Software Requirement Specification

for

Campus Placer

Deeraj S
2217077
MCA
Aloysius Institute of Management and IT
Mangalore
30-05-2024

Under the guidance of

Mr. Shine
MCA Department
Aloysius Institute of Management and IT
Mangalore

Submitted to

ST ALOYSIUS INSTITUTE OF MANAGEMENT AND INFORMATION TECHNOLOGY (AIMIT)
ST ALOYSIUS COLLEGE (AUTONOMOUS)
MANGALORE, KARNATAKA

1. Introduction

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) document is to detail the functional and non-functional requirements for the development of "Campus Placer," a modern solution aimed at optimizing college placement processes. It includes features for student job applications, job listings, administrative management, and communication between stakeholders.

1.2 Document Conventions

This document follows standard conventions for specifying software requirements, including priority levels for requirements and formatting guidelines for clarity and consistency.

1.3 Intended Audience and Reading Suggestions

The intended audience for this document includes developers, project managers, quality assurance testers, system architects, and stakeholders involved in the development and deployment of the "Campus Placer" system. The document is organized to provide an overview followed by detailed sections tailored to different reader types, with a suggested sequence for reading based on relevance to each role.

1.4 Project Scope

The "Campus Placer" system is a software solution designed to facilitate campus placement activities within our college. It aims to provide students, administrators, HODs, and placement officers with tools to manage job listings, applications, and communication-related to campus placements. The scope of this SRS document covers the specific functionalities and requirements of the "Campus Placer" system.

1.5 References

References to related documents, standards, or external resources that influenced the development of the "Campus Placer" system are listed here. This includes user interface style guides, system requirements specifications, use case documents, and any other relevant materials.

IEEE Std 830-1998: IEEE Standard for Software Requirements Specifications
Web Address: IEEE Std 830-1998
MongoDB Documentation
Web Address: MongoDB Documentation
React Documentation
Web Address: React Documentation
Express.js Documentation
Web Address: Express.js Documentation
Node.js Documentation
Web Address: Node.js Documentation
JSON Web Token (JWT) Documentation
Web Address: JWT Documentation

2. Overall Description

2.1 Product Perspective

The "Campus Placer" system is a standalone application that integrates with existing college systems (such as student databases) to provide a comprehensive platform for managing campus placements. It is designed to complement and enhance existing processes without replacing core systems.

2.2 Product Features

Key features of the "**Campus Placer**" system include:

- Job listing management
- Application tracking and status updates
- User profile management (students, administrators, HODs, placement officers)
- Insights and analytics for decision-making
- Communication tools for stakeholders

2.3 User Classes and Characteristics

Here are four user classes (users) and their characteristics.

Students: Students are job seekers with varying technical skills, and need easy access to job listings, application tracking, and profile management within the system.

Administrators: Administrators are college staff proficient in system management, and responsible for user account administration, system configuration, and generating reports.

Heads of Departments (HODs): HODs are faculty members overseeing academic departments, requiring insights into departmental placement trends and student profiles for effective decision-making.

Placement Officers: Placement Officers are experienced staff managing campus placements, responsible for creating job postings, reviewing applications, and communicating updates to students.

2.4 Operating Environment

The "Campus Placer" system will operate in a web-based environment, accessible via standard web browsers (Chrome, Firefox, Safari) on desktop and mobile devices. The system will be compatible with commonly used operating systems (Windows, macOS, Linux) and will leverage cloud infrastructure for scalability and availability.

2.5 Design and Implementation Constraints

Constraints include hardware compatibility, security considerations, integration with existing college systems, and adherence to data protection regulations.

2.6 User Documentation

User documentation components such as user manuals, online help resources, and tutorials will accompany the system to assist users in navigating and utilizing its functionalities.

2.7 Assumptions and Dependencies

Assumptions made during the requirements gathering phase, such as availability of internet connectivity for system operation, and dependencies on external systems (e.g., student databases, authentication services) are documented to clarify project constraints and risks.

3. System Features

3.1 Job Viewing and Application (Student Module)

3.1.1 Description and Priority

The "Job Viewing and Application" feature enables students to browse available job listings, apply for positions of interest, and track the status of their applications. This feature is of high priority as it is central to the core functionality of the "Campus Placer" system, facilitating student engagement with placement opportunities.

3.1.2 Stimulus/Response Sequences

Stimulus:
A user (student) accesses the job listing interface within the system.
Response:

- The system presents a list of available job opportunities with relevant details (e.g., job title, company, description, requirements).
- The user selects a job listing to view more details.
- The user decides to apply for the selected job by submitting an application through the system.
- Upon successful application submission, the system updates the application status to "Submitted."

3.1.3 Functional Requirements

Secure Login/Authentication: Users must authenticate themselves securely to access the job viewing and application features. Authentication methods should include username/password login or integration with college/university authentication systems (e.g., Single Sign-On).

