Software Requirements Specification (SRS)

Project: Dynamic Resume Analyzer

Version: 1.0

Authors: Rajesh Banginwar/

Vineet Sharma	PES1UG23CS690
Vishwambhara Hebbalalu	PES1UG23CS700
Vrishank Amembal	PES1UG23CS705
Prerana M P	PES1UG23CS903

Date: 31-08-2025

Status: Model Answer / Approved

Revision history

Version	Date	Author	Change summary	Approval
1.0	31-08-2025	Instructor	SRS with diagrams embedded	

Approvals

Role	Name	Signature / Email	Date
Course Coordinator			

Table of Contents

1. Introduction

1.1 Purpose

This document is a Software Requirements Specification (SRS) for an **Dynamic Resume Analyzer system**. It defines functional and non-functional requirements, interfaces, and verification criteria intended for instructors and students to use as a reference.

1.2 Scope

Covers parsing of candidate resumes, extracting key sections (Skills, Education, Work Experience, Projects), and evaluating them against job requirements. The system checks for required keywords and highlights gaps in the candidate's profile. It includes regex/NLP-based parsing and section mapping but excludes advanced semantic analysis, ATS integration, and internal HR processes beyond resume screening.

1.3 Audience

- Developers
- QA Engineers
- Recruiters / HR Teams
- Assessment Evaluators

1.4 Definitions

- **Keyword Matching** – Process of checking if resume content includes job-specific terms (e.g., technologies, certifications).
- **Section Mapping** – Logic to identify resume sections such as Education, Skills, and Work Experience.
- **ATS (Applicant Tracking System)** – Software used by recruiters to manage, filter, and evaluate candidates.
- **NLP (Natural Language Processing)** – Techniques to analyze text beyond regex-based matching (optional future scope).

2. Overall description

2.1 Product perspective

The system is a resume evaluation service that interacts with candidate resumes and job descriptions.It consists of

- Parsing Engine – extracts structured data (skills, education, experience) in terms of sections from resumes.
- Job Requirement Module – processes the role description and identifies required keywords/skills.
- Matching Engine – compares extracted resume data against job requirements and performs gap analysis.
- Feedback Generator – produces a structured report highlighting strengths, missing skills, and overall fit.

This product is primarily designed as a standalone web application for recruiters. In future extensions, it may integrate with existing ATS (Applicant Tracking Systems) or career platforms, though such integration is excluded from the current scope.

2.2 Major product functions

- Upload resume of the candidate in (Text / PDF / Docx) format
- Extract sections: Education, Skills, Work Experience, Projects, Certifications.
- Parse job descriptions to extract required skills/keywords.
- Perform keyword-based matching between resume and job profile.
- Highlight missing skills/sections for the given job.
- Generate a fit score/report (percentage or qualitative feedback).
- Provide recruiters with a summary dashboard of candidate-job alignment.

2.3 User Roles and Characteristics

Recruiter / HR Team

- Expects quick, clear results on candidate-job fit.
- Limited time for resume screening
- Needs actionable feedback (who to shortlist, what's missing).

Developers (Project Team)

- Responsible for building the resume parsing, keyword matching, and feedback generation modules.
- Need to ensure robustness in handling varied resume formats and job descriptions.
- Must design the system for scalability (handling many resumes at once) and maintainability (easy updates to parsing rules or keywords).

QA Engineers (Project Team)

- Validate that the system correctly extracts sections and identifies missing skills.
- Ensure accuracy of keyword matching and gap analysis.
- Test the system under different inputs (varied resume formats, incomplete resumes, long/short job descriptions).
- Ensure usability and reliability of the system for recruiters.

Assessment Evaluators

- Stakeholders such as professors, project reviewers, or pilot users who evaluate the product.
- Focus on accuracy, usability, completeness of features, and clarity of reports.
- Provide feedback for future improvements and enhancements.

2.4 Operating Environment

- Web application running on cloud or on-prem servers.
- Supported input formats: PDF, DOCX, TXT resumes.
- Database: Stores parsed resumes and job requirement templates.
- Network environment: Secure HTTPS/TLS for data transfer
- Supported platforms: Desktop and mobile browsers for recruiters.

