Software Requirements Specification

for

JobNova

Smart Job Recruitment Platform

Version 1.0

Prepared by Kirinda GRTD

E2448006

BIT University of Moratuwa

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document describes the functional and nonfunctional requirements for the Smart Job Recruitment Platform, Version 1.0. The platform is a web-based recruitment system designed to streamline the hiring process for Small and Medium-sized Enterprises (SMEs), educational institutions, and job seekers in localized markets, particularly targeting the Sri Lankan context. This SRS is intended to serve as the primary reference for the development team, testers, and stakeholders throughout the software development lifecycle.

1.2 Document Conventions

This document follows IEEE standard formatting conventions. All requirements are assigned unique identifiers using the format REQ-[FEATURE]-[NUMBER] where FEATURE represents the system feature category and NUMBER is a sequential identifier. Priority levels are indicated as High (H), Medium (M), or Low (L). High priority requirements are critical for system operation, medium priority requirements are important but not critical, and Low priority requirements are desirable enhancements.

1.3 Intended Audience and Reading Suggestions

This document is intended for multiple audiences. Software developers should focus on Section 3 (System Features) and Section 4 (External Interface Requirements) for implementation details. Project managers should review Section 1 (Introduction), Section 2 (Overall Description), and Section 5 (Nonfunctional Requirements) for project planning and resource allocation. Testers should concentrate on Section 3 for functional testing requirements and Section 5 for performance and security testing criteria Stakeholders and end users may benefit from reading Section 1 and Section 2 to understand the system's purpose and capabilities. Documentation writers should review all sections to prepare user manuals and help documentation. It is recommended to begin with Section 1 and 2 for an overview, then proceed to sections relevant to your specific role.

1.4 Project Scope

The Smart Job Recruitment Platform aims to address critical inefficiencies in modern recruitment processes by providing an intelligent, automated, and user-friendly web application. The system serves two primary user groups: job seekers and recruiters. For job seekers, the platform enables creation of comprehensive digital profiles, automatic generation of professional resumes in PDF format, job recommendations based on skills and preferences, and streamlined on click application submission. For recruiters, the system provides tools for posting job openings, viewing applicants ranked by relevance to job requirements, communicating via emails and managing the entire recruitment workflow efficiently. The platform leverages user profiling to accelerate hiring timelines, improve candidate engagement, and enhance the overall recruitment experience. By focusing on SMEs, startups, and educational institutions in underrepresented markets, the system fills a gap left by large-scale global platforms that are often too complex or inaccessible for local hiring needs.

1.5 References

Project Proposal Document: "Smart Job Recruitment Platform," E2448006, University of Moratuwa, 2025

React.js Documentation: https://react.dev/

Node.js Documentation: https://nodejs.org/

Express.js Documentation: https://expressjs.com/

MongoDB Documentation: https://www.mongodb.com/docs/

JWT (JSON Web Tokens) Specification: RFC 7519

Glassdoor Recruiter Strategies: https://www.glassdoor.com/blog/9-ways-recruiters-use-glassdoor/

Candidate Experience Statistics 2025: https://www.jobtwine.com/blog/candidate-experience-statistics-for-hiring-success/

K. E. Wiegers, Software Requirements, Copyright © 2002. Permission granted for use, modification, and distribution of the document

Data-Driven Recruitment Best Practices: https://www.aihr.com/blog/data-driven-recruitment/

2. Overall Description

2.1 Product Perspective

The Smart Job Recruitment Platform is a standalone web-based system designed specifically for the needs of local recruitment markets in developing regions like Sri Lanka. It targets SMEs, educational institutions, startups, and entry-level job seekers, offering a modular and scalable architecture that supports future integrations and feature expansions.

- Purpose-built for SMEs, educational institutions, startups, and fresh job seekers.
- Operates as a standalone system with a modular architecture for future integrations.
- Integrates easily with external systems such as educational databases and professional networks.
- Uses React.js for the frontend, ensuring a responsive UI.
- Uses Node.js/Express.js for backend logic and API handling.
- Stores all persistent data in MongoDB.
- Supports user interaction through standard web browsers.
- Sends notifications via emails.
- Generates PDF resumes through server-side rendering.

2.2 Product Features

The Smart Job Recruitment Platform provides a complete set of features designed to streamline the hiring process for both job seekers and recruiters. It includes secure authentication, comprehensive profile management, automated resume generation, job posting tools, an efficient application workflow, real-time communication via emails and notification systems. Each feature is built to improve usability, reduce manual effort and enhance the overall recruitment experience.

- Secure Authentication & Authorization
    - Two user roles: job seekers and recruiters
    - JWT-based session management

- Profile Management
    - Job seekers can create detailed digital profiles
    - Includes personal info, education, experience, skills and preferences

- Resume Generation
    - Creates professional PDF resumes from profile data
    - Removes repetitive manual data entry

- Job Posting & Management
    - Recruiters can create detailed job listings
    - Includes descriptions, requirements and deadlines

- Application Management
    - One-click job applications for seekers
    - Organized applicant tracking for recruiters

- Integrated Communication via Email
    - Email between recruiters and candidates
    - Helps with interview coordination and offers

- Role-Specific Dashboards
    - Job seekers: applied jobs, profile management
    - Recruiters: posted jobs, applicant counts and status updates

- Notification System
    - Real-time in-platform alerts
    - Updates on application status, messages and new job posts

2.3 User Classes and Characteristics

The system serves two main user groups, Job Seekers and Recruiters. each with different needs and behavior patterns. Job seekers range from recent graduates to early-career professionals and require simple, intuitive tools for creating profiles, generating resumes, and applying for jobs. Recruiters, including HR personnel and business owners, need efficient tools to post jobs, evaluate candidates, and manage communication. Both groups expect strong security, reliability and privacy with the platform primarily optimized to enhance the job seeker experience to attract more qualified applicants.

User Classes and Key Characteristics

1. Job Seekers

- Includes recent graduates, early-career professionals, and individuals seeking local opportunities.
- Technical expertise: basic to intermediate.
- High usage frequency during active job searches.

Needs:

- Simple profile creation
- Easy resume generation
- Job search and filtering tools
- Clear and quick application process
- Motivated to present themselves effectively.
- Value features that reduce repetitive tasks and increase visibility to recruiters.

2. Recruiters

- Includes HR professionals, hiring managers, and SME/startup/business owners.
- Technical expertise: intermediate to advanced.
- Higher usage during active hiring cycles.

Needs:

- Fast job posting tools
- Ability to review many applications efficiently
- Streamlined communication with multiple applicants
- Prioritize reducing time-to-hire and finding best-fit candidates.

3. Shared Expectations

- Strong security, privacy, and system reliability are essential.
- Platform design gives priority to job seeker experience as their participation drives recruiter engagement.

2.4 Operating Environment

The Smart Job Recruitment Platform operates in a modern web environment accessible through standard web browsers. On the client side, the system requires Chrome version 90 or higher, Firefox version 88 or higher, Safari version 14 or higher, or Edge version 90 or higher. The client-side application is responsive and adapts to desktop computers, laptops, tablets, and smartphones with screen resolutions ranging from 320px mobile devices to 4K displays. Users require a stable internet connection with minimum recommended bandwidth of 2 Mbps for optimal performance. On the server side, the application runs on a Node.js runtime environment version 18 or higher hosted on a Linux-based server (Ubuntu 20.04 LTS or higher recommended) or compatible cloud platform services such as AWS, Google Cloud Platform, or Microsoft Azure. The MongoDB database operates as a separate service, either on the same server infrastructure or as a managed database service (MongoDB Atlas). The system peacefully coexists with standard web infrastructure components including reverse proxy servers (Nginx or Apache), SSL/TLS certificate management systems and monitoring tools for system health tracking. The platform is designed to scale horizontally across multiple server instances behind a load balancer to handle increased traffic.

2.5 Design and Implementation Constraints

The Smart Job Recruitment Platform is built under several technical, regulatory and procedural constraints that ensure security, maintainability and future scalability. The system must adhere to the approved technology stack, follow strict data protection standards, maintain mobile-first usability and align with best practices such as RESTful API design and Agile development. These constraints guide how version 1.0 is developed while keeping room for future advanced features.

