|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Software Design Specification**  TalentMatch - Applicant Tracking System  (ATS)  21F-ATS  Version: 1.0   |  |  | | --- | --- | | Project Code |  | | Supervisor | Sir Rizwan Ali Abro  \_\_\_\_\_\_\_\_\_\_ | | Co-Supervisor | NA | | Project Team | Vishaka Pahuja  Mohammad Yaqoob | | Submission Date | Feb-2025 | |

**Document History**

|  |  |  |
| --- | --- | --- |
| **Name of Person** | **Date** | **Description of change** |
| Vishaka Pahuja | 27/12/2024 | Document Created |
| Mohammad Yaqoob | 31/12/2024 | Added Non-functional requirements |
| Vishaka Pahuja | 4/1/2024 | Added Case Diagrams (HR & Candidate) |
| Mohammad Yaqoob | 09/01/2025 | Added ER-Diagram |
| Vishaka Pahuja | 15/01/2025 | Added Sequence Diagrams |
| Mohammad Yaqoob | 18/01/2025 | Added Architecture Design |
| Vishaka Pahuja | 24/01/2025 | Added Database Diagram |
| Mohammad Yaqoob | 29/01/2025 | Added Class Diagram |
| Vishaka Pahuja | 5/02/2025 | Added Interface Design |
| Mohammad Yaqoob | 15/02/2025 | Added Test Cases |

**TABLE OF CONTENTS**

[1. Introduction 1](#_Toc195742631)

[1.1 Purpose of Document 1](#_Toc195742632)

[1.2 Intended Audience 1](#_Toc195742633)

[2. Overall System Description 1](#_Toc195742634)

[2.1 Project Background 1](#_Toc195742635)

[2.2 Project Objectives 1](#_Toc195742636)

[3.Entity Relationship Diagram (ER-Diagram) 2](#_Toc195742637)

[4. USE CASE DIAGRAMS 2](#_Toc195742638)

[4.1 USE CASE DIAGRAM (For HR) 2](#_Toc195742639)

[4.2 USE CASE DIAGRAM (For Candidate) 4](#_Toc195742640)

[5. Sequence Diagrams 4](#_Toc195742641)

[5.1 User Login 5](#_Toc195742642)

[5.2 Job Posting 5](#_Toc195742643)

[5.3 Job Application 6](#_Toc195742644)

[5.4 Resume Parsing & AI Analysis 6](#_Toc195742645)

[5.5 Interview Scheduling 7](#_Toc195742646)

[6. Architecture Design 8](#_Toc195742647)

[7. Database Diagram 10](#_Toc195742648)

[8. Class Diagram 11](#_Toc195742649)

[9. Interface Design 12](#_Toc195742650)

[10. Test Cases 14](#_Toc195742651)

# 

# Introduction

## 1.1 Purpose of Document

The purpose of this document is to provide the Software Design Specifications (SDS) for the AI-based Applicant Tracking System (ATS). It outlines the overall architectural design of the system, detailing how various components interact with each other, the data flow, and the key functional modules within the system. This document serves as a guideline for developers, stakeholders, and quality assurance teams to ensure that the ATS is developed in alignment with the defined requirements and follows best practices in software design.

## 1.2 Intended Audience

This document has been intended for HR professionals, recruiters, developers and Q&A team. It is designed to make sure that the design of the system fits the needs of HR department, facilitate recruitment, enhance the power of decision-making, and deliver effective user experience.

# 2. Overall System Description

## 2.1 Project Background

This is a time-consuming process due to manual resume screening and interview scheduling which causes delays, bias and lacks efficiency. The increased volume of applications has put huge strains on organizations who are struggling to manage all large volumes of applications, thus increasing costs and, of course, missed opportunities.

These challenges are tackled by the AI-Modelled Applicant Tracking System (AI-ATS) which automates resume screening, candidate evaluation and ranking, using AI and ML. This speeds up the hiring process even more and delivers unbiased and more efficient hiring while reducing HR workload for better selection accuracy.

## 2.2 Project Objectives

The Applicant Tracking System powered by the AI can perform automated resume screening, automate candidate evaluation, and schedule interviews so that the recruitment process is faster and unbiased.

Key objectives include:

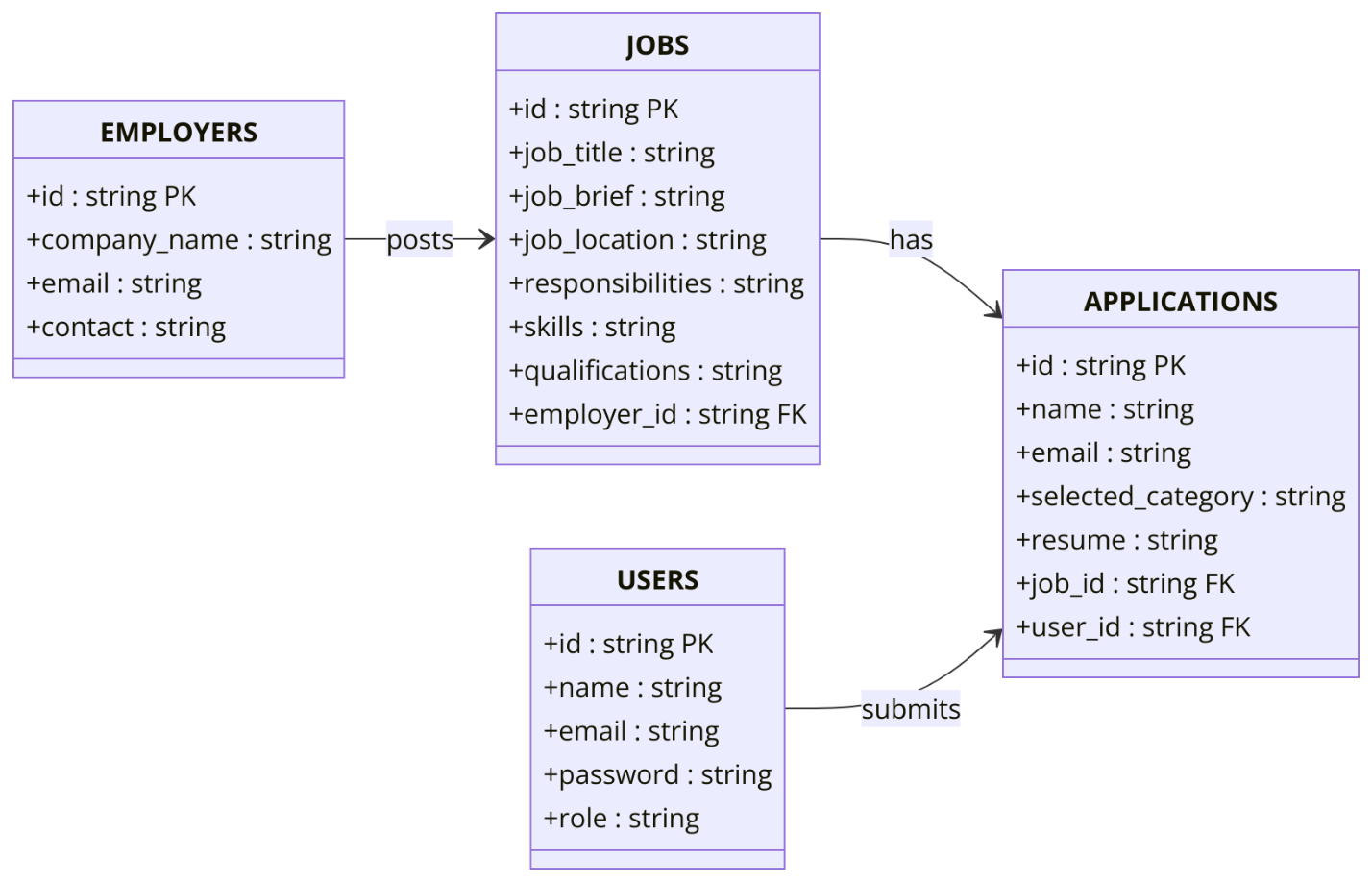
* **Automate Resume Screening**
* **Enhance Candidate Evaluation**
* **Streamline Interview Scheduling**
* **Reduce Hiring Bias**
* **Boost HR Productivity**

