Bilkent University
Department of Computer Engineering

Senior Design Project
T2406
Recruit4Me

Analysis and Requirement Report

22101007 Kanan Zeynalov kanan.zeynalov@ug.bilkent.edu.tr
22003108 Musa Yiğit Yayla yigit.yayla@ug.bilkent.edu.tr
22103449 Burak Efe Öğüt efe.ogut@ug.bilkent.edu.tr
22003416 Mehmet Ege Kayaselçuk mehmet.kayaselcuk@ug.bilkent.edu.tr
22003672 Bahadır Günenc bahadir.gunenc@ug.bilkent.edu.tr
Fazlı Can
Mert Bıçakçı, Atakan Erdem

13.12.2024

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfilment of the requirements of the Senior Design Project course CS491/2.

Contents

Analysis and Requirement Report
Recruit4Me

1 Introduction

Recruit4Me aims to provide a web based software platform for hiring IT & tech professionals while hosting numerous features enhanced by technologies including AI & ML. Project proposes high innovation through two main stages, automated candidate elimination and pay as you go interviewer expert hiring done by companies. This way companies with limited HR resources and large applicant pools will reduce costs by AI automation, followed by HR & technical expert hiring for conducting candidate elimination. Thus, with pay as you go hiring, further employment opportunities will be introduced for interviewer experts, who also may be employed in some full time job, proposing extra revenue. First stage of Recruit4Me will consist of automation tools, which companies can use, including CV parsing ML model labeling candidates' technical and soft skills, AI generated technical questions relevant to the job position, and speech to text models. This first phase will reduce HR labor cost by performing automated and accurate preliminary candidate elimination. Furthermore, companies will be able to customize these models with respect to their professional and ethical principles. Subsequently, the second phase, interview expert hiring, will involve companies employing HR and technical experts for managing their recruitment. Given permission, interview experts will also manage candidate pools and introduce new potential candidates. Recruit4Me will also include searching and filtering candidates' profiles by their skills, job listings by required expertise and skills, enabling further interaction.

2 Current System

Existing technical recruitment platforms, such as Codility and HackerRank, primarily focus on automated coding tests to evaluate candidates' technical abilities. While this technique has widespread usage, they often limit the scope to assessing programming skills under time constraints, which some candidates find stressful and differ from real-world development scenarios. Additionally, these solutions may rely on a smaller candidate pool that contains only the people who apply for the job whereas our solution aims to have a larger CV bank. Recruit4Me will also integrate cultural fit assessments.

Our approach differs by extending beyond coding exams and by creating a structured hiring flow, we incorporate thorough cultural and soft skill evaluations into the screening process. Moreover, we actively ask for detailed

feedback from candidates regarding their interview experience to continuously develop our system and ensure fairness, accessibility, and candidate satisfaction.

3 Proposed System

3.1 Overview

Our analysis shows that an integrated AI-driven recruitment platform will do a better job in selecting suitable candidates than the current solutions based on coding tests. By integrating automated resume parsing, technical and cultural assessments, and AI-driven interviews, all informed by continuous candidate feedback, the system aims to reduce recruiter workload, expand the candidate pool, improve evaluation quality, and provide a more positive candidate experience.

3.2 Functional Requirements

- Users should upload their resumes in either pdf or word (docx) format.
- The required skill sets should be extracted from the resumes
- Candidates should be filtered based on the companies' requirements
- Technical and non-technical questions should be generated based on company's need and required skill set
- Online tests should be created
- Candidates answers should be evaluated
- While answering the questions, candidates should be recorded and those records should stored at most 2 weeks
- Real time speech-to-text should be used
- Analyze the candidates answer's relevance and correctness
- Candidates should also be rated based on their communication skills
- The system should discard some of the candidates without human interaction. While eliminating some candidates, the system should consider resumes and candidates' test scores.
- The candidates who are successful in automated elimination should be called for an HR interview.
- Both candidates and HR specialists should be informed about the interview by notifications.
- Candidates should be informed about their performance and get feedback.
- System should be IT specific and evaluations are made on following specialities:
    - Data Scientist
    - Data Engineer
    - Machine Learning Engineer
    - Big Data Engineer

    - Software Engineer
    - Fullstack Engineer
    - DevOps Engineer
    - Embedded Systems Engineer
    - Cloud Engineer
    - Cybersecurity Engineer
- Based on candidates performance, reports should be generated
- Authentication should be used to distinguish the HRs, candidates, companies and tech specialists
- External tools also be used for retrieving more information about candidates
- Integration with third party scheduling programs should be implemented.

3.3 Non-functional Requirements

- **Usability**

  Recruit4Me prioritizes a user-friendly and intuitive interface to ensure smooth navigation for all users, including candidates, HR personnel, and experts. The system will feature responsive design, accessible functionality compliant with WCAG standards, and real-time feedback through form validation and tooltips. Multi-language support and a built-in FAQ system are expected to enhance user experience and accessibility, making the platform approachable for a diverse audience.

- **Reliability**

  The application will be implemented to have a high reliability, which should be fault-tolerant. Robust error handling and proper data management using ACID-compliant databases will assist these to make an advancement for reliability. Load balancers, failover servers and automated backups with efficient fallback mechanisms are planned to be implemented to maintain an uninterrupted service experience for the users.

- **Performance**

  Performance optimization is a core part of Recruit4Me, with fast response times for key features like login, CV parsing and getting candidate results. Asynchronous operations and caching will reduce server load, while database optimization ensures efficiency even with large datasets. Stress testing guarantees the system can handle significant user traffic and heavy workloads without degrading performance.

- **Supportability**

  Recruit4Me is built with modularity and maintainability in mind, ensuring easy debugging, updates, and enhancements. Comprehensive logging, monitoring, and documentation support developers, while CI/CD pipelines and extensive test coverage enable safe and seamless deployment of updates without disrupting user operations.

- **Scalability**

  The application will be designed to handle increased workload during peak times and user-load through proper cloud integration(AWS, Google) and database optimization methods such as sharding and replication. It will support horizontal scaling and microservices, to ensure the system can manage growth of workload without affecting performance.

3.4 Pseudo Requirements

1. React will be used for programming the web application's frontend (UI).
2. NodeJs will be used for constructing the web application's backend and logic (API).
3. Machine Learning models will be developed using Python, PyTorch, Transformers, GPT openAI, Llama 3.3, Google cloud GPU to train models.
4. The web application will be hosted on AWS together with the database.
5. The project's source code should be trackable on Azure DevOps using Git.
6. As for the database, MySQL will be used.
7. Machine learning and natural language processing components will be built using Python. NLP pipelines and data annotation will leverage spaCy and Prodigy.

3.5 System Models

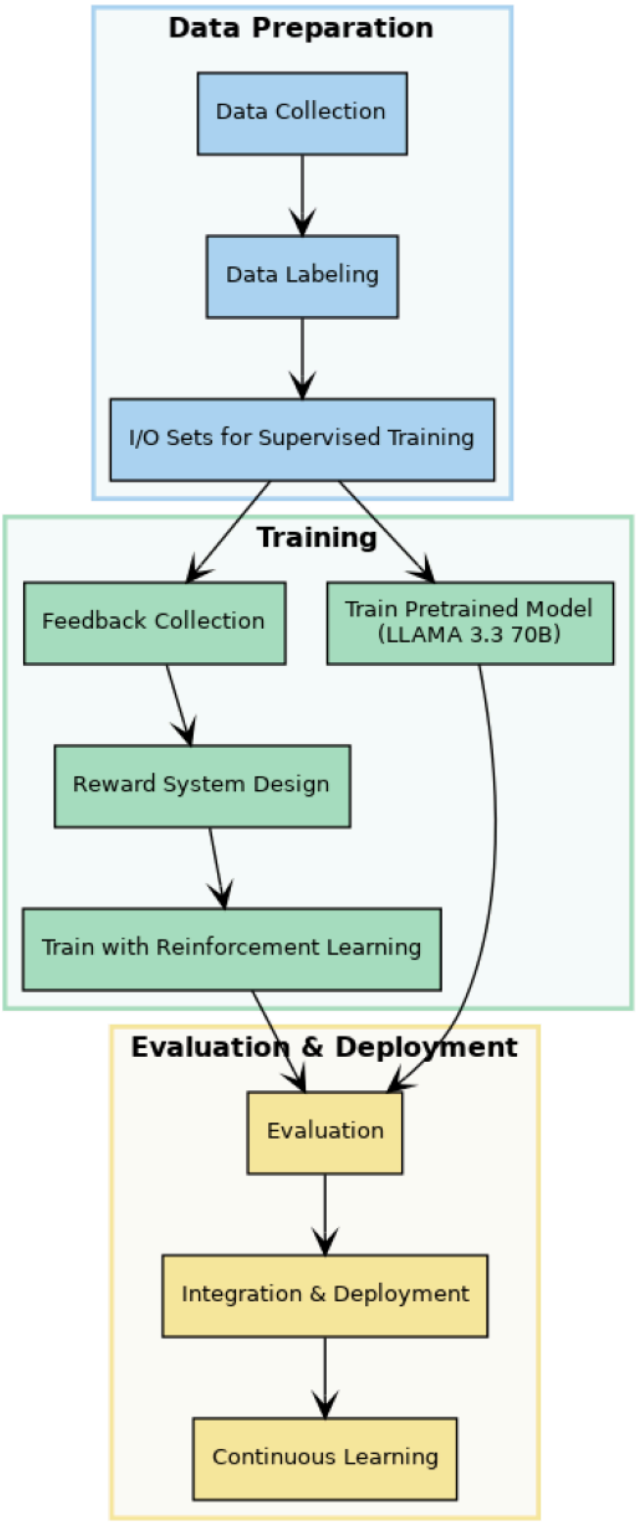

Figure 1. ML Training Pipeline

Data Preparation

We begin with the collection of raw data from various sources, including resumes, job descriptions, and HR records, in order to have a wide variety. Then, specific attributes are annotated on the structured datasets, such as job titles, skills, experience levels, and qualifications, after this raw data has been labeled. These are converted into I/O pairs for supervised learning, with the input including features such as keywords, years of experience, and education, among others, while the output will include job suitability scores, role classification, and a ranked list of candidates. A model is initially set on a path by which it learns the domain-specific pattern that enables it to make an accurate prediction.

Training

We use a pre-trained model, such as LLAMA 3.3 70B, as the base and fine-tune it with our HR-focused dataset to specialize it for certain applications like CV classification and candidate ranking. We need a huge amount of GPU so we are planning to train our model in a cloud service such as Amazon or Google. Since 10GB or more GPU is required for model training it is not possible to handle this in local computers. After fine-tuning we are planning to use reward system by collecting feedback and doing Reinforcement Learning to increase the model performance. Further training by reinforcement learning will allow the model to learn iteratively from that feedback to make better decisions for complex scenarios, like ambiguous profiles or different descriptions of jobs.

Evaluation & Deployment

Testing Performance of the Fine-Tuned Model: We will test the model in various aspects that are necessary in metrics such as accuracy, precision, recall, and speed. This will ensure the capability to perform real-world tasks on things like filtering large pools of candidates. Upon validation, this model will be integrated into an HR system and deployed onto platforms, where automation of tasks on such systems, including CV filtering, question generation, and candidate ranking, can occur. The model deploys into a continuous learning phase, changing and adapting to new incoming data, changing trends in the job market, and user feedback so that over time, it stays accurate and efficient.

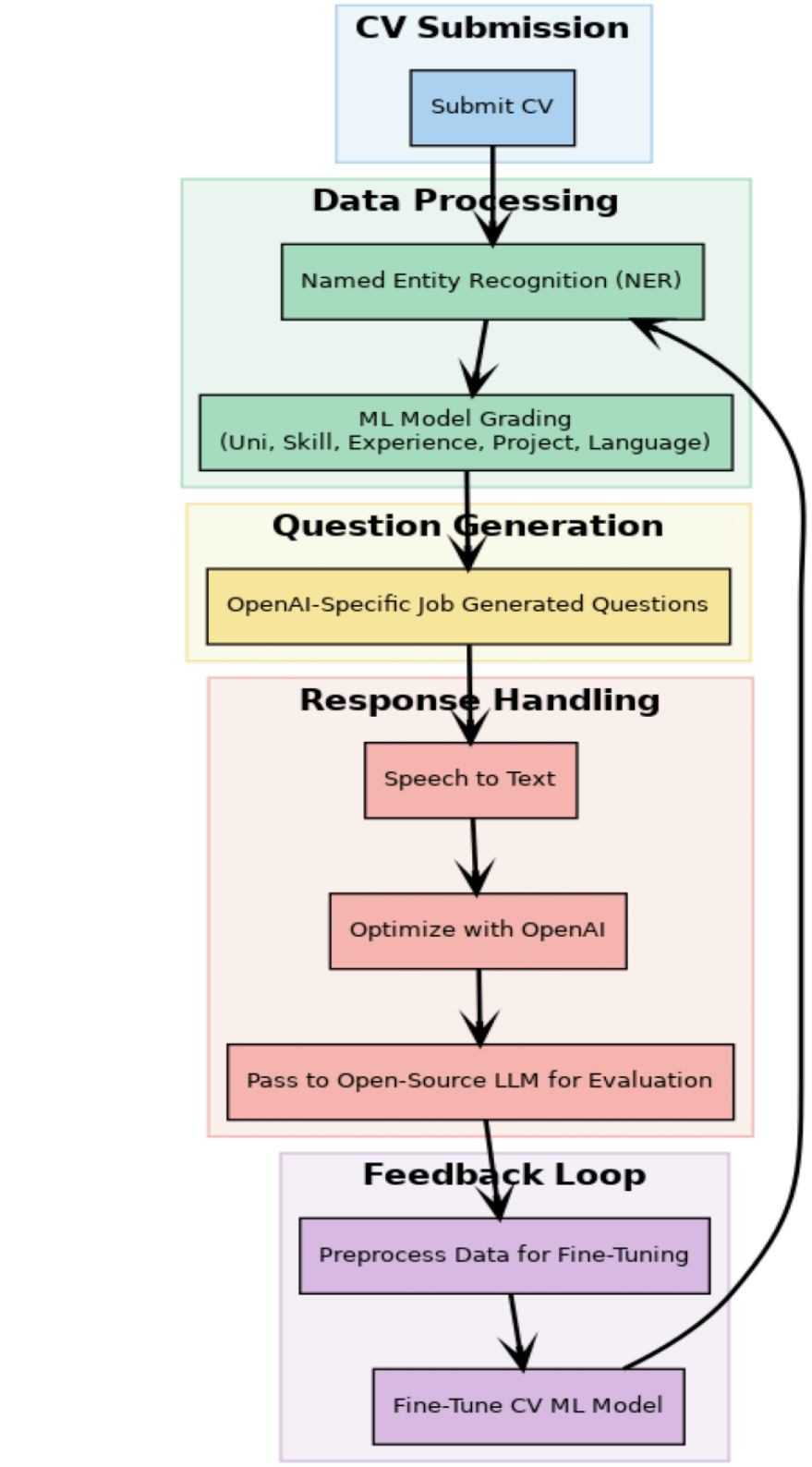

Figure 2. Real-time usage of system

CV Submission and Data Processing

It starts with the candidate sending the CV via a dedicated portal within our automated application handling system that is designed to be compatible with various file formats. With just a few clicks, after it arrives, the system automatically forwards the process into the Data Processing phase and uses advanced NER mechanisms in order to extract and group key elements from the given CV. It includes information about the candidate's educational background, skills, years of experience, project history, and language proficiencies. The extracted information goes through an ML-supported grading module where each component will be checked against set metrics. Scores could be given, for example, based on university ranking, relevance, and depth of skills, quality and size of past projects, or fluency in relevant programming and human languages. These scores form altogether a candidate profile, in which the applicants can be compared on a quantitative basis.

Question Generation and Response Handling

Once the grading phase is complete, the system moves on to Question Generation, utilizing the API provided by OpenAI for the generation of job-specific questions tailored to the candidate's profile and also to the requirements of the target position. The questions are designed to check technical expertise and problem-solving skills and are framed in such a way that they are appropriate for the demands of the role. Candidates answer these questions verbally, and their responses are captured using a robust speech-to-text system. We will use either Amazon’s API or Google’s API integration to handle these. This module guarantees high transcription accuracy even for non-native speakers or individuals with an accent, hence making it all-inclusive. The responses are then optimized using OpenAI tools to ensure that they are clear and relevant, after which they are passed to the LLM open-source for a detailed review. It checks the depth of the answers, logical structure, and overall quality to come up with a final assessment for each candidate.

Feedback Loop and Continuous Improvement

The system has a reinforcing loop to maintain and advance its accuracy and adaptiveness in the long run: the data being processed, aggregated and pre-processed for further use encompasses the CVs, results on entity extraction, grading scores obtained, and evaluation feedback available. Further, this serves as a fine-tunement for the underlying Machine Learning models so that ultimately it has evolved to meet the altering parameters of industry standards and fresh changing attitudes and expectations of candidates at large. Through continuous updates regarding the grading mechanism, question-generation logic, and criteria for marking a response, the system is developed to handle any applicant pool with accurate, consistent scores. In other words, this would indeed ensure that the automata mechanism will stay robust and scalable, allowing a tremendous number of applications without actually hindering quality.

