Software Requirements Specification

for

Elective Course/Project Registration System

Version 1.0

Prepared by

Group Name: SmartSelect

Kushal Devineni	**SE22UARI043**	se22uari043@mahindrauniversity.edu.in
Akshith Reddy	**SE22UARI013**	se22uari013@mahindrauniversity.edu.in
Dheeraj Madgula	**SE22UARI088**	se22uari088@mahindrauniversity.edu.in
N Prajay	**SE22UARI105**	se22uari105@mahindrauniversity.edu.in

Instructor: Vijay Rao

Course: Software Engineering

Lab Section: 1st Floor

Teaching Assistant: Swapna Mam

Date: 10/03/2025

Contents

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
Draft Type Full Name and Number		Information about the revision. This table does not need to be filled in whenever a document is touched, only when the version is being	00/00/00

1. Introduction

<TO DO: Please provide a brief introduction to your project and a brief overview of what the reader will find in this section.>

1.1. Document Purpose

This **Software Requirements Specification (SRS)** document defines the requirements for the **Elective Course/Project Registration System (Version 1.0)**. The system is designed to simplify and automate the process of selecting and registering for elective courses and academic projects. It ensures a seamless experience for students while providing faculty and administrators with tools to manage course offerings and enrollments efficiently.

This SRS covers the **entire system**, detailing its functionalities for **students, faculty, and administrators**. It serves as a reference for developers, testers, and stakeholders to ensure that the software meets all specified requirements, including authentication, course catalog management, registration automation, waitlist handling, faculty approvals, admin controls, notifications, and reporting.

1.2. Product Scope

The **Elective Course/Project Registration System** is a web-based platform designed to streamline the selection and enrollment process for elective courses and academic projects. It enables students to browse available courses, check prerequisites, and register based on eligibility and seat availability. Faculty members can manage enrollments, approve project registrations, and oversee student selections, while administrators have control over course offerings and registration policies.

The system aims to **enhance transparency, reduce manual effort, and improve accessibility** in the registration process. Key objectives include **automating enrollment, managing waitlists, sending real-time notifications, and generating insightful reports** to track student preferences and course demand. By digitizing and optimizing the registration workflow, the system ensures efficiency and fairness in elective and project allocation.

1.3. Intended Audience and Document Overview

This document is intended for **software developers, faculty members, university administrators, and testers** involved in the development and deployment of the **Elective Course/Project Registration System**. Faculty members and administrators will use it to understand the system's functionalities and ensure that it meets academic requirements,

while developers and testers will use it to implement and validate the system according to the specified requirements.

The document is structured to provide a clear understanding of the system, starting with an **introduction to its purpose and scope**, followed by **detailed functional and non-functional requirements, interface descriptions, use case models, and design constraints**. Readers should begin with the **introduction** and **overall description** before moving on to specific requirements relevant to their roles, ensuring a smooth and systematic understanding of the system.

1.4. Definitions, Acronyms and Abbreviations

Below is a list of key terms, abbreviations, and acronyms used in this document:

- **Admin** – System Administrator responsible for managing courses, users, and registrations.
- **API (Application Programming Interface)** – A set of protocols and tools that allow different software components to communicate.
- **DB (Database)** – A structured collection of data used for storing course, student, and registration details.
- **Enrollment** – The process of a student registering for an elective course or academic project.
- **Faculty Dashboard** – A system interface for faculty members to manage student enrollments and approve project registrations.
- **SRS (Software Requirements Specification)** – A document that outlines the functional and non-functional requirements of a software system.
- **UI (User Interface)** – The part of the system that users interact with, including screens, menus, and input fields.
- **Waitlist** – A feature that automatically places students in a queue when a course/ project reaches maximum capacity.

1.5. Document Conventions

This **Software Requirements Specification (SRS)** document follows standard IEEE formatting guidelines to ensure clarity and consistency. The following conventions are used throughout the document:

1.5.1 Formatting Conventions

- The document uses **Arial font, size 11**, with **single-line spacing** and **1-inch margins** on all sides.

- **Section headings** are bolded for emphasis, while **subsections** are also bolded but in a smaller font size.
- Key terms and important points may be **bolded** for emphasis.
- Comments or placeholders for future modifications are written in *italics*.

1.5.2 Naming Conventions

- All system components, such as **Admin Dashboard, Student Portal, Faculty Dashboard**, are capitalized for clarity.
- Functional requirements are labeled as **F1, F2, F3, etc.**, while non-functional requirements are labeled as **NF1, NF2, NF3, etc.**
- Use case identifiers follow the format **U1, U2, U3, etc.**

These conventions ensure that the document remains structured, easy to read, and consistent for all stakeholders.