This system aims to reduce hiring time, improve candidate quality, and optimize recruitment efficiency.

# 3.Entity Relationship Diagram (ER-Diagram)

This **Entity-Relationship Diagram (ERD)** represents a job application system that connects HR with job seekers.

* **HR** creates job postings with details like title, location, skills, and qualifications.
* **Users** (job seekers) register, browse job listings, and submit applications.
* **Applications** store candidate details, resumes, and job references, linking each application to a specific job and user.



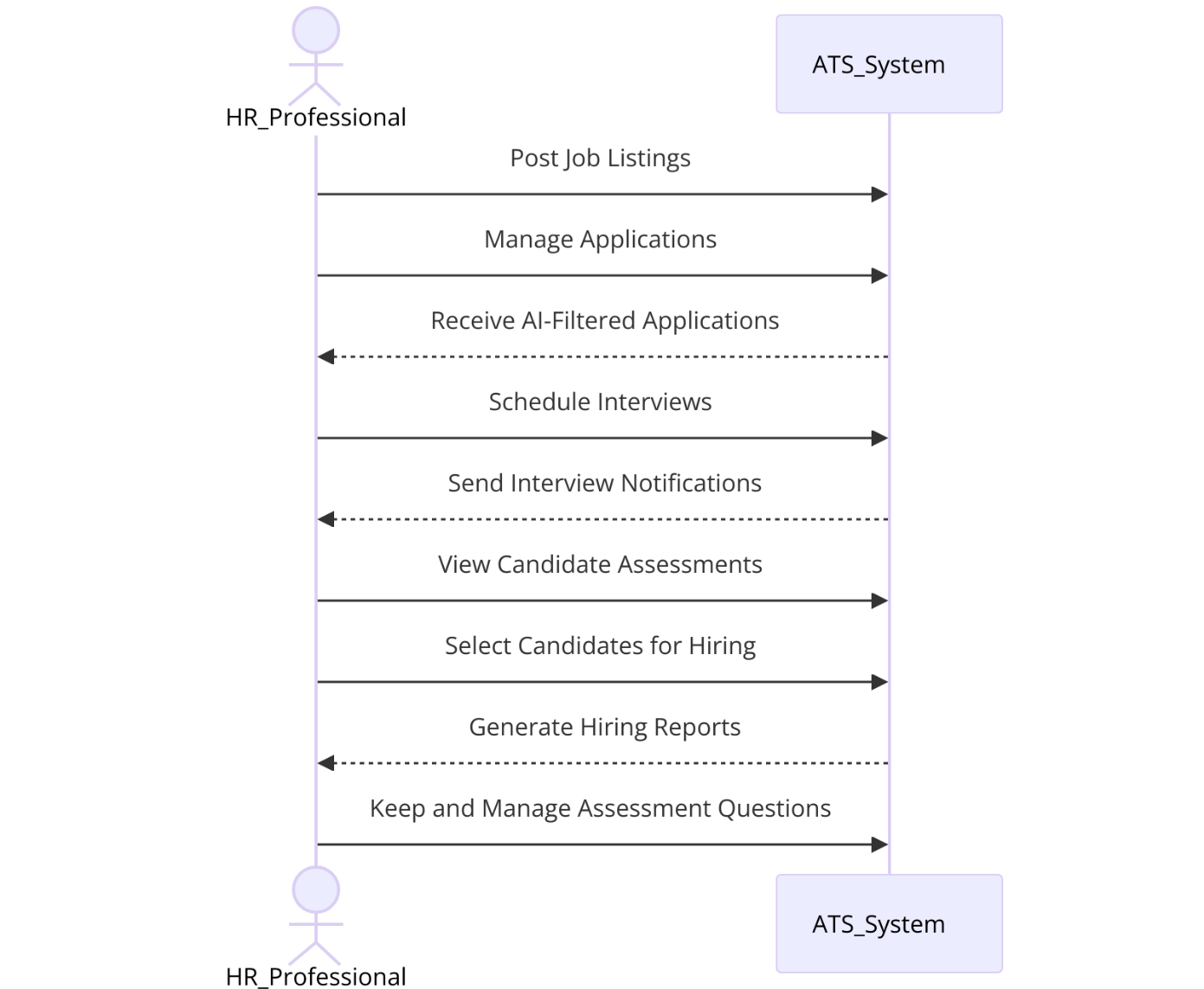
**Figure 1 : Entity Relationship Diagram of** **ATS**

# 4. USE CASE DIAGRAMS

A **Use Case Diagram** visually represents the interactions between users (actors) and a system, depicting different functionalities (use cases) the system provides.

