

SOFTWARE REQUIREMENTS SPECIFICATION (SRS)

Online Student Management System (OSMS)

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document provides a detailed description of the functional and non-functional requirements of the Online Student Management System (OSMS). The document is intended to serve as a reference for software developers, system architects, quality assurance engineers, and researchers involved in requirement analysis and training-based system modelling.

This SRS is also used as an input document for SRS segmentation and software feature extraction, where requirements are analysed, classified, and prepared for training intelligent systems.

1.2 Scope

The Online Student Management System (OSMS) is a web-based application designed to automate and manage academic and administrative operations of an educational institution. The system facilitates student registration, course enrolment, grade management, and academic reporting while ensuring secure and controlled access to system resources.

The OSMS aims to improve data accuracy, reduce manual workload, and provide timely access to academic information for administrators, lecturers, and students.

1.3 Definitions, Acronyms, and Abbreviations

Term	Description
OSMS	Online Student Management System
SRS	Software Requirements Specification
RBAC	Role-Based Access Control
CRUD	Create, Read, Update, Delete

1.4 Intended Audience

This document is intended for:

- Software developers implementing the system
- System analysts validating requirements
- Testers designing test cases
- Researchers performing requirement segmentation and training-based analysis

2. Overall Description

2.1 Product Perspective

The OSMS is implemented as a client-server web application following a three-tier architecture, consisting of:

- Presentation Layer (Web UI)
- Application Layer (Business Logic)
- Data Layer (Relational Database)

The system integrates with external services such as email notification systems and authentication services.

2.2 Product Functions

At a high level, the system provides the following functions:

- User authentication and authorization
- Student profile management
- Course and enrolment management
- Academic grading and reporting
- Role-based access control

2.3 User Classes and Characteristics

User Class	Description
Administrator	Manages system configuration, users, and institutional data
Lecturer	Manages courses, student grades, and reports
Student	Views academic records and enrolled courses

Users are expected to have basic computer literacy and internet access.

2.4 Operating Environment

The system shall operate on **Linux-based servers** and shall be accessible through modern web browsers such as Google Chrome, Mozilla Firefox, and Microsoft Edge.

2.5 Design and Implementation Constraints

The system must comply with institutional IT policies and data protection regulations. Development must follow secure coding standards and best practices.

3. Functional Requirements

3.1 User Authentication and Authorization

- The system shall allow users to register using a valid institutional email address.
- The system shall authenticate users using encrypted credentials.
- The system shall allow users to reset their passwords securely.
- The system shall enforce role-based access control (RBAC).

3.2 Student Information Management

- The system shall allow administrators to create student profiles.
- The system shall allow administrators to update student personal and academic information.
- The system shall allow administrators to delete student records when required.
- The system shall persist student records in a centralized database.

3.3 Course Management

- The system shall allow lecturers to create and update course information.
- The system shall allow administrators to assign lecturers to courses.
- The system shall allow students to enrol in available courses.

3.4 Academic Records and Grading

- The system shall allow lecturers to submit grades for enrolled students.
- The system shall allow lecturers to update submitted grades before finalization.
- The system shall calculate grade point averages automatically.
- The system shall generate academic performance reports for students.

3.5 Reporting and Notifications

- The system shall generate reports in PDF format.
- The system shall notify students via email when grades are published.
- The system shall log report generation activities for audit purposes.

4. Non-Functional Requirements

4.1 Performance Requirements

- The system shall respond to user requests within 2 seconds under normal operating conditions.

- The system shall support concurrent access by at least 500 users.

4.2 Security Requirements

- The system shall encrypt sensitive data at rest and in transit.
- The system shall log all authentication and authorization attempts.
- The system shall restrict access to system resources based on user roles.

4.3 Usability Requirements

- The system shall provide a user-friendly interface.
- The system shall be accessible to non-technical users.

4.4 Reliability and Availability

- The system shall maintain an uptime of 99%, excluding scheduled maintenance.
- The system shall recover from failures without data loss.

5. System Constraints

- The system must be developed using web-based technologies.
- The system must be hosted on Linux-based servers.
- The system must comply with institutional data protection policies.

6. Assumptions and Dependencies

- It is assumed that users have continuous internet connectivity.
- The system depends on third-party email services for notifications.
- The system assumes availability of institutional authentication services.

7. Future Enhancements

Future versions of the system may include:

- Mobile application support
- Advanced analytics dashboards
- Integration with learning management systems