Job Listing Display: The system should display a comprehensive list of job opportunities available for students. Job listings should include relevant information such as job title, company, location, description, eligibility criteria, and application deadlines.

Application Submission: Students should be able to submit job applications directly through the system. The application form should capture the necessary details (e.g., resume upload, cover letter, additional documents) as required by the job posting.

Status Tracking: After submission, students should be able to track the status of their applications (e.g., "Submitted," "Under Review," "Shortlisted," "Rejected," "Selected").

The system should update application statuses in real time based on actions taken by placement officers or recruiters.

3.2 User Account Management (Admin Module)

3.2.1 Description and Priority

The "User Account Management" feature within the Admin Module enables administrators to manage user accounts for students, HODs (Heads of Departments), and placement officers. This feature is of high priority as it ensures proper access control and user management within the "Campus Placer" system.

3.2.2 Stimulus/Response Sequences

Stimulus:
The administrator accesses the user management interface within the system.
Response:
The system presents a dashboard displaying user account management options.

- The administrator selects the "Create New User" or "Manage Existing Users" functionalities.
- Upon selecting "Create New User," the system prompts for user details (e.g., username, role, department).
- After submitting user details, the system creates a new user account and updates the user list.
- For "Manage Existing Users," the system allows editing or deactivating user accounts based on administrator actions.

3.2.3 Functional Requirements

User Account Creation and Deletion: Administrators should be able to create new user accounts for students, HODs, and placement officers. Ability to deactivate or delete user accounts as needed (e.g., upon graduation or change in roles).

User Profile Management: Administrators can view and edit user profiles, including personal details, contact information, and department affiliations.

Access Control and Permissions: Define and manage user roles and permissions (e.g., admin, standard user) for accessing different system features.

User Activity Monitoring: Track user activity logs (e.g., login/logout timestamps, actions performed) for audit and security purposes.

3.3 Departmental Insights (HOD Module)

3.3.1 Description and Priority

The "Departmental Insights" feature within the HOD Module provides department heads with valuable insights into student profiles, job applications, and placement trends within their

respective departments. This feature is of medium to high priority, supporting informed decision-making and department-specific placement strategies.

3.3.2 Stimulus/Response Sequences

Stimulus:
HOD accesses the departmental insights dashboard.
Response:

- The system displays analytics and reports related to student profiles, job applications, and placement trends within the department.
- The HOD selects specific metrics or filters to view detailed insights (e.g., student GPA distribution, top recruiters).
- The system generates customized reports based on HOD's selections and preferences.
- HOD can export or share departmental insights for collaborative decision-making with faculty and placement officers.

3.3.3 Functional Requirements

Student Profile Analytics: HODs should be able to view aggregated data on student profiles, including academic performance, skills, and career interests.

Job Application Tracking: Access department-specific job application data, including application statuses and historical trends.

Placement Trend Analysis: Generate reports and analytics on placement trends within the department, helping HODs identify areas for improvement and optimization.

3.4 Job Posting Management (Placement Officer Module)

3.4.1 Description and Priority

The "Job Posting Management" feature within the Placement Officer Module empowers placement officers to create, manage, and track job postings on the platform. This feature is of high priority as it directly facilitates the recruitment process and communication with students.

3.4.2 Stimulus/Response Sequences

Stimulus:
The placement officer accesses the job posting management interface.
Response:

- The system presents a dashboard displaying current job postings and recruitment activities.
- Placement officer selects "Create New Job Posting" or "Manage Existing Postings" functionalities.
- For "Create New Job Posting," the system prompts for job details (e.g., job title, description, application deadline).

- After submitting job details, the system adds the new job posting to the platform and notifies students.
- Placement officer can view and manage applications, update job statuses, and communicate with applicants through the system.

3.4.3 Functional Requirements

Job Creation and Editing: Placement officers can create new job postings with detailed descriptions, requirements, and application deadlines. Ability to edit or update existing job postings as needed (e.g., extend deadlines, modify requirements).

Application Review and Communication: Review job applications submitted by students, shortlist candidates, and communicate updates or interview schedules.

Real-time Job Status Updates: Track the status of posted jobs (e.g., open, closed, filled) and update job statuses based on recruitment progress.

4. External Interface Requirements

4.1 User Interfaces

The user interfaces of "Campus Placer" will adhere to modern GUI (Graphical User Interface) standards, ensuring an intuitive and user-friendly experience for students, administrators, HODs, and placement officers. Each interface component, including job listings, application forms, user profiles, and navigation menus, will feature clear layouts and consistent design elements. Standard buttons such as Home, Job Listings, and Profile will be prominently displayed to facilitate easy navigation and access to key features. Screen layouts will be optimized for various device sizes, from desktops to mobile devices, ensuring a responsive and accessible interface across platforms.