## 4.1 USE CASE DIAGRAM (For HR)

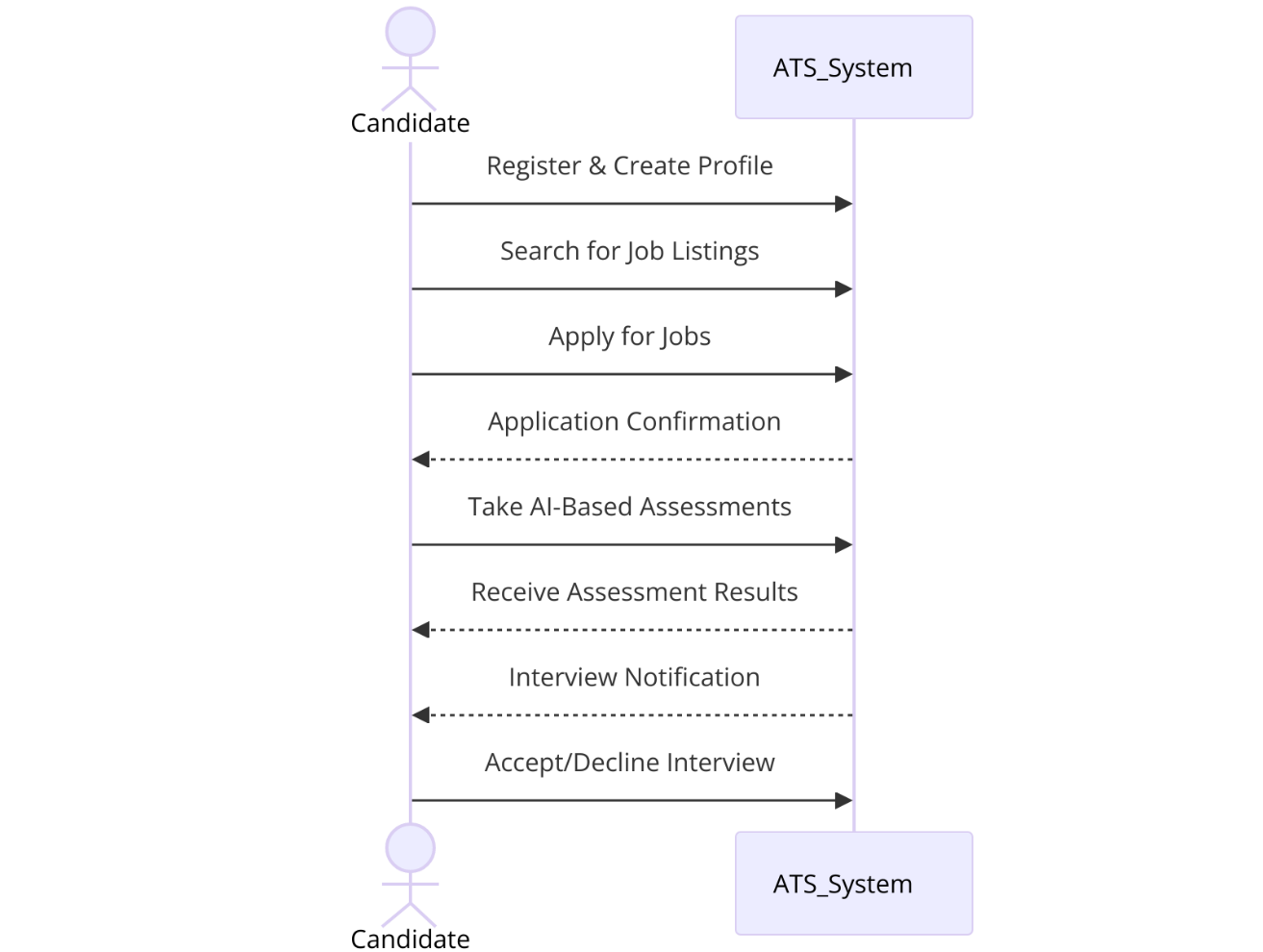
The HR use case diagram describes the interactions between the HR personnel and the AI based Applicant Tracking System (ATS). It breaks down important tasks, including checking user applications, shortlisting candidates, scheduling interviews, conducting test tests, and making the final hiring decisions.



**Figure 2 : Use Case Diagram of HR**

## 4.2 USE CASE DIAGRAM (For Candidate)

Use case diagram of the candidate involves the interactions of the candidate with the AI based Applicant Tracking system (ATS). The main actions included in it are applying for jobs, uploading resumes, taking assessments, etc., and receiving the application status informed to you. This diagram presents how the candidates interact with the system during the whole hiring system, clearly illustrating the steps of their experience from the application submission to getting feedback.



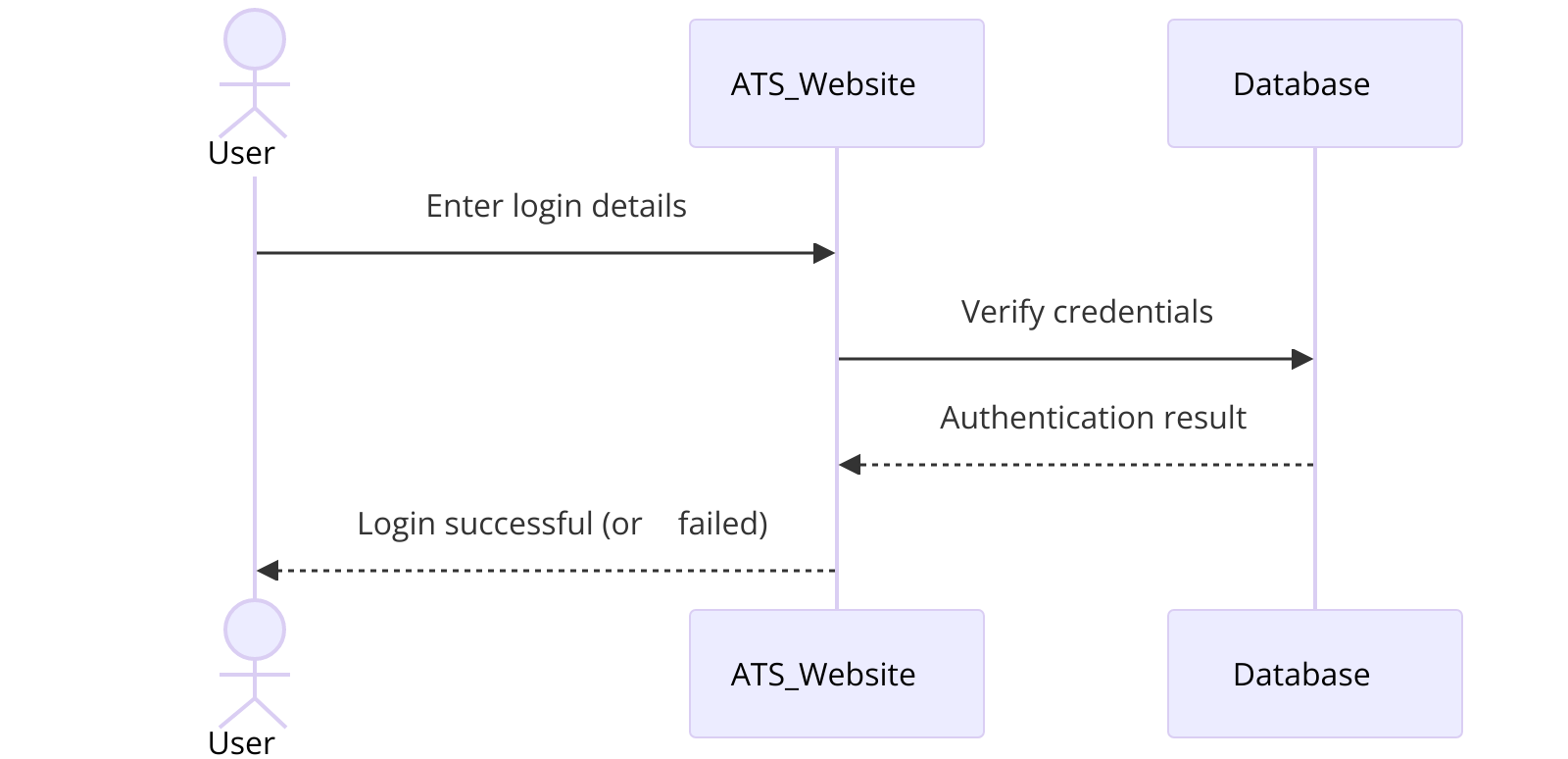
**Figure 3 : Use Case Diagram of Candidate**