1.6. References and Acknowledgments

This **SRS** document is based on the **Statement of Work (SOW)** provided by **Mahindra University** for the **Elective Course/Project Registration System**. It follows industry standards to ensure clarity and consistency.

Key references include:

- **IEEE Std 830-1998** – Software Requirements Specification guidelines.
- **University Academic Policies** – Rules for elective and project registration.
- **Database Management Standards** – Best practices for handling student records securely.
- **UI Guidelines** – Ensuring a user-friendly and accessible interface.

2. Overall Description

2.1. Product Overview

The **Elective Course/Project Registration System** is a **new, self-contained web-based platform** designed to automate and simplify the process of elective and academic project registration at **Mahindra University**. Currently, the registration process involves **manual paperwork, email approvals, and administrative intervention**, leading to inefficiencies and delays. This system **eliminates manual dependencies** by providing a **centralized, automated platform** for students, faculty, and administrators.

The system comprises several interconnected modules:

- **Student Portal** – Allows students to view electives, check prerequisites, and register.
- **Faculty Dashboard** – Enables faculty to approve/disapprove project enrollments.
- **Admin Panel** – Provides administrators control over course creation, enrollment limits, and waitlists.
- **Notification System** – Sends automated emails/SMS for registration updates.

This system integrates with the university's **existing academic database** to ensure real-time updates on course availability and enrollment status.

2.2. Product Functionality-

The **Elective Course/Project Registration System** provides the following key functionalities:

- **User Authentication** – Secure login for students, faculty, and administrators.
- **Course/Project Browsing** – Students can view available electives and projects with descriptions and prerequisites.
- **Automated Registration** – Students can enroll in electives or projects based on eligibility and seat availability.
- **Waitlist Management** – Automatically places students on a waitlist if a course/project is full.
- **Faculty Approval** – Faculty can approve or reject student registrations for academic projects.
- **Course Management** – Administrators can add, modify, or remove courses and manage enrollment limits.
- **Notification System** – Sends email/SMS alerts for registration confirmations, deadlines, and updates.
- **Reporting & Analytics** – Generates reports on enrollment trends and student preferences.

2.3. Design and Implementation Constraints

The **Elective Course/Project Registration System** must adhere to the following constraints:

- **Software Design Methodology** – The system will be designed using the **COMET (Collaborative Object Modeling and Enterprise Transformation) method**, ensuring a structured approach to software development.
- **Modeling Standards** – **UML (Unified Modeling Language)** will be used for system design, including use case diagrams, sequence diagrams, and class diagrams.
- **Technology Stack** – The system will be developed as a **web-based application** using **HTML, CSS, JavaScript (frontend), Python/Java (backend), and MySQL/PostgreSQL (database)**.
- **Hardware Limitations** – The system must be optimized to handle **peak loads during registration periods**, ensuring fast response times without exceeding **server memory or processing limitations**.
- **Database Integration** – The system must be compatible with the **university's academic database** to retrieve and update course and student information.
- **Security Considerations** – The system must implement **role-based access control (RBAC)**, **data encryption**, and **secure login authentication (OAuth or LDAP)** to protect sensitive student and faculty data.

- **Communication Protocols** – **HTTPS** will be enforced for secure communication, and **email/SMS notifications** will be sent using third-party APIs.
- **Parallel Operations** – The system should support **multiple concurrent users** without performance degradation, ensuring a seamless experience for students, faculty, and administrators.

These constraints will guide development to ensure **scalability, security, and efficiency** while meeting the university's requirements.

2.4. Assumptions and Dependencies

The development and functionality of the **Elective Course/Project Registration System** rely on the following key assumptions and dependencies:

- **Student and Faculty Data Availability** – The system assumes that accurate and updated student, faculty, and course data will be provided by the university's academic database.
- **Stable Internet Connectivity** – Users must have a reliable internet connection to access the web-based platform seamlessly.
- **Institutional Policy Compliance** – The system will operate under predefined university rules regarding elective selection, enrollment limits, and faculty approvals.
- **Third-Party Service Integration** – The system depends on external **email/SMS notification services** for communication, which may affect performance if unavailable.
- **Security Infrastructure** – It is assumed that the university will provide a **secure server environment** with proper authentication mechanisms to host the system.
- **Scalability Requirements** – The system is expected to handle a predefined number of concurrent users during peak registration periods; if this number exceeds expectations, performance optimizations may be required.

If any of these assumptions change, the system's design and implementation may need to be adjusted accordingly.

