CSC1097

HireTrack Functional Specification

Siri Nandipaty - 21449384

Anushree Umak - 21343003

0. Table of contents

1. Introduction

1.1 Overview

HireTrack is a platform designed to optimise and match candidates' resumes with specific job descriptions, simplifying the recruitment process. By offering valuable data insights on job search progress, users will have a clearer path to success in landing their desired roles.

Many job seekers often overlook the need to tailor their CVs to specific job postings, facing challenges due to different industries' unique skills and keyword requirements. This can result in missed opportunities and make finding a job more difficult.

To address this, HireTrack will provide advanced AI-driven insights on how well a CV matches job postings using techniques such as BERT[1], TF-IDF[2], and cosine similarity[3]. The platform will also simplify job search management, allowing users to track applications, visualise their progress, and schedule important milestones by integrating Google Calendar. Direct API connections with leading job platforms will offer users a wide array of opportunities.

Built with Django, React, and Firebase, HireTrack will prioritise security, user-friendliness, and effectiveness to help job seekers perfect their applications and maximise their success.

1.2 Business Context

Operating within the dynamic recruitment and job search industry, HireTrack solves some of the most significant pain points for job seekers and recruiters by making the hiring process faster and more relevant.

Through AI-powered CV matching, and integrations with leading job boards such as Indeed[1], LinkedIn[2], and Glassdoor[3], HireTrack aims to provides insights to candidates on optimising their applications and staying on top of their application pipeline.

This approach makes time-consuming work, like adjusting one's CV a lot faster, to fit a particular post. It also allows the recruiters to connect with a qualified candidate in a much more targeted manner. Ultimately, HireTrack aims to improve job placement success, simplify recruitment efforts, and offer users a clear path to achieve professional goals effectively.

[1] https://ie.indeed.com/?r=us
[2] https://www.linkedin.com/
[3] https://www.glassdoor.com/index.htm

1.3 Glossary

AI (Artificial Intelligence) - A technology that enables computers to learn from data, make decisions, and perform tasks that typically require human intelligence.

Analytics Dashboard - A visual display that shows useful data and insights to help users understand their job search progress.

APIs (Application Programming Interfaces) - Tools that allow different software programs to communicate and share information.

Application Tracker - A tool that helps users keep track of job applications, interviews, and deadlines in one place.

Backend - The behind-the-scenes part of a website or app that processes data, manages user accounts, and performs tasks users don't see.

BERT (Bidirectional Encoder Representations from Transformers) - A model that helps computers understand the meaning of words in context, improving their ability to comprehend human language.

Calendar Integration (Google Calendar Sync) - A feature that connects with your calendar to automatically add events like interviews and reminders.

Cloud Services - Online storage and computing tools that allow users to access and use data and applications over the internet.

Cosine Similarity - A method to measure how similar two things are by comparing their content, often used to see how closely a CV matches a job description.

Django - A high-level Python web framework that encourages rapid development and clean, pragmatic design. It's used for building the server side of web applications.

Encryption - The process of converting data into a coded form to prevent unauthorised access.

Firebase - A platform developed by Google for building and managing mobile and web applications, offering services like real-time databases, authentication, and hosting.

Frontend - The part of a website or app that users see and interact with, like buttons, text, and layout.

GDPR (General Data Protection Regulation) - A set of rules in Europe designed to protect people's data and privacy online.

GPT (Generative Pre-trained Transformer) - An AI model that can generate human-like text, often used to create suggestions or responses based on user input.

Natural Language Processing (NLP) - A type of AI that helps computers understand, interpret, and respond to human language.

Static Database - A database that is manually updated and not constantly changed by external or real-time inputs.

TF-IDF (Term Frequency-Inverse Document Frequency) - A technique used to identify important words in a text by measuring how frequently they appear, helping to highlight key terms.

User Authentication - The process of verifying a user's identity, typically using a combination of a username and password.

User Experience (UX) - The overall experience a person has when interacting with an application or website, including ease of use and satisfaction.

User Interface (UI) - The design and layout of a website or app that users see and interact with.

2. General Description

2.1 Product / System Functions

HireTrack is a platform that makes job searching and recruiting easier with AI-powered analysis, simple tracking tools, and smooth integrations. Each feature is built to engage users, offer useful insights, and simplify complex tasks when it comes to entering the job market.

The main functions of this app are:

- **AI-Powered CV-Job posting Comparison**: The platform compares CVs with job postings using AI techniques such as BERT, TF-IDF, and cosine similarity to show users if their experience and skill set matches with the job posting.
- **Job Tracking & Analytics**: Users can track their applications, visualise progress, and get data-driven insights to understand how well their CV performs against different job descriptions, highlighting gaps in skills and areas that are in need of improvement.
- **Calendar Integration**: The website synchronises with Google Calendar, allowing users to manage deadlines, interviews, and application milestones.
- **Job Search API Integration**: The website integrates with job platforms Google Jobs and Glassdoor, giving users access to a wide range of job opportunities straight from the source.
- **Collaborative CV Editing**: Users can share their CVs for real-time feedback from mentors or colleagues.

2.2 User Characteristics and Objectives

2.2.1 Job Seekers

Target Users: Users looking to optimise and improve their CVs to better align with job descriptions, enhancing their chances of securing their dream job.

User Expertise: Job seekers will be expected to have a basic understanding of skills in their desired area of application. They will also be expected to be familiar with managing CVs, applying for jobs, and using online platforms.

User Persona Example: Anna, a recent graduate, struggles to tailor her CV to each job posting. HireTrack would offer her tailored suggestions, saving her hours of manual editing.

Objectives and Responsibilities:

Primary Requirements:

- Ability to easily upload and manage multiple versions of CVs.
- Simple, user-friendly interface for uploading job descriptions.
- Instant feedback on CV-job matching.
- Efficient tracking of job applications and interview schedules.
- Visual progress reports to track job search success.

“Wish List” Features:

- Instant collaboration tools for sharing and receiving feedback on CVs from mentors or professors..
- Integration with LinkedIn to pull job data directly, and advanced filters for job searches.
- Auto suggestions for relevant job openings based on CV content.
- Advanced and personalised GPT suggestions.

Feasible Solutions:

- Focus on CV-job matching and providing personalised suggestions using AI tools like BERT and GPT.
- Develop a straightforward application tracker with Google Calendar integration to help job seekers better organise and manage their job search tasks.

2.2.2 Recruiters

Target Users: Recruiters aiming to quickly identify and evaluate CVs that match the job openings they've posted.

User Expertise: Recruiters will be generally expected to have expertise in evaluating CVs and prefer intuitive solutions that simplify candidate screening.

User Persona Example: John, a recruiter for a tech firm, needs to evaluate hundreds of applicants efficiently.

Objectives and Responsibilities:

Primary Requirements:

- Easy-to-use search interface to filter and find CVs that match job descriptions.
- Ability to view detailed match scores between CVs and job postings, highlighting key skills and experience.

“Wish List” Features:

- Advanced filters based on industry, skills, and experience to refine candidate searches.

Feasible Solutions:

- Implement a simple but effective CV search and filtering system with relevant skills.