Key Constraints

Technology Stack

- Must use React.js (frontend), Node.js, Express.js (backend) and MongoDB (database).

Authentication & Security

- JWT required for stateless, secure session management.
- Passwords must be hashed using bcrypt (minimum 10 salt rounds).

Regulatory Compliance

- Must comply with Sri Lankan data protection regulations.
- Should follow GDPR principles for future international expansion.

Resume/PDF Constraints

- PDF resumes must follow standard formats readable by common viewers and ATS.

Design Constraints

- All features must work on smartphones, tablets and desktops.
- API must follow RESTful principles for maintainability and future integrations.

Functional Restrictions

- Communication module must NOT store payment information.
- No premium or financial transactions in version 1.0.

Development Process

- Must follow Agile methodology with iterative releases.

Future Scalability Constraints

- Should be designed to support future additions such as:
    - AI-powered matching
    - Resume parsing

2.6 User Documentation

The Smart Job Recruitment Platform will provide essential user documentation focused on two key areas: Frequently Asked Questions (FAQ) and a Privacy Policy Agreement. The FAQ section will guide users through common issues, troubleshooting steps, platform features, and best practices for both job seekers and recruiters. A clear and accessible Privacy Policy will be presented to users, requiring them to review and agree before proceeding with major actions on the platform, such as account creation, job posting, or job application. This policy agreement may appear as pop-up confirmations or consent checkboxes during important workflow steps. All documentation will be written in simple, clear English and updated regularly to remain accurate and relevant.

2.7 Assumptions and Dependencies

The Smart Job Recruitment Platform is based on several assumptions regarding user behavior, technology stability and legal compliance along with dependencies on external systems, services and resources. These factors influence development planning and system performance and any major changes may require adjustments in scope or design.

Key Assumptions

- Users have access to internet-connected devices with modern web browsers.
- Users (job seekers and recruiters) possess basic digital literacy.
- Reliable hosting infrastructure will be available throughout development and deployment.
- Job seekers provide accurate, complete profile data.
- Recruiters create clear and detailed job descriptions.
- Core technology stack (React.js, Node.js, MongoDB) will remain stable during the project.
- Development will follow the approved timeline.
- Users will provide consent for data processing under applicable privacy regulations.
- Recruitment-related legal requirements in Sri Lanka will remain stable.
- Platform adoption will grow through partnerships with educational institutions and local businesses.

Key Dependencies

- Availability and reliability of hosting infrastructure (cloud).
- Compatibility of server-side PDF generation libraries (PDFKit, Puppeteer) with Node.js.
- Quality and completeness of input data for matching and ranking algorithms.
- Third-party services.
- Stable external environment for user acquisition and institutional partnerships.

3. System Features

3.1 User Authentication and Authorization

3.1.1 Description and Priority

The user authentication and authorization system provide secure account creation, login, and role-based access control for both job seekers and recruiters. This is a High Priority feature as it forms the foundation for all other system features and ensures data security and privacy. The system must distinguish between user types and grant appropriate permissions for accessing role-specific features.

3.1.2 Stimulus/Response Sequences

Registration Flow:

- User navigates to registration page and selects account type (Job Seeker or Recruiter)
- User enters required information (email, password, full name, account type)
- System validates input format and checks for existing accounts
- System creates account with encrypted password

Login Flow:

- User enters email and password on login page
- System validates credentials against database
- System generates JWT token and returns it to client
- User is redirected to role-appropriate dashboard
- System maintains session until logout or token expiration

3.1.3 Functional Requirements

REQ-AUTH-1: The system shall provide separate registration forms for Job Seekers and Recruiters with fields appropriate to each user type.

REQ-AUTH-2: The system shall validate email addresses using standard email format validation and check for uniqueness in the database before account creation.

REQ-AUTH-3: The system shall enforce password strength requirements including minimum 8 characters, at least one uppercase letter.

REQ-AUTH-4: The system shall hash all passwords using bcrypt algorithm with a minimum of 10 salt rounds before storing in the database.

REQ-AUTH-5: The system shall generate JWT tokens upon successful authentication containing user ID, user type, and expiration time set to 24 hours.

REQ-AUTH-6: The system shall validate JWT tokens on every API request to protected endpoints and return 401 Unauthorized statuses if token is invalid or expired.

REQ-AUTH-7: The system shall implement role-based access control preventing job seekers from accessing recruiter-only features and vice versa.

REQ-AUTH-8: The system shall provide a logout function that invalidates the current JWT token and redirects to the login page.

REQ-AUTH-9: The system shall display appropriate error messages for invalid credentials without revealing whether the email exists in the system.

3.2 Job Seeker Profile Management

3.2.1 Description and Priority

The profile management system allows job seekers to create, edit, and maintain comprehensive digital profiles containing all relevant information for job applications and resume generation. This is a High Priority feature as it serves as the foundation for resume generation and job matching functionalities.

3.2.2 Stimulus/Response Sequences

Profile Creation Flow:

- New job seeker logs in and is prompted to complete profile
- User enters personal information section (name, contact details, location)
- User adds education entries with institution, degree, field, and dates
- User adds work experience entries with company, position, responsibilities, and dates
- User adds skills with proficiency levels
- User saves profile and system validates all required fields
- System stores profile data and confirms successful save

Profile Editing Flow:

- Job seeker navigates to profile page
- System displays current profile information in editable form
- User modifies desired fields
- User saves changes and system updates database
- System confirms successful update and refreshes display

3.2.3 Functional Requirements

REQ-PROFILE-1: The system shall provide a profile form for job seekers including sections for Personal Information, Education, Work Experience, Skills, and Career Preferences.

REQ-PROFILE-2: The Personal Information section shall include fields for full name, email (pre-filled from registration), phone number, location (city/country), professional summary, and optional profile photo upload.

REQ-PROFILE-3: The Education section shall allow users to add multiple education entries, each containing institution name, degree type, field of study, start date, end date (or "Present"), and GPA (optional).

REQ-PROFILE-4: The Work Experience section shall allow users to add multiple experience entries, each containing company name, job title, employment type (full-time, part-time, internship, contract), start date, end date (or "Present"), and description of responsibilities.

REQ-PROFILE-5: The Skills section shall allow users to add multiple skills with associated proficiency levels (Beginner, Intermediate, Advanced, Expert).

REQ-PROFILE-6: The Career Preferences section shall include fields for desired job titles, preferred industries, preferred locations, job type preferences (remote, on-site, hybrid), and salary expectations (optional).

REQ-PROFILE-7: The system shall validate all date fields ensuring start dates are before end dates and that dates are not in the future (except for end dates marked as "Present").

REQ-PROFILE-8: The system shall validate phone numbers according to standard international formats.

REQ-PROFILE-9: The system shall support profile photo uploads in JPEG or PNG format with maximum file size of 5MB and automatically resize images to 200x200 pixels.

REQ-PROFILE-10: The system shall mark certain fields as required (name, email, phone, at least one education entry, at least one skill) and prevent saving incomplete profiles.

3.3 Job Posting and Management

3.3.1 Description and Priority

The job posting and management system enables recruiters to create, publish, edit, and manage job listings with comprehensive details about positions and requirements. This is a High Priority feature as it provides the core functionality for recruiters and generates content for job seekers to browse and apply.

3.3.2 Stimulus/Response Sequences

Job Creation Flow:

- Recruiter navigates to "Post Job" page
- Recruiter enters job details (title, company, location, type, description)
- Recruiter specifies requirements (education, experience, skills)
- Recruiter sets application deadline and other parameters
- Recruiter previews job posting
- Recruiter submits job posting
- System validates input and publishes job
- System confirms posting and displays job ID

Job Management Flow:

- Recruiter navigates to "My Jobs" dashboard
- System displays list of all posted jobs with status and applicant counts
- Recruiter selects a job to view details

- System displays job details and list of applicants
- Recruiter can edit job details, close job posting, or delete job

3.3.3 Functional Requirements

REQ-JOB-1: The system shall provide a job posting form for recruiters including fields for job title, company name, location, job type (full-time, part-time, contract, internship), salary range (optional), and application deadline.

