**Software Requirements Specification**

**for**

**Online Course Enrollment System**

**Version 1.0 approved**

**Prepared by Damon Alexander**

**CST499: Capstone for Computer Software Technology**

**Joseph Rangitsch**

**February 17, 2025**

**Table of Contents**

1. Introduction

1.1 Purpose

1.2 Document Conventions

1.3 Intended Audience and Reading Suggestions

1.4 Project Scope

1.5 References

2. Overall Description

2.1 Product Perspective

2.3 User Classes and Characteristics

2.4 Operating Environment

2.5 Design and Implementation Constraints

2.6 User Documentation

2.7 Assumptions and Dependencies

3. System Features

3.1 User Registration and Profile Creation

3.1.1 Description and Priority

3.1.2 Stimulus/Response Sequences

3.1.3 Functional Requirements

3.2 User Login

3.2.1 Description and Priority

3.2.2 Stimulus/Response Sequences

3.2.3 Functional Requirements

4. External Interface Requirements

4.1 User Interfaces

4.2 Hardware Interfaces

4.3 Software Interfaces

4.4 Communications Interfaces

5. Other Nonfunctional Requirements

5.1 Performance Requirements

5.3 Security Requirements

5.4 Software Quality Attributes

6. Other Requirements

Appendix A: Glossary

Appendix C: Issues List

**1. Introduction**

**1.1 Purpose**

This document outlines the requirements for the Online Course Enrollment System (OCES), a web-based platform that simplifies student registration, enrollment, and management of online courses over three semesters: Spring, Summer, and Fall.

**1.2 Document Conventions**

Bold text indicates requirement identifiers (REQ-1).

Priorities: High (H), Medium (M), Low (L).

**1.3 Intended Audience and Reading Suggestions**

* Developers: Concentrate on Sections 3 (System Features) and 4 (External Interfaces).
* Project Managers: Examine Sections 1–2 (Introduction, Overall Description).
* Testers: Refer to Section 3 (Functional Requirements) and Appendix C (Issues List).

**1.4 Project Scope**

The system will allow students to register, log in, view courses each semester, enroll in courses with a waitlist feature, and cancel their enrollments. Administrators will manage course offerings and capacity.

**1.5 References**

Geeksforgeeks (2023, September 20). Software Requirement Specification (SRS) format. Geeksforgeeks.org. <https://www.geeksforgeeks.org/software-requirement-specification-srs-format/>

Lambdatest (2023, September, 11). What is Requirement Analysis: Best Practices and Examples. Lambdatest.com. https://www.lambdatest.com/learning-hub/requirement-analysis

Tsui, F., Karam, O., & Bernal, B. (2018). Essentials of software engineering (4th ed.). Jones & Bartlett Learning.

**2. Overall Description**

**2.1 Product Perspective**

A new web-based system will replace manual enrollment processes. It integrates with email services to send notifications.

**2.3 User Classes and Characteristics**

* Students: the primary users who enroll in courses.
* Administrators: manage course offerings, capacities, and user accounts.
* Guests: They can browse course listings but must register to enroll.

**2.4 Operating Environment**

* Frontend: Web browsers (Chrome, Firefox, Safari).
* Backend: Linux/Windows servers, relational database (MySQL/PostgreSQL).

**2.5 Design and Implementation Constraints**

Secure communication must utilize HTTPS. Passwords should be stored using industry-standard encryption methods, such as bcrypt.

**2.6 User Documentation**

* Online help guide and FAQ section.
* Admin and student user manuals.

**2.7 Assumptions and Dependencies**

* Assumes that users have access to the internet.
* Relies on third-party email services to send notifications.

**3. System Features**

**3.1 User Registration and Profile Management**

**3.1.1 Description and Priority**

Allows new users to create accounts. Priority: High (H).

**3.1.2 Functional Requirements**

* REQ-1: The system must ensure unique user IDs upon registration.
* REQ-2: User profiles must include name, email, phone number, and academic background.
* REQ-3: Passwords must be at least 8 characters, including one special character.

**3.2 Course Enrollment and Waitlist Management**

**3.2.1 Description and Priority**

Allows enrollment in courses that have a waitlist feature. Priority: High (H).

**3.2.2 Functional Requirements**

* REQ-4: Display courses available each semester (Spring, Summer, Fall).
* REQ-5: Enforce maximum enrollment limits specific to each course.
* REQ-6: Automatically add students to a waitlist when a course reaches full capacity.
* REQ-7: Notify waitlisted students via email when a seat becomes available.

**3.3 Enrollment Cancellation**

**3.3.1 Description and Priority**

Allows students to cancel enrollments. Priority: Medium (M).

**3.3.2 Functional Requirements**

* REQ-8: Upon cancellation, free the seat and notify the first waitlisted student.

**4. External Interface Requirements**

**4.1 User Interfaces**

* Online forms for registration, login, and course enrollment.
* The dashboard shows enrolled courses and the waitlist status.

**4.3 Software Interfaces**

* Database: MySQL or PostgreSQL for storing user and course data.
* Email Service: Integrated SMTP for sending notifications.

**4.4 Communications Interfaces**

* HTTPS for secure data transmission.
* Email notifications will be sent using SMTP protocols.

**5. Other Nonfunctional Requirements**

**5.1 Performance Requirements**

* Accommodate 1,000 users simultaneously during peak enrollment periods.
* Database transactions, such as enrollment and waitlist updates, must be completed within acceptable response times to ensure real-time updates.

**5.3 Security Requirements**

* Ensure that sensitive data, such as passwords, is encrypted both when it is stored (at rest) and when it is being transmitted (in transit).

**5.4 Software Quality Attributes**

* Usability: An intuitive interface for students requiring minimal training.
* Reliability: Maintained 99.9% uptime during enrollment periods.
* Maintainability: The code should be modular and well-documented to support future enhancements.
* Scalability: The design must accommodate a growing number of users and course enrollments.

**6. Other Requirements**

* The system will be developed utilizing industry-standard web technologies, including HTML5, CSS, JavaScript, and server-side scripting.
* The application will comply with accessibility guidelines to support users with disabilities.
* Decisions related to the database and integration components, such as selecting a relational database management system (RDBMS) and email services, should be finalized during the design phase.

**Appendix A: Glossary**

* Waitlist: A queue for students to join when a course is full.
* Enrollment Limit: The maximum number of students allowed in a course.

**Appendix C: Issues List**

* TBD: Finalize the email service provider.
* TBD: Define the logic for calculating course capacity.