# Software Requirements Specification (SRS)

**Project:** HR Hiring & Onboarding System

**Prepared for:** [Your IT Company]

**Prepared by:** [Author]

**Date:** 2025-09-17

## 1. Introduction

### 1.1 Purpose

This SRS describes the requirements for an HR Hiring & Onboarding System designed to systemize and automate the candidate selection workflow for an IT company. The system will ingest existing candidate records from the database, support bulk communications, run technical assessments, collect and score responses, schedule and evaluate interviews, and automate onboarding tasks (offer letters, ID generation, onboarding scheduling).

### 1.2 Scope

The system will: - Import/use existing candidate records in the database. - Send bulk and individual emails with attachments (technical assessments, links). - Provide secure candidate assessment submission forms. - Allow admin review and scoring of assessments and interviews. - Automatically shortlist candidates based on configurable mark thresholds. - Send interview invites (individual) with date/time and meeting link. - Automate offer emails and onboarding schedule notifications. - Allow hired candidates to complete personal profiles and upload required documents. - Generate offer letters and ID cards automatically (PDF).

### 1.3 Definitions, Acronyms, Abbreviations

* Admin: HR staff or system user with permission to manage hiring.
* Candidate: Person who has applied and is in the database.
* Assessment: Technical test document and answer submission.
* ATS: Applicant Tracking System (the existing database + new system).
* OTP: One Time Password (optional for verification).

## 2. Overall Description

### 2.1 User Classes and Characteristics

* **Admin (HR / Hiring Manager):** Full system access for recruiting workflow, scoring, thresholds, templates.
* **Interviewer / Panel Member:** Access to assigned interviews and scoring screens.
* **Candidate:** Can receive emails, click assessment links, submit answers, and later (if selected) complete profile & documents.
* **System (automated processes):** Background tasks that send emails, generate documents, and run automatic shortlisting.

### 2.2 Operating Environment

* Web-based application (responsive)
* Backend: Laravel / Node.js / chosen stack
* Database: MySQL / PostgreSQL (existing candidate DB import supported)
* File storage: local filesystem or cloud (S3)
* Mail service: SMTP / SendGrid / Mailgun

### 2.3 Constraints and Assumptions

* Candidate personal data already exists in an accessible database.
* Email sending volume conforms to the chosen mail provider limits.
* Admins have necessary permissions and training.
* Candidate answer submissions limited to specified file types/size.

## 3. Functional Requirements

Each requirement is labeled FR-xxx for traceability.

### 3.1 Candidate Data Integration

* **FR-001**: System shall import existing candidate records (name, email, phone, resume link, position applied, status, source, notes) from the current database via a one-time migration tool or scheduled sync.
* **FR-002**: System shall display candidate profile pages showing all imported fields and activity history.

### 3.2 User Authentication & Authorization

* **FR-003**: System shall provide secure admin login (email + password) with role-based access control (Admin, Interviewer).
* **FR-004**: System shall support candidate access via secure, single-use assessment links (tokenized) — no mandatory candidate accounts for assessments.

### 3.3 Email Templates & Bulk Email

* **FR-005**: System shall allow admins to create/edit email templates with placeholders (e.g., {{first\_name}}, {{assessment\_name}}, {{meeting\_link}}, {{date}}, {{time}}).
* **FR-006**: System shall allow admins to select a set of candidates and send bulk emails attaching documents (e.g., assessment PDF) and unique assessment links.
* **FR-007**: System shall queue emails and track delivery status (sent, delivered, bounced, opened if supported by provider).

### 3.4 Technical Assessment Delivery & Submission

* **FR-008**: System shall host or attach technical assessment documents for candidates to download.
* **FR-009**: System shall generate a unique assessment submission link per candidate.
* **FR-010**: System shall provide submission form(s) where candidates can upload files and/or enter answers (supports file upload, text entry, code block, multiple-choice).
* **FR-011**: System shall validate submission formats and store submissions securely linked to the candidate record.

### 3.5 Assessment Review & Scoring

* **FR-012**: System shall present submissions to admin/interviewers with an interface to review and enter marks per evaluation criteria.
* **FR-013**: System shall support configurable marking rubrics (e.g., criteria: correctness, code quality, explanation — each with a weight and max score).
* **FR-014**: System shall compute total assessment score automatically using configured weights.

### 3.6 Automated Shortlisting (Assessment)

* **FR-015**: System shall allow admin to set a minimum assessment mark (absolute or percentile) and auto-shortlist candidates whose assessment score >= threshold.
* **FR-016**: System shall provide a dashboard/listing filter for shortlisting results and reasons (e.g., “Assessment score 82/100 >= threshold 75”).