2.3 Operational Scenarios

Use Case 1: Job Seeker Uploads CV and Job Posting
Actors: - Job Seeker - System(HireTrack)
Trigger: - The user wants to determine the match score between their CV and a job description.
Pre-Conditions: - The user has internet access. - The user is logged in to the HireTrack platform.

Post-Conditions:

- The match score is displayed.
- Suggestions are generated for the user's CV.

Success Scenario:

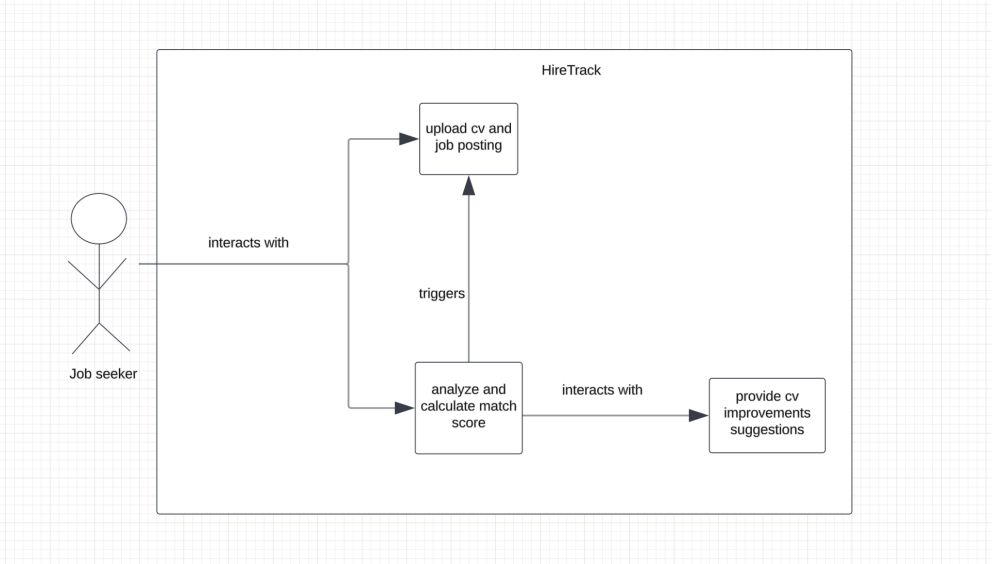
Users: Uploads their CV and a job description from a job board (e.g., Google Jobs) to the HireTrack website.

System: The platform analyses both the CV and job description using BERT or TF-IDF based on the user's choice.

Users: Receives the match score and suggestions for improving the CV to match the job description better.

Quality Requirements:

- The analysis process should typically be completed in a few seconds.
- Suggestions provided should generally align with most of the input job description keywords.



(fig 1)

Use Case 2: Job Seeker Edits Their CV
Actors: - Job Seeker - System(HireTrack)
Trigger: - The user wants to update their CV on the platform.
Pre-Conditions: - The user is logged in to the HireTrack platform. - The user has an existing CV uploaded to the platform.
Post-Conditions: - The CV is updated with the user's changes. - A new version of the CV is ready for download.
Success Scenario: - User: Edits their CV within the website. - System: Saves the changed CV and provides an option for the user to download the revised file as a PDF. - User: Downloads the updated CV in PDF format.
Quality Requirements: - The CV should be saved without errors. - The PDF download should be available immediately after saving.

(Fig 2)

Use Case 3: Job Seeker Tracks Application

Actors:

- Job Seeker
- System(HireTrack)

Trigger:

- The user applies for a job and wants to track interview dates and deadlines.

Pre-Conditions:

- The user has applied for at least one job.
- The user has set up their calendar and connected it with the platform.

Post-Conditions:

- The application is logged in the tracker.
- Interview dates and deadlines are synchronised with the user's Google Calendar.

Success Scenario:

- User: Inputs the interview date and deadlines into the platform after applying for a job.
- System: Logs the application and syncs interview dates with Google Calendar.
- User: Receives reminders for upcoming interviews and deadlines.

Quality Requirements:

- Synchronisation with Google Calendar should occur without delay.
- Reminders should usually be sent roughly a day in advance.

fig (3)

Use Case 4: Recruiter Searches for Matching CVs

Actors:

- Recruiter
- System(HireTrack)

Trigger:

- The recruiter wants to find candidates whose CVs match a specific job description.

Pre-Conditions:

- The recruiter is logged into the HireTrack platform.
- The recruiter has a job description or search criteria.

Post-Conditions:

- The system displays a list of CVs that match the search criteria.
- The recruiter can view the match scores and detailed profiles of the CVs.

Success Scenario:

- User: Logs into the platform and searches for CVs based on specific skills and job requirements.
- System: Displays CVs that best match the job description, with a match score and detailed profile of each CV.
- User: Reviews the displayed CVs and evaluates suitable candidates.

Quality Requirements:

- The system should prioritise the most relevant CVs based on the match score.
- The profiles should load without significant delays.

fig(4)

Use Case 5: User Requests CV Feedback from Mentor

Actors:

- Job Seeker
- Mentor
- System(HireTrack)

Trigger:

- The job seeker wants to get feedback on their CV.

Pre-Conditions:

- The user has an existing CV uploaded to the platform.
- The user has selected a mentor for feedback.

Post-Conditions:

- The mentor receives a request for feedback on the CV.
- The user receives actionable feedback from the mentor.

Success Scenario:

- User: Share their CV with a mentor for feedback through the platform.
- System: Sends a link to the mentor, allowing real-time comments and suggestions.
- User: Receives actionable feedback and can make changes directly in the system.

Quality Requirements:

- The feedback should be delivered within about a day of the request.
- The system should allow real-time collaboration and comment tracking.

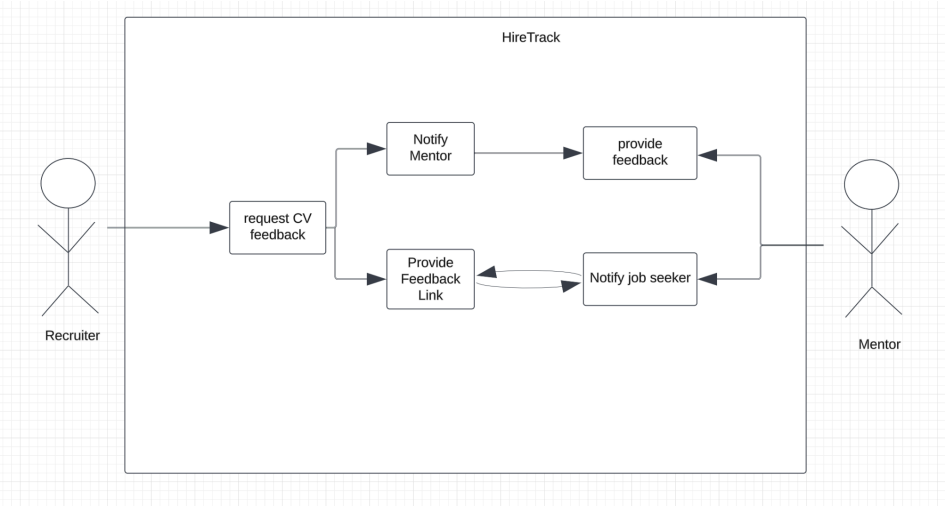


fig (5)