# 5. Sequence Diagrams

The Sequence Diagrams are used to show the interactions of objects in a specific use case in sequential order. The Sequence Diagram for our ATS project describes how the different components would interact during the stage of candidate evaluation. There is User Login, Job Posting, Job Application, Resume Parsing & AI Analysis, and Interview Scheduling. These diagrams help visualize the step-by-step flow of data and actions.

## 5.1 User Login

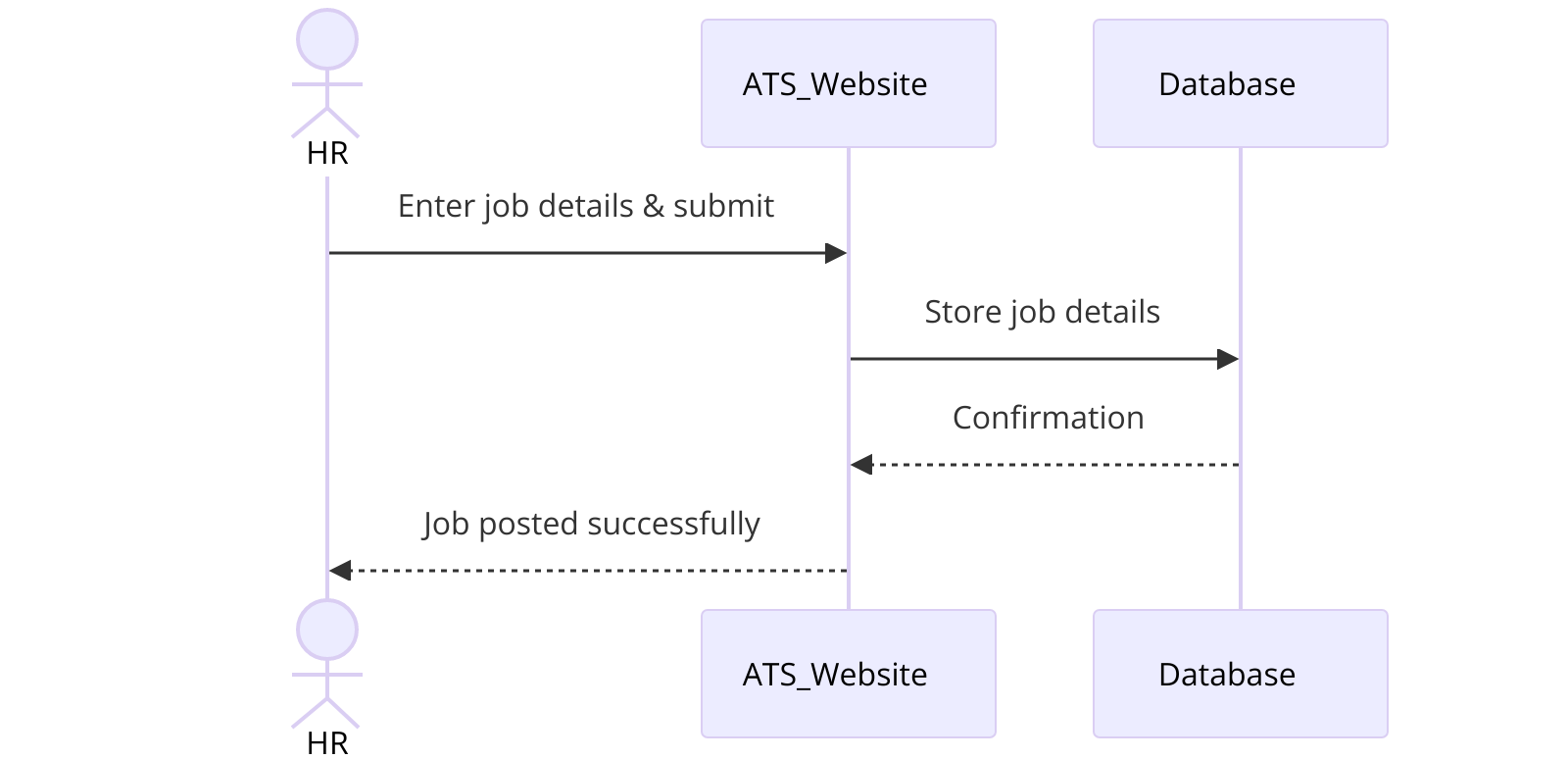
This diagram illustrates the user authentication process in the ATS. The user enters their credentials, which are verified against the database. The system then returns either a successful login or a failure message.



**Figure 4: User login**

## 5.2 Job Posting

This diagram shows how an HR manager posts a job on the ATS. The HR user enters job details, which are stored in the database. The system then confirms that the job posting is successful.



**Figure 5: Job Posting**

## 5.3 Job Application

This sequence diagram represents how an **applicant applies for a job**. The user selects a job and submits an application. The system stores the application data and confirms the submission.

A diagram of a website

Description automatically generated

**Figure 6: Job Application**

## 5.4 Resume Parsing & AI Analysis

After an applicant uploads a resume, the system processes it using a resume parser. The extracted skills are analyzed by an AI engine, which ranks the candidate and stores the results. The applicant receives a confirmation message.