REQ-JOB-2: The job posting form shall include a rich text editor for the job description section allowing formatting such as bold, italic, bullet points, and paragraphs.

REQ-JOB-3: The system shall allow recruiters to specify required qualifications including minimum education level, years of experience required, and specific skills needed.

REQ-JOB-4: The system shall allow recruiters to add preferred (but not required) qualifications separately from required qualifications.

REQ-JOB-5: The system shall assign a unique job ID to each posting for tracking and reference purposes.

REQ-JOB-6: The system shall allow recruiters to save job postings as drafts without publishing them publicly.

REQ-JOB-7: The system shall publish job postings immediately upon submission unless saved as draft, making them visible to all job seekers.

REQ-JOB-8: The system shall provide a job management dashboard for recruiters displaying all their posted jobs with status indicators (Active, Closed, Draft), number of applicants, and posting date.

REQ-JOB-9: The system shall allow recruiters to edit job postings after publication, with changes taking effect immediately.

REQ-JOB-10: The system shall allow recruiters to close job postings, which hides them from job seeker searches but retains all applicant data.

REQ-JOB-11: The system shall allow recruiters to delete job postings, which permanently removes the posting and all associated application data after confirmation.

REQ-JOB-12: The system shall automatically close job postings when the application deadline is reached.

REQ-JOB-13: The system shall validate that application deadlines are set to future dates when creating or editing job postings.

REQ-JOB-14: The system shall display job postings with timestamps showing when they were posted (e.g., "Posted 2 days ago").

3.4 Application Submission System

3.4.1 Description and Priority

The application submission system enables job seekers to apply for jobs with a streamlined one-click process and allows recruiters to manage received applications efficiently. This is a High Priority feature as it facilitates the core transaction of the recruitment process.

3.4.2 Stimulus/Response Sequences

Application Submission Flow:

- Job seeker browses jobs and selects one to view details
- System displays full job description and requirements
- Job seeker clicks "Apply Now" button
- System verifies profile completeness
- System submits application using profile data
- System sends confirmation notification
- Job seeker sees application status updated to "Applied"

Application Viewing Flow (Recruiter):

- Recruiter selects a job posting from their dashboard
- System displays list of applicants with relevance scores
- Recruiter clicks on an applicant to view full details
- System displays applicant's profile, resume, and match breakdown
- Recruiter can mark applicant as "Shortlisted," "Rejected," or "Under Review"

3.4.3 Functional Requirements

REQ-APP-1: The system shall allow job seekers to apply to any active (non-closed) job posting with a single click without requiring additional form filling.

REQ-APP-2: The system shall automatically attach the job seeker's current profile data and generated resume to each application.

REQ-APP-3: The system shall prevent duplicate applications by checking if a job seeker has already applied to a specific job posting.

REQ-APP-4: The system shall record the application timestamp and assign a unique application ID to each submission.

REQ-APP-5: The system shall display application status to job seekers with possible values: "Submitted," "Under Review," "Shortlisted," "Rejected," or "Accepted."

REQ-APP-6: The system shall provide recruiters with an applicant management interface for each job posting showing all applications.

REQ-APP-7: The system shall allow recruiters to change application status for individual applicants (Under Review, Shortlisted, Rejected, Accepted).

REQ-APP-8: The system shall display a summary count of applications by status (Total, Under Review, Shortlisted, Rejected, Accepted) on the recruiter dashboard.

REQ-APP-9: The system shall allow recruiters to view full applicant profiles including all sections (education, experience, skills) directly from the application list.

REQ-APP-10: The system shall allow recruiters to download applicant resumes as PDF files from the application interface.

REQ-APP-11: The system shall maintain a complete application history for both job seekers and recruiters even after job postings are closed.

REQ-APP-12: The system shall display a "My Applications" page for job seekers showing all their applications with job title, company, application date, and current status.

REQ-APP-13: The system shall allow recruiters to add private notes to applications that are visible only to the recruiter.

3.5 Application Matching and Ranking

3.5.1 Description and Priority

The intelligent matching and ranking system analyze job requirements against candidate profiles to rank applicants by relevance for recruiters and recommend suitable jobs to job seekers. This is a Medium Priority feature as it directly addresses the core problem of inefficient matching in traditional recruitment systems.

3.5.2 Stimulus/Response Sequences

Job Seeker Job Recommendation Flow:

- Job seeker logs into platform
- System retrieves job seeker profile including skills, experience, and preferences
- System queries active job postings
- System calculates match scores between profile and each job
- System displays recommended jobs sorted by match score (highest first)
- Job seeker can view match score
- System confirms posting and displays job

Recruiter Applicant Ranking Flow:

- Recruiter views applicants for a specific job posting
- System retrieves all applications for that job
- System retrieves full profiles for all applicants
- System calculates relevance scores based on job requirements
- System displays applicants ranked by relevance score (highest first)

3.5.3 Functional Requirements

REQ-MATCH-1: System shall compare job requirements (skills, education, experience) against candidate profiles.

REQ-MATCH-2: System shall calculate a simple match score based on the number of requirements met.

REQ-MATCH-3: System shall display match score as a percentage of requirements met.

REQ-MATCH-4: System shall show which requirements are met or not met.

REQ-MATCH-5: System shall sort recommended jobs for job seekers and applicants for recruiters based on match score.

REQ-MATCH-6: System shall highlight highly matching jobs or applicants (e.g., above 70% of requirements met).

REQ-MATCH-7: System shall allow job seekers to see the top recommended jobs.

REQ-MATCH-8: System shall update recommendations when the profile or job requirements are updated.

3.6 Communication Module

3.6.1 Description and Priority

The communication module enables direct email-based communication between recruiters and job seekers. When a recruiter contacts a candidate, the platform sends an email on behalf of the system and replies are handled through standard email clients. This is a Medium Priority feature that supports seamless communication without requiring an internal chat system.

3.6.2 Stimulus/Response Sequences

Message Initiation Flow (Recruiter to Candidate):

- Recruiter views the applicant's profile or application details.
- Recruiter clicks the "Send Email" button.
- System opens an email composition interface.
- Recruiter writes the message and clicks Send.
- System sends an email to the job seeker with recruiter details and job title.
- Job seeker receives the email through their email inbox.

Message Reply Flow:

- User receives an email notification from the system.
- User opens the email in their personal email account.
- User clicks Reply in their email client.
- Email is delivered to the opposite party's registered email address.

3.6.3 Functional Requirements

REQ-EMAIL-1: The system shall allow recruiters to send email messages to any candidate who has applied to their job postings.

REQ-EMAIL-2: The system shall allow job seekers to reply to recruiter emails through their personal email clients.

REQ-EMAIL-3: The system shall record only meta-information but shall NOT store the email content.

REQ-EMAIL-4: The system shall include the corresponding job title in the email subject line for context.

3.7 Notification Module

3.7.1 Description and Priority

The notification system keeps users informed about important events and updates through in-platform alerts. This is a Medium Priority feature that improves user engagement and ensures timely responses to recruitment activities.

3.7.2 Stimulus/Response Sequences

Application Status Change Notification:

- Recruiter changes application status to "Shortlisted"
- System detects status change event
- System creates in-platform notification for job seeker
- Job seeker logs in and sees notification badge
- Job seeker clicks notification and is directed to application details

New Job Recommendation Notification:

- New job is posted matching job seeker's profile
- System runs matching algorithm
- System identifies high-match job seekers
- System creates notification for matched job seekers

New Application Notification:

- Job seeker submits application

- System creates notification for recruiter
- Recruiter sees notification badge and application count update

3.7.3 Functional Requirements

REQ-NOTIF-1: The system shall provide a notification center accessible from the main navigation displaying all notifications in reverse chronological order.

REQ-NOTIF-2: The system shall display a notification count badge on the notification icon showing the number of unread notifications.

REQ-NOTIF-3: The system shall generate in-platform notifications for job seekers when their application status changes.

REQ-NOTIF-4: The system shall generate in-platform notifications for job seekers when they receive a new message from a recruiter.

REQ-NOTIF-5: The system shall generate in-platform notifications for recruiters when a new application is received for their job postings.