2.4 Constraints

Performance	The system should handle multiple users uploading and analysing CVs simultaneously without performance degradation.
Industry Protocols	The system must comply with data privacy regulations, especially regarding CV uploads (e.g., GDPR, if applicable), to ensure the protection of user data, especially regarding CV uploads. All sensitive data, including CVs, personal information, and user communications, must be stored securely
Speed Requirements	The system should provide real-time matching results within seconds (after the CV and job description are uploaded).
Security	The system must ensure data security, with encryption in place for CV uploads, user data, and communication. All sensitive information must be stored securely.
User Accessibility:	The system must adhere to accessibility standards,

	including text-to-speech capabilities, high-contrast modes, and keyboard navigability, to ensure it is usable by individuals with disabilities. This includes compliance with WCAG (Web Content Accessibility Guidelines) and offering features such as customisable font sizes and alternative navigation methods for ease of use.
Platform Compatibility	The system is optimised for Chrome, Firefox, and Safari to reach a broad range of users.
Platform Compatibility	The system should be compatible with all major browsers (Chrome, Firefox, Safari).
Testing and Device Availability:	Testing will be conducted across all supported browsers and platforms, with particular attention to device compatibility and screen resolutions. Ensuring compatibility across devices with different network conditions and processing power is critical. Limited access to certain devices may affect performance optimisation across the broadest user base.

3. Functional Requirements

FR001 User Authentication and Registration:

Description: - The system will ensure that users can create an account through our application. Each account must be connected to the user's email address and include a unique username and a secure password. To complete the registration process, the account must be authenticated via the user's email. Existing users will have the ability to log in using their credentials. Additionally, users will have the option to reset their password if they forget it.
Criticality: - High. - This is a critical requirement because, without the ability for users to register on the application, there would be no way for them to interact with it. Authentication is also a vital feature, as it ensures user privacy is maintained. Without proper authentication, sensitive information could be compromised.
Technical Issues: - The system must restrict the usernames that users can choose to ensure that every user has a unique username. Additionally, the system will require

users to select strong and secure passwords. These passwords must include a combination of capital letters, numbers, and special characters to enhance security.
Dependencies: - All subsequent features, such as user profiles, data uploads, and CV analysis, are dependent on the successful completion of user registration and authentication.

FR002 User Profiles:

Description: - When a user registers, they should have the ability to set a profile picture, a bio, and a display name to personalise their experience on the application. Additionally, when a user clicks on their profile, they should see options to either log out or delete their account.
Criticality: - Low. - The users do not need to customise their profiles to access the full functionality of the application.
Technical Issues: - There is an issue with scalability, as the population of the website grows we will need to accommodate for more existing profiles and information linked to it.
Dependencies: - This feature is dependent on the successful implementation of the registration and authentication processes.

FR003 CV Upload & Job Posting Upload:

Description: - The system should allow users to upload their CVs and job postings for analysis. Users must be able to upload a PDF version of their CV so that the platform can process it and calculate a match score; this will be based on algorithms such as BERT, TF-IDF, and cosine similarity.
Criticality: - High - This feature is of high criticality because all other functionalities of the application depend on its successful implementation.

Technical Issues: - The system must ensure the proper parsing of various file formats, such as PDF and DOCX. Additionally, there may be challenges related to API integration and the management of the static database used for processing.
Dependencies: - This feature depends on successful user authentication and registration. It is also reliant on the file upload functionality and the successful integration of APIs.

FR004 AI Matching (CV to Job Description):

Description: - The system should allow users to upload their CVs and job postings for analysis. Users must be able to upload a PDF version of their CV so that the platform can process it and calculate a match score. This analysis will be based on algorithms such as BERT, TF-IDF, and cosine similarity.
Criticality: - High. - This feature is crucial, as the successful implementation of all other application functionalities depends on it.
Technical Issues: - The system must ensure the proper parsing of various file formats, such as PDF and DOCX. Additionally, there may be challenges related to API integration and the management of the static database used for processing.
Dependencies: - This feature depends on successful user authentication and registration. It is also reliant on the file upload functionality and the successful integration of APIs.

FR005 Application Tracker & Google Calendar Sync:

Description: - The system will allow users to track their job applications and sync interview dates with Google Calendar. When a user has completed a job application, they can record it. The system will display their application status and highlight any upcoming interview dates.
Criticality: - Medium. - While it is not a core functionality, it enhances user organisation and engagement, thereby improving the overall job-seeking experience.

Technical Issues: - There are potential challenges with integrating the system with the Google Calendar API and ensuring accurate tracking of job application statuses and interview dates.
Dependencies: - This feature requires the user to input their job application data for tracking purposes.

FR006 Analytics Dashboard:

Description: - The system will provide users with visual insights into their job search process and the performance of their CV. These insights will help users understand how their job hunt is progressing and identify areas for improvement.
Criticality: - Medium. - It enhances user engagement by providing actionable data and progress tracking, making the job-seeking process more interactive and informative.
Technical Issues: - There are challenges in developing visualisations that are both intuitive and useful for users. Furthermore, implementing accurate models to generate reliable insights may pose technical difficulties.
Dependencies: - This feature depends on data generated by AI matching algorithms and requires users to update the application tracker with their job application details.

FR007 CV Search for Recruiters:

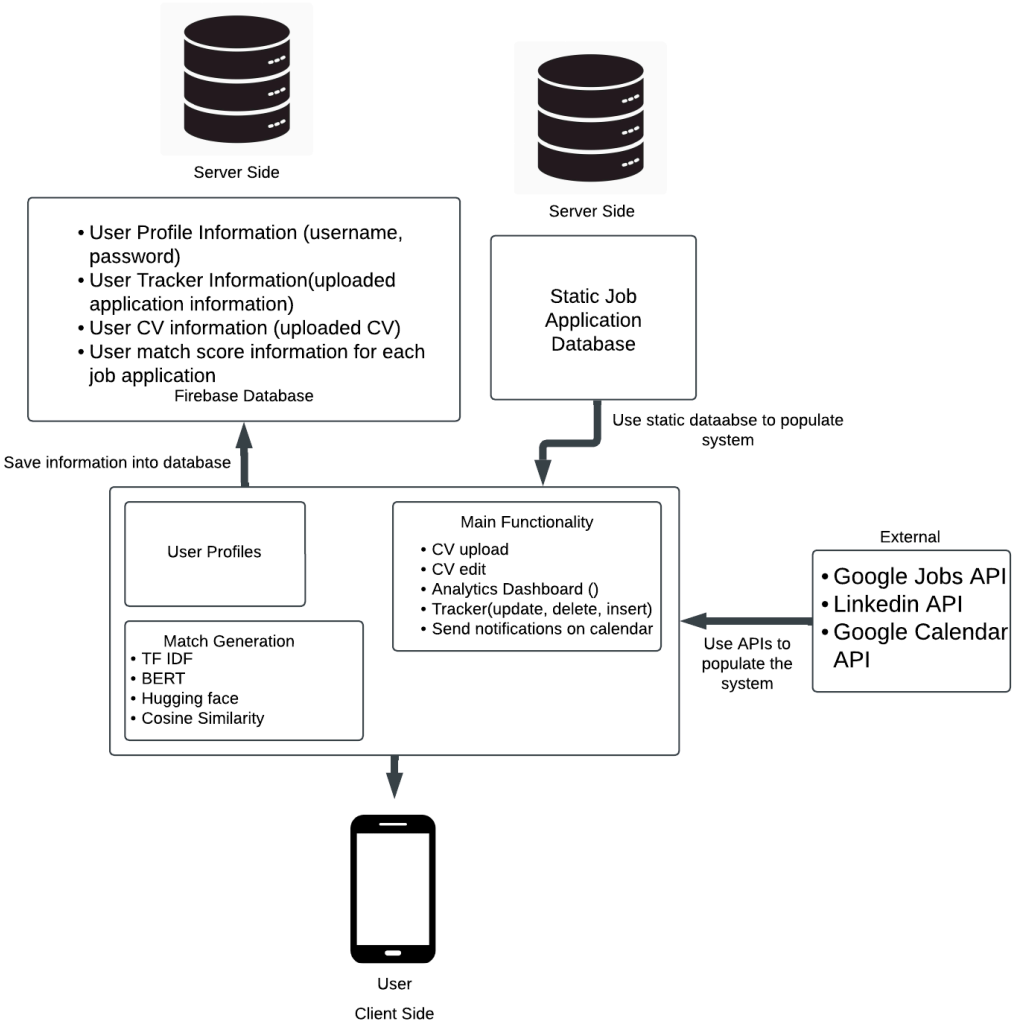
Description: - The system will provide a search interface for recruiters to find CVs that match specific job postings. This feature will enable recruiters to spend less time searching for suitable candidates by leveraging the AI functionalities implemented within the system.
Criticality: - Medium. - It supports the expanded functionality of the system by catering to a secondary user base, specifically recruiters.
Technical Issues: - There are challenges in developing search algorithms that can

accurately rank CVs by relevance to the job postings.
Dependencies: - This feature is dependent on the successful upload of CVs and the performance of the AI matching algorithms.

FR008 Personalised CV Suggestions:

Description: - The system will use GPT to provide feedback and suggestions for optimising a CV based on the requirements outlined in the job description.
Criticality: - High. - This feature directly enhances the quality of the user's CV and improves their job-seeking potential.
Technical Issues: - There are challenges in integrating GPT models and ensuring that the suggestions provided are both relevant and useful to the user.
Dependencies: - This feature is dependent on the results of the job-CV matching process.

4. System Architecture



Client Side:

The client side is the part of the application that users interact with. It provides an interface where users can:

- Upload CVs
- Edit CVs
- Compare CVS with job descriptions
- Track job applications
- Search for relevant job opportunities or candidates.

This client-side functionality will be built using web technologies such as HTML, CSS, Javascript and React

Server Side:

The server-side architecture will manage data processing and storage. It includes:

- Data processing
- Match scoring
- Job application tracker
- User Authentication

This will be built using a backend framework on Python/Django and Firebase.

Database:

All content created by users, along with their interactions and application data, is stored in a Firebase database.

If the external job APIs fail, we have a static database with CV information to keep the app functional.

Third-Party Integrations:

- LinkedIn API: Allows users to connect to LinkedIn to find job listings
- Google Jobs API: Provides access to job postings so users can find job listings directly within the app.

Components which are being reused:

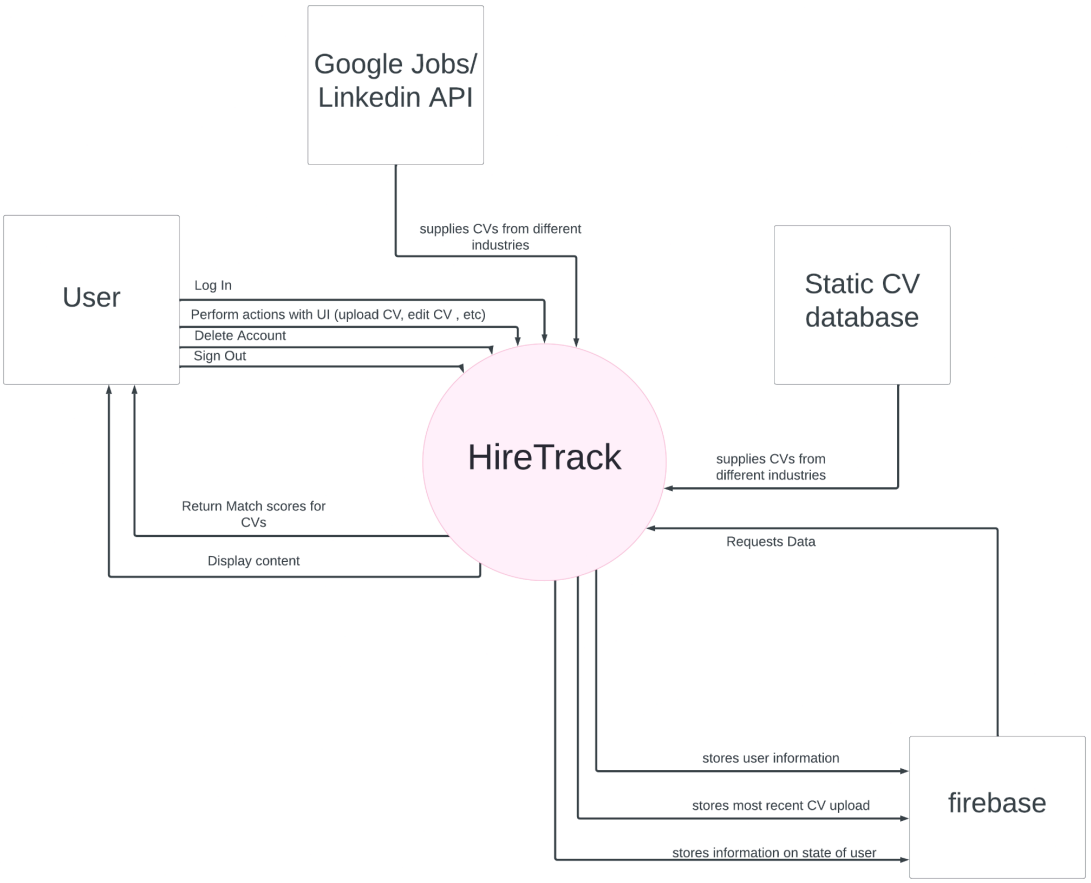
- Firebase Authentication: Used for secure user sign-in and access control across the platform.
- BERT/TF-IDF Models: Pre-trained models from third-party libraries that will be used to analyse and compare CVs with job descriptions.