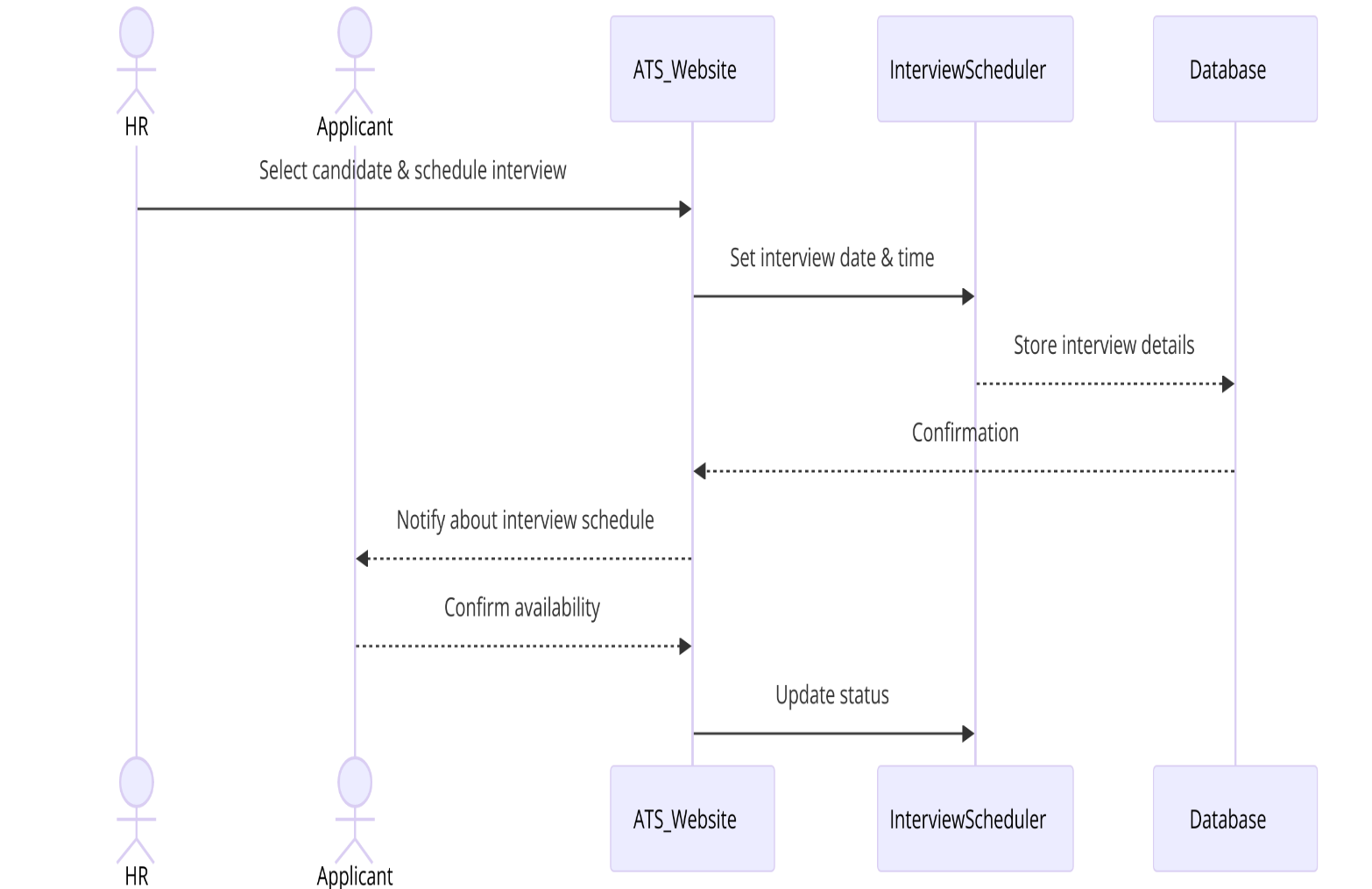
A diagram of a diagram

Description automatically generated

**Figure 7: Resume Parsing & AI Analysis**

## 5.5 Interview Scheduling

This diagram depicts how an HR manager schedules an interview. The system sets the interview date, updates the database, and notifies the applicant. The applicant can confirm availability, and the status is updated accordingly.



**Figure 8: Interview Scheduling**

# 6. Architecture Design

This Automated Recruitment System architecture enables seamless job matching and candidate assessment. The User Interface interacts with the Core Backend via an API Gateway with authentication. A Questions Database stores assessment tests, while Database Services handle user data, job applications, and AI-driven job matching using an NLP Model. Monitoring, Logging, and File Storage ensure system efficiency, and a Notification Service keeps users updated. The system is hosted on a scalable infrastructure for reliability and security.