2.5 Constraints

- Must comply with data privacy laws (GDPR, local regulations).
- Resume parsing limited to textual content (no image/PDF scans in MVP).
- Job requirement matching limited to keyword presence (not semantic similarity in MVP).
- Only supports English resumes and job descriptions initially.

3. External interface requirements

3.1 User Interfaces

UI-001: Web Application Interface

- Description: Browser-based responsive web interface for users to upload resumes and view results
- Components:
  - Upload page with drag-and-drop functionality
  - Progress indicator during processing
  - Results dashboard showing parsed sections and feedback
  - Download interface for improved resume format
- Standards: HTML5, CSS3, responsive design compatible with Chrome 90+, Firefox 88+, Safari 14+, Edge 90+
- Screen Resolution: Minimum 1024x768, optimized for mobile devices (320px width minimum)

UI-002: Admin Dashboard Interface

- Description: Administrative interface for system monitoring and configuration
- Components:
  - System performance metrics
  - User activity logs
  - Resume processing statistics
  - Configuration management for regex patterns

3.2 Hardware Interfaces

HW-001: Server Hardware Requirements

- Description: Cloud-based or on-premise server infrastructure
- Specifications:
  - Minimum 4 CPU cores, 8GB RAM
  - 100GB storage for temporary file processing
  - Network interface supporting HTTPS/TLS 1.2+
- Storage: Temporary file storage with automatic cleanup after processing

3.3 Software Interfaces

SW-001: Document Processing Libraries

- Interface: PyPDF2/pdfplumber for PDF text extraction
- Version: PyPDF2 v3.0+ or pdfplumber v0.9+
- Data Exchange: Binary PDF files input, plain text output
- Error Handling: Graceful handling of corrupted or password-protected files

SW-002: DOCX Processing Interface

- Interface: python-docx library for Word document processing
- Version: python-docx v0.8.11+
- Data Exchange: Binary DOCX files input, structured text output
- Functions: Extract text while preserving section structure and formatting cues

SW-003: Database Interface

- Interface: PostgreSQL database connection
- Version: PostgreSQL 12+
- Protocol: SQL over TCP/IP using psycopg2 driver
- Data Exchange: JSON-formatted parsed resume data, user feedback logs
- Connection: Connection pooling with maximum 50 concurrent connections

SW-004: Caching Interface

- Interface: Redis for temporary data caching
- Version: Redis 6.0+
- Protocol: Redis Protocol (RESP)
- Usage: Cache regex compilation results and temporary processing data
- TTL: 1-hour expiration for cached processing results

3.4 Communication Interfaces

COM-001: HTTP/HTTPS Protocol

- Protocol: HTTPS with TLS 1.2+ encryption
- Port: 443 for HTTPS, 80 for HTTP redirect
- Data Format: JSON for API responses, multipart/form-data for file uploads
- Request Methods: GET, POST, PUT, DELETE
- Authentication: JWT tokens with 24-hour expiration

COM-002: File Upload Interface

- Supported Formats: PDF, DOCX, DOC, TXT
- Maximum File Size: 10MB per upload
- Upload Method: Multipart form data via HTTPS
- Security: File type validation, virus scanning integration
- Progress Tracking: Real-time upload progress via WebSocket connection

COM-003: API Interface

- Description: RESTful API for integration with external systems
- Endpoints:
    - POST /parse - Submit resume for parsing
    - GET /results/{job_id} - Retrieve parsing results
    - GET /health - System health check

- Authentication: API key-based authentication
- Rate Limiting: 100 requests per minute per API key
- Response Format: JSON with standardized error codes

COM-004: Email Notification Interface

- Service: SMTP integration for result delivery
- Protocol: SMTP over TLS (port 587)
- Usage: Send parsing results and system notifications
- Format: HTML emails with PDF attachments for detailed reports
- Security: OAuth 2.0 authentication with email service provider

3.5 External System Interfaces

EXT-001: Cloud Storage Interface

- Service: AWS S3 or equivalent cloud storage
- Purpose: Temporary storage of uploaded files during processing
- Protocol: HTTPS REST API
- Security: Server-side encryption, pre-signed URLs for secure access
- Retention: 24-hour automatic deletion policy

EXT-002: Antivirus Scanning Interface

- Service: ClamAV or cloud-based scanning service
- Integration Point: File upload validation
- Protocol: HTTP API calls for real-time scanning
- Response: Binary safe/unsafe status with threat details

4. System features

Each requirement below includes acceptance criteria. IDs follow FR-### for functional and NFR-### for non-functional.