3.5.1 Scenarios

1. Use Case Name: Register as a Candidate

Participating Actors: Candidate
Flow of Events:

1. User navigates to the registration page and selects "Candidate" as his/her role.
2. User fills in their details, including personal information (e.g., name, email, password).
3. User submits the form, and the system sends a verification email to the provided address.
4. User clicks the verification link to confirm their email.
5. System activates the account, allowing the Candidate to log in and use the platform.
6. Users upload their CV and optionally fill in additional profile information after registration page(e.g., skills, experience).

Entry Conditions: The user lands on the registration page and selects "Candidate" as their role.
Exit Conditions: Candidate account is created, verified, and active.
Quality Requirements: A valid email must be provided and verified via the confirmation link. CV upload must be completed to enable account activation.

2. Use Case Name: Register as a Company Representative

Participating Actors: Company Representative
Flow of Events:

1. User navigates to the registration page and selects "Company Representative" as their role.
2. Users fill in their details, including personal information and a domain-specific email address. (@companyname included in mail address)
3. User submits the form, and the system sends a verification email to the domain-specific company email address.
4. User clicks the verification link to confirm their email.
5. System activates the company account and flags it as "Pending Verification."
6. User completes the company profile by providing required information (e.g., company name, description, and location).
7. Admins optionally review the company profile and assign a "Verified Company" badge after approval.

Entry Conditions: The user lands on the registration page and selects "Company Representative" as their role.
Exit Conditions: Company account is created, email is verified, and profile is active.
Quality Requirements: Email address must be domain-specific and verified

via the confirmation link. Company Name, Description, and Location must be completed before profile activation.

3. Use Case Name: Register as an HR Expert

Participating Actors: HR Expert
Flow of Events:

1. User navigates to the registration page and selects "HR Expert" as their role.
2. User fills in their details, including personal information (e.g., name, email, password).
3. User submits the form, and the system sends a verification email to the provided address.
4. User clicks the verification link to confirm their email.
5. System activates the HR Expert's account, allowing them to log in and use the platform.
6. Users select their domain expertise (e.g., technical or non-technical) and upload a professional profile or CV.

Entry Conditions: The user lands on the registration page and selects "HR Expert" as their role.
Exit Conditions: HR Expert account is created, verified, and active.
Quality Requirements: A valid email must be provided and verified via the confirmation link. Uploaded CV or professional profile must meet the required format. Domain expertise must be selected.

4. Use Case Name: Add Candidate to Company Candidate Pool

Participating Actors: Candidate, Company Representative
Flow of Events:

1. System suggests potential candidates for the company based on job post criteria (e.g., skills, experience, location).
2. Company Representative reviews the suggested candidates and selects one or more candidates to add to their candidate pool.
3. Company Representative adds the selected candidate(s).
4. Candidates are added to the Company's candidate pool.
5. Companies can remove candidates from their candidate pool whenever needed.

Entry Conditions: No entry condition needed. Companies can add a candidate whenever required or needed.
Exit Conditions: Candidate is added to the Company's candidate pool.
Quality Requirements: Suggested candidates must align with the job post or skill set of the company. Candidate addition success or failure must be logged, and duplicate additions of the same candidate should be prevented.

5. Use Case Name: Create Job Post

Participating Actors: Company Representative
Flow of Events:

1. Company Representative logs into their account.
2. Company Representative navigates to the "Job Listings Pane" section and clicks on + button indicating "Create Job Post."
3. Company Representative fills in the job details (Skills, submission date begin, end) and submits the job post.
4. System verifies the input, saves the job post, and makes it visible to candidates and HR experts.

Entry Conditions: Company Representative is logged in and navigates to the "Job Listings" section.
Exit Conditions: Necessary fields must be filled with proper information (i.e. skills, submission begin, end dates, job description). Otherwise job posts will not be created. Job posts are successfully created and visible to candidates.
Quality Requirements: All required fields must be filled before submission. Duplicate job postings for the same position should be flagged.

6. Use Case Name: View Candidate Page

Participating Actors: Company Representative
Flow of Events:

1. Company Representative logs into their account.
2. Company Representative navigates to the "Candidates" section.
3. Companies can view a candidate briefly via the view CV button which shows brief oversight of the candidate's CV.
4. If more information about a candidate is needed, the company representative selects a specific candidate and clicks on "View Person" their detailed profile.
5. System displays the candidate's profile.
6. Company Representative reviews the candidate's information and decides whether to send a request to add them to their candidate pool.

Entry Conditions: Candidate data exists in the system, and the Company Representative has access to the "Candidate Pool" or "Suggested Candidates" section.
Exit Conditions: Company Representative has reviewed the candidate's profile and optionally can add them to their candidate pool or send a request to the candidate. For a shortcut, if a candidate has applied to a job post of the company the information with the job post can be reviewed, giving the company an option to reject or accept a candidate in that page.
Quality Requirements: Candidate profiles must be displayed accurately and securely. Sensitive information should only be shared with mutual consent.

7. Use Case Name: Hire HR Expert

Participating Actors: Company Representative, HR Expert
Flow of Events:

1. Company Representative logs into their account and navigates to the "Experts" section.
2. System displays a list of available HR Experts, including their experience level, and brief info of their CV's linked with the "view CV" button.
3. If needed, the company can view the full CV of the HR expert via the "view Person" button. Such as in candidate that page will view all details of HR expert's CV, with options to send a hiring request to the expert.
4. Company Representative reviews the list and selects an HR Expert.
5. Company Representative sends a hire request to the selected HR Expert, specifying the job post or purpose for which they are being hired.
6. HR Expert receives the hire request and decides to accept or reject it.
7. If the HR Expert accepts, the system establishes a connection between the Company and the HR Expert for further collaboration.

Entry Conditions: HR Experts are registered in the system, and the Company Representative is logged in.
Exit Conditions: HR Expert is hired and connected with the Company for the specified purpose.
Quality Requirements: HR Expert profiles must be displayed with accurate and relevant details. Duplicate or invalid hire requests should be prevented.

8. Use Case Name: Filter HR Experts by Department or Skill

Participating Actors: Company Representative
Flow of Events:

1. Company Representative logs into their account and navigates to the " Experts" section.
2. Available Experts are listed and Company Representative opens the filter options and selects a department (e.g., IT, Network)) from the menu.
3. In addition, skill set filtering can be used, filtering skills of experts such as Java, C++, Azure ex. with a list and removal for each filtering. This can work simultaneously with department filtering.
4. System filters the list of Experts to display only those belonging to the selected department and/or skill set.
5. Company Representative reviews the filtered list and optionally selects an Expert for further action (e.g., viewing their profile or sending a hire request).

Entry Conditions: Experts are registered in the system, and the Company Representative is logged in.
Exit Conditions: The filtered list of Experts is displayed based on the selected department.

Quality Requirements: Filter functionality must return accurate results. Filter selections must not affect system performance or introduce errors.

9. Use Case Name: Review Candidate CV

Participating Actors: HR Expert
Flow of Events:

1. HR Expert logs into their account and navigates to the "Candidates"
2. HR Experts can briefly view the CV of candidates.
3. If needed, HR Expert selects a candidate from the list to view the detailed profile.
4. System displays the candidate's profile, including their uploaded CV, skills, experience, and other relevant details.
5. HR Expert reviews the CV and other information to assess the candidate's qualifications.

Entry Conditions: Candidate data exists in the system, and the HR Expert has access to their assigned candidates.
Exit Conditions: Candidate CV has been reviewed.
Quality Requirements: CVs and candidate information must be displayed accurately and securely.

10. Use Case Name: Create Automated Appointment for Candidate

Participating Actors: Company Representative, Candidate
Flow of Events:

1. Company Representative logs into their account and navigates to the "Candidates" section.
2. Company Representative selects a candidate and clicks "Request Appointment."
3. System automatically assigns the earliest available date and time for the appointment based on predefined scheduling rules.
4. System sends the appointment details (date, time) as a request to the candidate.
5. Candidate receives a notification and reviews the appointment in their "Appointments" section.
6. Candidate accepts or rejects the request.
7. If accepted, the appointment is finalized and shown in the Candidate's "Appointments" section.

Entry Conditions: Candidate exists in the system, and the Company Representative is logged in.
Exit Conditions: The appointment is confirmed and displayed in the Candidate's "Appointments" section if they accept the request.
Quality Requirements: The automated scheduling system must avoid conflicts. Notifications about the appointment must be sent reliably. Requesting change in date, time of appointments should be managed properly to ensure flexibility.

11. Use Case Name: View Results

Participating Actors: Candidate
Flow of Events:

1. Candidate logs into their account and navigates to the "Results" section.
2. System displays a list of completed applications submitted by the Candidate.
3. Candidate selects a specific application to view its result.
4. System displays the result or status of the selected item
5. Candidate reviews the result and takes further action if applicable (e.g., applying to another job, contacting the company).
6. If a candidate is rejected, a “Failed” message is displayed and with the “See Results” button, the candidate can access more detailed information about the result.

Entry Conditions: Candidate has applied to job posts and automated pipeline results or statuses are available in the system.
Exit Conditions: Candidate views the results or statuses of their appointments and applications.
Quality Requirements: Results and statuses must be displayed accurately and updated in real-time. Sensitive information must remain secure.

12. Use Case Name: View Company Profile

Participating Actors: Candidate
Flow of Events:

1. Candidate logs into their account and navigates to the "Jobs” section.
2. Candidate selects a specific company from the list or associated with a job posting.
3. System displays the company’s profile.
4. Candidate reviews the company’s profile to gather information.
5. Candidates can see the recent job posts of the company and view them or follow the company.

Entry Conditions: Company profiles exist in the system, and the Candidate is logged in.
Exit Conditions: Candidate views the selected company’s profile.
Quality Requirements: Company profiles must be displayed accurately and securely. Links to external websites must be functional and safe.