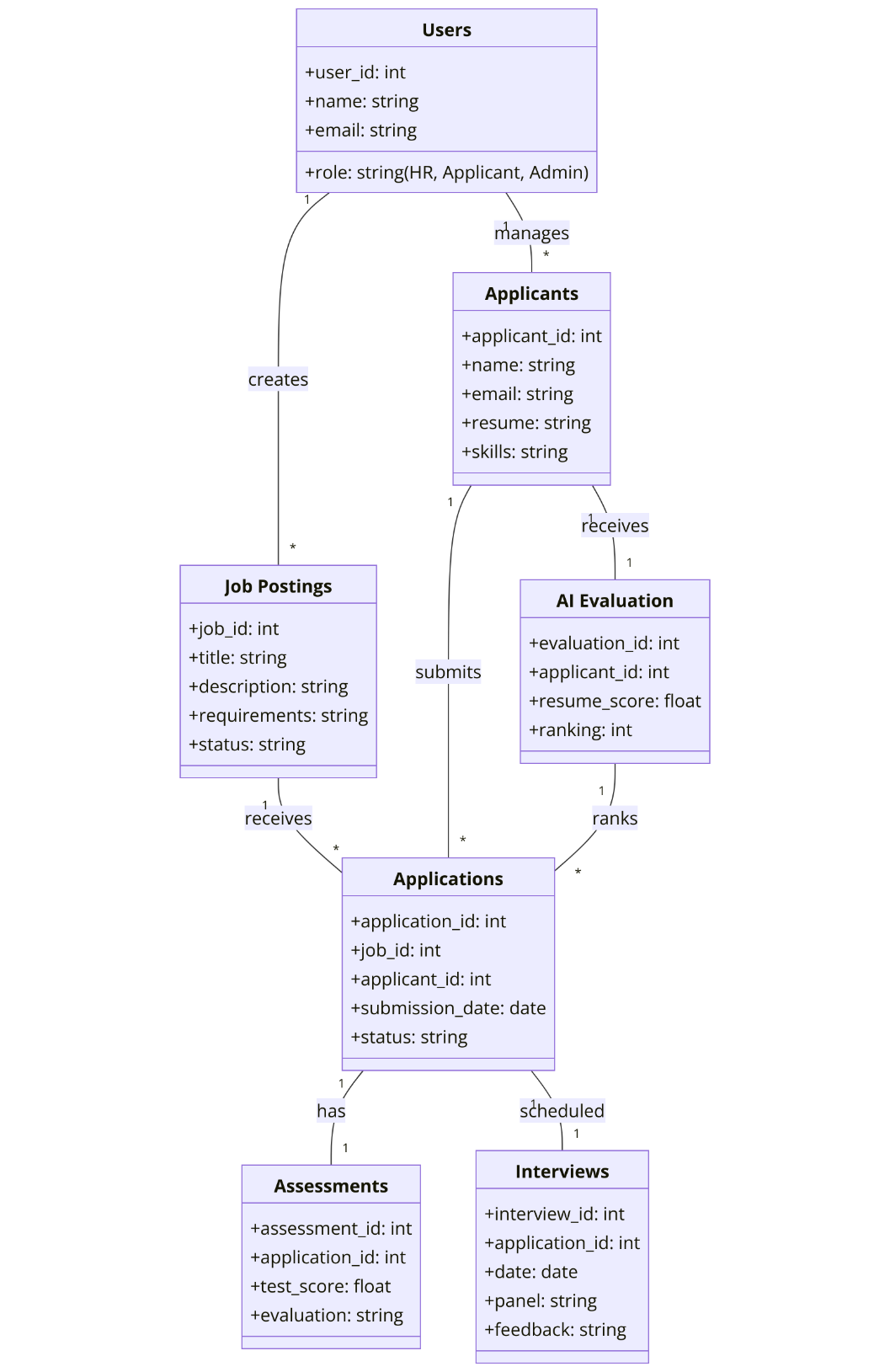
A diagram of a software system

Description automatically generated with medium confidence

**Figure 9: Architecture Design**

# 7. Database Diagram

The **Database Schema Diagram** for the **AI-Based ATS** outlines key entities and relationships. HR create Job Postings, while Applicants submit Applications. AI Evaluation ranks candidates, Assessments store test results, and Interviews manage scheduling.



**Figure 10: Database Diagram**

# 8. Class Diagram

This class diagram represents a job application system with four main entities: Employer, Job, User, and Application.