4.1 Resume Upload & Validation

Description: System accepts resumes in supported formats, validates structure, and ensures completeness.

Req ID	Requirement	Type	Priority	Source/Stakeholder	Acceptance criteria / Test case ref	Comments / Dependencies
FR-001	The system shall allow users to upload resumes in .txt, .pdf, or .docx formats and reject unsupported types.	Functional	High	End User / Business	AC-FR-001: Valid formats accepted with confirmation; unsupported rejected with error. Test: TC-FR-001-A	File upload module dependency

FR-002	The system shall flag missing sections (e.g., “Skills”).	Functional	High	End User	AC-FR-002: Missing sections listed in feedback. Test: TC-FR-002-A	Parsing engine
FR-003	The system shall highlight incomplete sections (e.g., “Education” with no details).	Functional	Medium	End user	AC-FR-003: Sections with headers but no content flagged. Test: TC-FR-003-A	Parsing engine

4.2 Report Generation & Privacy

Req ID	Requirement	Type	Priority	Source/Stakeholder	Acceptance criteria / Test case ref	Comments / Dependencies
FR-004	The system shall generate a structured summary of identified sections and allow PDF download.	Functional	High	End User	AC-FR-004: Summary lists found/missing/incomplete sections; downloadable as PDF. Test: TC-FR-004-A	Requires PDF generator
FR-005	The system shall anonymize personal data (email, phone) in reports.	Functional	High	Operations	AC-ATM-F-011: Dispensed amount equals requested; cassette counts decremented. Test: TC-WD-02	Hardware interaction with dispenser
FR-006	The system shall allow batch processing of multiple resumes.	Functional	Medium	Recruiter / Business	AC-FR-006: Multiple files processed per session. Test: TC-FR-006-A	File handler scalability

4.3 Parsing & Skill Extraction

Req ID	Requirement	Type	Priority	Source/Stakeholder	Acceptance Criteria / Test Case Ref	Comments / Dependencies

FR-007	The system shall recognize and extract key skills via keyword matching and synonyms.	Functional	High	End User / Recruiter	AC-FR-007: Extracted skills listed in feedback. Test: TC-FR-007-A	Parsing engine NLP
FR-008	The system shall detect duplicate section headings and merge contents.	Functional	Low	End User	AC-FR-008: Duplicate headers merged. Test: TC-FR-008-A	Parsing engine
FR-009	The system shall detect inconsistent date formats and validate presence/format of contact info.	Functional	High	End User	AC-FR-009: Feedback shows date inconsistencies & invalid/missing contact info. Test: TC-FR-009-A	Validation library

4.4 Resume Quality Analysis

Req ID	Requirement	Type	Priority	Source/Stakeholder	Acceptance Criteria / Test Case Ref	Comments / Dependencies
FR-010	The system shall perform grammar & spelling checks.	Functional	High	End User	AC-FR-010: Feedback shows grammar/spelling errors. Test: TC-FR-010-A	Requires language API
FR-011	The system shall provide a scoring system (e.g., out of 100).	Functional	Low	End User	AC-FR-011: Numerical score shown in report. Test: TC-FR-011-A	Scoring rules configurable
FR-012	The system shall detect employment gaps (>6 months) and prompt user for explanation.	Functional	Medium	Recruiter	AC-FR-012: Report highlights gaps with suggestion. Test: TC-FR-012-A	Requires date parsing

