

# AI HR Recruiter – Project Documentation

## AI HR Recruiter – Project Documentation

### Introduction

AI HR Recruiter is a proof-of-concept application that leverages modern natural language processing and machine learning to streamline the early stages of hiring. Traditional resume screening is manual, time-consuming and prone to inconsistency; by contrast, this tool parses resumes automatically, ranks candidates against a job description and generates personalised follow-up questions. Recruiters are thus freed from repetitive administrative tasks and can focus on higher-value activities such as interviewing and relationship building.

### Purpose

The primary purpose of this project is to demonstrate how artificial intelligence can be integrated into real business workflows. By automatically extracting structured information from resumes and matching it to job requirements, the system reduces manual effort while improving data accuracy and consistency. AI-powered resume parsers use natural language processing (NLP) to quickly interpret data such as work history, skills, education and contact details. Such automation enables recruiters to process large volumes of applications with speed and reliability, offering faster candidate screening, improved matching and a better candidate experience.

### Project Overview

The AI HR Recruiter consists of a web-based front end and a Python back end. Users provide a job description and upload candidate CVs through a clean SaaS-style interface. The back end parses each CV, extracts the candidate's name, contact information and skills, then compares the CV content to the job description using semantic similarity techniques. Candidates receive a numerical score and a recommendation (e.g. *Consider* or *Reject*). For those who meet the threshold, the system automatically generates follow-up questions to gather more context about their experience. Recruiters can review the candidate profiles, scores and follow-up responses through an admin dashboard. Emails are sent automatically to candidates with a secure link for answering follow-up questions.

## Key Features

Feature	Description
<b>AI-based CV parsing</b>	Automatically extracts names, contact details and skills from uploaded CVs. The parser supports PDF and DOCX formats.
<b>Candidate ranking</b>	Compares CVs content to the job description and calculates a similarity score to prioritise applicants.
<b>Follow-up question generation</b>	Automatically generates personalised questions for candidates who meet the threshold, eliciting more context around their experience.
<b>Email workflow</b>	Sends follow-up questions and receives responses via secure email links, collecting answers for review.
<b>Admin dashboard</b>	Provides recruiters with a web-based dashboard to view candidate profiles, scores, recommendations and follow-up responses.
<b>Modern UI</b>	A clean, responsive front end built with React (Vite) delivers a smooth user experience.

## Technical Architecture

Layer	Technologies	Role
<b>Frontend</b>	React (Vite)	Implements a responsive SaaS-style UI for uploading résumés, entering job descriptions and reviewing results.
<b>Backend</b>	FastAPI, OpenAI API, SMTP	Handles résumé parsing, semantic matching, scoring logic, question generation and email sending.
<b>Database</b>	SQLite	Stores candidate records, scores, recommendations and follow-up responses.