Architectural Flow:

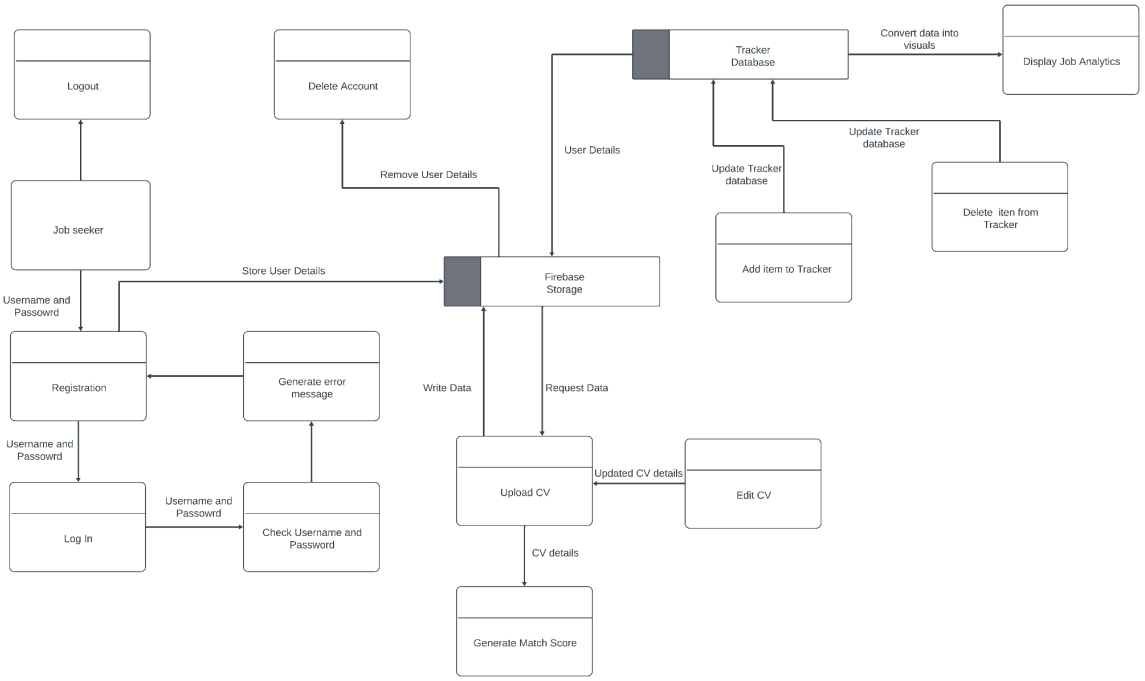
1. Client Side: Users interact with the platform through the web interface.
2. API Layer: User actions (e.g., uploading a CV) trigger API calls to the server.
3. Server-Side Logic: The server processes the user's request, runs AI algorithms, and fetches job data from third-party APIs.
4. Database: Relevant user data is stored and updated in Firebase for persistence.
5. Third-Party Integrations: APIs like Google Jobs and LinkedIn provide additional functionality and data for the platform.