3.5.2 Use-Case Model

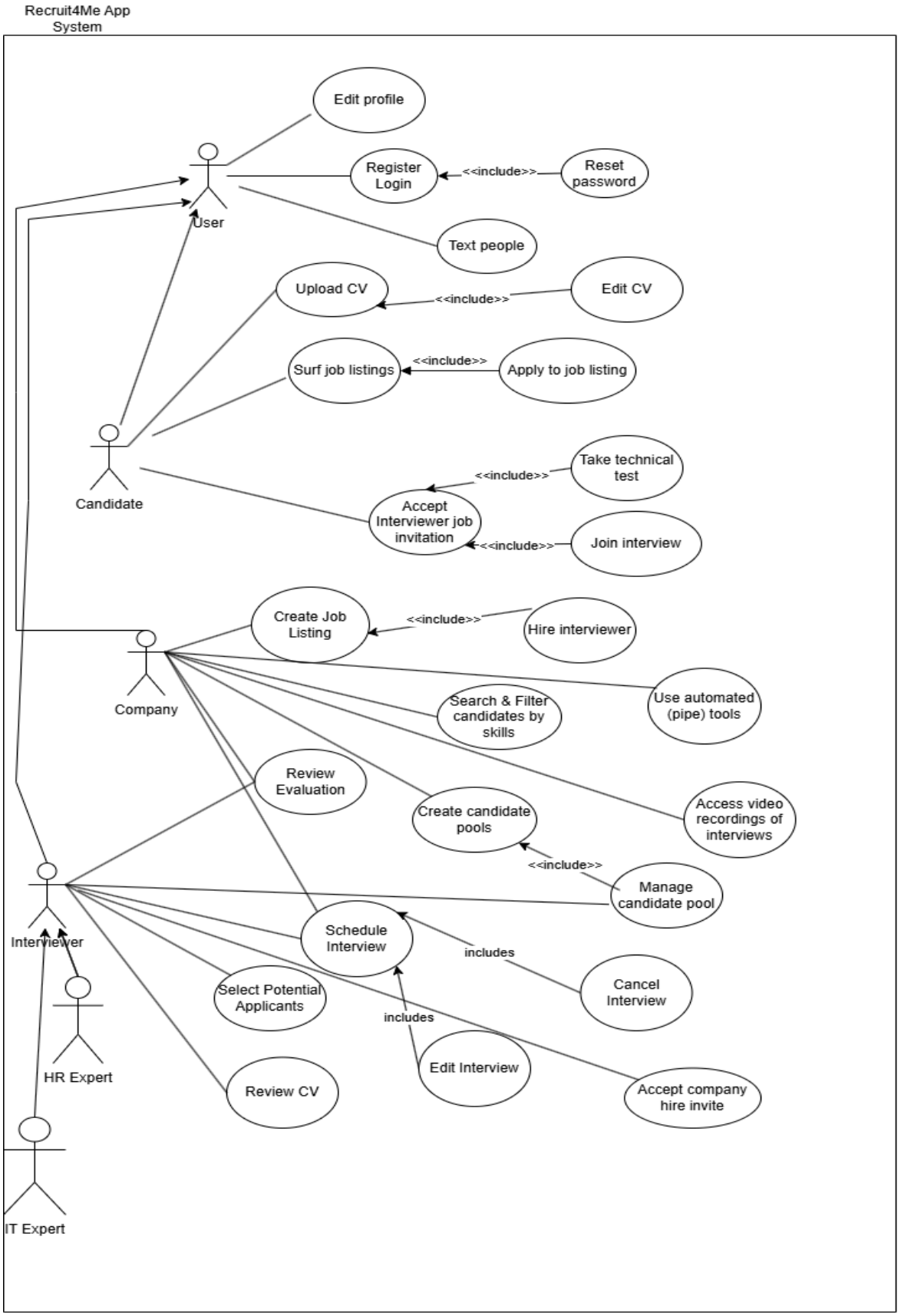

Figure 3. Use case diagram

3.5.3 Object and Class Model

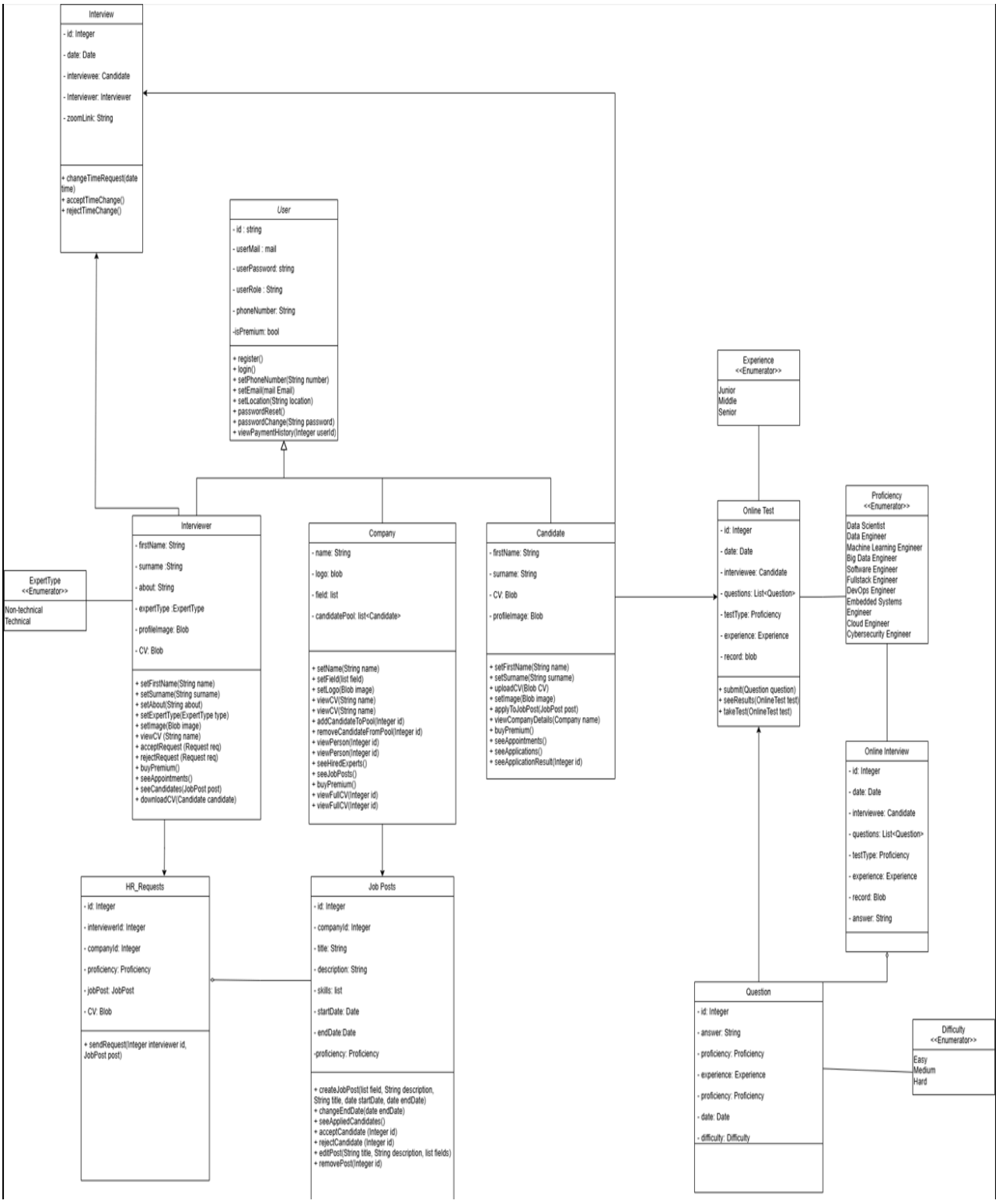

Figure 4. Object & Class diagram

3.5.4 Dynamic Models

3.5.4.1 ER Diagram

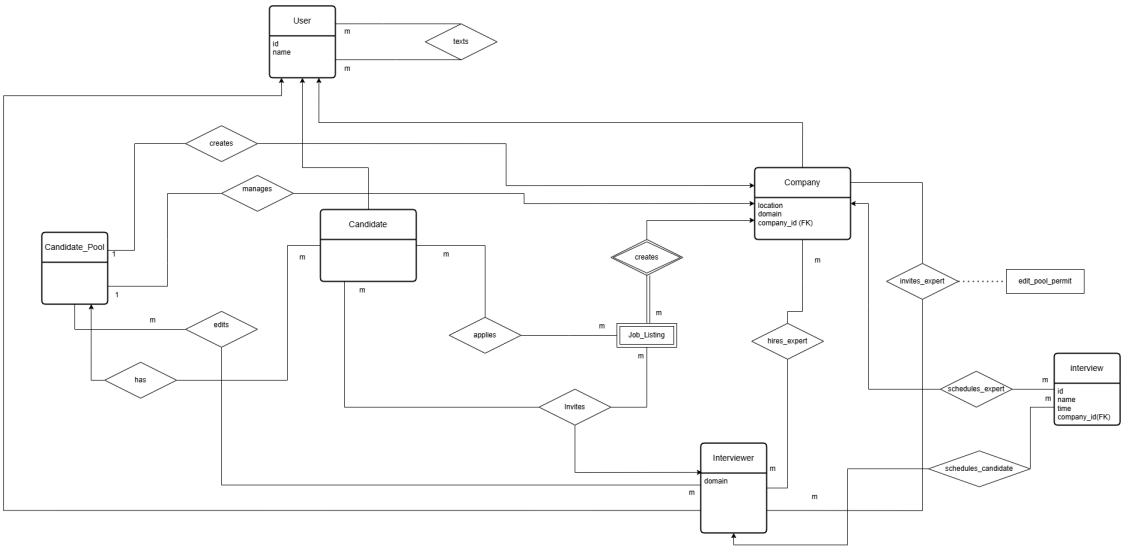

Figure 5. ER diagram

This ER diagram represents the very foundation of our system for automated application handling, capturing all major entities and their relationships in order to manage user interactions and data seamlessly. The entity User is the parent table for all types of users of the system, including Candidates, Companies, and Interviewers; it provides shared attributes such as id, name, and role to enforce consistent authentication and authorization. Each user has certain roles, be it a business representative, a candidate looking out for a job, and an expert interviewer, attributed through constraints to ensure the proper integrity of the data. Companies, considered a certain role, include the creation and management of different job listings while hiring or inviting their interviewers, which they are allowed to do because in this system, they're showing administrative privileges and many responsibilities.

The Candidate entity corresponds to every candidate seeking employment and thus stores information about their personal data such as: first_name, surname, mail, associated cv, and profile_image. As in the following, each of them is assigned to one candidate_pool. This pool shall allow every company to maintain its proprietary database of candidates that apply for job vacancies in their firm and organize all the profiles according to demands that may arise at some later stage in their recruitment pipeline. This is achieved through the pool_has_candidate table, which joins candidate pools to candidates in a many-to-many relationship. The Job_Listing entity, created and maintained by the companies, will track all open jobs with attributes such as title, about, and date listed, which uniquely identify the job posting. A separate job_listing_skills table allows job listings to include required skills, while the **Predefined_Skills** table is managed by system administrators to suggest common skills that streamline the job search process. The candidate_applications table is a bridge table to link candidates to job postings. This table will be responsible for the application status, such as pending,

rejected, or approved, thus updating the status dynamically as one advances through the stages of hiring.

Companies can also hire experts or interviewers for technical evaluations through the hired_experts table, which stores hiring details like date_hired and specific permissions for accessing and editing the candidate pool. Alternatively, companies can send invitations to potential interviewers via the expert_hire_invites table, including details like the date_sent, permissions offered, and a personalized message to facilitate the recruitment of expert evaluators. The Interviewer entity represents those that help companies assess candidates in various ways. Interviewers of different types exist, further classified based on their **expert_type**, for example - IT, HR. This captures valid classifications of interviewers. Detailed profiles of these interviewers are maintained through attributes like about and profile_image, which companies can base their decisions on to hire them or invite them for selected roles. Interviewers, upon hiring, can schedule interviews of candidates, which is reflected by the Interview entity as a relationship between companies, interviewers, and candidates being scheduled for events with attributes like id, name, time, and company_id. This allows delineation of a well-controlled and traceable process about interviews.

This schema will be entirely normalized according to database optimization techniques such as 3NF, 4NF or further algorithms for minimal data redundancy and more efficiency. It uses the relationship through primary and foreign keys in order to maintain the integrity of data for consistency across tables. For example, the user_mail field in the User table is used for reference by all user-specific sub-entities: Candidates, Companies, Interviewers. Unique constraints, such as on the **Job_Listing** table (company_id, title, listed), prevent duplicate entries of jobs while allowing companies to reopen positions after a certain period. Additionally, predefined check constraints are there to validate data, such as the fact that an interviewer's expert_type is either IT or HR, or an application status is within the defined states.

This ER model is extendable and supports complex relationships among the users, job listing, candidate, and interviewer with scope for enhancement. For instance, other roles or features, like analytics for companies or candidate recommendations, can be integrated within the schema without breaking it at the core. Since this is a first version, it still can be refined and optimized based on feedback from implementation in order to meet the dynamic requirements of the automated application handling system.

3.5.4.2 Sequence Diagrams

This is the section where you can find the sequence diagram for different processes.

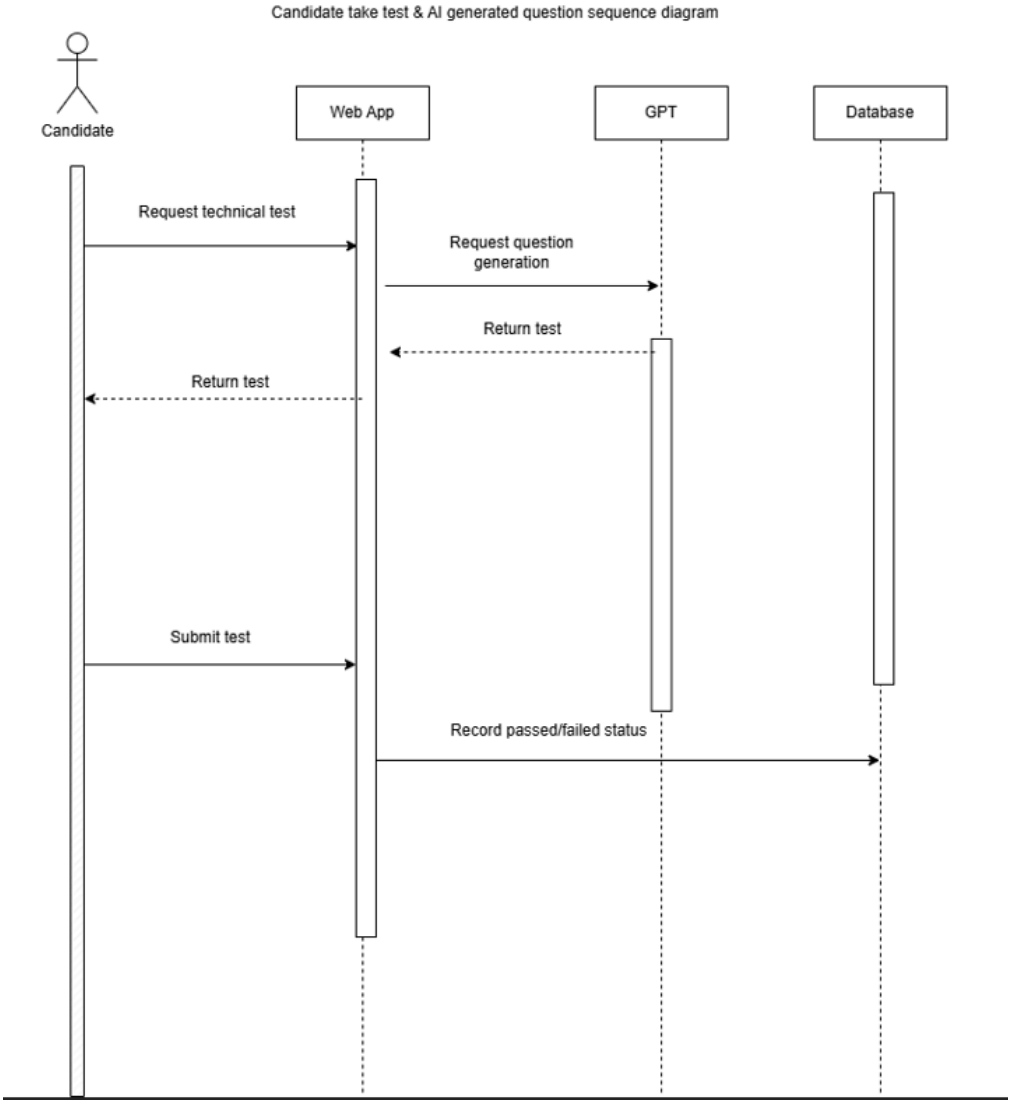

Figure 6. Sequence diagram for test

Explanation: This sequence diagram describes the interaction in a technical test taken by a candidate, where questions will be generated according to both job and company specifications by GPT model. It begins when the candidate requests a test by web application. In response, the web application talks to the GPT model to generate the required questions and delivers the test back to the candidate. Once the candidate has finished and submitted the test, the web application checks the responses and decides on the candidate's status. Finally, the result is recorded in the database for future reference.

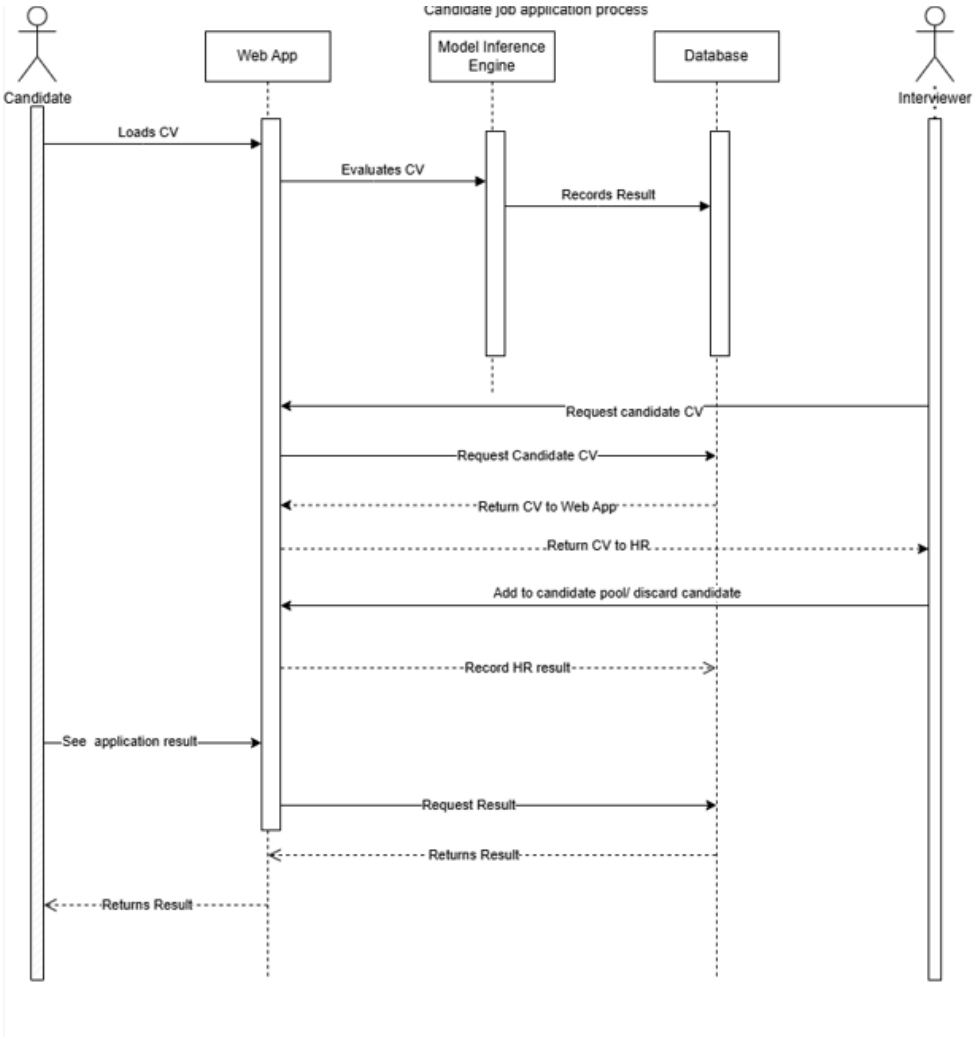

Figure 7. Sequence diagram for job application

Explanation: Sequence diagram above shows the candidate job application process involving the interactions of the Candidate, Web App, Model Inference Engine, Database, and Interviewer, respectively. Firstly, the candidate uploads the CV through the web app. The web sends this forward to the model inference engine for evaluation. In return, the results from processing the CV are passed to the web app, where this gets recorded in the database. The interviewer may then ask for the candidate's CV through the web application, have a look, and decide to add or discard the application into/from the company's pool. The decision will be updated into the database. Finally, the candidate will be informed about the application status through the web app, by retrieving the result from the database and showing it to the user.This workflow effectively incorporates multiple components, highlighting an AI-powered evaluation system, a centralized database for keeping records, and real-time status updates to make the process more effective for both candidates and interviewers.

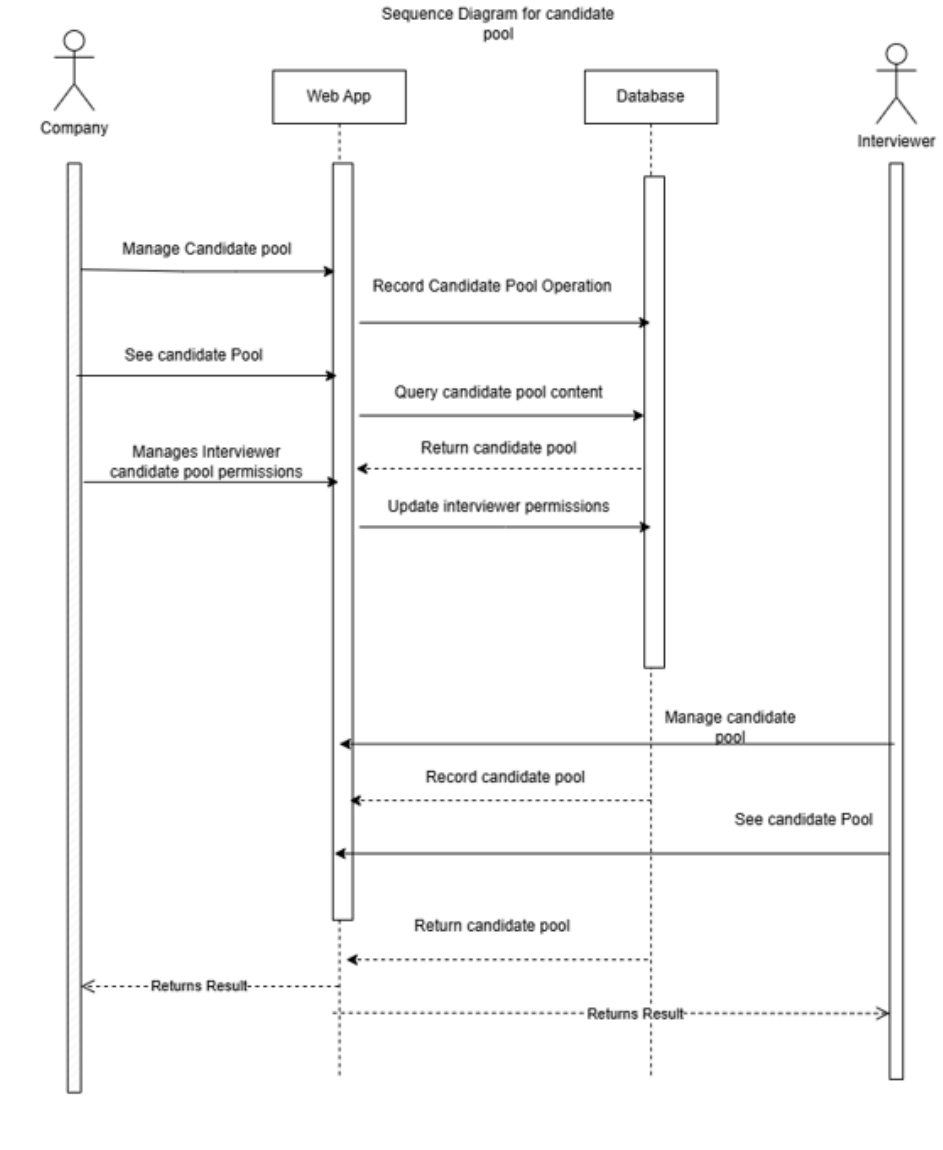

Figure 8. Sequence diagram for candidate pool management

Explanation: Above is the sequence diagram for candidate pool management. This presents the interaction of the Company with the Web App, the Database, and the Interviewer. The company initiates a candidate pool management process through the web app, and this gets registered in the database. It will also query what the pool contains, create candidates, and update the list of interviewers that are going to view or change the contents of the pool. The results are fetched from the database, which the web application is going to request, and updates regarding permissions of the Interviewer. The web applications will also provide options for the Interviewer to enable and manage the candidate pool, viewing it directly, but their actions performed are recorded and justified in the database. Results from such operations are then returned to the respective stakeholders, ensuring smooth candidate pool management with controlled access for all parties involved.