FR-013	The system shall suggest resume templates by industry.	Functional	Medium	End User	AC-FR-013: Links to industry-specific templates in report. Test: TC-FR-013-A	Template library required
FR-014	The system shall allow users to provide a job description and compare against resume.	Functional	High	End User / Recruiter	AC-FR-014: Report shows match % and missing skills. Test: TC-FR-014-A	Requires JD parsing
FR-015	The system shall detect weak/passive wording and suggest stronger action verbs.	Functional	Medium	End User	AC-FR-015: Report highlights weak verbs and suggests alternatives. Test: TC-FR-015-A	Requires NLP
FR-016	The system shall calculate resume length (word count) and flag deviations from standards.	Functional	Medium	End User	AC-FR-016: Report shows word count and warns if outside 400–600 words/page. Test: TC-FR-016-A	Text analysis

5. Non-functional requirements (detailed)

Req ID	**Requirement**	**Category**	**Priority**	**Acceptance Criteria / Measurement**	**Comments / Dependencies**
NFR-001	The system shall process each resume in ≤10s and handle ≥50 concurrent requests.	Performance	High	Load tests confirm ≤10s with 50 users. Test: TC-NFR-001	Requires scalable backend
NFR-002	The system shall delete resumes after	Security/Privacy	High	No files retained post-analysis.	Storage cleanup verification

	processing.			Test: TC-NFR-002	
NFR-003	The system shall encrypt all data in transit via HTTPS.	Security	High	Verified secure communications. Test: TC-NFR-003	SSL/TLS configuration
NFR-004	The system shall display meaningful error messages.	Usability	Medium	All errors descriptive. Test: TC-NFR-004	Error handling framework
NFR-005	The system shall balance critical and appreciative feedback in reports.	UX	High	User feedback shows balanced tone. Test: TC-NFR-005	Content design guideline
NFR-006	The system's feedback must minimize cognitive overload.	UX	High	Usability study confirms digestible output. Test: TC-NFR-006	UX research
NFR-007	The UI and reports shall comply with WCAG 2.1 AA.	Accessibility	High	Accessibility audit passes. Test: TC-NFR-007	Requires accessible UI design

5.1. Security

5.1.1 Security Objectives

Confidentiality – Ensure that all resumes and candidate data are kept private, accessible only to authorized users and not leaked or exposed.

Integrity – Guarantee that uploaded data (resumes, parsed results) cannot be altered, tampered with, or corrupted during storage or transmission.

Availability – Ensure that the system remains accessible and functional for legitimate users without security breaches causing downtime.

Compliance & Data Protection – Adhere to relevant data protection standards (e.g., GDPR, local IT security guidelines), ensuring secure handling and timely deletion of sensitive data.

5.1.2 Security Requirements

Req ID	Requirement	Category	Priority	Acceptance Criteria	Comments / Dependencies
SR-001	The system shall enforce TLS 1.2 or higher for all communications.	Security	High	All data in transit is encrypted.	Requires HTTPS configuration and valid SSL/TLS certificates.
SR-002	The system shall require multi-factor authentication (MFA) for administrator access.	Security	High	Admin accounts cannot log in without MFA.	Integration with MFA provider.
SR-003	The system shall enforce automatic session timeout after 15 minutes of inactivity.	Security	Medium	Idle user sessions are logged out automatically.	Session management in the web application.
SR-004	The system shall delete uploaded resumes immediately after processing.	Privacy/Security	High	No resumes remain on the server once analysis is complete.	Requires secure file cleanup mechanism.

6. Quality Attributes and Acceptance Tests

6.1 Quality Attributes

Quality attributes are the non-functional requirements that define the system's operational excellence. For this project, the key attributes are:

- **Performance:** The system must process resumes (up to 3 pages) in 10 seconds or less, even while handling 50 concurrent user requests. This ensures a fast and responsive user experience.
- **Reliability:** The service is expected to be highly available, with a target of 99.5% uptime per month.
- **Security:** This is a critical attribute with several layers:
  - **Confidentiality:** Uploaded resumes must be automatically deleted after analysis to protect user privacy. All data in transit must be encrypted using TLS 1.2+.
  - **Integrity:** Passwords must be securely hashed and salted using bcrypt to protect user accounts.
  - **Access Control:** Administrative accounts require multi-factor authentication (MFA), and all user sessions will time out after 15 minutes of inactivity to prevent unauthorized access.