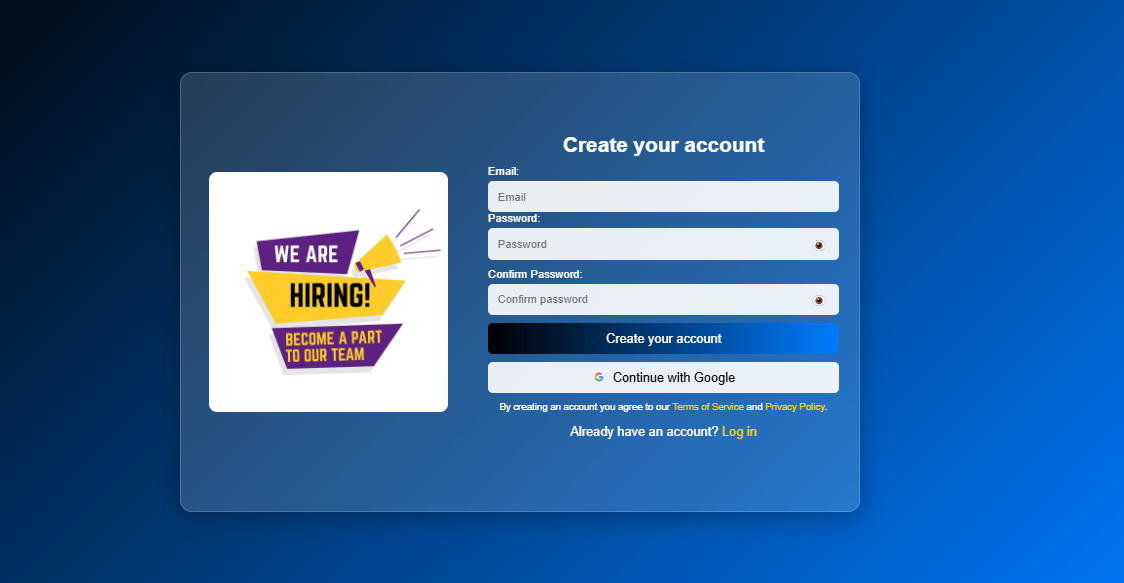
A diagram of a job application

Description automatically generated

**Figure 11: Class Diagram**

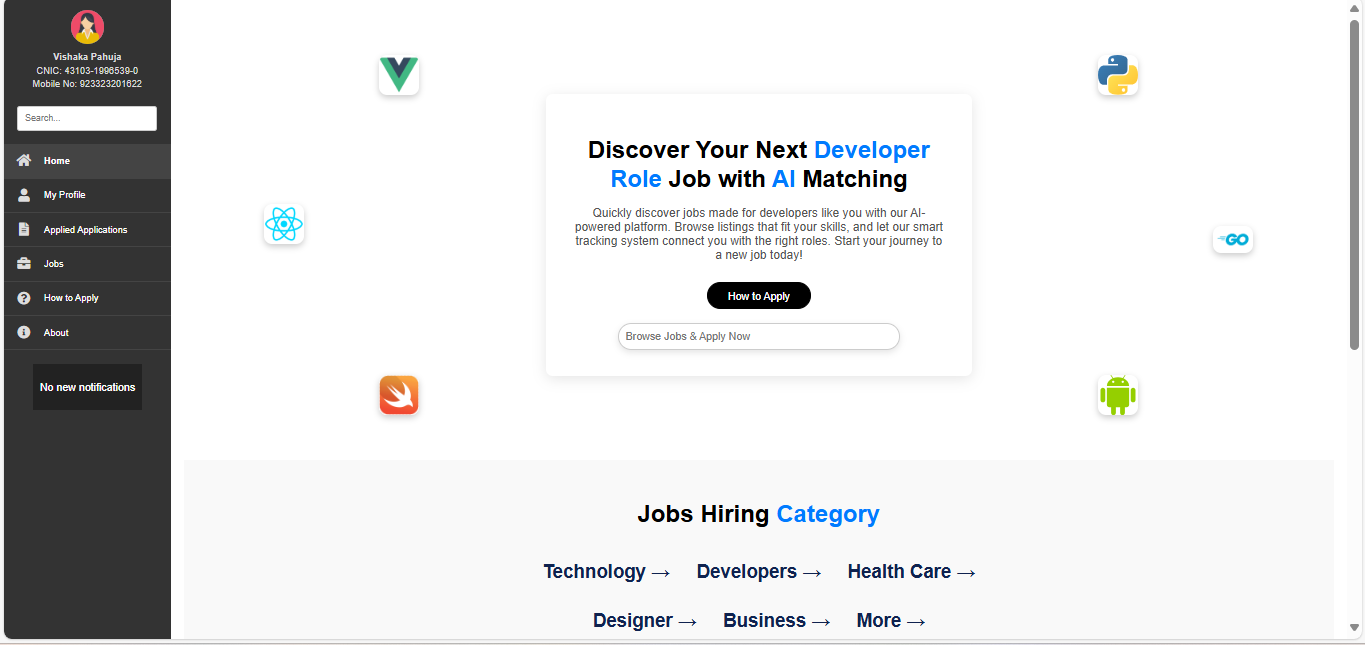
# 9. Interface Design

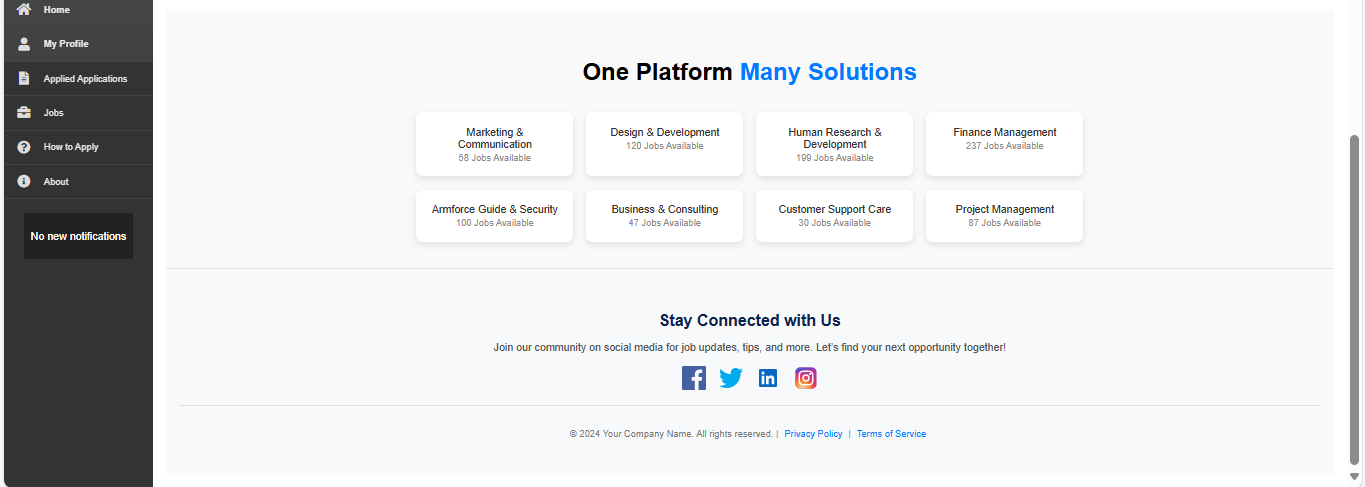
**Sign Up Page**



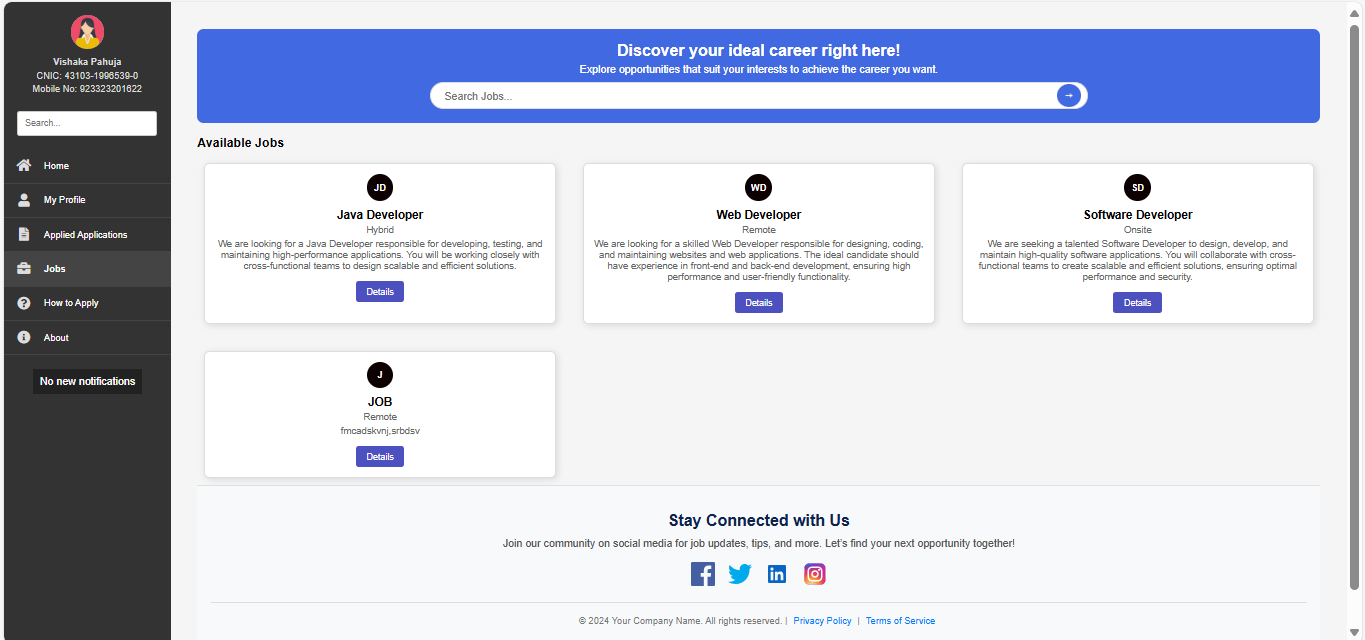
**FOR CANDIDATE**

**Home Page**

****

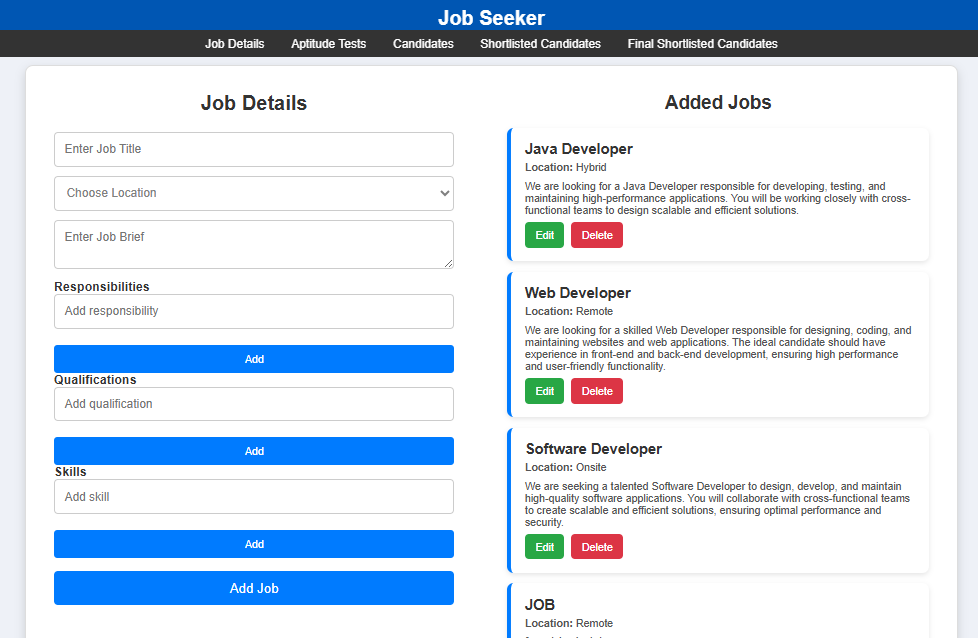


**Jobs Page**



**FOR HR**

**Job Details Page**



# 10. Test Cases

In this section of document, some of the test cases have been written.

|  |  |  |
| --- | --- | --- |
| **TC-01 Valid User Login** | | |
| **Test case Id** | | **TC-01** |
| **Test Scenario:** Verify user login with valid credentials. | | |
| **Pre-condition** | | User must be registered in the system. |
| **Steps #** | **Test Steps** | |
| **1.** | Go to the login screen | |
| **2.** | Enter valid email/username | |
| **3.** | Enter valid password | |
| **4.** | Click the Login button | |
| **Test data:** | | |
| * **Username:** hr\_manager@example.com * **Password:** Hr@1234 | | |
| **Expected outcome:** | | User is successfully logged into the system and redirected to the dashboard. |
| **Actual results:** | | TBD |
| **TC-02 Invalid User Login** | | |
| **Test case Id** | | **TC-02** |
| **Test Scenario: Verify user login with invalid credentials.** | | |
| **Pre-condition** | | User must be registered in the system. |
| **Steps #** | **Test Steps** | |
| **1.** | Go to the login screen | |
| **2.** | Enter invalid email/username | |
| **3.** | Enter invalid Password | |
| **4.** | Click the Login button | |
| **Test data:** | | |
| * **Username:** wrong\_user@example.com * **Password:** Wrong@123 | | |
| **Expected outcome:** | | User is not logged in.  The system should display an error message:  "Invalid credentials, please try again." |
| **Actual results:** | | TBD |
| **TC-03 JOB POSTING** | | |
| **Test case Id** | | **TC-03** |
| **Test Scenario:** Verify that an HR manager can successfully post a job. | | |
| **Pre-condition** | | HR must be logged into the system. |
| **Steps #** | **Test Steps** | |
| **1.** | Navigate to the Job Posting section | |
| **2.** | Enter job details (title, description, skills, qualifications) | |
| **3.** | Click the Post Job button | |
| **Test data:** | | |
| * **Job Title:** Software Engineer * **Skills:** Python, Django, SQL | | |
| **Expected outcome:** | | Job is successfully posted, and confirmation is displayed. |
| **Actual results:** | | TBD |
| **TC-04 Job Application Submission** | | |
| **Test case Id** | | **TC-04** |
| **Test Scenario:** Verify that a candidate can successfully apply for a job. | | |
| **Pre-condition** | | Candidate must be logged into the system. |
| **Steps #** | **Test Steps** | |
| **1.** | Navigate to the Job Listings section | |
| **2.** | Select a job | |
| **3.** | Upload resume | |