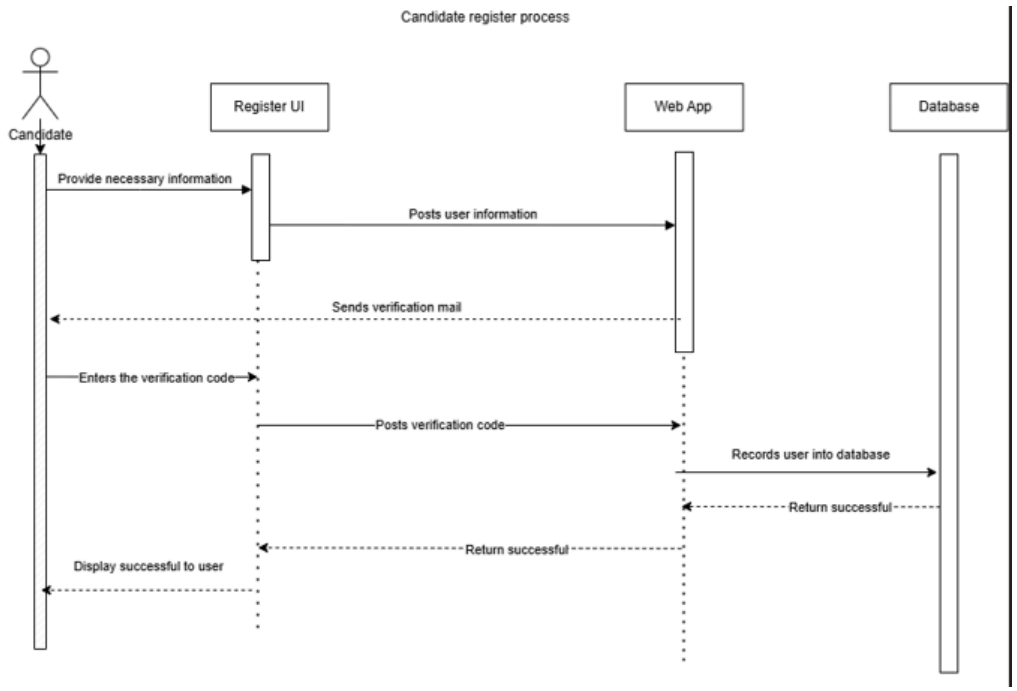

Figure 9. Sequence diagram for candidate registration process

Explanation: This is a sequence diagram for candidate registration, denoting the interactions between different entities like Candidate, Register UI, Web App, and Database. The process starts by having the candidate provide the information needed by Register UI, which posts the provided user information to the Web App. Then, web app sends an email to the candidate for verification purposes. The candidate writes the verification code to the Register UI. Then candidate write down this verification code which was inside the verification email. The verification of the code is done by the Web App, while the user information is stored in the database. Then, the success message is sent to the Register UI and displayed to the candidate, completing the registration action.

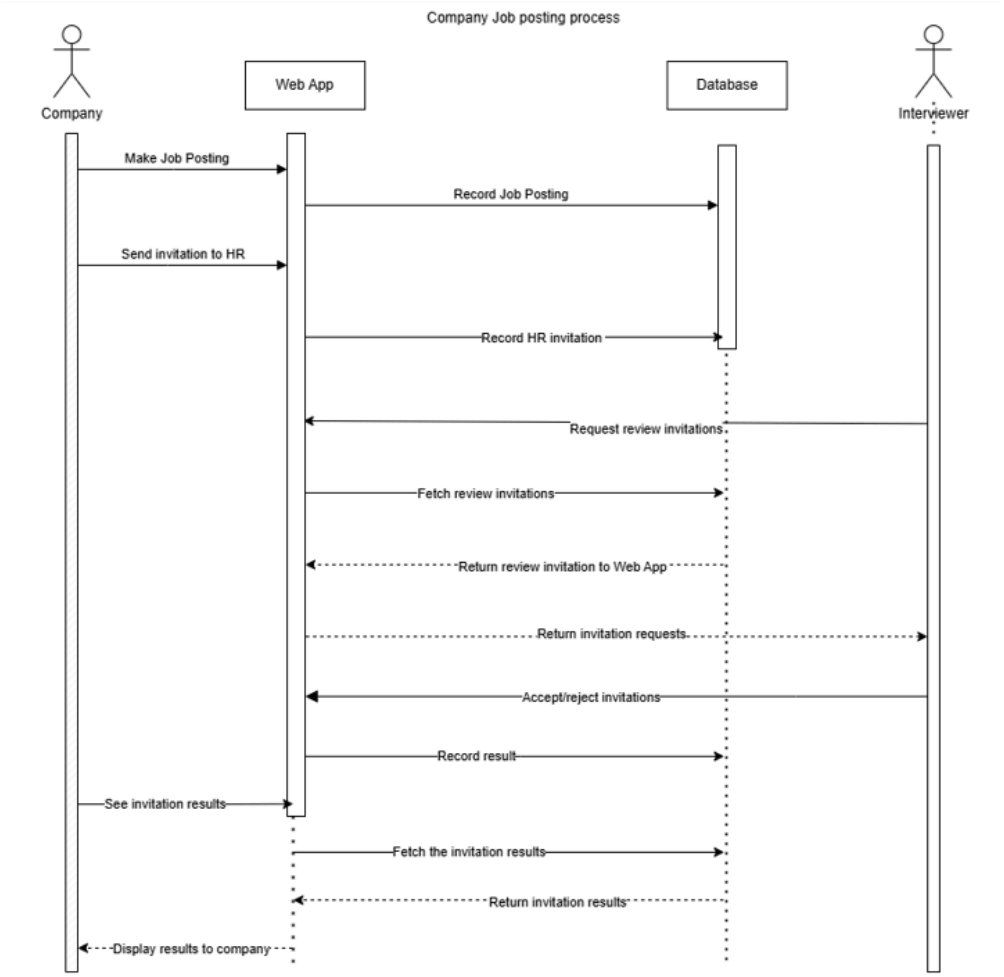

Figure 10. Sequence diagram for job posting process

Explanation: This sequence diagram represents the Company Job Posting Process, which includes interactions between the Company, Web App, Database, and Interviewer. The whole process starts when the company, through the Web App, initiates a job posting that records the details of the post in the database. The company invites HR or an interviewer to consider the posting. The Web App records the invitation of HR in the database. These invitations can be requested and fetched by interviewers using Web App, which fetches the data from the database and returns it to the interviewer. Interviewers then can review and either accept or reject the invitations; these decisions are recorded in the database. The company can then fetch and view the invitation results via the Web App, which queries the database for this information and displays the outcomes, completing the job posting process. This workflow ensures smooth communication and coordination among all parties involved in job posting and review.