- **Usability:** The user experience is paramount. This is addressed by providing clear error messages, balancing positive and critical feedback, designing a report that avoids cognitive overload, and ensuring the entire system is accessible according to WCAG 2.1 AA standards.

6.2 Acceptance Tests

Acceptance tests are user-focused scenarios designed to verify that the system meets its requirements.

Test 1: Flagging Inconsistent Data (for RPRS-F-006)

This test confirms the system can identify formatting inconsistencies.

Scenario: A user uploads a resume containing inconsistent date formats.

- **Given:** A user has a resume with work experience dates written as "Jan 2023 – Dec 2024" and "01/05/2022 - 31/12/2022".
- **When:** The user uploads the resume for analysis.
- **Then:** The system's feedback report must explicitly flag the date format inconsistency to the user.

Test 2: Comparing Resume to a Job Description (for RPRS-F-015)

This test validates one of the core reporting features.

Scenario: A user compares their analyzed resume against a specific job description.

- **Given:** The system has generated a feedback report for the user's resume.
- **When:** The user pastes a job description for a "Software Engineer" into the comparison field.
- **Then:** The report must update to show a match percentage and list key skills from the job description (e.g., "Kubernetes," "Go") that are missing from the resume.

Test 3: Session Timeout Enforcement (for PRJ-SR-003)

This test verifies a critical security requirement.

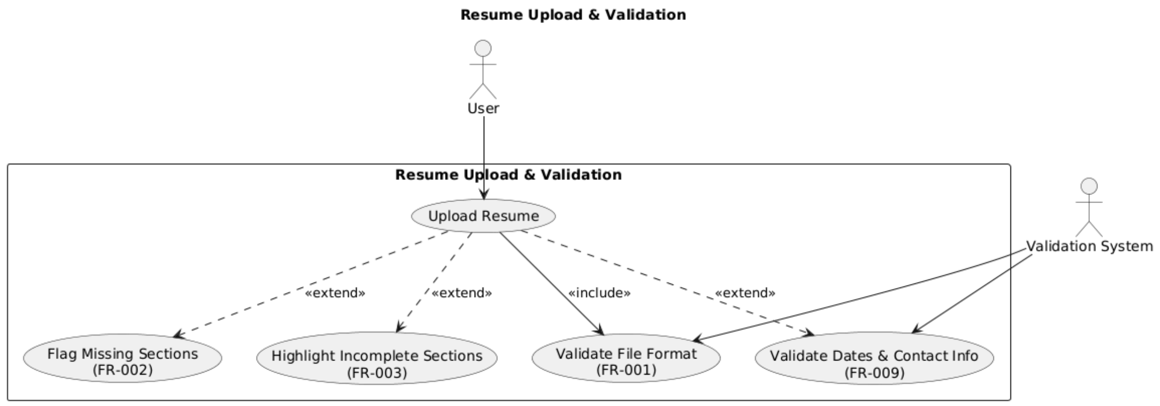
Scenario: A user's inactive session is automatically terminated for security.

- **Given:** A user is logged into their account.
- **When:** The user's browser tab remains idle for more than 15 minutes.
- **Then:** The system must automatically log the user out; any subsequent action must redirect them to the login page.