REQ-NOTIF-6: The system shall generate in-platform notifications for recruiters when they receive a message reply from a job seeker.

REQ-NOTIF-7: The system shall mark notifications as read when clicked by the user.

REQ-NOTIF-8: The system shall provide a "Mark all as read" function in the notification center.

REQ-NOTIF-9: The system shall make notifications clickable, directing users to the relevant page (application details, message thread).

REQ-NOTIF-10: The system shall include notification content describing the event

3.8 Resume Generation

3.8.1 Description and Priority

The resume generation feature creates professional, formatted PDF resumes directly from job seeker profile data, eliminating manual resume creation and ensuring consistency. This is a Low Priority feature as it is a core value proposition differentiating the platform from traditional job boards.

3.8.2 Stimulus/Response Sequences

Resume Generation Flow:

- Job seeker navigates to resume generation page or clicks "Generate Resume" button
- System retrieves complete profile data from database
- System applies selected resume template formatting
- System generates PDF document with formatted content
- System presents preview of generated resume
- User reviews and clicks "Download" button
- System initiates download of PDF file to user's device

3.8.3 Functional Requirements

REQ-RESUME-1: The system shall generate a PDF resume document containing all information from the job seeker's profile formatted in a professional layout.

REQ-RESUME-2: The generated resume shall include sections for Personal Information, Professional Summary, Work Experience (in reverse chronological order), Education (in reverse chronological order), and Skills.

REQ-RESUME-3: The system shall include the profile photo in the resume header if the user has uploaded one.

REQ-RESUME-4: The generated PDF shall be formatted for standard A4 size with appropriate margins.

REQ-RESUME-5: The system shall name generated PDF files using the format "FirstName_LastName_Resume.pdf" for easy identification.

REQ-RESUME-6: The system shall provide a preview feature allowing users to view the generated resume before downloading.

REQ-RESUME-7: The system shall display an error message if profile data is incomplete and indicate which required sections must be completed before resume generation.

REQ-RESUME-8: The system shall allow users to regenerate their resume at any time to reflect updated profile information.

4. External Interface Requirements

4.1 User Interfaces

The Smart Job Recruitment Platform provides distinct web-based user interfaces for job seekers and recruiters, designed following modern UI/UX principles with emphasis on responsiveness and accessibility.

General UI Requirements:

The user interface shall be built using React.js components with a responsive design adapting to screen sizes from 320px (mobile) to 2560px+ (large desktop displays). The interface shall use a consistent color scheme with primary brand colors. All interactive elements shall provide visual feedback on hover and click states. The navigation bar shall be persistent across all pages containing the platform logo, main navigation links, user profile dropdown, notifications icon with badge, and messages icon with badge.

Job Seeker Interface:

The Job Seeker Dashboard shall display a personalized greeting, profile completion indicator, quick statistics (applications submitted, interviews scheduled), recommended jobs carousel showing top matches, and recent activity feed. The Profile page shall use a multi-section form layout with collapsible sections for Personal Information, Education, Work Experience, Skills, and Career Preferences, with inline editing capability and real-time validation feedback. The Job Search page shall feature a search bar with filters for location, job type, salary range, and posting date, a results grid showing job cards with company logo, job title, location, match score badge, and save/apply buttons. The Job Details page shall display full job description, requirements section with matching indicators, company information panel, and prominent "Apply Now" call to-action button. The My Applications page shall use a card layout showing job title, company, application date, status badge with color coding, and quick action buttons to view details or withdraw application. The Resume Generator page shall show download button.

Recruiter Interface:

The Recruiter Dashboard shall display active jobs count, total applications received count, pending reviews count, quick access cards to create new job posting and view recent applications, and analytics chart showing application trends over time. The Post Job page shall use a step-by-step wizard interface with progress indicator for sections including Basic Information, Job Description, Requirements, and Review & Publish. The My Jobs page shall display job listings in card or table format showing job title, status indicator, applicant count with breakdown by status, posted date, application deadline, and action buttons for view, edit, and close. The

Applicant Management page shall show applicant table with columns for name with profile photo thumbnail, relevance score with colored badge, application date, current status dropdown, and actions to view full profile, send message, or change status. The filtering toolbar shall include options to filter by status. The Applicant Profile View shall display complete candidate information in an organized layout mirroring the job seeker profile structure, with additional sections showing match score breakdown, application date, attached resume download button, and status change controls.

Common UI Elements:

Error messages shall be displayed using non-intrusive toast notifications appearing with auto-dismiss after 5 seconds and color-coded by severity (red for errors, yellow for warnings, green for success). Form validation shall provide inline error messages beneath form fields immediately upon blur. Loading states shall use skeleton screens or spinner components to indicate data fetching operations. All models and dialogs shall include clear close buttons and submit/cancel action buttons.

4.2 Hardware Interfaces

The Smart Job Recruitment Platform is a web-based application that does not directly interface with specialized hardware components. However, the system operates within standard computing hardware environments with the following considerations:

Client-Side Hardware:

The system shall be accessible from any device with a modern web browser including desktop computers, laptop computers, tablets, and smartphones. The client device must have sufficient processing power to run a modern web browser with JavaScript enabled. A minimum screen resolution of 320x568 pixels is supported for mobile devices, with optimal experience on devices with 1920x1080 or higher resolution. The system requires an input device (mouse or touchscreen) for interaction. File upload functionality requires the device to have local storage access for selecting resume photos and potential future document uploads.

Server-Side Hardware:

The backend application requires a server or cloud compute instance with minimum specifications of 2 CPU cores, 4GB RAM, and 50GB storage for the initial deployment, with the ability to scale based on user load. Since the system uses MongoDB Atlas, the database does not require separate on-premise storage; instead, it will rely on Atlas's fully managed cloud storage. Atlas also ensures built-in backups, auto-scaling, and distributed storage management. The infrastructure should include network connectivity capable of handling high concurrent requests, while horizontal scaling through load balancing must be supported to distribute traffic across multiple backend server instances as the user base grows.

4.3 Software Interfaces

The Smart Job Recruitment Platform interfaces with several external software components and systems to deliver its functionality:

Frontend Framework:
Component: React.js
Purpose: Client-side user interface rendering and state management
Interface: Component-based architecture with props and state management
Data Exchanged: User interactions, form data, application state

Backend Framework:
Component: Node.js with Express.js
Purpose: Server-side application logic and RESTful API endpoints
Interface: HTTP/HTTPS protocols handling JSON request/response payloads
Data Exchanged: API requests and responses including user data, job postings, applications, messages

Database Management System:
Component: MongoDB Atlas
Purpose: Persistent data storage for all application data
Interface: MongoDB driver for Node.js using connection string authentication
Data Exchanged: Documents stored as JSON objects including users, profiles, jobs, applications, messages, notifications
Data Sharing: Database access is exclusive to the backend application; no direct client access is permitted

Authentication System:
Component: JSON Web Tokens (JWT)
Purpose: Stateless authentication and authorization
Interface: Token generation on login, token validation middleware on API requests
Data Exchanged: Encoded tokens containing user ID, user type, and expiration timestamp

PDF Generation Library:
Component: PDFKit or Puppeteer
Purpose: Server-side generation of formatted PDF resume documents

Interface: Node.js library API accepting structured data and template specifications
Data Exchanged: Profile data input; PDF binary output

Web Browser:
Component: Google Chrome, Mozilla Firefox or Microsoft Edge
Purpose: Client-side application runtime environment
Interface: Standard web APIs (Fetch API, Local Storage, Session Storage)
Data Exchanged: HTML, CSS, JavaScript, JSON data via HTTP/HTTPS

4.4 Communications Interfaces

The Smart Job Recruitment Platform requires several communication interfaces to enable data exchange and user interactions:

HTTP/HTTPS Protocol:
The system shall use HTTPS (Hypertext Transfer Protocol Secure) for all client-server communications to ensure data encryption in transit. All API endpoints shall be accessible via HTTPS on standard port 443. The system shall implement TLS (Transport Layer Security) with strong cipher suites. HTTP requests shall use standard methods including GET for data retrieval, POST for resource creation, PUT/PATCH for updates, and DELETE for resource removal.