### 3.7 Technical Interview Scheduling & Invitations

* **FR-017**: System shall allow admin to schedule technical interviews per candidate with date, time, interviewer(s), and meeting link (e.g., Zoom/Teams link).
* **FR-018**: System shall be able to send individualized interview invitation emails including date/time and meeting link.
* **FR-019**: System shall support rescheduling and send updated emails automatically.

### 3.8 Interview Scoring & Automated Shortlisting (Interview)

* **FR-020**: System shall enable interviewers to enter marks on interview criteria and comments.
* **FR-021**: System shall allow admin to set interview mark thresholds and apply automatic shortlisting based on interview scores (e.g., combined assessment + interview weighted score, or interview-only threshold).

### 3.9 Final Hiring Decision & Offer

* **FR-022**: System shall allow admin to mark a candidate as “Offer Extended / Hired / Rejected”.
* **FR-023**: Upon acceptance, the system shall automatically send an offer email to the candidate with the offer letter attached (PDF) and next steps.

### 3.10 Candidate Onboarding (post-offer)

* **FR-024**: System shall provide a secure portal for selected candidates to login and complete their personal profile (full name, profile image, personal details, bank details, emergency contact, signed documents).
* **FR-025**: System shall validate required onboarding fields and present progress (e.g., 60% complete).
* **FR-026**: System shall mark candidates who have completed required documents as “Documents Submitted” and show in admin dashboard.
* **FR-027**: System shall allow admin to set onboarding start date(s) and send automated onboarding invitation emails.
* **FR-028**: On the onboarding date (configured), the system shall send final reminders and attach the offer letter and automatically generated ID card PDF to the candidate.

### 3.11 Document Generation & ID Card

* **FR-029**: System shall generate personalized offer letters as PDF using templates and candidate data (position, salary, start date, manager, etc.).
* **FR-030**: System shall generate employee ID cards (PDF or image) with company logo, employee name, position, employee ID, and QR code linking to the employee profile.
* **FR-031**: System shall store generated documents and log generation timestamps.

### 3.12 Audit Trail & Activity Logs

* **FR-032**: System shall record actions (emails sent, assessment submitted, scores entered, status changes) with timestamps and admin user ids.

### 3.13 Notifications & Reminders

* **FR-033**: System shall support scheduled reminders (e.g., assessment pending, interview reminder 24hrs before) and allow admins to configure timing.

### 3.14 Reporting & Dashboards

* **FR-034**: System shall present dashboards for HR: candidate pipeline counts (Applied, Assessment Pending, Shortlisted, Interview Scheduled, Hired), average scores, time-to-hire metrics, and source effectiveness.
* **FR-035**: System shall export candidate lists, scores, and reports as CSV/PDF.

### 3.15 Security & Privacy

* **FR-036**: System shall store candidate PII encrypted at rest where supported and use HTTPS for all traffic.
* **FR-037**: System shall support role-based access and only expose PII to authorized users.
* **FR-038**: System shall comply with data retention policies configurable by admin (e.g., purge candidate data after X months).

## 4. Non-Functional Requirements

### 4.1 Performance

* **NFR-001**: System shall handle up to 5k candidate records and bulk-send up to 500 emails per batch (scalable via queuing).
* **NFR-002**: Email-sending tasks and document generation shall be performed asynchronously (job queue) and not block admin UI.

### 4.2 Availability & Reliability

* **NFR-003**: System should have 99.5% uptime during business hours.

### 4.3 Scalability

* **NFR-004**: System will be designed to horizontally scale email workers and file storage.

### 4.4 Usability

* **NFR-005**: Admin workflows should complete common tasks (send assessment, shortlist) within 3 clicks where possible.

### 4.5 Security

* **NFR-006**: Support password policies, 2FA for admins (optional), and secure token generation for candidate links.
* **NFR-007**: File uploads limited to safe types and scanned for viruses (if integrated).

### 4.6 Maintainability

* **NFR-008**: Codebase modular and well-documented. APIs documented with OpenAPI/Swagger.

### 4.7 Compliance

* **NFR-009**: System shall include mechanisms for data export and deletion to comply with data protection laws.

## 5. Data Model (high-level)

Tables (examples): Candidates, Assessments, Assessment\_Submissions, Assessment\_Scores, Interviews, Interview\_Scores, Admin\_Users, Email\_Templates, Document\_Templates, Audit\_Logs, Employee\_Profiles.