3.5.5 User Interface

Login Page

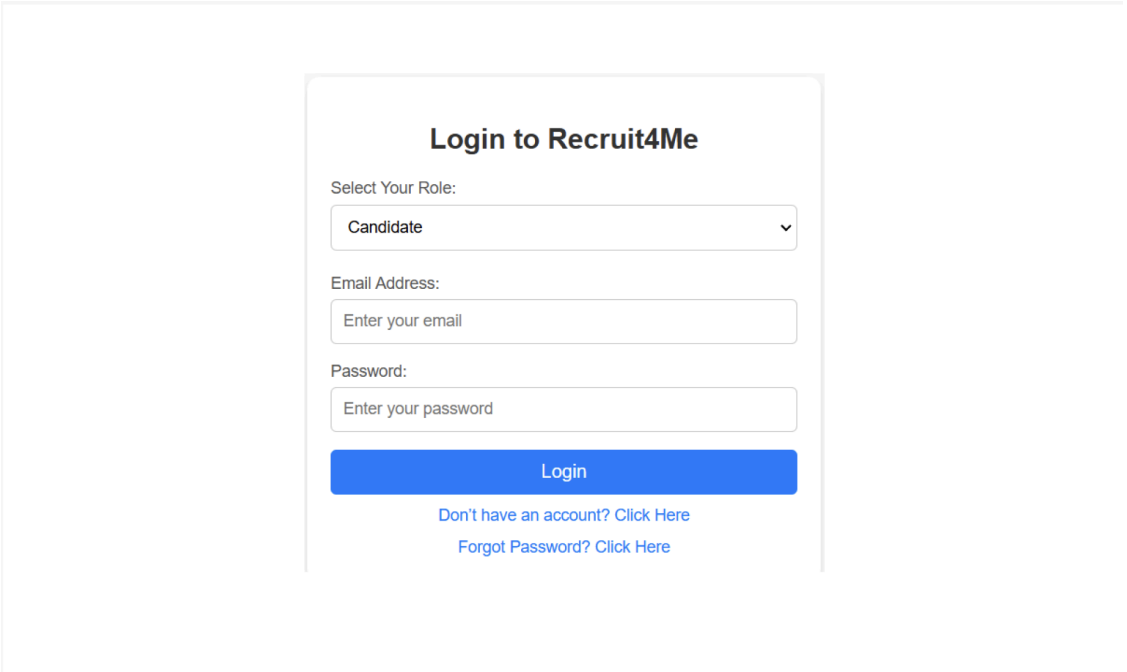

Figure 11. Login screen

Candidate Home Page

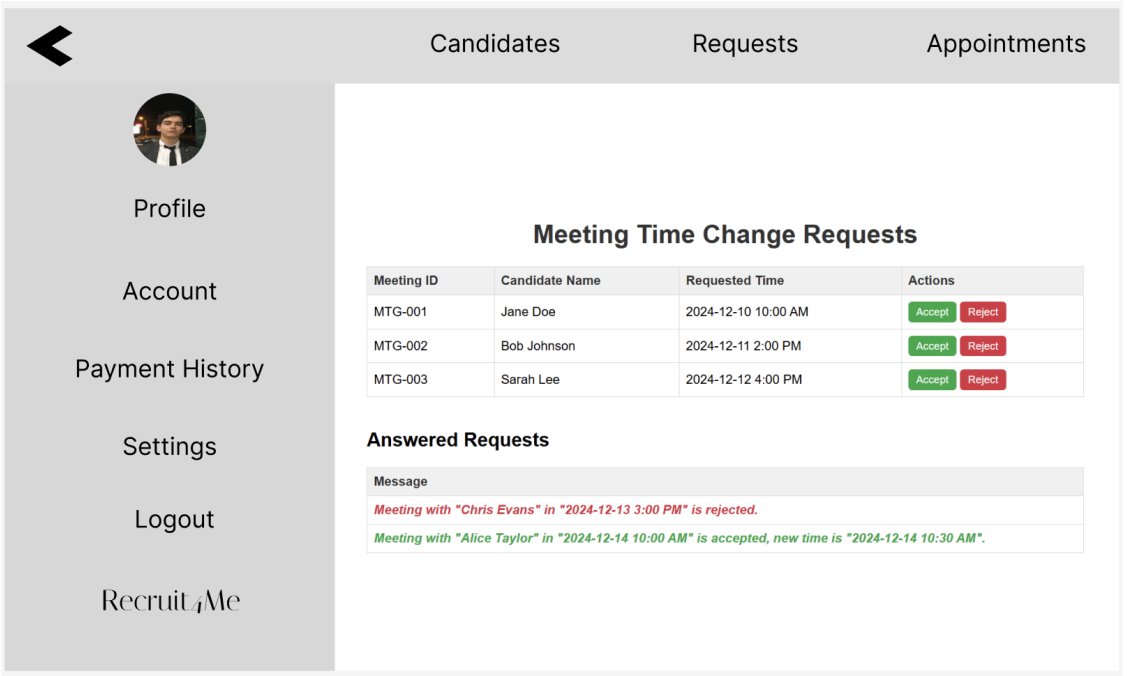

Figure 12. Candidate Home Page

Candidate Appointments Page

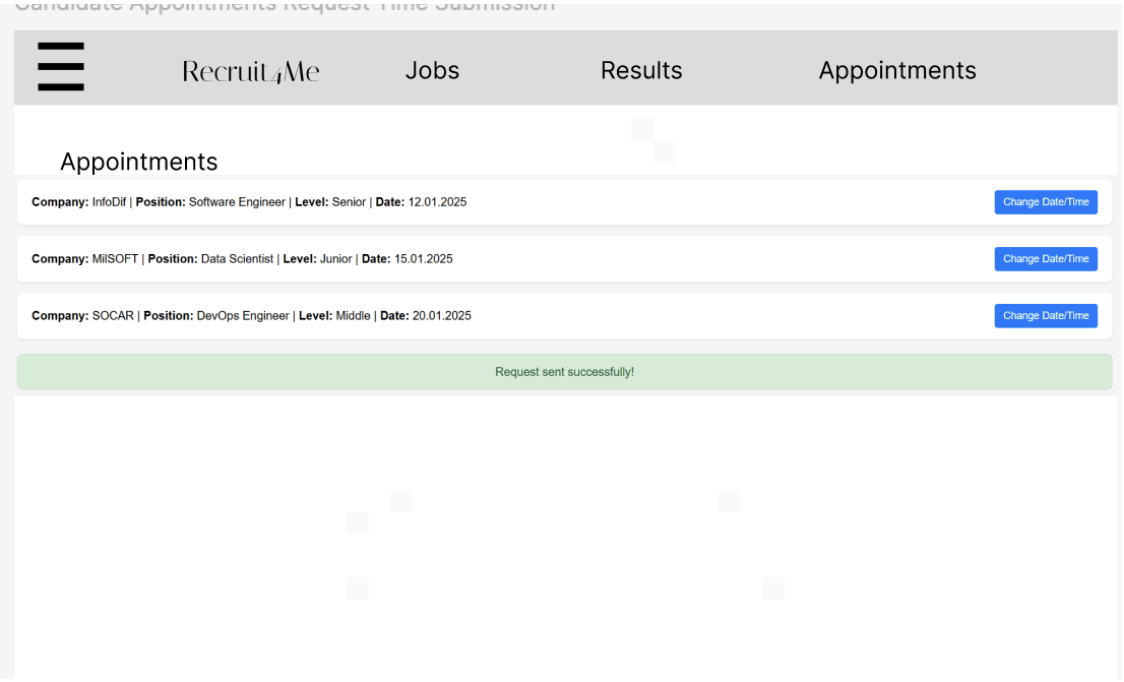

Figure 13. Candidate Appointments Page

Candidate Results Page

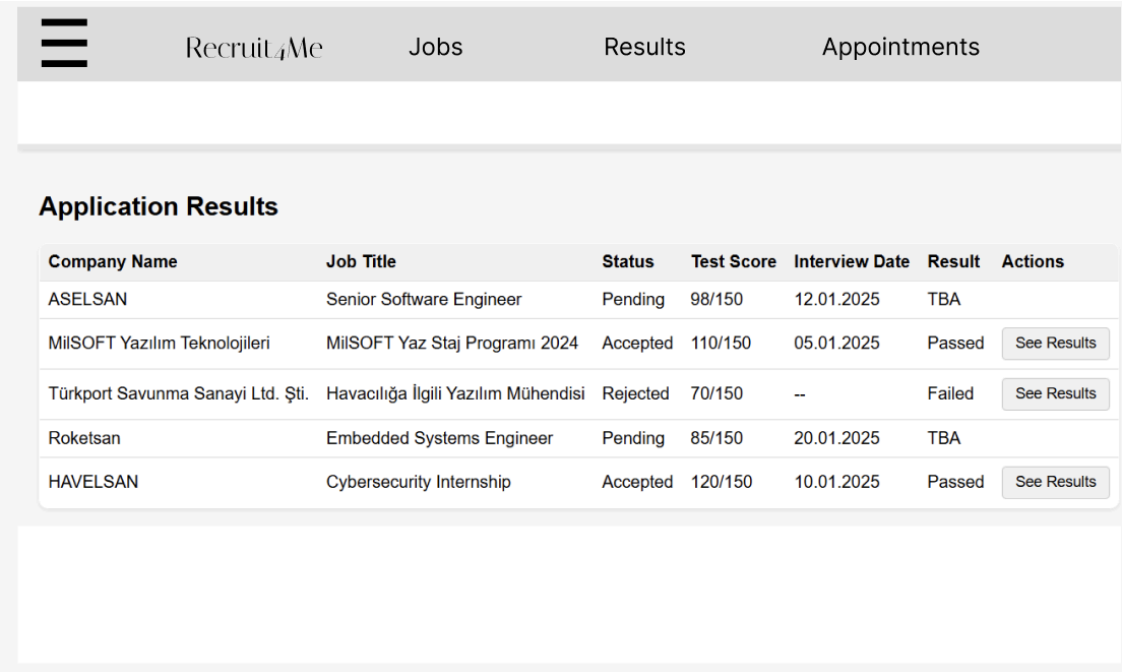

Figure 14. Candidate Results Page

Candidate Online Interview Pages

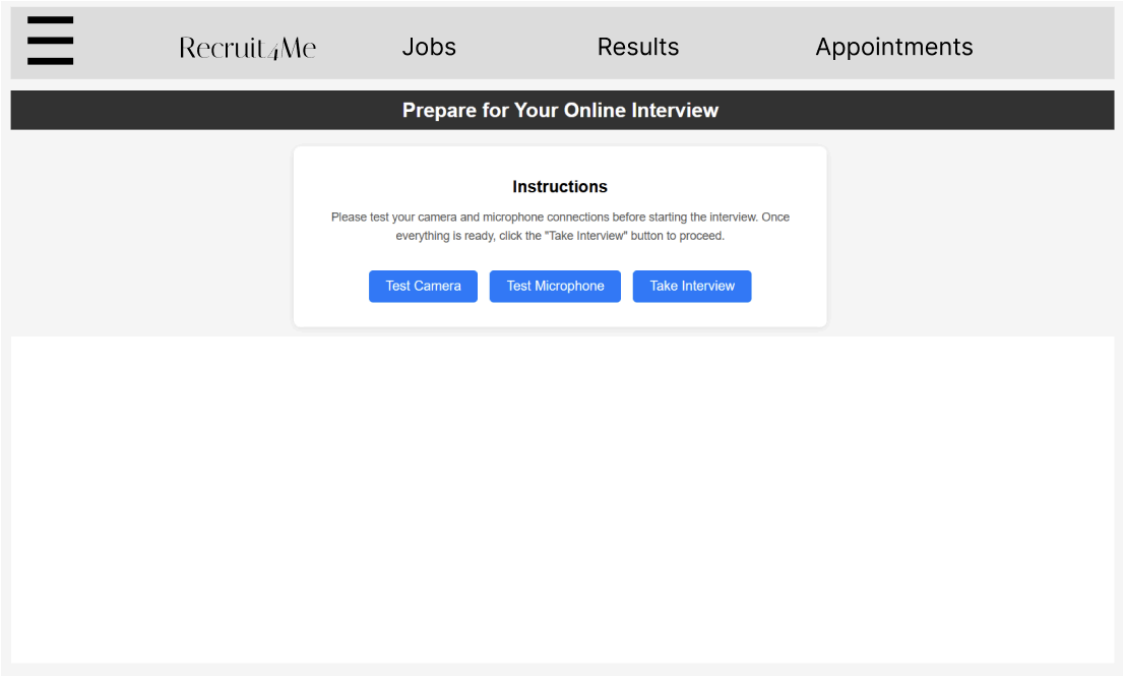

Figure 15. Candidate Online Interview Page 1

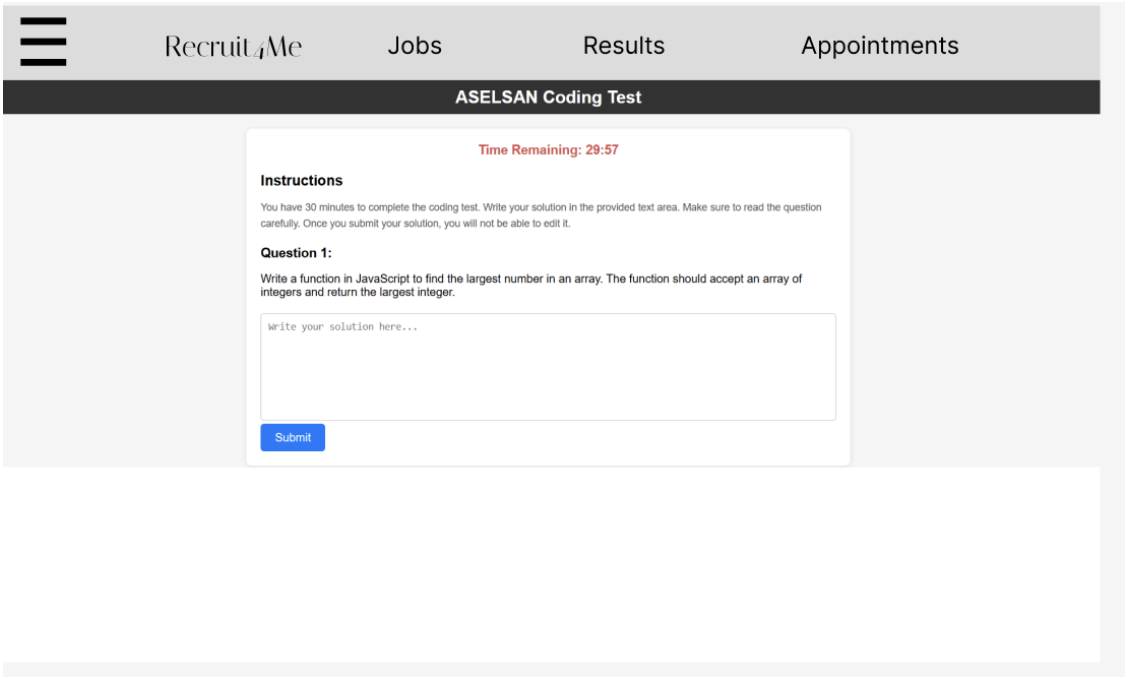

Figure 16. Candidate Online Interview Page 2

Candidate Edit Profile Page

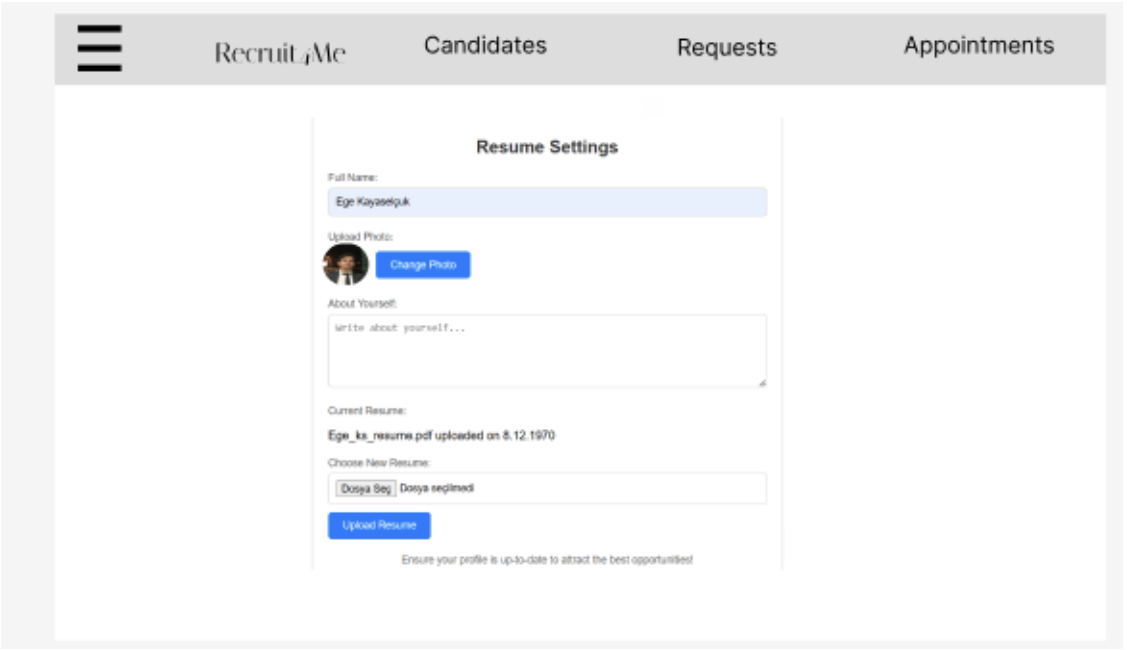

Figure 17. Candidate Edit Profile Page

Candidate View Company Page

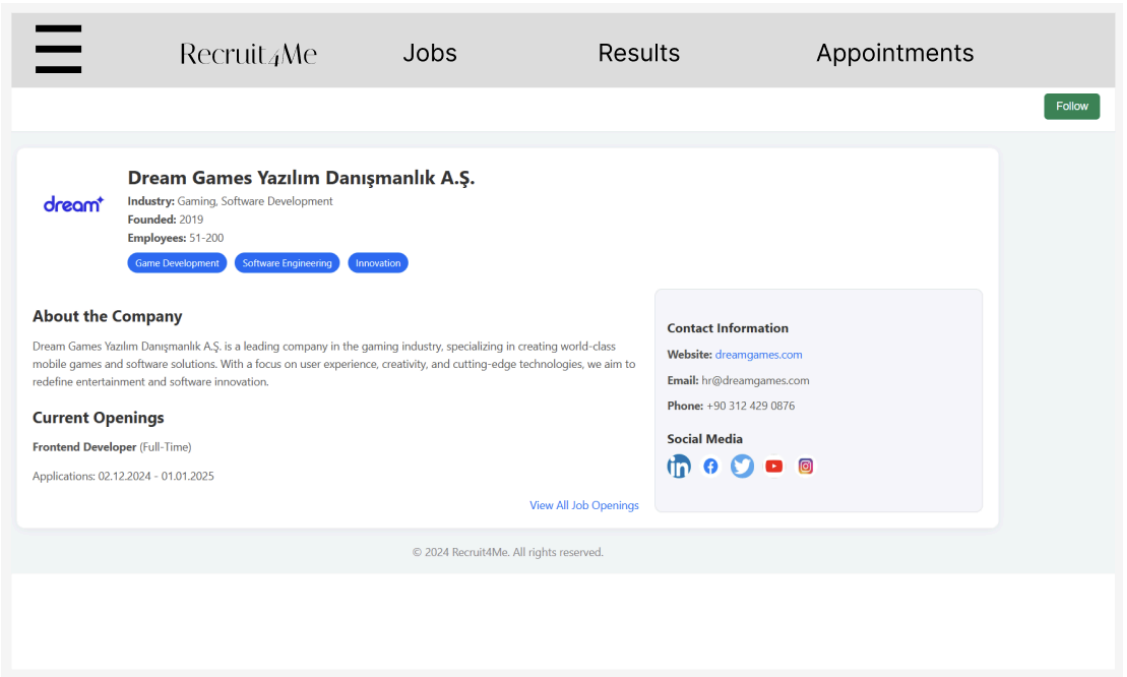

Figure 18. Candidate View Company Page

Company Home Page

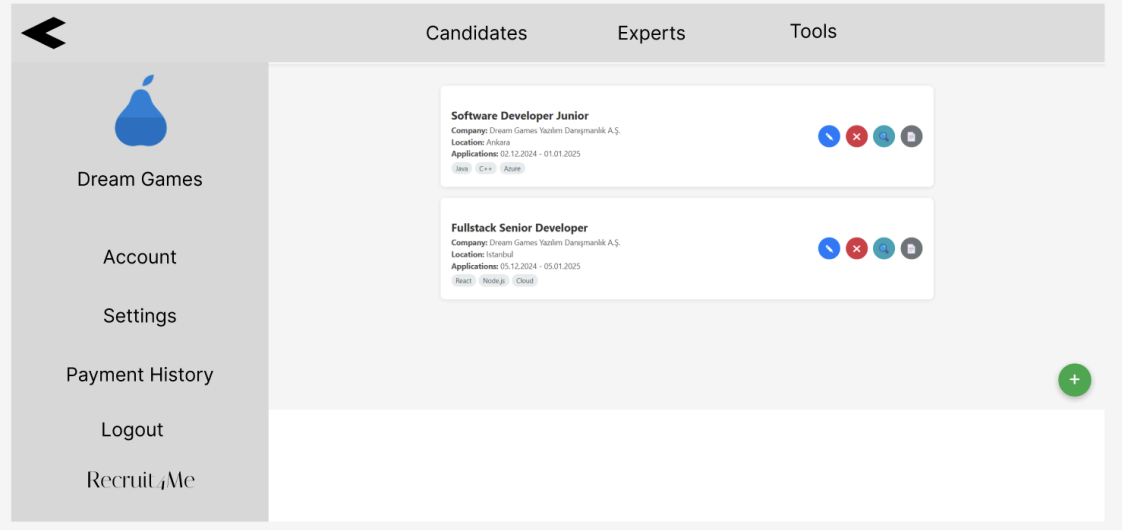

Figure 19. Company Home Page

Company Experts Page

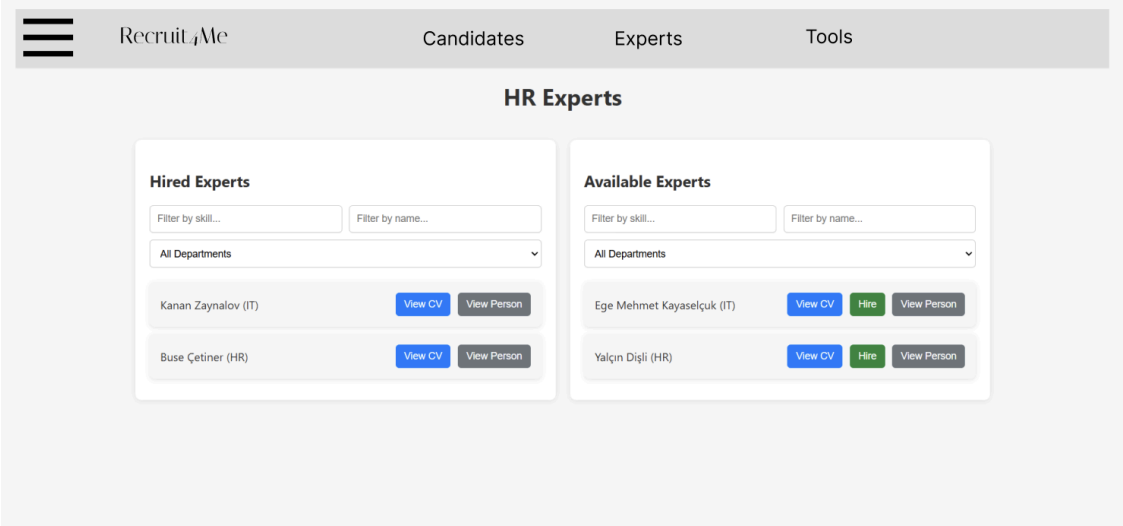

Figure 20. Company Experts Page

View Expert Profile

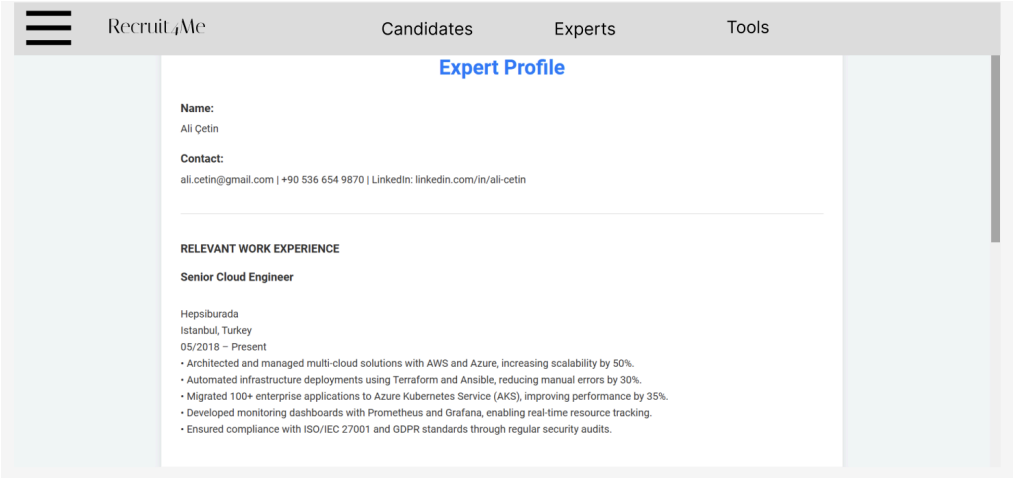

Figure 21. View Expert Profile

Company Add Job Posting Pages

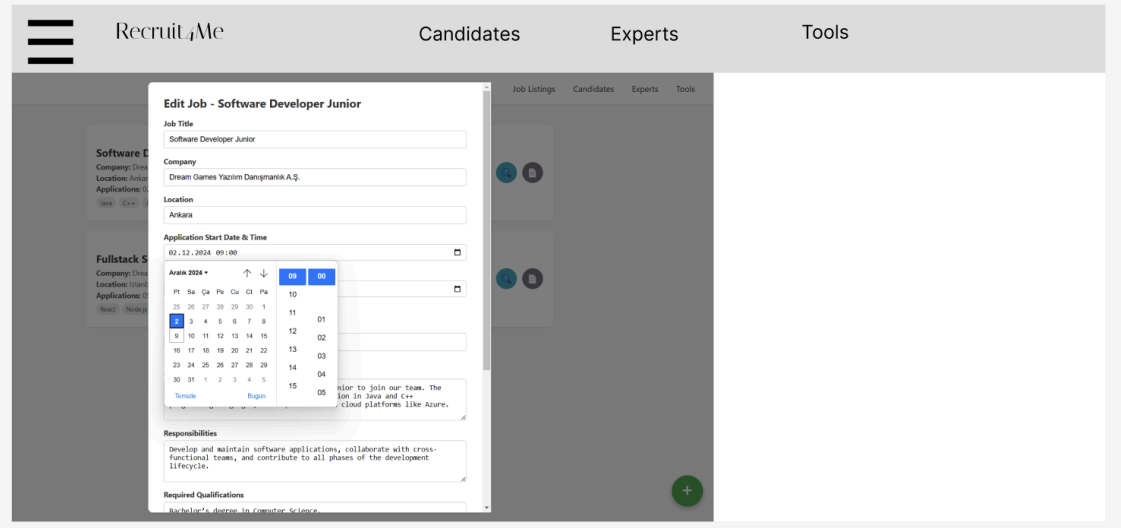

Figure 22. Company Add Job Posting Page 1

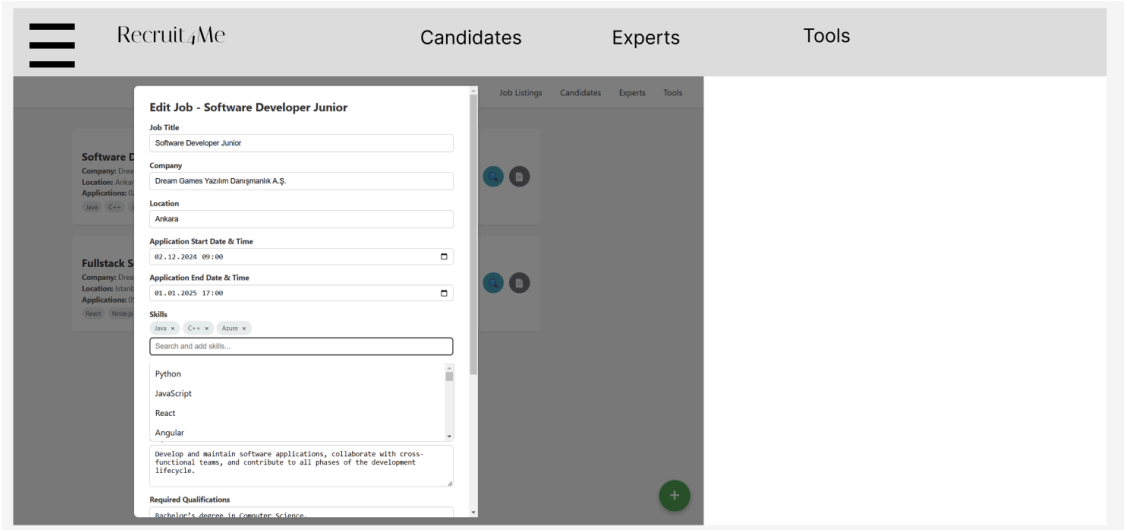

Figure 23. Company Add Job Posting Page 2

Company Candidates Page

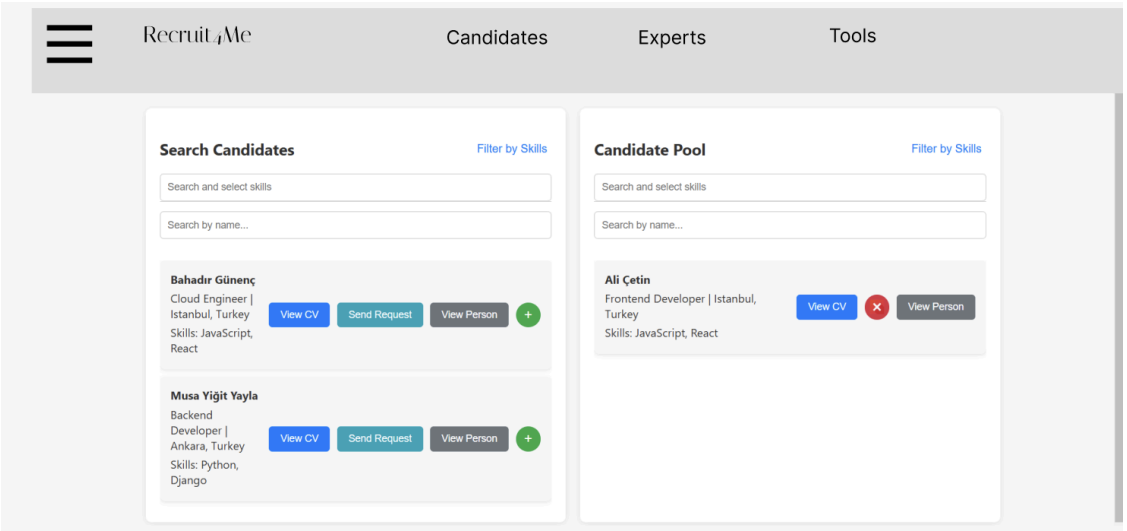

Figure 24. Company Add Job Posting Page 2

HR Expert Home Page

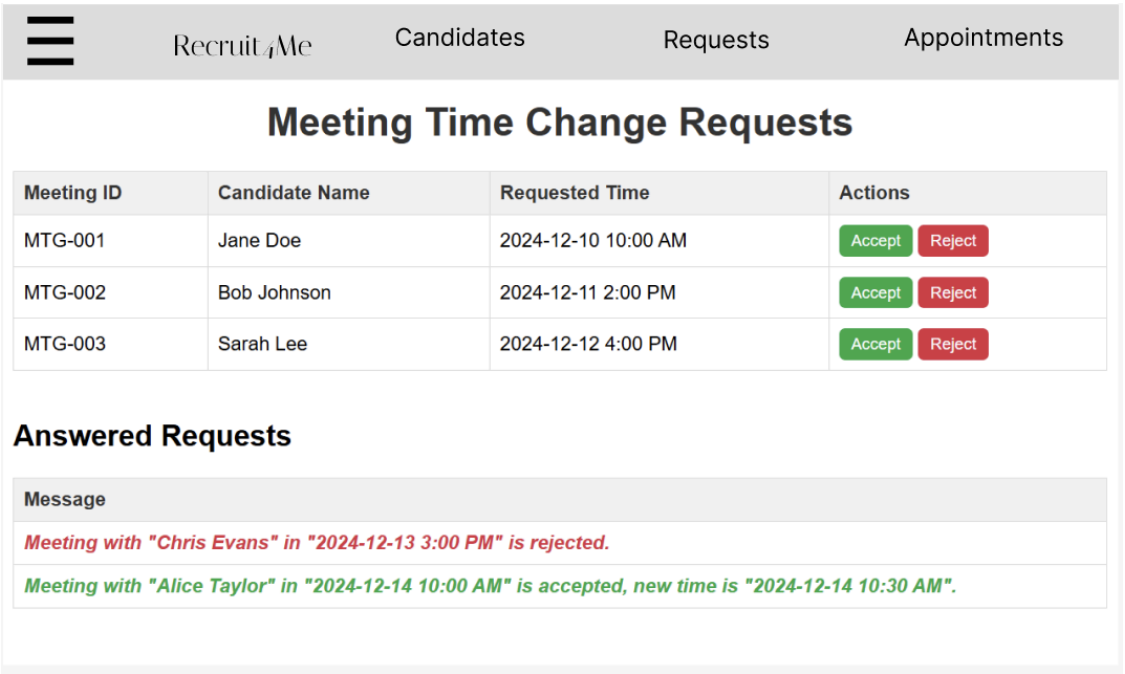

Figure 25. HR Expert Home Page

HR Expert Appointments Page

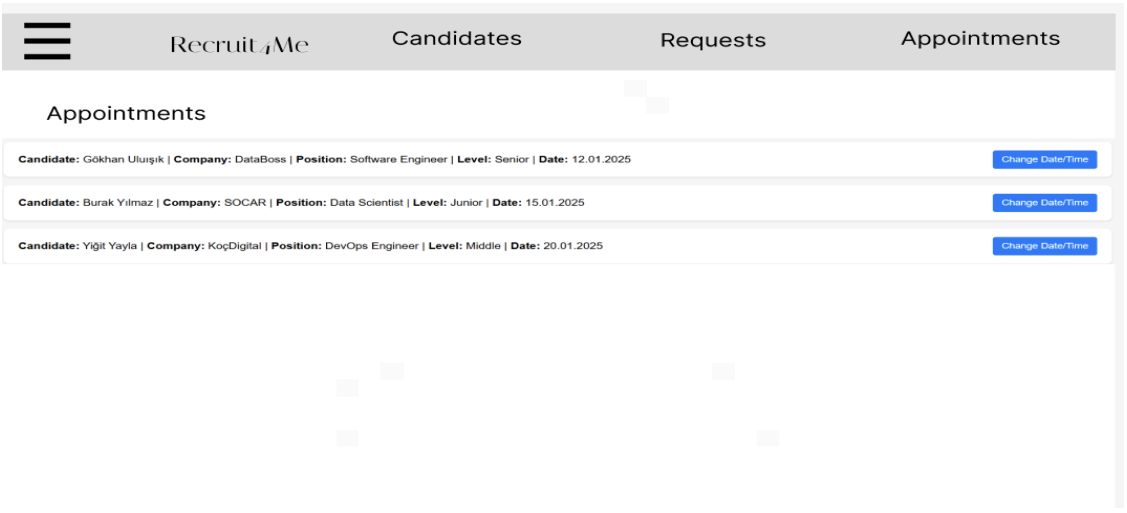

Figure 26. HR Expert Appointments Page

HR Expert Change Interview Time Page

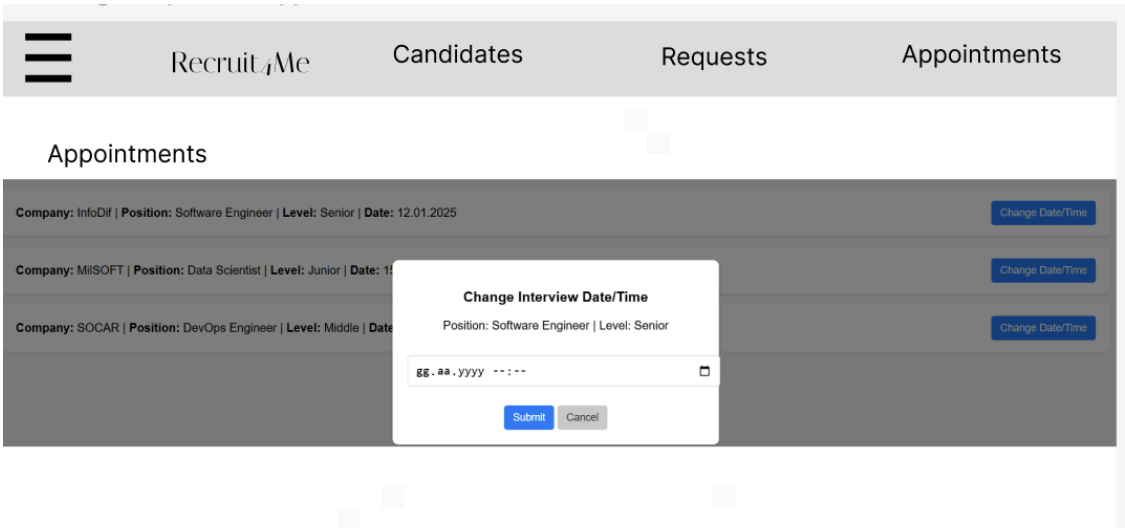

Figure 27. HR Expert Change Interview Time Page

HR Expert View Candidate Profile Page

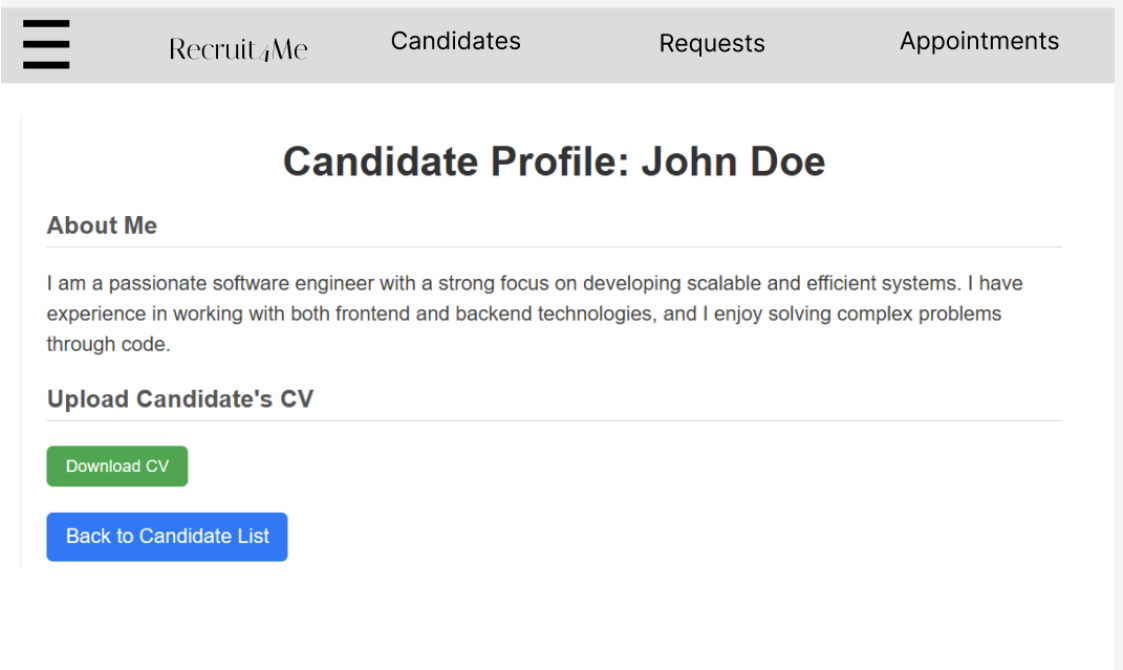

Figure 28. HR Expert View Candidate Profile Page

HR Expert Accept/Reject Hiring Request Pages

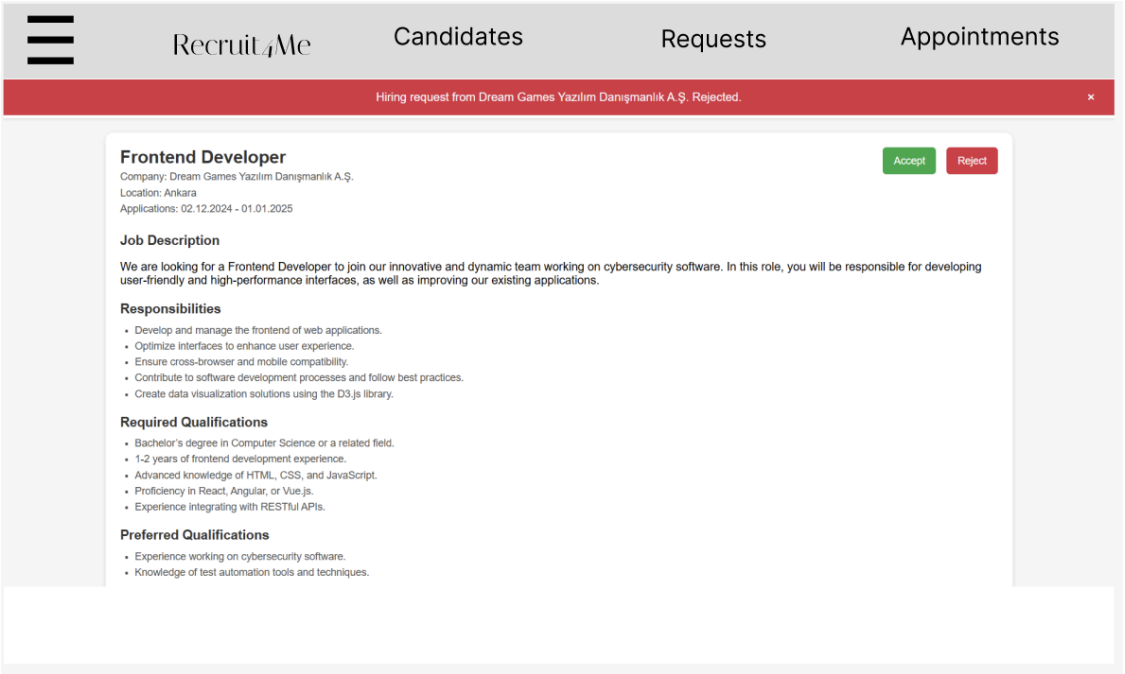

Figure 29. HR Expert Accept/Reject Hiring Request Page 1

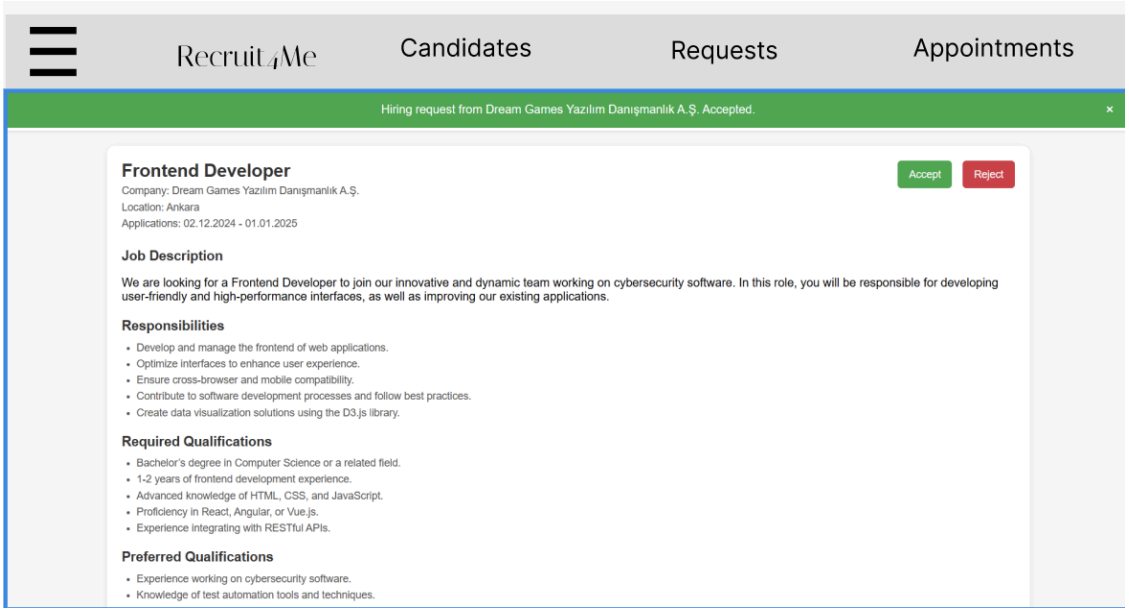

Figure 30. HR Expert Accept/Reject Hiring Request Page 2

HR Expert View Hiring Request

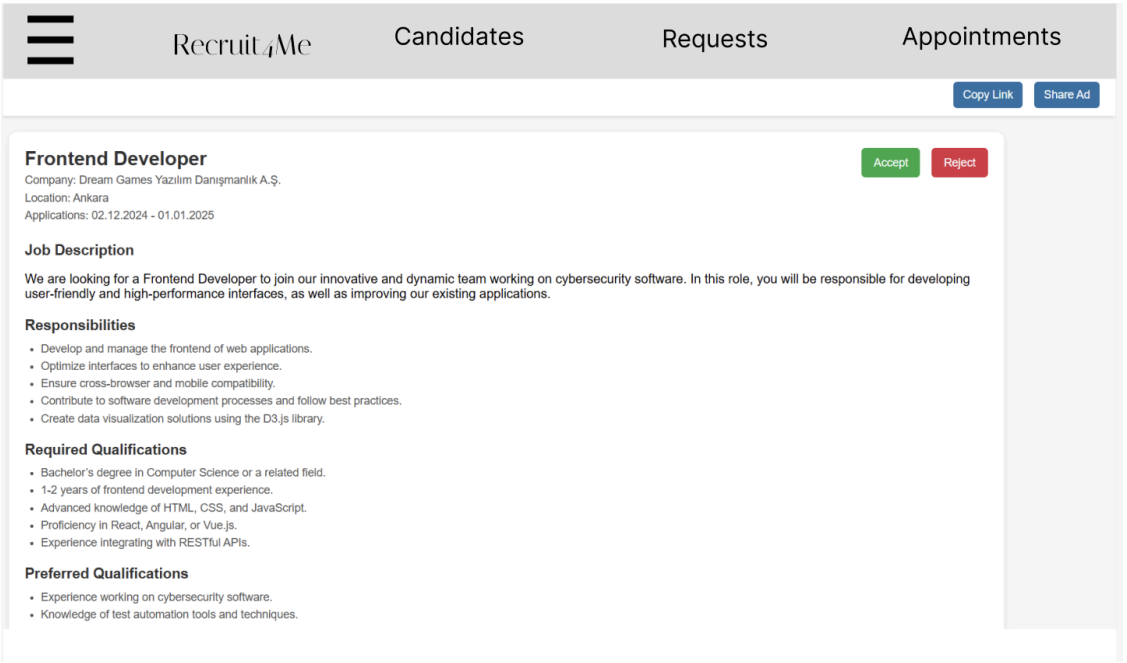

Figure 31. HR Expert View Hiring Request

HR Expert Candidate Results Page

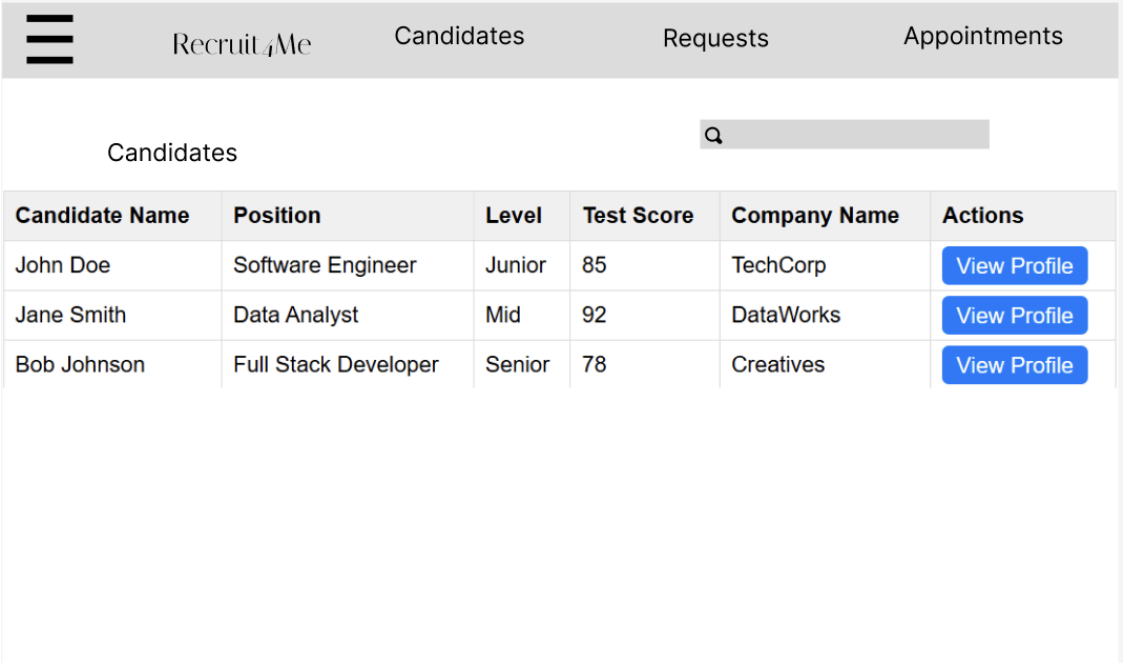

Figure 32. HR Expert Candidate Results Page

4 Other Analysis Elements

4.1 Consideration of Various Factors in Engineering Design

4.1.1 Constraints

- **Implementation Constraints**
  - System will be designed using React for front-end, Python or PHP for backend development.
  - Predefined APIs are needed to utilize speech-to-text processes such as Google Speech-to-text or AWS Transcribe.
  - Parsing resumes will require skill extraction thus integration of external tools (SpaCy, third-party APIs).
  - A secure database is needed(PostgreSQL or MongoDB) to store candidate records and test results.
  - Platform should be scalable to handle large data of candidates and companies.
  - Protocols making the application secure will be used such as OAuth2 or SSO.
  - System will be hosted by a cloud platform (AWS, Azure or GCP) to increase availability and reliability.
  - In a certain(now undefined) period video data must be deleted from the database to ensure high availability in the backend.
- **Economic Constraints**
  - In resume parsing, with support of NLP and predefined models from the team, potential API costs will be minimized.
  - Open-source solutions will be implemented first to see effectiveness before using large scale APIs like Google Speech-to-text or AWS Transcribe.
  - Google cloud platform will be used in free trial initially, which utilizes Compute Engine instances and BigQuery for data analysis.
  - Amazon Web Services(AWS) offers free EC2 instances, Lambda functions and RDS for a limited duration, which will be analyzed to ensure if it is proper for early-stage development.
  - Google Speech-to-text requires approximately 50 TL per hour for real-time transcription. AWS Transcribe has a similar pricing, lower volumes can be managed however large scale candidate pools will require budget evaluation.
  - Nvidia T4 GPUs on GCP cost approximately 30 TL per hour for on-demand usage. With sustained usage discounts, costs could drop to around 21000 TL for a single GPU instance.
  - Amazon EC2 P4 instances with Nvidia A100 GPUs are more powerful but cost around 72000 TL, which is suitable for high-performance tasks but may be excessive for lighter AI workloads.

- Efforts should be made to optimize these resources by scheduling GPU usage only during model training or evaluation processes.
- A modestly scaled deployment (e.g., 2 vCPUs, 8GB RAM, and 50GB storage) on AWS or GCP costs around 650 to 2000 TL per month. Adding auto-scaling capabilities for peak periods could slightly increase this price.
- Managed database services like AWS RDS or GCP Cloud SQL cost approximately 1000 to 3500 TL per month depending on storage and performance needs.
- Regular updates, monitoring, and bug fixes will require developer hours. Allocating team resources efficiently is crucial to avoid unnecessary expenditures.