RESTful API Communication:
The backend shall expose a RESTful API following REST architectural principles with resource-based URL. Request and response payloads shall use JSON (JavaScript Object Notation) format with Content-Type header set. API responses shall include appropriate HTTP status codes (200 for success, 201 for created, 400 for client errors, 401 for unauthorized, 404 for not found, 500 for server errors). The API shall implement versioning to support future updates without breaking existing clients.

Network Security:
All communications shall implement appropriate security headers. API authentication shall use Bearer token authorization with JWT tokens included in the Authorization header of requests.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

Response Time:

The system shall load the main dashboard quickly under normal internet conditions. API requests should respond fast enough to ensure smooth user interaction. Database queries must be optimized with proper indexing so that common operations return results promptly. Page navigation should feel instant to the user by using techniques. Generating PDF resumes should also be completed within a reasonable time, allowing users to download their files without delay.

Throughput:

The system shall smoothly support multiple users accessing the platform at the same time, with the ability to increase capacity as usage grows. The database should efficiently manage frequent read and write operations under normal conditions. Application submissions should be processed quickly even during busy periods, and file uploads should allow several users to upload documents at once without slowing down the system.

Resource Utilization:

The backend application shall keep CPU and memory usage at healthy levels during normal operation. The system must run reliably for long periods without memory leaks. Database connections should be managed through proper pooling to avoid connection issues. On the client-side, the application should maintain smooth and responsive animations and interactions.

Scalability:

The system should be designed to scale horizontally by adding more server instances behind a load balancer. The database must support read replicas and allow sharding in the future as the user base expands. Static files such as images and scripts should be delivered through a CDN for faster global performance. The overall architecture should maintain good performance even as the system grows to support a large number of active users.

5.2 Safety Requirements

Error Handling:

The system shall implement comprehensive error handling preventing display of stack traces or sensitive system information to end users. All errors shall be logged with sufficient detail for debugging while user-

facing error messages remain generic and helpful. Critical errors shall trigger automated alerts to system administrators for immediate investigation.

System Monitoring:

The system shall implement health check endpoints allowing monitoring systems to verify application availability. Resource utilization (CPU, memory, disk space, network traffic) shall be continuously monitored with alerts triggered when thresholds are exceeded. Application logs shall be centralized and retained for at least 90 days for auditing and troubleshooting.

Failure Recovery:

The system shall automatically restart failed application processes without manual intervention. Database connections shall implement automatic reconnection logic to recover from temporary network issues. The system shall implement circuit breaker patterns for external service calls to prevent cascading failures. User sessions shall be preserved across server restarts using persistent session storage.

5.3 Security Requirements

Authentication Security:

The system shall enforce strong password requirements including minimum length, complexity rules, and prevention of common passwords. Passwords shall be hashed using bcrypt algorithm with minimum cost factor of 10 before database storage. The system shall implement account lockout after 5 consecutive failed login attempts within 15 minutes.

Authorization and Access Control:

The system shall implement role-based access control (RBAC) strictly enforcing separation between job seeker and recruiter functionality. All API endpoints shall validate JWT tokens and verify user permissions before processing requests. Job seekers shall only access their own profile and application data, not other users' information. Recruiters shall only access applications for their own job postings and profiles of applicants to those jobs. Administrative functions shall be protected by separate admin role not accessible to regular users.

Data Privacy:

The system shall implement privacy controls ensuring users can only view data they are authorized to access. Personal information shall be stored with appropriate encryption at rest for sensitive fields. The system shall provide functionality for users to request deletion of their accounts and associated data in compliance with data

protection regulations. Profile visibility shall be controlled with job seeker profiles visible only to recruiters viewing their applications.

Network Security:

All client-server communications shall use HTTPS with valid SSL/TLS certificates. The system shall implement security headers. API endpoints shall implement rate limiting to prevent brute force attacks and denial of service. The system shall sanitize all user inputs.

Session Management:

User sessions shall be managed using secure, HTTP-only cookies for web applications. Session tokens shall be regenerated after authentication to prevent attacks. Logout functionality shall properly invalidate tokens both client and server-side.

5.4 Software Quality Attributes

Reliability:

The system shall maintain 99% uptime during business hours (8 AM to 8 PM local time, Monday through Friday) excluding planned maintenance. The mean time between failures (MTBF) shall exceed 720 hours (30 days) of continuous operation. The system shall detect and recover from transient failures automatically without data loss. Critical user actions (application submissions, job postings) shall be confirmed with success messages and failure cases shall provide clear guidance for retry.

Maintainability:

The codebase shall follow consistent coding standards and style guidelines documented in the project repository. Code shall be modular with clear separation of concerns following MVC architectural patterns. All functions and components shall include inline documentation explaining purpose and usage. The system shall maintain minimum 70% code coverage with automated unit and integration tests.

Usability:

The user interface shall be intuitive requiring minimal training for users with basic web literacy. Critical user tasks (profile creation, job application, job posting) shall be completable within 5 minutes by first-time users. The system shall provide contextual help and tooltips explaining complex features. Error messages shall be clear, specific, and provide actionable guidance for resolution. The system shall support keyboard navigation and screen readers for accessibility compliance.

Portability:

The application shall run on any operating system supporting Node.js without code modifications. The client-side application shall function consistently across all supported browsers without browser-specific code. Configuration shall be externalized through environment variables enabling deployment to different environments (development, staging, production) without code changes.

Testability:

The system architecture shall support automated testing at unit, integration, and end-to-end levels. APIs shall be documented with example requests and responses enabling automated API testing. The system shall include test fixtures and seed data for consistent testing environments.

Flexibility:

The system architecture shall support addition of new features without requiring major refactoring. The API shall be versioned allowing backward-compatible updates. The database schema shall accommodate new fields and collections without breaking existing functionality. The matching algorithm shall be configurable allowing adjustment of scoring weights through configuration rather than code changes.

Interoperability:

The API shall follow REST principles enabling future integration with mobile applications or third-party services. Data exports shall use standard formats compatible with common tools.

6. Other Requirements

Internationalization:

While the initial version targets English-speaking users in Sri Lanka, the system architecture shall support future internationalization. All user-facing text shall be externalized from code into resource files enabling translation without code changes. Date and time displays shall use formats appropriate for the user's locale. The system shall support UTF-8-character encoding.

Legal and Compliance:

The system shall include Terms of Service and Privacy Policy documents accessible from the footer of all pages. Users shall be required to accept these terms during registration. The system shall comply with applicable data protection regulations in Sri Lanka. The platform shall include disclaimer text clarifying that it facilitates connections between recruiters and candidates but is not responsible for hiring decisions or employment outcomes. The system shall implement age verification ensuring users are at least 18 years old during registration.

Analytics and Reporting:

The system shall track basic usage metrics including number of registered users by type, number of jobs posted and number of applications submitted. The recruiter dashboard shall display analytics showing application trends over time for their job postings. The system shall track match score effectiveness by monitoring which score ranges correlate with successful hires. Administrative users shall have access to system-wide analytics dashboard showing platform growth and engagement metrics

Help and Support:

The system shall include an integrated help center with FAQ documentation covering common user questions and tasks. Each major feature shall include contextual help accessible through help icons or links. Error messages shall include reference IDs enabling support staff to locate relevant error logs quickly.