3. Specific Requirements

3.1. External Interface Requirement

The **Elective Course/Project Registration System** is a **web-based platform** with an intuitive and structured user interface (UI) designed for **students, faculty, and administrators**. The key UI elements include:

- **Login Page:** Users (students, faculty, admins) must authenticate using credentials before accessing the system.
- **Dashboard:** Displays an overview of available electives/projects for students, pending approvals for faculty, and course management options for admins.
- **Navigation Menu:** Located on the sidebar, it includes links to **Dashboard, My Courses, Notifications, and Profile** for easy access.
- **Student Interface:** Allows students to **browse courses, check prerequisites, register, and view waitlist status** with interactive filters.
- **Faculty Interface:** Faculty can **approve/reject student project enrollments** and track student registrations.
- **Admin Panel:** Enables admins to **add, modify, or remove courses, manage enrollment limits, and handle waitlists**.
- **Notifications System:** Pop-up alerts and emails inform users about **registration confirmations, deadlines, and waitlist updates**.

The UI follows a **clean and responsive design**, ensuring accessibility across **desktops, laptops, and tablets**.

3.1.1. Hardware Interfaces

The **Elective Course/Project Registration System** is a **web-based application**, meaning it primarily interacts with **computers, servers, and networking devices** rather than physical sensors. Below are the key hardware interfaces:

- **User Devices:** The system supports access via **desktop computers, laptops, tablets, and smartphones** through a web browser.
- **Servers:** The system relies on a **dedicated or cloud-based server** to host the database and web application, ensuring data storage and processing.
- **Database Server:** Stores and manages **student, course, and registration data**, ensuring secure access and retrieval.

- **Networking Infrastructure:** Requires **stable internet connectivity** to handle multiple concurrent user requests efficiently.
- **Notification System (Optional Hardware):** If integrated with **SMS gateways or email servers**, the system may interact with communication hardware to send real-time notifications.

3.1.2. Software Interfaces

The **Elective Course/Project Registration System** interacts with various software components to ensure smooth functionality. The key software interfaces include:

- **Web Browser Interface:** The system is accessible through modern web browsers such as **Google Chrome, Mozilla Firefox, Microsoft Edge, and Safari** for students, faculty, and administrators.
- **Database Management System (DBMS):** The system interacts with a **relational database (MySQL or PostgreSQL)** to store and manage student, course, and registration records securely.
- **Authentication System:** Supports **OAuth, LDAP, or institutional login systems** to verify user credentials and enforce role-based access control.
- **Notification Services:** The system integrates with **email servers (SMTP)** and **SMS APIs (Twilio, Firebase Cloud Messaging)** to send registration confirmations and reminders.
- **University Academic System (If Applicable):** Can be integrated with the university's **existing student information system (SIS)** to fetch student eligibility details and update course enrollments.

Since this is a **web-based application**, it does not have a dedicated mobile app at the moment, but it can be extended to support one in the future.

3.2. Functional Requirements

The **Elective Course/Project Registration System** must fulfill the following functional requirements to ensure a seamless user experience:

3.2.1 User Authentication and Access Control

- **F1**: The system shall allow **students, faculty, and administrators** to log in using secure credentials.
- **F2**: The system shall enforce **role-based access control (RBAC)** to restrict functionalities based on user roles.

3.2.2 Course/Project Browsing and Selection

- **F3**: Students shall be able to **view available elective courses and academic projects** with details such as prerequisites, credit hours, and seat availability.
- **F4**: The system shall provide **filtering and sorting options** to help students find suitable courses/projects.

3.2.3 Registration and Enrollment Management

- **F5**: The system shall allow students to **register for courses/projects** based on eligibility and seat availability.
- **F6**: The system shall **automatically place students on a waitlist** if a course/project is full.
- **F7**: The system shall allow students to **drop a registered course/project** before the registration deadline.

3.2.4 Faculty Approval and Management

- **F8**: Faculty members shall be able to **approve or reject project enrollments** from students.
- **F9**: The system shall allow faculty to **view the list of students enrolled** in their projects.

3.2.5 Administrative Controls

- **F10**: Administrators shall be able to **add, modify, or remove elective courses/projects** from the system.
- **F11**: The system shall allow administrators to **set enrollment limits and waitlist capacities** for courses.
- **F12**: Administrators shall be able to **view reports on student registration trends and course demand**.