7. System models and diagrams

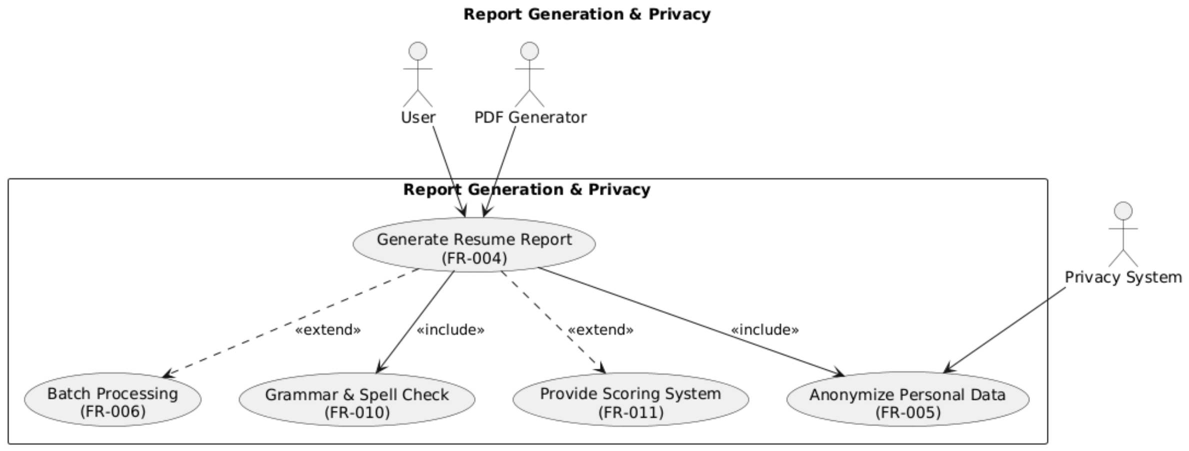
7.1 UML Use-Case diagram

UML 1: Resume Upload and Validation


Resume Upload & Validation
User
Resume Upload & Validation
Upload Resume
Validation System
«extend»
«extend»
«include»
«extend»
Flag Missing Sections (FR-002)
Highlight Incomplete Sections (FR-003)
Validate File Format (FR-001)
Validate Dates & Contact Info (FR-009)

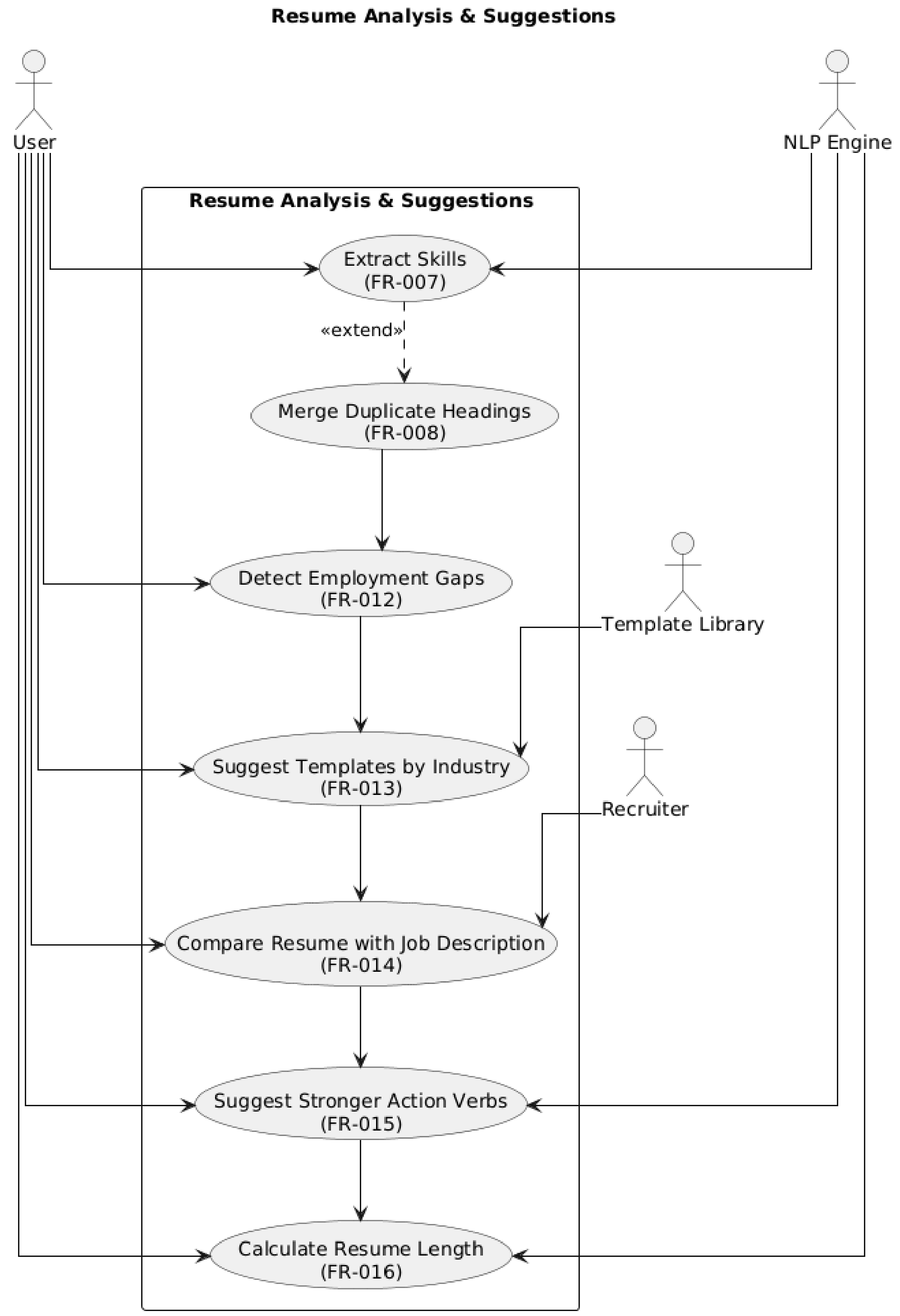
Actors: User, Validation System
Pre-condition: User has a resume ready for upload.
Post-condition: Resume is validated for format, completeness & correctness.
Main Flow:
User uploads resume → System checks format → Flags missing/incomplete sections → Validates dates and contact info.