4.2 Hardware Interfaces

"Campus Placer" will be accessible through standard hardware devices such as desktop computers, laptops, tablets, and smartphones. The software will leverage web-based technologies and will not require specific hardware components or physical connections beyond standard computing devices. Users will interact with the application using web browsers, ensuring compatibility with different hardware configurations and operating systems without the need for specialized equipment or peripherals.

4.3 Software Interfaces

The software components of "Campus Placer" will interface with several technologies and tools to deliver its functionality. The system will utilize MongoDB as the database for storing user data, job listings, and application records. It will be compatible with modern web browsers on Windows, macOS, and Linux operating systems. The development will leverage the MERN stack (MongoDB, Express.js, React.js, Node.js) for web application development, enabling efficient data retrieval and real-time updates. Interaction with external services and libraries,

such as authentication frameworks like Passport.js, will be facilitated through RESTful APIs (Application Programming Interfaces) to ensure seamless integration and data exchange.

4.4 Communications Interfaces

Communication functions within "Campus Placer" will encompass email notifications, HTTP(S) protocols for secure data transmission, and optimized data transfer rates. The system will utilize email notifications to update users on application statuses and system announcements. All data exchanges will adhere to HTTP(S) standards, ensuring secure communication between users and the system. Data transfer rates will be optimized to minimize loading times for job listings, application forms, and user profiles, enhancing the overall user experience. Security measures such as encryption (e.g., HTTPS) will be implemented to protect sensitive user data during transmission and communication.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The "Campus Placer" system must perform efficiently under varying loads. The system should be capable of handling concurrent access by up to 500 users without performance degradation. Response time for user actions, such as viewing job listings or submitting applications, should not exceed 2 seconds. The system should also support scalability to accommodate future increases in user numbers and data volume, ensuring that performance remains optimal. Real-time updates, such as job posting notifications and application status changes, must be processed and reflected within 1 second.

5.2 Safety Requirements

The "Campus Placer" system must safeguard against data loss and ensure data integrity. Regular backups must be scheduled daily, with redundant storage solutions to prevent data loss. Access control mechanisms should be implemented to prevent unauthorized data modification or deletion. The system should have safeguards to handle data input errors and recover gracefully from unexpected failures, ensuring no data corruption occurs. Compliance with safety regulations and standards specific to educational institutions is mandatory.

5.3 Security Requirements

Security is paramount for the "Campus Placer" system. User authentication must be robust, using strong passwords and, where possible, two-factor authentication. All data, particularly sensitive information like student profiles and job applications, must be encrypted both in transit and at rest. Access to different system functionalities should be role-based, ensuring users can only access relevant features. The system must be protected against common security threats, such as SQL injection, cross-site scripting (XSS), and cross-site request forgery (CSRF). Regular security audits and updates must be performed to address potential vulnerabilities.

5.4 Software Quality Attributes

The "Campus Placer" system must exhibit high adaptability, allowing easy integration with other college systems. Availability should be maintained at 99.9%, minimizing downtime. Correctness is essential; all functionalities must perform as specified without errors. Flexibility is necessary to accommodate changing requirements and user needs. Interoperability with existing college databases and systems is crucial. Maintainability should be ensured through clear documentation and modular design, facilitating easy updates and bug fixes. Portability across various devices and platforms, reliability in all functionalities, reusability of components, robustness to handle unexpected inputs gracefully, testability through automated testing, and overall usability are critical quality attributes.

6. Other Requirements

The "Campus Placer" system must support multiple languages to cater to a diverse user base, enhancing accessibility. Legal compliance with data protection laws, such as GDPR, must be ensured. Database requirements include using MongoDB for its scalability and flexibility, with structured schema to manage various data types. The system should aim for the reusability of code components to facilitate future developments and integrations.

7. Analysis Models

For the "Campus Placer" project, various analysis models will be used to ensure comprehensive understanding and design:

Data Flow Diagrams (DFD): To represent the flow of data within the system, detailing how data is processed at different stages.

Entity-Relationship Diagrams (ERD): To illustrate the database structure, showing the entities, relationships, and attributes involved.

Appendix A: Glossary

MERN Stack: A JavaScript stack used for developing web applications, consisting of MongoDB, Express.js, React, and Node.js.
HOD: Head of Department, responsible for overseeing departmental activities.
Placement Officer: College staff responsible for managing campus placements.
SSO: Single Sign-On, an authentication process that allows a user to access multiple applications with one set of login credentials.
CSRF: Cross-Site Request Forgery, a type of security exploit.
XSS: Cross-site scripting, a security vulnerability typically found in web applications.

Appendix B: Issues List

Issue 1: Integration with existing college databases is pending final approval from the IT department.
Issue 2: The decision on the specific encryption method to be used for data at rest is still pending.

Issue 3: User feedback on the job application form design is awaited to finalize the UI/UX design.
Issue 4: Determination of backup frequency and storage solutions is under review.
Issue 5: Finalization of role-based access control specifics is in progress.