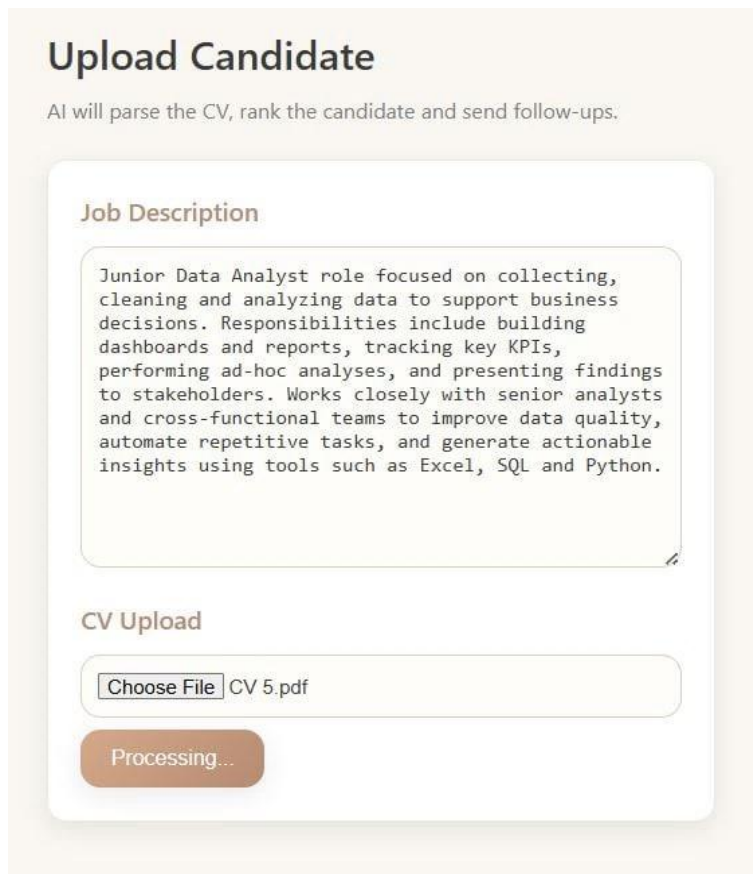
## NLP and Machine Learning

To match candidates with job descriptions, the system uses sentence embeddings generated by transformer models. This allows the algorithm to measure semantic similarity, going beyond simple keyword counts. The project uses tools such as PyMuPDF, pdfminer and python-docx for parsing PDFs and DOCX files, and leverages the OpenAI API to generate follow-up questions. According to a case study of a similar resume-screening system, a combination of React.js on the front end, FastAPI on the back end and a SQLite database enables a lightweight yet powerful full-stack application.

## Workflow

1. **Input job description:** The recruiter enters a job description describing skills and responsibilities for the role.
2. **Upload CVs:** Candidate CVs are uploaded in PDF or DOCX format.
3. **Parse CVs:** The back end extracts names, contact information and keywords from each document.
4. **Match and score:** A semantic similarity model compares each résumé to the job description, assigning a score between 0 and 100.
5. **Recommend:** The system assigns a recommendation (e.g. *Reject*, *Consider*) based on the score threshold.
6. **Generate questions:** For candidates marked as “Consider,” follow-up questions are created automatically.
7. **Email follow-up:** Candidates receive an email with a secure link to answer the follow-up questions. Their responses are stored in the database.
8. **Review:** Recruiters review the scored résumés and follow-up responses on the admin dashboard for final decision making.

Below are some representative screens from the application:



The screenshot shows a web interface titled "Upload Candidate". Below the title is a subtitle: "AI will parse the CV, rank the candidate and send follow-ups." The main content area is divided into two sections. The first section, "Job Description", contains a text box with the following text: "Junior Data Analyst role focused on collecting, cleaning and analyzing data to support business decisions. Responsibilities include building dashboards and reports, tracking key KPIs, performing ad-hoc analyses, and presenting findings to stakeholders. Works closely with senior analysts and cross-functional teams to improve data quality, automate repetitive tasks, and generate actionable insights using tools such as Excel, SQL and Python." The second section, "CV Upload", features a "Choose File" button next to the text "CV 5.pdf". Below this is a brown button labeled "Processing...".

*Upload Candidate page showing job description entry and CV upload.*

**AI Result**

**ID:** 8

**Score:** 85

**Recommendation:** Consider

**Status:** followup\_sent

The image shows a white card with rounded corners on a light beige background. The card contains the following text: 'AI Result' in bold, 'ID: 8', 'Score: 85', 'Recommendation: Consider', and 'Status: followup\_sent'. The 'followup\_sent' text is enclosed in a light orange pill-shaped box.

*AI Result page displaying candidate ID, score, recommendation and status.*

**Follow-up Questions**

Hi Renata Golemovic, please answer the questions below.

1. Can you describe any experience you have with automating repetitive data tasks, either using Python or other tools?

2. How comfortable are you with your current Python skills, and what specific areas are you focusing on to improve?

The image shows a white card with rounded corners on a light beige background. The card contains the following text: 'Follow-up Questions' in bold, 'Hi Renata Golemovic, please answer the questions below.', and two numbered questions. Each question is followed by a large, empty text input area with a light beige background and a small cursor icon at the bottom right.

*Follow-up Questions page requesting additional information from the candidate.*

## Conclusion

AI HR Recruiter demonstrates how artificial intelligence can be embedded into everyday business processes to streamline and standardize candidate screening.

The application combines a modern, SaaS-style frontend with a Python-based backend and large language models to perform structured CV parsing, semantic matching against job descriptions, candidate scoring, and automated follow-up communication. This approach helps improve consistency in early-stage screening while reducing manual workload, allowing recruiters to focus on higher-value, strategic activities.

Although presented as a demonstration project, the underlying principles – automated data extraction, semantic evaluation, decision support, and workflow automation – are directly applicable to real-world recruitment and HR systems.

Potential future enhancements include multi-language CV support, integration with external applicant tracking systems (ATS), richer analytics for recruiter decision-making, and more advanced recommendation and ranking logic based on historical hiring outcomes.