UML 2: Report Generation & Privacy


Report Generation & Privacy
User
PDF Generator
Report Generation & Privacy
Generate Resume Report (FR-004)
Privacy System
«extend»
«include»
«extend»
«include»
Batch Processing (FR-006)
Grammar & Spell Check (FR-010)
Provide Scoring System (FR-011)
Anonymize Personal Data (FR-005)

Actors: User, PDF Generator, Privacy System
Pre-condition: Resume has been uploaded, parsed, and validated.
Post-condition: A final report is generated in PDF format, with anonymized personal data and optional scoring.
Main Flow:
User requests a resume report → System generates a PDF → Privacy system anonymizes data → Final report with grammar check & optional scoring is shared.

UML 3: Resume analysis and Sugeestions



Actors: User, Recruiter, NLP Engine, Template Library
Pre-condition: Resume is uploaded and validated.
Post-condition: Suggestions and insights are provided to improve the resume.
Main Flow:
System extracts skills → Detects duplicate headings → Identifies employment gaps → Suggests industry templates → Compares with job description → Suggests stronger action verbs → Calculates resume length.

8. Requirements Traceability Matrix (RTM)

Req ID	Requirement (shall...)	Type	Priority	Source / Stakeholder	Acceptance Criteria / Test Case Ref
---	**Functional Requirements: Resume Upload & Validation (Section 4.1**) ---				
RPRS-F-001	The system shall allow users to upload resumes in .pdf, .docx, and .txt formats and reject unsupported types.	Functional	High	User	AC-RPRS-F-001: Valid file uploads succeed; unsupported file (.jpg) rejected. Test: TC-UP-01
RPRS-F-002	The system shall anonymize sensitive personal data (email, phone) in reports.	Functional	Medium	Privacy/User	AC-RPRS-F-002: Report masks contact info. Test: TC-UP-02
---	**Functional Requirements: Resume Parsing & Section Identification (Section 4.2**) ---				
RPRS-F-003	The system shall parse resumes to identify standard sections (Education, Skills, Experience, Projects).	Functional	High	User/Recruiter	AC-RPRS-F-003: Sections detected. Test: TC-PA-01
RPRS-F-004	The system shall recognize and extract key skills using keyword matching and synonyms.	Functional	High	Recruiter	AC-RPRS-F-004: “Python, Java” extracted. Test: TC-PA-02
RPRS-F-005	The system shall detect duplicate section headers and merge their contents.	Functional	Low	User	AC-RPRS-F-005: Duplicate “Skills” merged. Test: TC-PA-03
RPRS-F-006	The system shall detect and flag inconsistent date formats and missing/incorrect contact details.	Functional	High	Recruiter	AC-RPRS-F-006: Flags “01/01/23” vs “Jan 2023”. Test: TC-PA-04
---	**Functional Requirements: Feedback & Suggestions (Section 4.3)** ---				
RPRS-F-007	The system shall flag missing sections (e.g., Skills, Education).	Functional	High	User	AC-RPRS-F-007: Missing flagged. Test: TC-FB-01
RPRS-F-008	The system shall highlight incomplete sections.	Functional	Medium	User	AC-RPRS-F-008: Empty “Education” flagged. Test: TC-FB-02
RPRS-F-009	The system shall analyze work experience language and suggest stronger action verbs.	Functional	Medium	User	AC-RPRS-F-009: Flags “responsible for” → suggests “led.” Test: TC-FB-03