5. High-Level Design

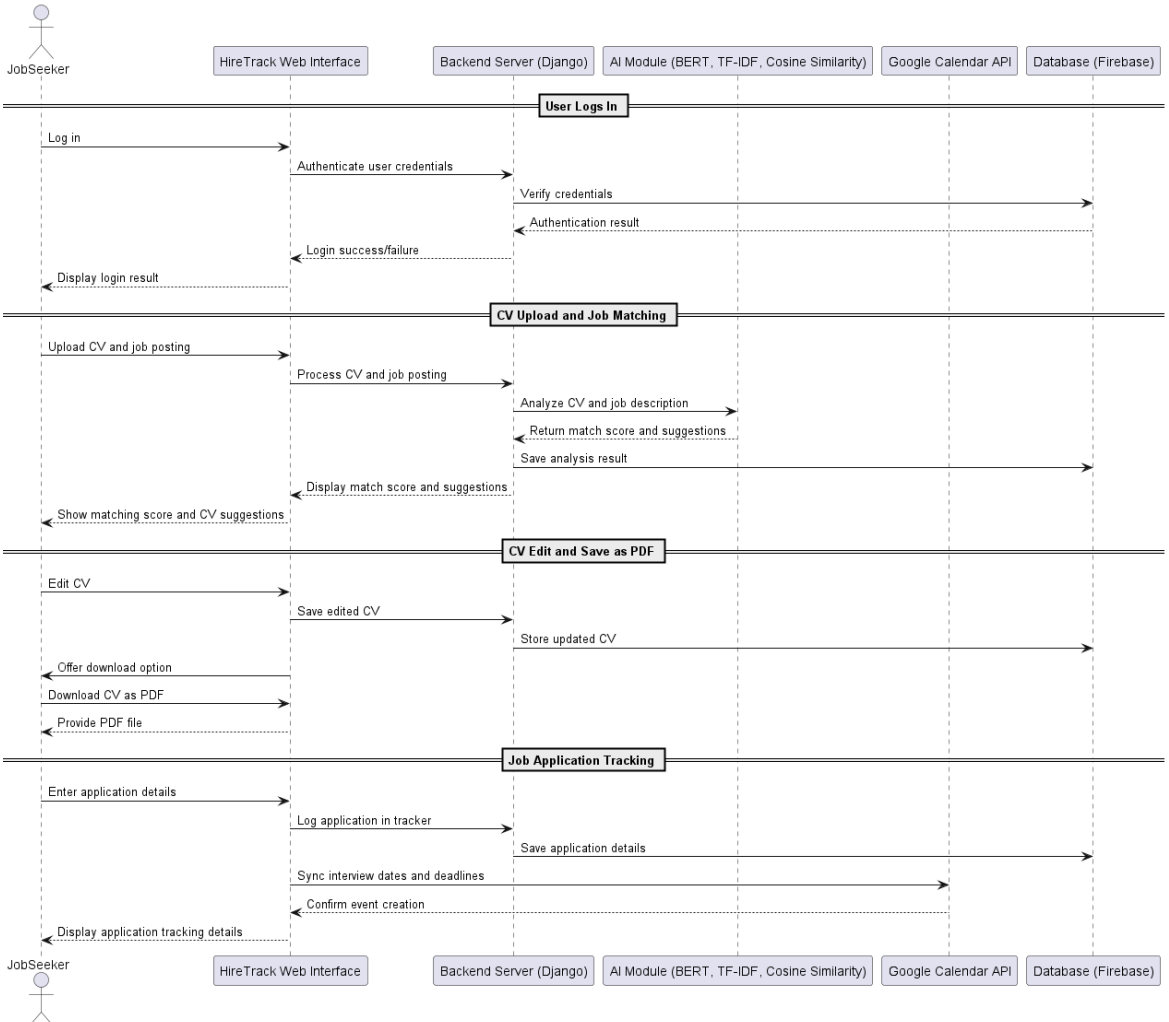
5.1 Context Diagram



5.2 Data Flow Diagram



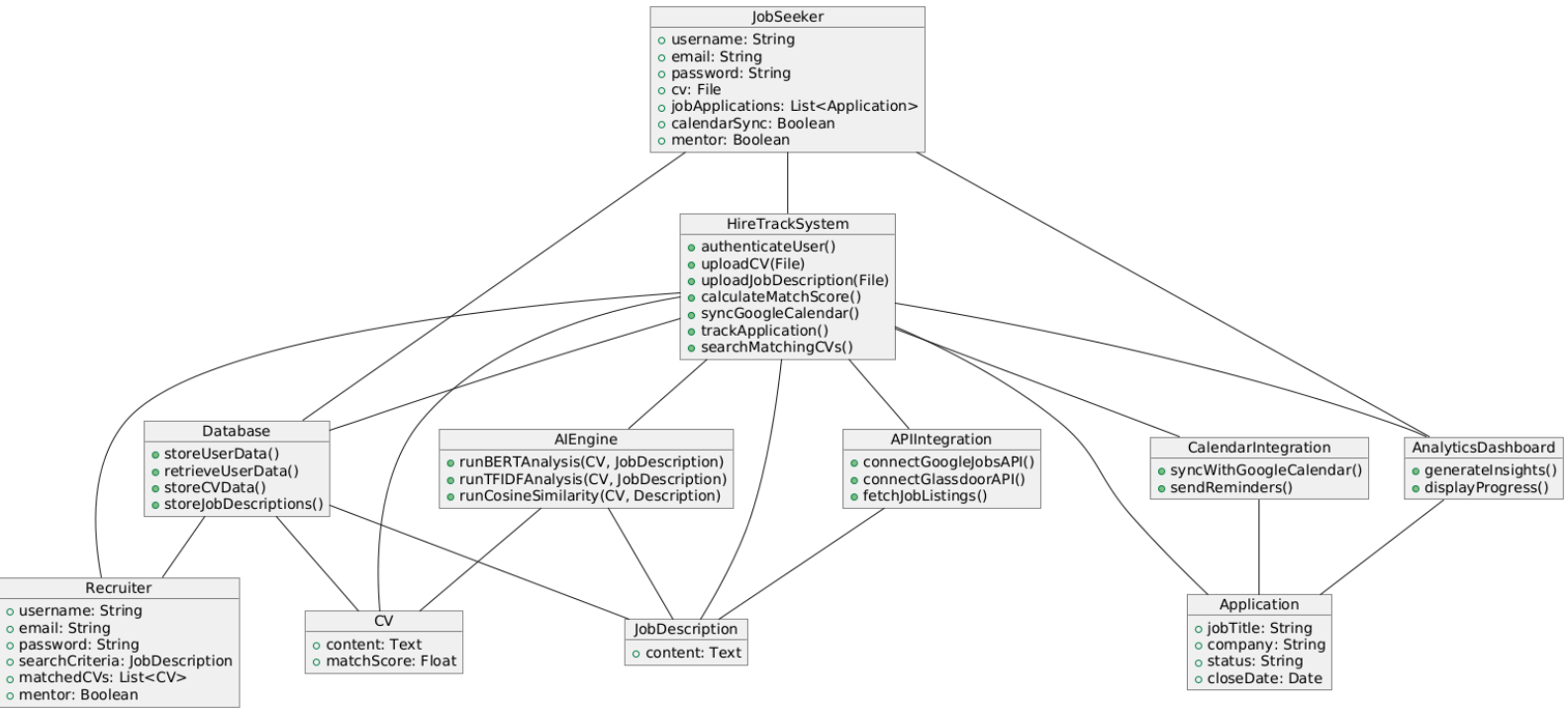
5.3 Sequence Diagram



1. **User Logs In**: The job seeker logs in, and the system verifies their credentials with Firebase.
2. **CV Upload and Job Matching**: The job seeker uploads their CV and a job posting, which is analysed by the AI module (using BERT, TF-IDF, etc.). The match score and improvement suggestions are saved and displayed to the user.
3. **CV Edit and Save as PDF**: The user can edit the CV, save the changes, and download it as a PDF.
4. **Job Application Tracking**: The job seeker logs an application, which includes syncing dates with Google Calendar for reminders. This data is stored and displayed in their application tracker.

Diagram was made using PlantUML

5.4 Object Diagram



Key Entities

1. **JobSeeker**: Users uploading CVs, tracking applications, and syncing calendars. They will also have the option of giving real time feedback to other CVs.
2. **Recruiter**: Professionals searching for CVs matching specific job descriptions. They can also opt to give real time feedback on CVs that are shared with them.
3. **HireTrackSystem**: Core platform managing authentication, CV uploads, job matching, calendar sync, and application tracking.

Supporting Components

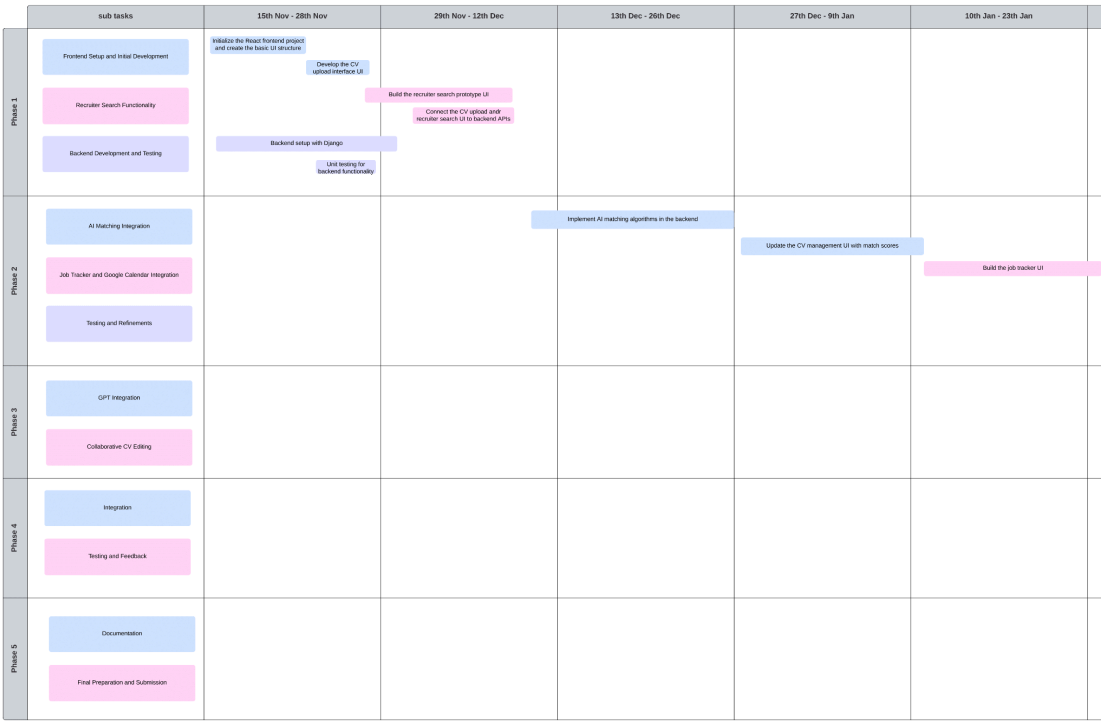
- **CV & JobDescription**: Represent user resumes and job postings, analyzed for compatibility.
- **Application**: Tracks job application details like status and deadlines.
- **CalendarIntegration**: Syncs applications with Google Calendar and sends reminders.
- **AnalyticsDashboard**: Displays user progress and insights.
- **APIIntegration**: Connects to job platforms like Google Jobs for listings.
- **AIEngine**: Analyses CVs and job descriptions using BERT, TF-IDF, and cosine similarity.
- **Database**: Stores user profiles, CVs, and application data.