| **4.** | Submit application | |
| **Test data:** | | |
| * **Job Title:** Software Engineer * **Resume:** resume.pdf | | |
| **Expected outcome:** | | Application is submitted successfully, and a confirmation message is displayed |
| **Actual result:** | | TBD |
|  | |  |
| **TC-05 Resume Parsing & AI Analysis** | | |
| **Test case Id** | | **TC-05** |
| **Test Scenario:** Verify that the system correctly parses the resume and ranks the candidate. | | |
| **Pre-condition** | | Candidate must have uploaded a resume. |
| **Steps #** | **Test Steps** | |
| **1.** | Upload resume | |
| **2.** | System processes the document using AI-based parsing | |
| **3.** | Candidate profile is updated with extracted data | |
| **Test data:** | | |
| **Resume File:** resume.pdf | | |
| **Expected outcome:** | | System extracts relevant data and ranks the candidate accordingly. |
| **Actual results** | | TBD |
| **TC-06 Interview Scheduling** | | |
| **Test case Id** | | **TC-06** |
| **Test Scenario:** Verify that an HR manager can schedule an interview. | | |
| **Pre-condition** | | Candidate must have passed initial screening. |
| **Steps #** | **Test Steps** | |
| **1.** | HR navigates to the Interview Scheduling section | |
| **2.** | Selects a candidate and assigns interview date/time | |
| **3.** | Clicks the Schedule Interview button | |
| **Test data:** | | |
| * **Candidate Name:** Vishaka Pahuja * **Interview Date:** 10th Feb 2025 | | |
| **Expected outcome:** | | Interview is scheduled, and confirmation is sent to the candidate. |
| **Actual results** | | TBD |
|  | |  |
| **TC-07 Notification System** | | |
| **Test case Id** | | **TC-07** |
| **Test Scenario:** Verify that candidates receive notifications for job updates. | | |
| **Pre-condition** | | Candidate must have applied for a job. |
| **Steps #** | **Test Steps** | |
| **1.** | HR updates the job application status | |
| **2.** | System generates a notification | |
| **3.** | Candidate receives an email/app notification | |
| **Test data:** | | |
| * **Job Title:** Software Engineer * **Notification Type:** Application Status Update | | |
| **Expected outcome:** | | Candidate receives notification successfully. |
| **Actual results** | | TBD |
| **TC-08 Candidate Profile Update** | | |
| **Test case Id** | | **TC-08** |
| **Test Scenario:** Verify that a candidate can update their profile information**.** | | |
| **Pre-condition** | | Candidate must be logged into the system |
| **Steps #** | **Test Steps** | |
| **1.** | Navigate to Profile Settings | |
| **2.** | Edit profile details (name, email, resume, etc.) | |
| **3.** | Click the Save button | |
| **Test data:** | | |
| * **Name:** Vishaka Pahuja * **Email:** vishaka@gmail.com * **Resume:** updated\_resume.pdf | | |
| **Expected outcome:** | | Profile is updated successfully. |
| **Actual results** | | TBD |