RPRS-F-010	The system shall calculate resume length and flag if too long or short.	Functional	Medium	Recruiter	AC-RPRS-F-010: Outside 400–600 words flagged. Test: TC-FB-04
RPRS-F-011	The system shall suggest industry-specific templates.	Functional	Medium	User	AC-RPRS-F-011: Suggests IT vs HR resume template. Test: TC-FB-05
---	**Functional Requirements: Reporting (Section 4.4) ---**				
RPRS-F-012	The system shall generate a structured feedback report listing identified, missing, and incomplete sections.	Functional	High	User	AC-RPRS-F-012: Report generated. Test: TC-REP-01
RPRS-F-013	The system shall allow users to download the feedback report as PDF.	Functional	Medium	User	AC-RPRS-F-013: Report saved as PDF. Test: TC-REP-02
RPRS-F-014	The system shall provide a numerical resume quality score (0–100).	Functional	Low	User	AC-RPRS-F-014: Score shown in report. Test: TC-REP-03
RPRS-F-015	The system shall allow users to compare resumes with job descriptions.	Functional	High	User	AC-RPRS-F-015: Match % + missing keywords shown. Test: TC-REP-04
RPRS-F-016	The system shall allow batch analysis of multiple resumes at once.	Functional	Medium	Recruiter	AC-RPRS-F-016: Multiple files processed in one run. Test: TC-REP-05
---	**Non-Functional & Security Requirements (Section 5.0)** ---				
RPRS-NF-001	The system shall process resumes (≤3 pages) in ≤10s and handle 50 concurrent requests.	Performance	High	User/Admin	Load test shows ≤10s at 50 users.
RPRS-NF-002	The system shall provide 99.5% uptime monthly.	Reliability	High	Admin	Monitoring logs show ≥99.5%.
RPRS-NF-003	Uploaded resumes shall be auto-deleted post analysis.	Security	High	User/Privacy	No resumes stored after the session.
RPRS-NF-004	The system shall display clear error messages.	Usability	Medium	User	Errors are descriptive.
RPRS-NF-005	Reports/UI shall balance positive + critical feedback.	Usability	High	User	User tests confirm motivation.
RPRS-NF-006	Feedback design shall minimize cognitive overload.	Usability	High	User	Reports confirmed digestible.
RPRS-NF-007	The system shall comply with WCAG 2.1 AA accessibility.	Usability	High	User	Audit passes keyboard + screen reader.
PRJ-SR-001	Enforce TLS 1.2+ encryption.	Security	High	Security Officer	Verified by penetration test (TC-SR-01).
PRJ-SR-002	Require MFA for admin accounts.	Security	High	Admin	MFA login succeeds, non-MFA fails (TC-SR-02).
PRJ-SR-003	Enforce session timeout after 15 min inactivity.	Security	Medium	User	Idle session auto logout (TC-SR-03).

PRJ-SR-004	Hash and salt passwords (bcrypt).	Security	High	Security Officer	Verified in code review (TC-SR-04).
PRJ-SR-005	Maintain audit logs of uploads/downloads/logins.	Security	High	Admin	Logs retrievable, tamper-proof (TC-SR-05).