- **Ethical Constraints**
  - System must ensure that all evaluations and eliminations are objective and free from bias based on factors such as gender, ethnicity, nationality or other non-technical factors. Which means that grading should be based on skill sets, experiences and test performances.
  - Communication skills ratings must be standardized by using well-defined criteria, such as clarity, grammar, and coherence, or some factors explicitly indicated by the company to avoid subjective judgments and potential biases.
  - AI models should be regularly checked for unintentional bias in the predefined algorithms and data handling to ensure fair treatment of all candidates.
  - All candidate data, including resumes, test results, and recordings, must comply with GDPR or other relevant data protection regulations, depending on the region.
  - Recordings should be stored for no longer than a certain period, as specified, and deleted securely afterward.
  - Notifications sent to candidates about results, feedback, or interviews must respect their privacy preferences and include clear opt-out options for future communication.
  - Encryption should be used to protect sensitive data during transmission and storage, ensuring the highest level of security.
  - For a possible built-in messaging in application encryption should be made to protect privacy.
  - For candidates eliminated by the system, the reasons must be explained in a transparent and professional manner, including data points such as skill gaps, test scores, or evaluation criteria.
  - All third-party tools integrated into the system, such as resume parsers, speech-to-text APIs, or scheduling platforms, must comply with strict ethical standards and avoid misuse of candidate data.

4.1.2 Standards

We use the Agile technique in the development of Recruit4Me HR Automation, dividing the project into two-week sprints that encourage flexibility in responding to changing requirements and iterative progress. Every sprint is devoted to the autonomous development of particular modules, which are then combined into the system as a whole at the end of each cycle. This strategy guarantees ongoing delivery of useful components and permits prompt modifications in response to stakeholder input. Recruit4Me HR Automation is carefully crafted to fully adhere to the ISO/IEC 25010 software quality model, which covers essential qualities including functionality, dependability, usability, efficiency, maintainability, and portability, in order to provide the highest standards of software quality [2]. At the same time, the web application component closely complies with Web Content Accessibility Guidelines (WCAG) 2.1, guaranteeing that the platform satisfies international accessibility standards and is usable by all users, including those with disabilities [3]. We have chosen UML 2.5.1 as our main modeling standard for system modeling, which makes it easier to create thorough and consistent diagrams that clearly convey the architecture, parts, and interactions of the system. Furthermore, the requirements documentation is painstakingly created in compliance with IEEE 830 criteria, guaranteeing that all functional and non-functional needs are precisely stated, organized, and simple enough for all parties involved to understand [4]. This combination of Agile practices and adherence to industry-recognized engineering standards ensures that Recruit4Me HR Automation is developed in a structured, high-quality manner, capable of meeting user needs, maintaining flexibility throughout its lifecycle, and delivering a robust, accessible, and maintainable HR automation solution.

4.2 Risks and Alternatives

Risks:

Model Performance Risks: The performance of Recruit4Me HR Automation's machine learning models is a major factor in its efficacy. Inaccurate predictions or classifications of candidate suitability by the algorithms may result in bad hiring choices, which would reduce the tool's usefulness and credibility. User confidence and the system's overall performance may suffer from misclassifications brought on by low accuracy, high bias, or high volatility in model predictions.

Data Privacy and Security Risks: When handling sensitive employee and candidate data, there are significant privacy and security risks. Financial penalties, a decline in consumer confidence, and legal ramifications can arise from unauthorized access, data breaches, and inappropriate handling of

personal data. Important challenges include defending against cyberattacks and making sure data protection regulations like the GDPR are followed.

API Integration and Compatibility Risks: Recruit4Me HR Automation's functionality depends on seamless integration with various third-party APIs, such as job boards, applicant tracking systems (ATS), and other HR tools. Incompatibilities, changes in third-party API structures, or downtimes can disrupt the tool's operations, leading to reduced functionality and user dissatisfaction.