**Candidate**: id, first\_name, last\_name, email, phone, resume\_url, applied\_position, status, source, notes, created\_at, updated\_at

**Assessment**: id, title, description, file\_url, created\_by, created\_at

**Assessment\_Submission**: id, candidate\_id, assessment\_id, submitted\_at, files[], answers\_json, reviewer\_status

**Assessment\_Score**: id, submission\_id, reviewer\_id, criteria\_scores\_json, total\_score, comments, scored\_at

**Interview**: id, candidate\_id, interviewer\_id, scheduled\_at, meeting\_link, status

**Interview\_Score**: id, interview\_id, interviewer\_id, criteria\_scores\_json, total\_score, comments

**Employee\_Profile**: id, candidate\_id, employee\_id, joining\_date, bank\_details\_encrypted, documents[], profile\_image\_url

## 6. Workflows & Use Cases

### 6.1 Bulk Assessment Workflow

1. Admin selects candidates (filter by position/status).
2. Admin attaches assessment template and clicks “Send Assessment”.
3. System generates unique assessment tokens/links and queues individualized emails.
4. Candidate receives email, downloads assessment, submits answers via link.
5. Submission appears in admin review queue for scoring.
6. Admin scores; system computes total and auto-shortlists per FR-015.

### 6.2 Interview Workflow

1. Admin schedules interview for shortlisted candidates.
2. System sends individualized interview invites.
3. Interviewer enters marks during/after interview.
4. System applies interview thresholds and updates candidate status.

### 6.3 Offer & Onboarding Workflow

1. Admin marks candidate as Hired / Offer Extended.
2. System generates offer letter based on template and candidate data.
3. System emails offer + instructions; candidate confirms.
4. Candidate logs into portal and completes required onboarding fields.
5. Admin sets onboarding date; system sends reminders and generates ID card.

## 7. UI/UX Requirements (high-level)

* Dashboard with quick actions (Send Assessment, Shortlist, Schedule Interview).
* Candidate list with filters and bulk action checkboxes.
* Candidate detail page with timeline of actions and documents.
* Scoring UI with rubric-based inputs and automatic totals.
* Onboarding portal for candidates: mobile-friendly, minimal steps.

## 8. APIs and Integration Points

* Import API for existing candidate DB (CSV import and REST endpoints).
* Mailer integration: SMTP / SendGrid / Mailgun.
* Calendar integration (optional): Google Calendar / Outlook to add interview events.
* Document generation: server-side PDF generator (e.g., wkhtmltopdf, Puppeteer, or library).
* Optional SSO (AzureAD / Google Workspace) for admin login.

## 9. Acceptance Criteria

* All imported candidates visible and searchable.
* Admin can send bulk assessment emails with unique links and attachments.
* Candidate can submit assessment and admin can score it.
* System shortlists correctly based on thresholds set by admin.
* Interview scheduling emails sent and interview scoring works.
* Offer letters and ID cards generated and emailed.
* Candidate onboarding portal accepts documents and updates candidate status.

## 10. Security, Privacy & Legal Considerations

* Use TLS for all web traffic and enforce strong admin passwords.
* Limit retention of candidate data per company policy; provide export and deletion tools.
* Ensure generated PDFs do not leak admin-only data.

## 11. Testing Requirements

* Unit tests for scoring & thresholds.
* Integration tests for email sending and document generation.
* User acceptance testing for HR workflows.
* Load testing for bulk email flows.

## 12. Deployment & Operational Notes

* Use job queue (Redis + workers) for asynchronous email/document tasks.
* Backups for database and file storage.
* Monitoring (errors, email bounce rate, job queue length).

## 13. Traceability Matrix (sample)

* FR-005 -> UI: Email Template Editor, API: /templates
* FR-006 -> UI: Bulk Send Modal, Job: EmailQueue
* FR-012 -> UI: Submission Review Page, DB: assessment\_submissions

## 14. Appendix

### 14.1 Sample Email Template Placeholders

* {{first\_name}}, {{last\_name}}, {{position}}, {{assessment\_link}}, {{meeting\_link}}, {{interview\_date}}, {{interview\_time}}, {{offer\_link}}

### 14.2 Example Thresholds

* Assessment pass: 70/100
* Interview pass: 60/100
* Combined score weight: Assessment 60% + Interview 40%

## 15. Next Steps / Recommendations

1. Review and confirm the SRS with stakeholders and HR.
2. Prepare data-migration plan for existing candidate DB.
3. Choose email provider and document-generation tools.
4. Prioritize MVP features: Candidate import, Bulk assessment send, Submission and scoring, Automatic shortlisting, Offer generation.

*End of SRS*