3.2.6 Notifications and Alerts

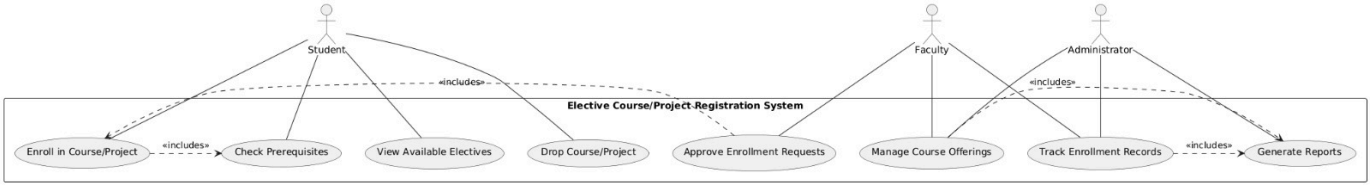
- **F13**: The system shall send **email/SMS notifications** to students upon successful registration or waitlisting.
- **F14**: The system shall notify faculty when a student applies for a project that requires approval.
- **F15**: The system shall provide reminders for **registration deadlines and seat availability updates**.

3.2.7 Security and Compliance

- **F16**: The system shall encrypt **user passwords and sensitive data** to ensure security.
- **F17**: The system shall log **all user activities** for auditing purposes.

These requirements ensure that the system is **efficient, secure, and user-friendly**, meeting the needs of students, faculty, and administrators.

3.3. Use Case Model



4. Other Non-functional Requirements

4.1. Performance Requirements

The **Elective Course/Project Registration System** must meet the following performance requirements to ensure **efficiency, scalability, and responsiveness**:

- **P1. The system shall load the login page within 3 seconds** under normal network conditions.
- **P2. Course search and filtering operations shall return results within 2 seconds** to ensure a smooth user experience.
- **P3. The system shall process student course registration requests within 5 seconds**, including eligibility checks and seat availability validation.
- **P4. The system shall support at least 500 concurrent users** during peak registration periods without performance degradation.

- **P5. Notifications (email/SMS alerts) shall be sent within 10 seconds** after a successful course registration or waitlist update.
- **P6. The database shall handle at least 10,000 records for students, faculty, and courses** while maintaining fast query response times.
- **P7. Faculty project approval decisions shall be updated in the system within 3 seconds**, ensuring real-time updates.
- **P8. Admin modifications to courses (add/edit/remove) shall reflect in the student portal within 5 seconds** to maintain accurate data synchronization.

These performance criteria ensure **fast, reliable, and efficient** system functionality, even during peak usage periods.

4.2. Safety and Security Requirements

The **Elective Course/Project Registration System** must ensure data integrity, user privacy, and protection against unauthorized access. Below are the key safety and security requirements:

4.2.1 Data Protection & Privacy

- **S1. All user data (students, faculty, administrators) shall be encrypted using AES-256** to prevent unauthorized access.
- **S2. The system shall comply with university data privacy policies** to protect student academic records.
- **S3. Personal information (e.g., student IDs, contact details) shall only be visible to authorized users** based on role-based access control (RBAC).

4.2.2 Authentication & Access Control

- **S4. The system shall require multi-factor authentication (MFA) for administrators and faculty** to enhance security.
- **S5. Students, faculty, and administrators shall have distinct roles with limited access to features** based on their permissions.
- **S6. After three failed login attempts, the system shall lock the user account for 10 minutes** to prevent brute-force attacks.

4.2.3 Network & Communication Security

- **S7. All data exchanges shall occur over a secure HTTPS connection** to prevent interception by third parties.
- **S8. The system shall automatically log out inactive users after 15 minutes** to prevent unauthorized access.
- **S9. Email and SMS notifications shall be sent using encrypted communication protocols** to protect sensitive information.

4.2.4 System Reliability & Compliance

- **S10. The system shall maintain daily database backups** to prevent data loss in case of system failure.
- **S11. A log of all system transactions (registrations, approvals, modifications) shall be maintained** for auditing purposes.
- **S12. The system shall comply with **university IT security policies and applicable data protection regulations**.

These requirements ensure the **safety, security, and reliability** of the system while protecting user data from threats.

-

4.3. Software Quality Attributes

The **Elective Course/Project Registration System** must meet the following quality attributes to ensure a smooth, reliable, and adaptable user experience.

4.3.1 Reliability

- **Q1. The system shall have an uptime of at least 99.5%** to ensure availability during peak registration periods.
- **Q2. Automatic database backups shall be performed **daily** to prevent data loss in case of system failure.
- **Q3. The system shall implement error-handling mechanisms** to recover from unexpected failures without data corruption.