Future Enhancement Considerations:

The system architecture shall consider future enhancements including integration with external job boards (such as LinkedIn and Indeed) for cross-posting, AI-powered resume parsing from uploaded documents, video interview capabilities, assessment test integration for technical positions, mobile native applications for iOS and Android, premium subscription tiers with advanced features, and an analytics dashboard with predictive hiring insights. The system shall also support generating in-platform notifications for job seekers when a newly

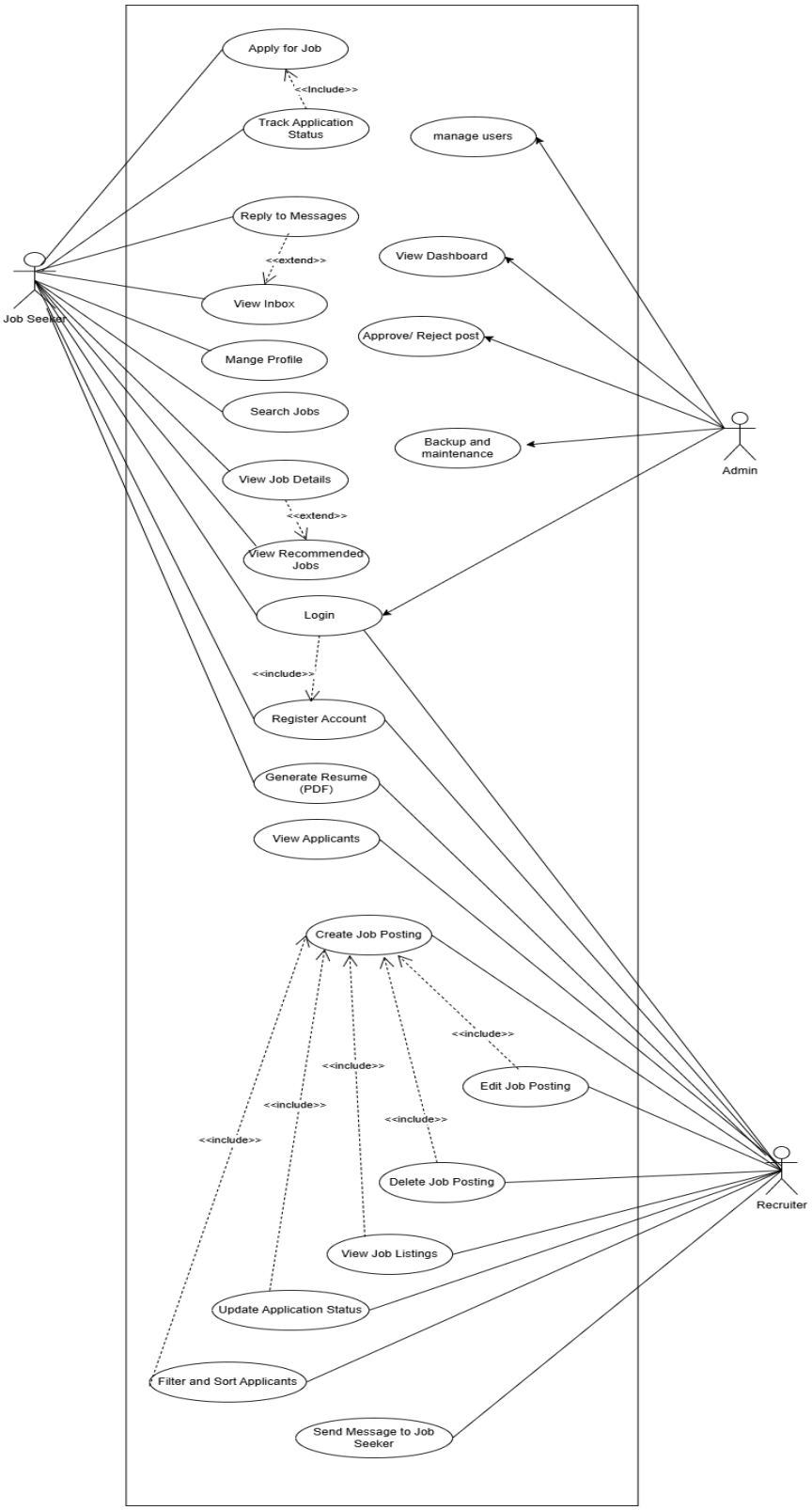
posted job matches their profile with a score above 70%. Additionally, future versions may allow job seekers to choose from multiple resume templates when generating or downloading their resumes. These features are out of scope for version 1.0 but the architecture should be designed so that they can be added seamlessly later.

Appendix A: Glossary

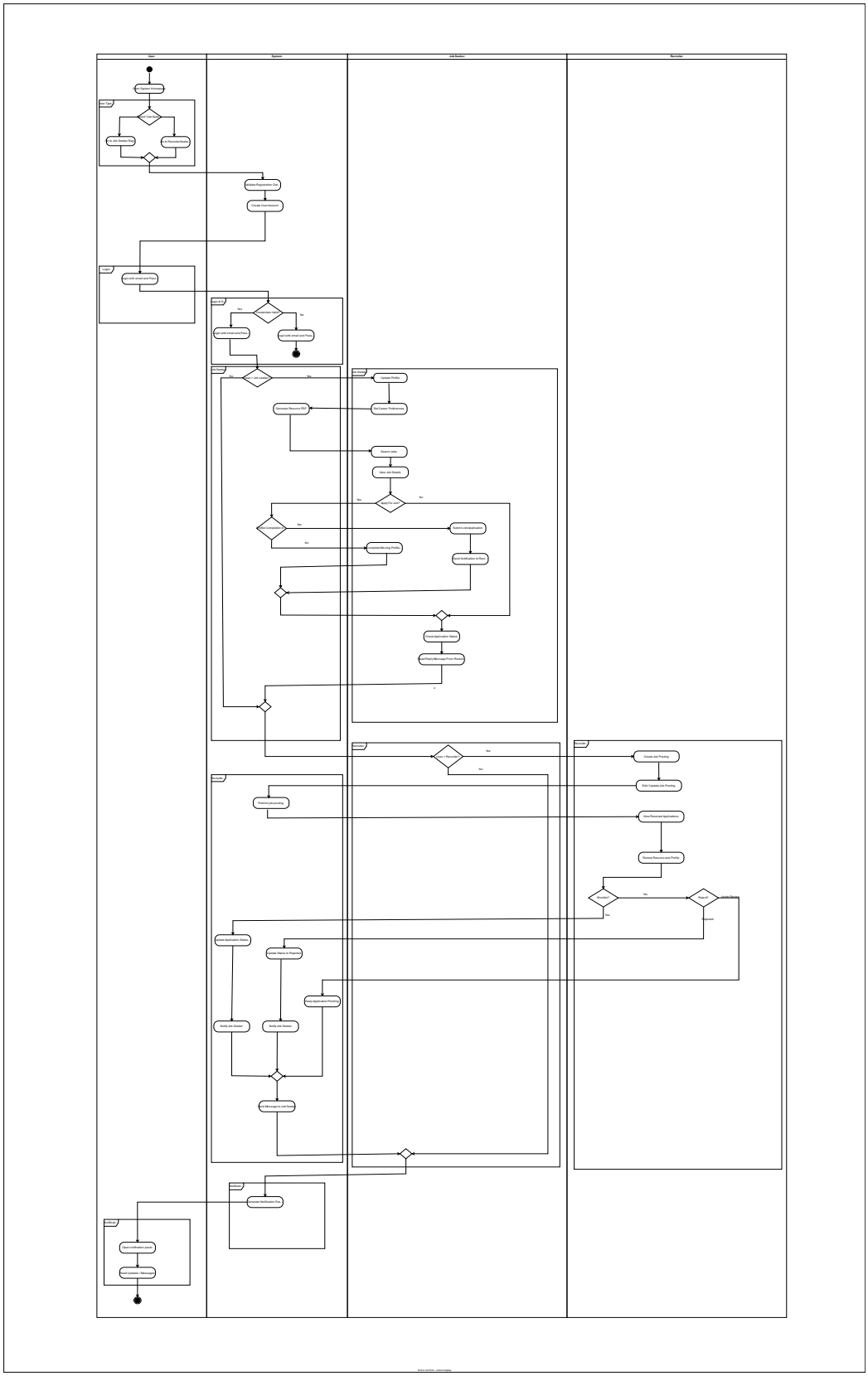
- ATS (Applicant Tracking System): Software applications that enable the electronic handling of recruitment and hiring needs, typically used by large corporations.
- API (Application Programming Interface): A set of protocols and tools for building software applications, specifying how software components should interact.
- Bcrypt: A password hashing function designed to be computationally expensive to resist brute-force attacks.
- JWT (JSON Web Token): An open standard for securely transmitting information between parties as a JSON object, commonly used for authentication.
- Job Seeker: A user of the platform who creates a profile to search for and apply to job opportunities.
- Match Score: A numerical value (0-100) representing how well a job seeker's profile aligns with a job posting's requirements.
- MongoDB: A NoSQL document-oriented database program that stores data in flexible, JSON-like documents.
- Node.js: An open-source, cross-platform JavaScript runtime environment that executes JavaScript code outside a web browser.
- PDF (Portable Document Format): A file format used to present documents independent of software, hardware, or operating systems.
- React.js: A JavaScript library for building user interfaces, particularly single-page applications, maintained by Meta.
- Recruiter: A user of the platform who posts job openings and reviews applications from job seekers.
- REST (Representational State Transfer): An architectural style for providing standards between computer systems on the web, making it easier for systems to communicate.
- Relevance Score: A numerical value indicating how closely an applicant's qualifications match a job's requirements.
- SME (Small and Medium-sized Enterprises): Businesses with employee counts or revenue below certain thresholds, varying by country.
- SSL/TLS: Security protocols that provide communications security over a computer network.
- UI (User Interface): The space where interactions between humans and machines occur, including screens, pages, and visual elements.