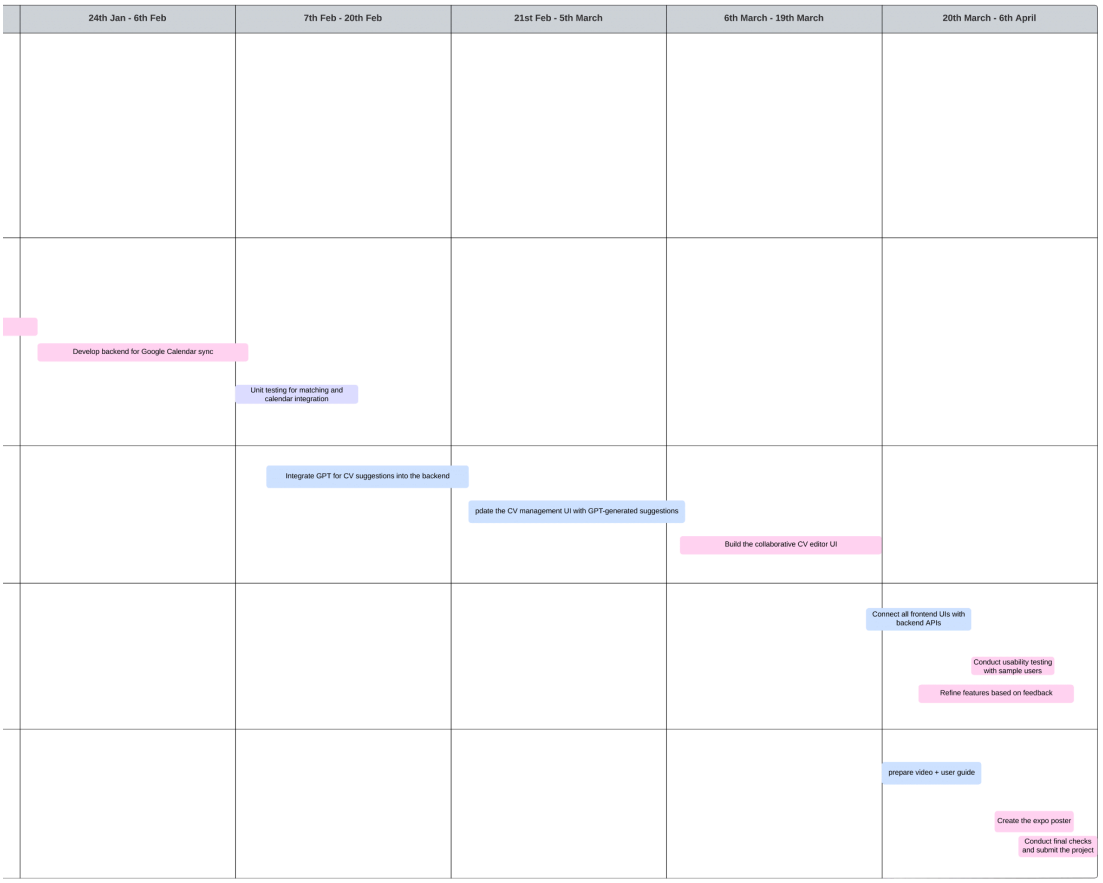
Relationships

- **JobSeeker** and **Recruiter** interact with the **HireTrackSystem**, which coordinates analysis, reminders, and external integrations.
- Supporting features like **CalendarIntegration**, **Mentor**, and **AnalyticsDashboard** enhance user experience.
- **AIEngine** and **APIIntegration** power intelligent matching and job search functionality.

Diagram was made using PlantUML

6. Preliminary Schedule





Phase 1: Basic Functionality (15th November - 10th December)

Task 1: Frontend Setup and Initial Development

- Subtasks:
    - Initialise the React frontend project and create the basic UI structure (15th Nov - 20th Nov).
    - Develop the CV upload interface UI (21st Nov - 25th Nov).

Task 2: Recruiter Search Functionality

- Subtasks:
    - Build the recruiter search prototype UI (26th Nov - 10th Dec).
    - Connect the CV upload and recruiter search UI to backend APIs (1st Dec - 10th Dec).

Task 3: Backend Development and Testing

- Subtasks:
    - Backend setup with Django (user authentication, CV upload, recruiter search endpoint) (15th Nov - 30th Nov).
    - Unit testing for backend functionality (26th Nov - 28th Nov).

Phase 2: Extended Functionality (11th December - 7th February)

Task 1: AI Matching Integration

- Subtasks:
    - Implement AI matching algorithms (cosine similarity, TF-IDF) in the backend (11th Dec - 27th Dec).
    - Update the CV management UI with match scores (28th Dec - 10th Jan).

Task 2: Job Tracker and Google Calendar Integration

- Subtasks:
    - Build the job tracker UI (11th Jan - 24th Jan).
    - Develop backend for Google Calendar sync (25th Jan - 7th Feb).

Task 3: Testing and Refinements

- Subtasks:
    - Unit testing for matching and calendar integration (8th Feb - 10th Feb).

Phase 3: Experimental Features (8th February - 20th March)

Task 1: GPT Integration

- Subtasks:
    - Integrate GPT for CV suggestions into the backend (8th Feb - 21st Feb).
    - Update the CV management UI with GPT-generated suggestions (22nd Feb - 6th Mar).

Task 2: Collaborative CV Editing

- Subtasks:
    - Build the collaborative CV editor UI (7th Mar - 20th Mar).

Phase 4: Integration and Final Testing (21st March - 1st April)

Task 1: Integration

- Subtasks:
    - Connect all frontend UIs with backend APIs (login, data retrieval, saved layout loading) (21st Mar - 25th Mar).

Task 2: Testing and Feedback

- Subtasks:
    - Conduct usability testing with sample users (26th Mar - 28th Mar).
    - Refine features based on feedback (29th Mar - 1st Apr).

Phase 5: Documentation and Submission (2nd April - 6th April)

Task 1: Documentation

- Subtasks:
    - Prepare the video walkthrough (2nd Apr - 3rd Apr).
    - Prepare the user guide (2nd Apr - 4th Apr).
    - Prepare the technical specification document (4th Apr - 5th Apr).

Task 2: Final Preparation and Submission

- Subtasks:
    - Create the expo poster (5th Apr - 6th Apr).
    - Conduct final checks and submit the project (6th Apr).

