Software Requirements Specification (SRS)

Project: HireWise

Version: 1.0

Date: August 25, 2025

# 1. Introduction

## 1.1 Purpose

To improve the efficiency and effectiveness of recruitment processes.

## 1.2 Scope

- Resume parsing, candidate matching, skill gap analysis

## 1.3 Intended Audience

- HR professionals, recruiters

# 2. System Overview

An AI-powered recruitment tool that streamlines the hiring process by automating resume screening, candidate matching, and skill gap identification.

# 3. Functional Requirements

## 3.1 User Account Creation [0]

**Description:**

The application should securely allow users to create accounts.

**Priority:**

High

**Subtasks:**

- Implement user registration form with input validation.

- Store user data securely (e.g., hashed passwords).

- Send confirmation email to newly created accounts.

- Allow users to reset their passwords.

- Integrate with email service for password reset functionality.

## 3.2 User Authentication [1]

**Description:**

The application should enable registered users to log in securely.

**Priority:**

High

**Subtasks:**

- Implement username/password authentication

- Integrate with a secure password hashing algorithm

- Provide password reset functionality

- Allow for multi-factor authentication (optional)

## 3.3 Resume Upload [2]

**Description:**

The application should allow users to upload their resumes in various formats.

**Priority:**

High

**Subtasks:**

- Implement a resume upload interface.

- Validate uploaded resume formats (docx, pdf, txt).

- Extract relevant information from uploaded resumes (name, skills, experience).

- Store uploaded resumes securely.

## 3.4 Resume Parsing [3]

**Description:**

The application should extract relevant information from uploaded resumes using natural language processing.

**Priority:**

High

**Subtasks:**

- Extract candidate name and contact information

- Identify work experience, including job titles, company names, and dates of employment

- Parse education history, including degrees, institutions, and graduation dates

- Identify relevant skills and keywords mentioned in the resume

- Store extracted information in a structured format for further processing

## 3.5 Job Posting Creation [4]

**Description:**

The application should allow users to create job postings with detailed descriptions and required skills.

**Priority:**

High

**Subtasks:**

- Define job title, company name, and location fields

- Implement a text editor for creating detailed job descriptions

- Enable users to select and add required skills from a searchable skills database

- Allow users to set salary range and other relevant job details

- Provide options for publishing the job posting to various job boards

## 3.6 Candidate Matching [5]

**Description:**

The application should use AI-powered algorithms to match candidate resumes with relevant job postings based on skills and experience.

**Priority:**

High

**Subtasks:**

- Develop a system to extract skills and experience from candidate resumes using Natural Language Processing (NLP) techniques.

- Create an AI-powered algorithm to analyze job postings and identify the required skills and experience.

- Implement a candidate matching engine that compares candidate profiles with job postings based on skill and experience similarity.

- Rank candidate matches based on the relevance and strength of the match.

- Provide a user-friendly interface to view and filter candidate matches.

## 3.7 Skill Gap Analysis [6]

**Description:**

The application should identify skill gaps between candidate profiles and job requirements.

**Priority:**

High

**Subtasks:**

- Compare candidate skills to job requirements extracted from job descriptions.

- Categorize skill gaps based on severity (e.g., missing essential skills, lacking advanced skills).

- Provide a visual representation of the skill gaps for easy understanding.

## 3.8 Personalized Recommendations [7]

**Description:**

The application should provide users with insights and recommendations for improving candidate profiles or job postings.

**Priority:**

High

**Subtasks:**

- Analyze candidate profile data to identify strengths, weaknesses, and potential areas for improvement.

- Suggest relevant skills, certifications, or experiences to enhance candidate profiles.

- Review job postings for clarity, keyword optimization, and alignment with target candidate profiles.

- Recommend revisions to job postings based on data-driven insights to attract a wider pool of qualified candidates.

## 3.9 Communication and Collaboration [8]

**Description:**

The application should allow users to communicate with potential candidates and manage the hiring process.

**Priority:**

High

**Subtasks:**

- Enable in-app messaging between recruiters and candidates

- Provide a centralized platform for scheduling interviews

- Allow for collaborative note-taking and feedback sharing among hiring team members

- Integrate with existing calendar systems for seamless scheduling

## 3.10 Data Security and Privacy [9]

**Description:**

The application should maintain the confidentiality and security of user data.

**Priority:**

High

**Subtasks:**

- Implement robust data encryption mechanisms for all sensitive user data.

- Establish role-based access control to restrict data access based on user permissions.

- Conduct regular security audits and penetration testing to identify vulnerabilities.

- Maintain a detailed log of all data access activities for auditing purposes.

## 3.11 Hiring Analytics [10]

**Description:**

The application should provide users with comprehensive reporting and analytics on the hiring process.

**Priority:**

High

**Subtasks:**

- Generate reports on time-to-hire metrics.

- Provide insights on candidate source effectiveness.

- Analyze hiring funnel drop-off points.

- Track key performance indicators (KPIs) such as cost-per-hire and candidate satisfaction.

- Enable custom report creation and data visualization.

# 4. Non-Functional Requirements

**NFR-1**: The application should support a high volume of concurrent users without significant performance degradation.

**NFR-2**: The application should be highly scalable to accommodate future growth in user base and data volume.

**NFR-3**: The application should have robust security measures to protect user data and prevent unauthorized access.

**NFR-4**: The application should provide clear and concise error messages to users in case of issues.

**NFR-5**: The application should be user-friendly and intuitive for both recruiters and candidates.

**NFR-6**: The application should be available with a high level of uptime and reliability.