Appendix B: Analysis Models

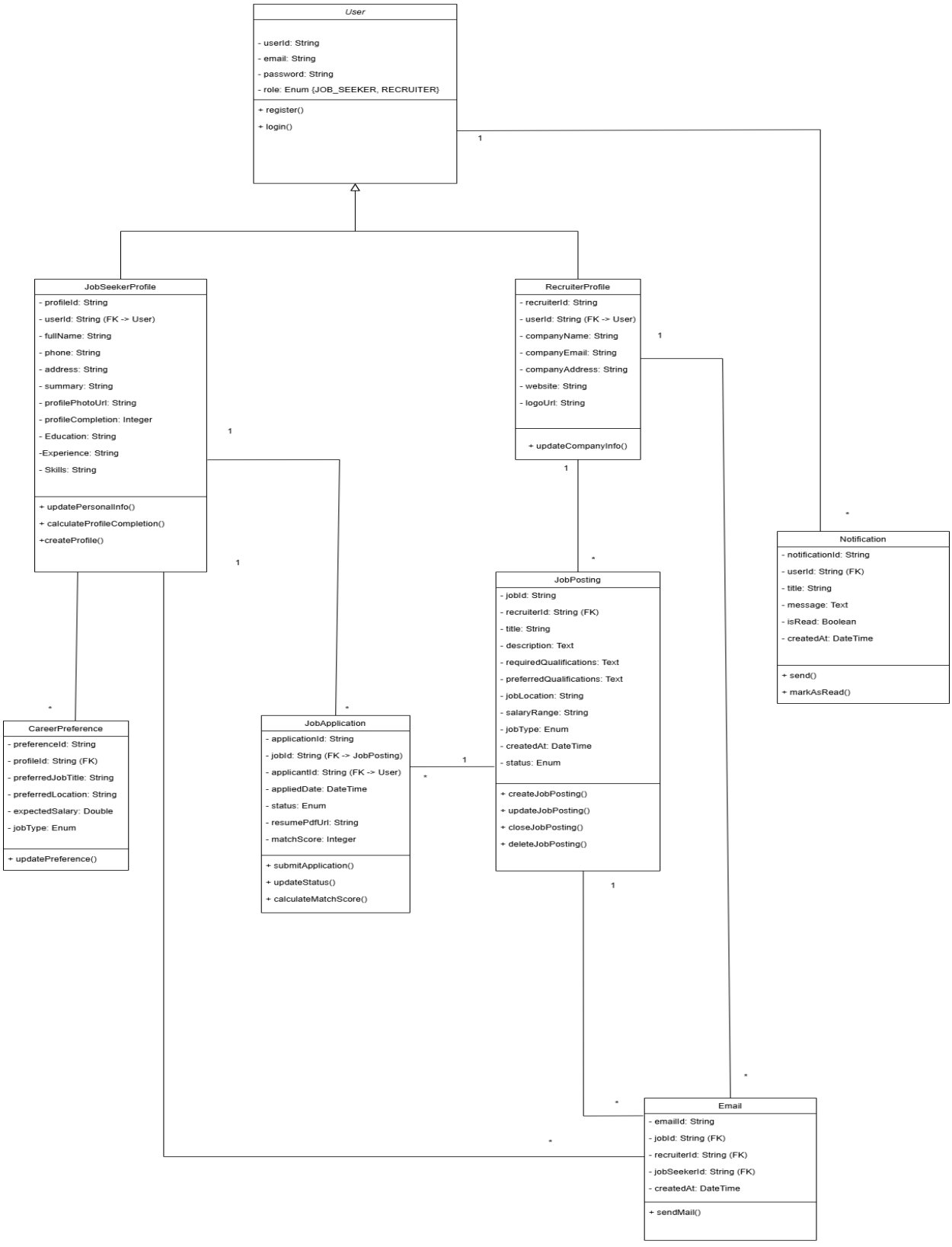
Use case Diagram



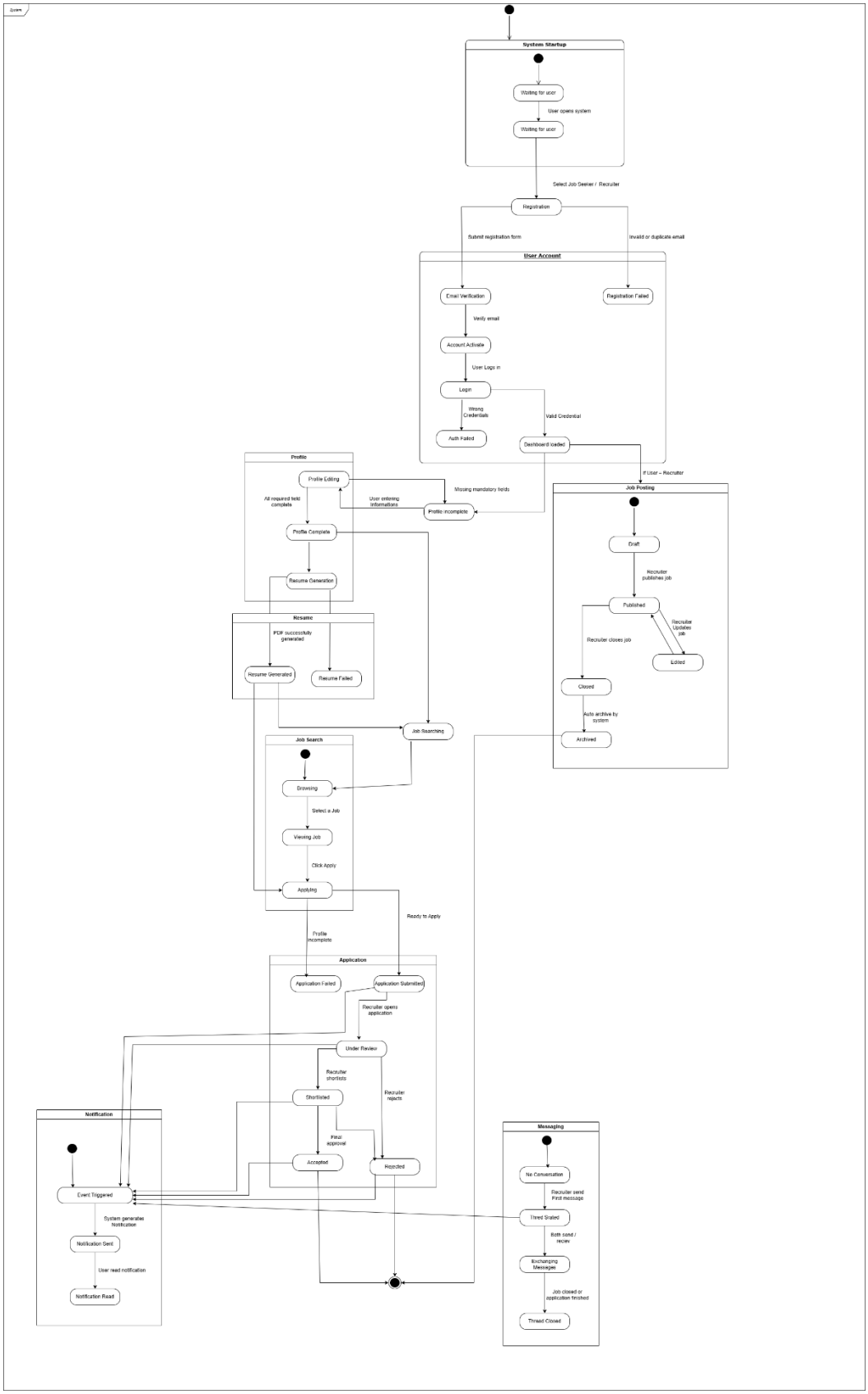
Activity Diagram



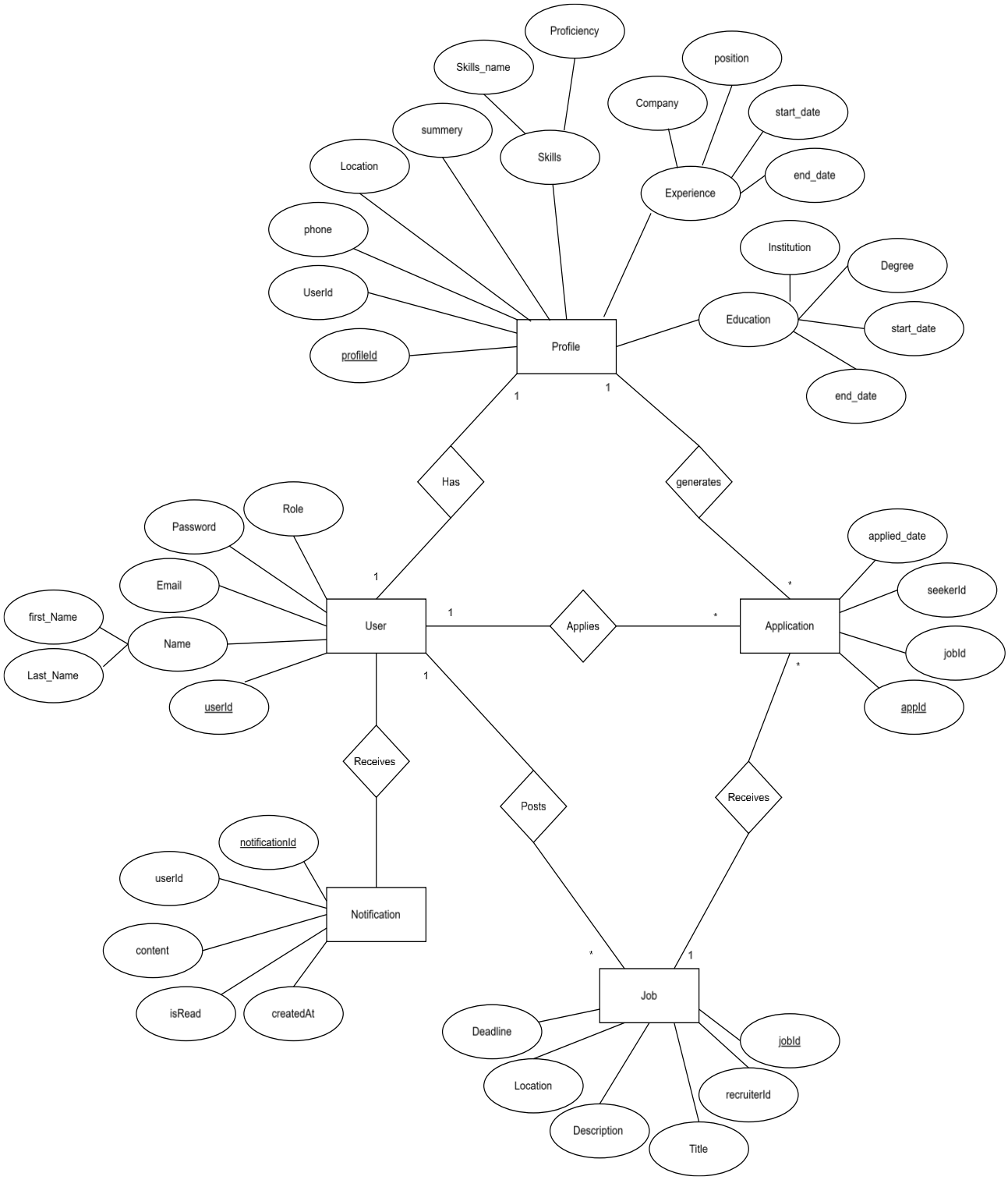
Class Diagram



State chart Diagram



ER Diagram


Proficiency
Skills_name
position
Company
start_date
summery
Skills
Location
Experience
end_date
phone
Institution
Degree
UserId
Education
start_date
Profile
profileId
end_date
1
1
Has
generates
Role
Password
applied_date
1
*
Email
seekerId
first_Name
User
1
Applies
*
Application
Name
jobId
Last_Name
1
*
userId
appId
Receives
Posts
Receives
notificationId
userId
Notification
content
*
1
isRead
createdAt
Job
Deadline
jobId
Location
recruiterId
Description
Title

Appendix C: Issues List

Open Requirements Issues:

1. PDF Template Design: Specific visual design and layout for resume templates has not been finalized.
2. Matching Algorithm Weights: Current specification uses 40% skills, 30% education, 30% experience weighting.
3. Search Functionality Scope: Job search filtering capabilities need detailed specification including which fields are searchable and how fuzzy matching will be implemented.
4. Application Withdrawal: Specification does not clearly define whether job seekers can withdraw applications after submission and what happens to related data.
5. Profile Visibility Controls: More detailed requirements needed for what information is visible to recruiters at different stages (before application, after application, etc.).
6. Mobile App Consideration: While mobile-responsive web is specified, decision needed on whether native mobile apps will be developed in future and how this affects current API design.
7. Admin Panel: Requirements do not specify administrative panel features for platform management. Decision needed on admin capabilities for user management, content moderation, and system monitoring in version 1.0.

Pending Decisions:

1. Data Retention Policy: Formal policy needed defining how long user data, applications, and messages are retained after account deletion or inactivity.
2. Multi-language Support Timeline: Decision needed on priority for internationalization beyond English and which languages to support first.
3. Integration APIs: Requirements for potential third-party integrations need specification if planned for future phases.