7. Appendices

[1] - https://arxiv.org/abs/1810.04805
[2] - https://arxiv.org/abs/2308.04037
[3] - https://www.datastax.com/guides/what-is-cosine-similarity
[4] - https://www.fepbl.com/index.php/csitrj/article/view/859

Fig 2-

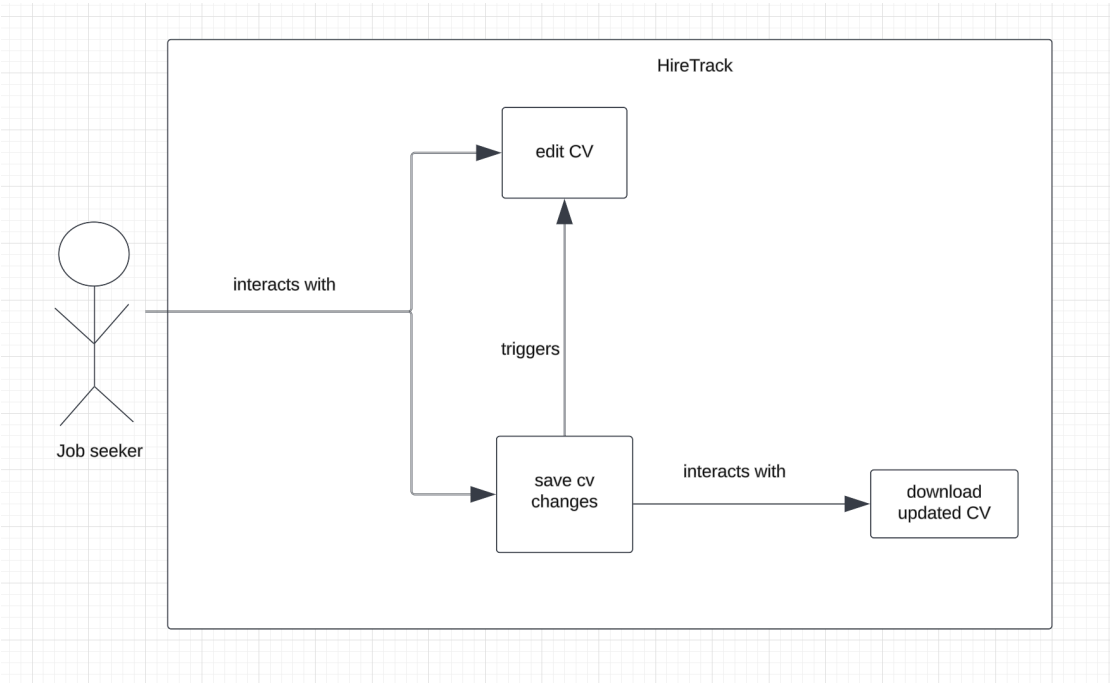


Fig 3 -

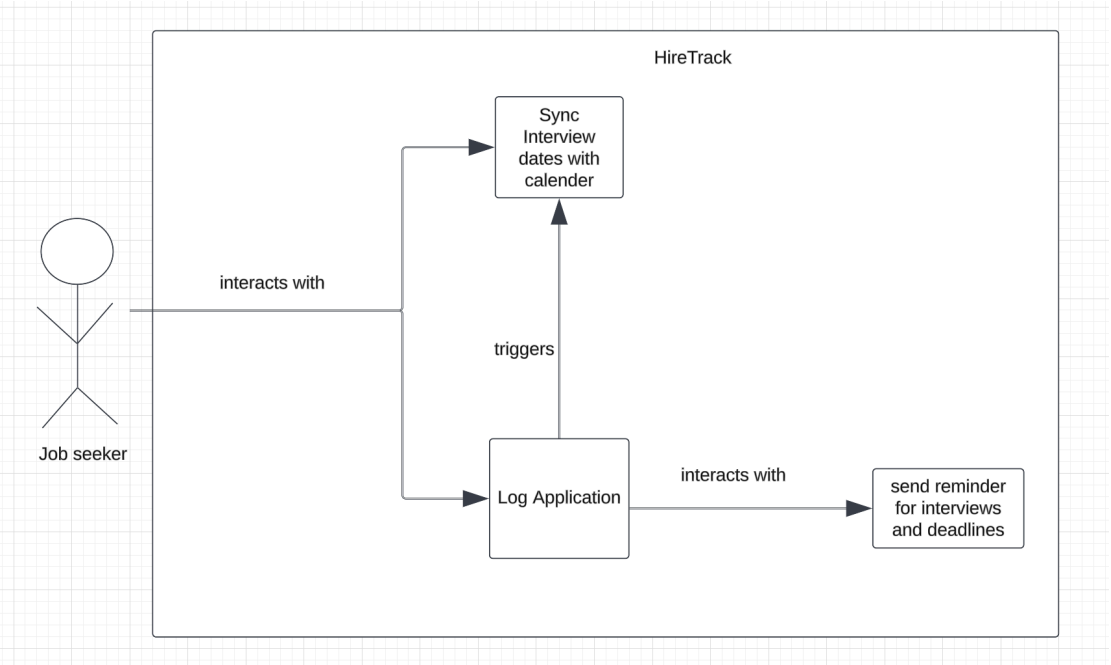
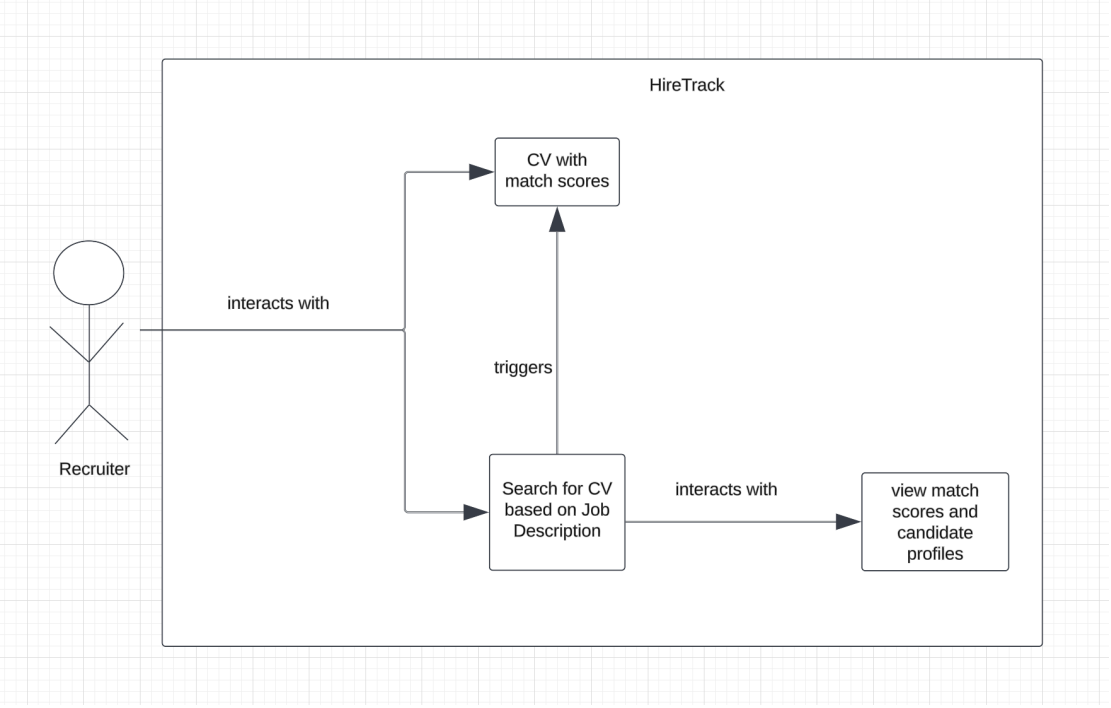

HireTrack
Sync Interview dates with calender
interacts with
triggers
Job seeker
Log Application
interacts with
send reminder for interviews and deadlines

Fig 4-


HireTrack
CV with match scores
interacts with
triggers
Recruiter
Search for CV based on Job Description
interacts with
view match scores and candidate profiles