User Adoption and Engagement Risks: The adoption and consistent use of Recruit4Me HR Automation by recruiters and HR professionals is essential to its success. The tool may have low adoption rates if it is not user-friendly, has no essential functionality, or does not fit in with current processes. Inadequate training and a bad user experience can also make it more difficult to engage and utilize a product consistently.

Technical Limitations and Scalability Risks: Technical constraints like server capacity, database speed, and response times may surface as Recruit4Me HR Automation grows to handle more users and bigger datasets. System slowdowns, higher latency, and decreased reliability brought on by inadequate scalability can have a detrimental effect on user pleasure and experience.

Dependency on External Services: There are risks associated with service availability, policy changes, and pricing fluctuations when crucial functionalities like cloud hosting, AI services, and third-party APIs are dependent on external services. Unexpected costs may result from Recruit4Me HR Automation's activities being disrupted by outages or changes in external services.

AI and Ethical Risks: The use of AI in taking applicants to a company may create ethical concerns, such as potential biases in hiring algorithms, socially and culturally, clearness in decision-making, and trustworthiness of automated decisions. By using AI models which works without bias, the problem of discrimination will be resolved and a fair environment will be produced.

Alternatives:

Enhancing Model Robustness: Next, we will be implementing the strategies to be taken on Model Performance Risks: scaling up the dataset with the collection of various and representative samples with more high-quality data would increase accuracy and generalization of model outputs; considering state-of-the-art machine learning methods or combination/ensemble techniques could improve prediction capability. Constant monitoring and evaluation of performance on models should go through continuous validation by techniques of cross-validation on small segments or natural environments of occurrences in practical and applicable areas. This will entail implementing techniques for detecting various biases in the training and model predictions to ensure very fair and accurate outcomes.

Privacy and Security Enhancements: We will introduce strong encryption techniques, which will be applied while dealing with sensitive information, in transit and when stored. Stringent access controls ensure unauthorized access of the sensitive information by the staff. Regular security audits and vulnerability tests shall be done to find out or reduce any prospective threat to security. Additionally, data protection regulations compliance like GDPR will be established with all implementations of necessary measures to achieve complete compliance and proper record maintenance.

Flexible Integration Approach: The architecture system will be modularly designed. An API abstraction layer shall, therefore, be set such that it can adapt changes from third-party APIs without affecting your system. This allows us to add new integrations easily when needed with minimal disruptions. To notice changes or downtimes in the API more quickly, continuous monitoring tooling will be provided so changes can be quickly addressed, and necessary adjustments can also be made. Mechanisms for fallbacks shall also ensure that key functionalities still function without dependency on specific APIs failing.

User-Centric Design and Feedback Loop: Improvements in user adoption and engagement will be driven through user-centered design by the provision of an intuitive and responsive interface, aligning with the workflows relevant to HR professionals and recruiters. This would also be supported through comprehensive training resources in tutorials and documentation that will help users apply the tool in an effective manner. By establishing a feedback loop, we can hear users' suggestions and what hurts if they use this tool. In that way, iterative improvements will enable this tool to catch up with the demands of users.

Scalability Planning: Technical limitations and scalability risks will be addressed by making use of scalable cloud infrastructure that dynamically allocates resources depending on demand. Load balancing ensures that traffic is evenly distributed across servers, hence preventing performance bottlenecks. The MySQL database will be optimized through indexing, query optimization, and regular maintenance for better performance. Moving to a microservices architecture will enable different system components to scale independently against reliability and efficiency as the user base grows.

Reducing External Dependencies: Technical limitations and scalability risks will be addressed by making use of scalable cloud infrastructure that dynamically allocates resources depending on demand. Load balancing ensures that traffic is evenly distributed across servers, hence preventing performance bottlenecks. The MySQL database will be optimized through indexing, query optimization, and regular maintenance for better performance. Moving to a microservices architecture will enable different system components to scale independently against reliability and efficiency as the user base grows.