4.3.2 Usability

- **Q4. The system interface shall follow UI/UX best practices** to ensure students, faculty, and administrators can navigate it easily.
- **Q5. All key functions (login, course browsing, registration) shall be accessible in **three or fewer clicks**.
- **Q6. The system shall support screen readers and keyboard navigation** to ensure accessibility for users with disabilities.

4.3.3 Maintainability

- **Q7. The system shall be designed using modular architecture** to allow future enhancements with minimal modifications.
- **Q8. Well-documented code and API references shall be provided** to help developers make changes efficiently.
- **Q9. System logs shall capture errors and exceptions** for easier debugging and maintenance.

4.3.4 Scalability

- **Q10. The system shall support at least 500 concurrent users** during peak hours without performance degradation.

- **Q11. The database design shall allow for expansion** to accommodate additional elective courses and student records.

These quality attributes ensure that the system remains **reliable, user-friendly, easy to maintain, and scalable**.

5. Other Requirements

*<This section is **Optional**. Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>*

Appendix A – Data Dictionary

Appendix A – Data Dictionary

The **Elective Course/Project Registration System** uses various variables, constants, inputs, and outputs to ensure proper functionality. The table below defines these elements, their descriptions, states, and associated operations.

Variable Name	Type	Description	Possible States/ Values	Related Operations	Requirement
`user_id`	Integer	Unique identifier for each user	Auto-incremented	User authentication,	F1, F5, S4
`user_role`	Enum	Defines the user type	`Student`,	Role-based access control	F2, S5
`course_id`	Integer	Unique identifier for each course/project	Auto-incremented	Course browsing, registration	F3, F5, F10
`course_status`	Enum	Status of a course	`Available`, `Full`, `Closed`	Registration checks, waitlist	F6, F11
`student_id`	Integer	Unique identifier for students	Auto-incremented	Course registration, project approval	F5, F8
`faculty_id`	Integer	Unique identifier for faculty	Auto-incremented	Project approval	F8, F9
`admin_id`	Integer	Unique identifier for administrators	Auto-incremented	Course management	F10, F11
`registration_status`	Enum	Status of student registration	`Enrolled`, `Waitlisted`,	Registration tracking	F5, F6, F7
`approval_status`	Enum	Faculty approval for project registration	`Approved`, `Rejected`,	Faculty approval workflow	F8, F9
`email_notification`	Boolean	Flag for email/SMS notifications	`True`, `False`	Sending alerts for enrollment updates	F13, F14, S9

`session_timeout`	Integer	Time (in minutes) before auto-logout	Default: 15	Security enforcement	S8
`backup_frequency`	Enum	Frequency of automatic database	`Daily`, `Weekly`	Data security and recovery	S10

Appendix B - Group Log

Appendix B – Group Log

This section includes the **meeting minutes, group activities, and progress updates** related to the development of the **Elective Course/Project Registration System SRS document**.

Meeting 1: Project Understanding & Task Allocation

Date: March 1, 2025
Attendees: Kushal Devineni, Akshith Reddy, Dheeraj Madgula, N Prajay
Discussion Points:

- Reviewed the **Statement of Work (SOW)** and discussed project scope.
- Assigned **sections of the SRS document** to each team member:
  - **Kushal** – Introduction, Functional Requirements, Use Case Diagrams
  - **Akshith** – System Overview, UI Design, Data Dictionary
  - **Dheeraj** – Security, Performance, Hardware/Software Interfaces
  - **N Prajay** – Non-functional Requirements, Group Log, Final Review
- Agreed on a **timeline for completion** (March 10, 2025).

Action Items:

✅ Each member to begin drafting assigned sections.

✅ Next meeting scheduled for March 5, 2025.

Meeting 2: Review & Refinement

Date: March 5, 2025
Attendees: All members
Discussion Points:

- Reviewed **progress on assigned sections** and provided feedback.
- Finalized **use case diagrams and UI wireframe**.
- Identified missing elements in security and performance sections.

Action Items:

✅ Dheeraj to refine **security requirements** (Add MFA & data encryption).

✅ Akshith to complete **Data Dictionary** section.

✅ Final review scheduled for March 8, 2025.

Meeting 3: Final Edits & Submission Prep

Date: March 8, 2025
Attendees: All members
Discussion Points:

- Conducted a **final review of the document**.
- Ensured **IEEE formatting guidelines** were followed.
- Added **missing references and acknowledgments**.

Action Items:

✅ N Prajay to compile and format the final document.

✅ Kushal to submit the SRS by **March 10, 2025**.

This group log documents the **collaborative effort** put into the **Elective Course/Project Registration System SRS**.