

8 Appendices

8.1 Acronyms and abbreviations

- CV: Curriculum Vitae
- ATS: Applicant Tracking System
- API: Application Programming Interface

- ACF: Automated Collaborative Filtering
- SSP: Social Signal Processing
- BOW: Bag-Of-Words
- BOVW: Bag-Of-Visual-Words
- LIWC: Linguistic Inquiry Word Count
- K-NN: K-Nearest Neighbors
- NLP: Natural Language Processing
- FACS: Facial Actions Coding System
- SVM: Support Vector Machine
- CNN: Convolutional Neural Network
- DNN: Deep Neural Network

8.2 Glossary

N/A