AI and Ethical Risk Mitigation: Explain the transparency of AI decision-making and bring ethical practice into AI in making explanations

available for automatic outcomes. Regular auditing of AI models will be performed to detect and deactivate any kinds of biases that could lead to hiring on the basis of fairness. Establish an accountability framework, including human oversight of automated decisions, to ensure standards are upheld. We will also design and adhere to guidelines on the ethical use of AI, ensuring that the product Recruit4Me HR Automation supports nondiscriminatory hiring practices.

4.3 Project Plan

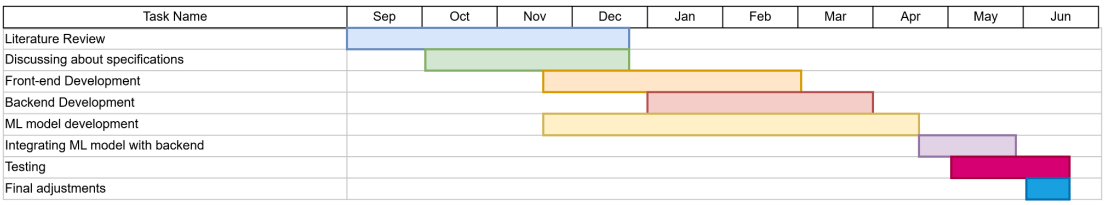

Figure 33. Gantt Chart for task time planning

Table 1: Factors that can affect analysis and design.

	Effect level	Effect
Public health	Low	Recruit4Me HR Automation does not directly impact public health. However, by enhancing HR processes, it can contribute to creating healthier work environments through better management practices and support systems. In situations like pandemics, the tool's remote capabilities ensure that HR functions continue smoothly, indirectly supporting employee health by maintaining stable employment and access to necessary resources.
Public safety	High	The primary concern related to public safety is the security of employee data CVs. Recruit4Me HR Automation must implement stringent security protocols to protect sensitive information from breaches and unauthorized access. We will use tokens and database security protocols to ensure CVs are protected from unauthorized sharing of data without permission of a user.

		Ensuring data integrity and confidentiality is essential to maintain user trust and prevent misuse of personal information, thereby safeguarding the overall safety of the system's users.
Public welfare	High	By automating HR tasks such as recruitment, onboarding, and employee management, Recruit4Me HR Automation can improve organizational efficiency and employee satisfaction. Enhanced HR processes can lead to better job placements, career development opportunities, and equitable treatment of employees, thereby positively influencing public welfare. HR experts will have a chance to be hired for candidate evaluations. Unemployed HR experts can use this application and profit from it.
Global factors	Low	Globalization introduces complexities such as varying labor laws, data protection regulations, and cultural differences in HR practices. Recruit4Me HR Automation must be adaptable to comply with international standards and support diverse workforce management. Features like multi-language support, compliance with global data privacy laws, and adaptability to different recruitment practices are essential for the tool's success in the global market.
Cultural factors	Medium	User interaction with Recruit4Me HR Automation is impacted by cultural variety. Different cultural conventions and preferences in HR procedures, including interview techniques, communication styles, and employee engagement tactics, must be accommodated by the tool. Culturally sensitive features and customizable user interfaces

		can improve user experience and guarantee the tool works well in a variety of cultural contexts.
Social factors	Medium	Social trends such as the emphasis on diversity and inclusion, remote work, and work-life balance influence the development of Recruit4Me HR Automation. The tool can address these trends by offering features that support inclusive hiring practices, facilitate remote workforce management, and promote employee well-being. Aligning with these social factors can increase the tool's relevance and acceptance in modern workplaces.
Environmental factors	Low	While Recruit4Me HR Automation does not have a direct impact on the environment, it contributes to sustainability by reducing the need for physical paperwork and enabling remote HR management. Hosting the application on environmentally responsible cloud platforms can further minimize its carbon footprint. Promoting digital transformation aligns with broader environmental sustainability goals by encouraging efficient resource utilization.
Economic factors	High	The acceptance and success of Recruit4Me HR Automation are significantly influenced by the state of the economy. Organizations may look for less expensive ways to streamline HR functions during recessions, which could raise demand for the technology. On the other hand, expansion and the demand for scalable HR solutions can be fueled by economic growth. Important elements influencing the tool's market competitiveness and uptake by businesses of all sizes and financial constraints are its

		price strategy, cost savings, and return on investment.

Table 2: Risks

Risk	Likelihood	Effect on Project	B Plan Summary
Inadequate Training Data Quality	Moderate	Reduced model performance, potential bias	Acquire additional data sources or add data to improve coverage
Model Accuracy Below Expectations	Moderate	Inaccurate candidate evaluations and mismatches	Pause model usage and let people handle interviews until the model's settings and rules are improved.
System Integration Delays	High	Postponed feature releases, extended development time	Break down the system into smaller steps, test each part with enough data, and simplify communication between parts.
Security and Privacy Breaches	Low	Compromised candidate data and trust	Temporarily suspend at-risk features, strengthen security measures, and perform inspection
Negative Candidate Feedback/Experience	Moderate	Diminished applicant number and employer brand damage	Revise interactions and interview flow, and incorporate user feedback in detail.
Infrastructure and Scaling Issues	Low	Reduced uptime, slow response times	Deploy additional servers, adjust scaling and introduce other techniques to improve performance

Table 3: List of work packages

WP#	Work package title	Leader	Members involved
WP1	Literature Review	Burak Efe Öğüt	Kanan Zeynalov, Musa Yiğit Yayla, Burak Efe Öğüt,

			Ege Mehmet Kayaselçuk, Bahadır Günenc
WP2	Discussing about specifications	Bahadır Günenc	Kanan Zeynalov, Musa Yiğit Yayla, Burak Efe Öğüt, Ege Mehmet Kayaselçuk, Bahadır Günenc
WP3	Front-end Development	Musa Yiğit Yayla	Musa Yiğit Yayla, Ege Mehmet Kayaselçuk, Bahadır Günenc
WP4	Backend Development	Musa Yiğit Yayla	Musa Yiğit Yayla, Bahadır Günenc
WP5	ML model development	Kanan Zeynalov	Kanan Zeynalov, Burak Efe Öğüt,Ege Mehmet Kayaselçuk
WP6	ML integration with backend	Kanan Zeynalov	Kanan Zeynalov, Musa Yiğit Yayla, Burak Efe Öğüt, Ege Mehmet Kayaselçuk, Bahadır Günenc
WP7	Testing	Ege Mehmet Kayaselçuk	Kanan Zeynalov, Musa Yiğit Yayla, Burak Efe Öğüt, Ege Mehmet Kayaselçuk, Bahadır Günenc
WP8	Final Adjustments	Ege Mehmet Kayaselçuk	Kanan Zeynalov, Musa Yiğit Yayla, Burak Efe Öğüt, Ege Mehmet Kayaselçuk, Bahadır Günenc

Table 4: List of work packages in detail

WP 1: *Lİterature Review*			
Start date: *16.09.2024* **End date:** *1.12.2024*			
Leader:	*Burak Efe Öğüt*	**Members involved:**	Kanan Zeynalov, Musa Yiğit Yayla, Burak Efe Öğüt, Ege Mehmet Kayaselçuk, Bahadır Günenc
Objectives: *Read and investigate about the what should be done for differentiating the project from other projects*			

Tasks: ***Task 1.1 Interview with HR*** **:** *Find HR and other people about what should be done for the project* ***Task 1.2 Internet research*:** *Find and detect similar applications and their pros/cons.* *…*			
Deliverables ***D1.1:*** *Websites of similar applications* ***D1.2:*** *Interview Records*			
WP 2: *Discussing project specifications*			
Start date: *7.10.2024* **End date:** *1.12.2024*			
Leader:	*Bahadır Günenc*	**Members involved:**	Kanan Zeynalov, Musa Yiğit Yayla, Burak Efe Öğüt, Ege Mehmet Kayaselçuk, Bahadır Günenc
Objectives: *Based on what we got from the literature review, detect and decide what should be added as a functionality. Make every member clear about what should be done in the future, generating general picture about the project*			
Tasks: ***Task 2.1 Finding fund*** **:** *Communicate with the firms about the fund to support project* ***Task 2.2 Determining requirements*** **:** *Add/Eliminate requirements based on the investigations* *…*			
Deliverables ***D2.1:*** *Company interview report* ***D2.2:*** *Project Specification report*			

WP 3: *Front-end development*			
Start date: *15.11.2024* **End date:** *31.02.2025*			
Leader:	*Musa Yiğit Yayla*	**Members involved:**	Musa Yiğit Yayla, Bahadır Günenc, Ege Mehmet Kayaselçuk
Objectives: *Deciding on UI elements, implementing the Front-end logic to program*			
Tasks: ***Task 1.1 UI Design*** **:** Create basic views for the application to outline the layout of the user interface and ensure all required components are included**.** ***Task 1.2 Integration with back-end*** **:** Connect the front-end with backend APIs to ensure data is fetched and displayed dynamically, enabling real-time updates. *…*			
Deliverables ***D1.1:*** *Figma designs* ***D1.2:*** User Manual			

WP 4: *Backend development*			
Start date: *15.12.2024* **End date:** *31.02.2025*			
Leader:	*Musa Yiğit Yayla*	**Members involved:**	Musa Yiğit Yayla,Bahadır Günenc, Ege Mehmet Kayaselçuk
Objectives: *Implementing back-end logic, ensuring security, reliability etc.*			
Tasks: ***Task 1.1 Plan and develop necessary logics*** **:** *Ensuring the efficiency and security of the application, implementing necessary functions* ***Task 1.2 Plan and develop appropriate database tables*** **:** *Establishing necessary connections among the tables. Writing appropriate views and triggers.* *…*			
Deliverables ***D1.1:*** *API documentation report* ***D1.2:*** *User Manual* ***D1.3:*** *Database schema* ***D1.4****: ER diagram* ***D1.5***: *Sequence Diagram*			

WP 5: *ML model development*			
Start date: *15.11.2024* **End date:** *15.04.2025*			
Leader:	*Kanan Zeynalov*	**Members involved:**	Kanan Zeynalov, Burak Efe Öğüt, Ege Mehmet Kayaselçuk
Objectives: *Developing appropriate model for CV evaluation, performing necessary operations for mıodel training such as data cleaning etc.*			
Tasks: ***Task 1.1 Find and adjust data*****:** *Collect CV datasets from various sources, clean and preprocess the data by removing inconsistencies, missing values, and redundant information to ensure it's suitable for training the ML model.* ***Task 1.2 Model Training*****:** *Train the selected model on the annotated dataset, ensuring that it learns to evaluate CVs based on the defined criteria*			
Deliverables ***D1.1:*** *Model documentation* ***D1.2:*** *Testing report*			

WP 6: *Integrating ML model with backend*			
Start date: *15.04.2025* **End date:** *10.05.225*			
Leader:	*Kanan Zeynalov*	**Members involved:**	Kanan Zeynalov, Musa Yiğit Yayla, Burak Efe Öğüt, Ege Mehmet Kayaselçuk, Bahadır Günenc

Objectives: *Integrating ML model to back-end system so that it can be usable inside the application*
Tasks: ***Task 1.1 Conduct Integrity Testing*** **:** *Making sure that there is no error in integration, making necessary tests* ***Task 1.2 Adding necessary changes on back-end*:** *Adds and adjust backend so that it both runs without interrupting other functionalities in back-end* *…*

WP 7: *Testing*			
Start date: *01.05.2025* **End date:** *10.06.2025*			
Leader:	*Mehmet Ege Kayaselçuk*	**Members involved:**	Kanan Zeynalov, Musa Yiğit Yayla, Burak Efe Öğüt, Ege Mehmet Kayaselçuk, Bahadır Günenc
Objectives: *Make necessary testing to make sure that app works in some extra conditions*			
Tasks: ***Task 1.1 Integrity Testing*** **:** *Investigates the relations among several different components, tries problematic cases* ***Task 1.2 Performance Testing*** **:** *Investigates application's response time in several edge cases* *…*			
Deliverables ***D1.1:*** *Integrity test report* ***D1.2:*** *Performance test report*			

WP 8: *Final adjustments*			
Start date: *20..05.2025* **End date:** *10.06.2025*			
Leader:	*Mehmet Ege Kayaselçuk*	**Members involved:**	Kanan Zeynalov, Musa Yiğit Yayla, Burak Efe Öğüt, Ege Mehmet Kayaselçuk, Bahadır Günenc
Objectives: *After dealing with bugs and development processes, makes final adjustments such as UI designs etc.*			
Tasks: ***Task 1.1 UI refinement*** **:** *Adjust UI so that it apples more user friendly* ***Task 1.2 Deployment Preparation*** **:** *Prepare the system for deployment by creating a deployment plan and ensuring all configurations are complete*			
Deliverables ***D1.1:*** *Deployment Plan* ***D1.2:*** *Suggested changes report*			

4.4 Ensuring Proper Teamwork

Proper teamwork is essential for collaboration. Therefore, noting the tasks are necessary. Because of this, we have decided to use planning applications. Notion is the sprint planning application where noting the tasks and sprint planning is possible. This application also has filters showing tasks based on the due time or person. With those filters, we can determine the workload of each person in the project. Another tool that we used is discord. With the help of the discord, we manage to make meetings even if we are far from each other. Another benefit of the discord is grouping the different tasks in different chat rooms. This feature of discord eases document management. We also have a WhatsApp group to enhance the daily communication among team members. This group has a huge positive effect on the relationship among team members. With the help of WhatsApp messaging, we are able to manage the positive energy among the team members. Lastly, workload is increasing and deliverable deadlines are coming up, we meet face to face to enhance the communication and keep motivation high among team members. With the help of face to face interactions, we are able to handle the tasks efficiently and faster.

4.5 Ethics and Professional Responsibilities

Recruit4Me will be implemented based on professional and ethical standards to be reliable in terms of fairness, transparency and accountability. Robust data privacy measures will protect user information along with dynamic token generation, authentication and authorization of the accounts and comply with GDPR kind of regulations. It is crucial to avoid biased decision-making in CV grading and candidate evaluation based on factors such as gender, ethnicity and race. AI datasets will utilize diverse datasets to resolve this issue. This application will ensure ethical use of external and third-party APIs and services which is a subject matter to be trustworthy and maintain integrity.

4.6 Planning for New Knowledge and Learning Strategies

Our team is eager to apply knowledge and technologies which we are already proficient with, but also to learn new knowledge and apply it to Recruit4Me. For learning new technologies and obtaining new knowledge, we will conduct individual research, collaborate with other teams, and share how we overcome difficulties and errors within our team. Moreover, we aim and hope to officially release Recruit4Me as a real product, and explore deployment and application lifecycle management domains by doing so. In brief, we will adaptively learn and apply new knowledge to overcome possible issues which we may face throughout our journey with Recruit4Me.

5 Glossary

- **HR (Human Resources):** HR department in an organization is responsible for managing employee-related functionalities, such as recruitment, training, rating performance and workplace policies. In the Recruit4Me application, HR is a major part of the system, handling large volumes of candidate applications and assisting automation. This system facilitates CV parsing, rating candidates, test generation and execution, scheduling interviews which minimize manual interventions. HR can focus on efficiently hiring a candidate with assistance of this automation. AI and ML will further enhance HR operations by providing unbiased data and evaluation of applicants.

- **AI (Artificial Intelligence):** Artificial intelligence aids in developing the systems which are capable of doing tasks which mostly require human intelligence, such as decision-making, natural language processing(NLP), and recognition of patterns. In Recruit4Me, AI will be crucial in automatization of the candidate evaluation process. Speech-to-Text for online interview response evaluation, generating questions based on level and domain of the candidate with the assistance of multiple APIs such as OpenAI. These will ensure HR workflows stay consistent and unbiased.

- **ML (Machine Learning):** ML is an AI component which focuses on building algorithms, models to learn from data and make predictions beforehand without explicit programming. In Recruit4Me, ML will be utilized in the CV evaluation pipeline, where the Model Evaluation Engine is used. These compose of tasks such as analyzing candidate CVs for requested qualifications by the company, rating candidates based on a predefined criteria, and give them ranks to develop assessment for further states of the laboring. ML algorithms provide feedback for improvement in addition, and continuously refine the evaluation methods as more data is gained, increasing accuracy and reliability of the system.

6 References

Object-Oriented Software Engineering, Using UML, Patterns, and Java, 2nd Edition, by Bernd Bruegge and Allen H. Dutoit, Prentice-Hall, 2004, ISBN: 0-13-047110-0.

[2] “ISO 25010.” Online. Available: https://iso25000.com/index.php/en/iso-25000-standards/iso-25010. [Accessed 11-Dec-2024].

[3] “Web Content Accessibility Guidelines (WCAG) 2.1,” Sep. 2023. Online. Available: https://www.w3.org/TR/WCAG21/. [Accessed 11-Dec-2024].

[4] IEEE Computer Society and Software Engineering Standards Committee, *IEEE Recommended Practice for Software Requirements Specifications*. IEEE Computer Society, 1998.