4. Video Interview Features: Scope and timeline for potential video interview capabilities need definition as mentioned in future enhancements.

Information Needed:

1. Hosting Environment: Confirmation needed on hosting solution.
2. Load Testing Parameters: Specific expected user load and concurrency requirements during initial deployment and at 6-month and 12-month projections.
3. Branding Guidelines: Company name, logo, color scheme, and other brand assets needed for UI

Conflicts Awaiting Resolution:

1. Session Duration vs Security: JWT token validity period of 24 hours balances user convenience with security, but feedback from security review may require shorter duration.
2. Profile Completeness Threshold: Current requirement is 75% profile completion before allowing applications. This may need adjustment based on user testing feedback on whether it's too restrictive or too lenient.
3. Notification Frequency: Balance needed between keeping users informed and avoiding notification fatigue. Current specification may generate high notification volume for active recruiters.

Technical Debt and Risks:

1. Scalability Testing: System has not been load tested at specified concurrent user levels. Performance requirements may need revision after testing.
2. Algorithm Validation: Matching algorithm effectiveness needs validation through user studies to ensure match scores actually predict successful hires.
3. Mobile UX: While responsive design is specified, actual mobile user experience should be validated through testing on various devices.

TBD Items:

1. TBD-001: Specific PDF generation library selection (PDFKit vs Puppeteer vs alternatives) pending performance and feature testing.
2. TBD-002: Exact field validations for profile sections (character limits, format requirements) to be finalized during UI design phase.
3. TBD-003: Specific error codes and messages for API responses to be documented in API specification document.
4. TBD-004: Performance monitoring tools and alerting thresholds to be configured during deployment phase.
5. TBD-005: Specific test cases and acceptance criteria for each requirement to be documented in test plan.
6. TBD-006: Database indexing strategy for optimal query performance to be determined after schema finalization.
7. TBD-007: API rate limiting exact thresholds to be determined after load testing.