Software Requirements Specification

for

Online Course Registration System

Version 1.0 approved

Prepared by Betina Driggers

University of Arizona – Global Campus

October 13, 2025

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Project Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Features 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

3. System Features 3

3.1 System Feature 1 3

3.2 System Feature 2 (and so on) 4

4. External Interface Requirements 4

4.1 User Interfaces 4

4.2 Hardware Interfaces 4

4.3 Software Interfaces 4

4.4 Communications Interfaces 4

5. Other Nonfunctional Requirements 5

5.1 Performance Requirements 5

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

6. Other Requirements 5

Appendix A: Glossary 5

Appendix B: Analysis Models 6

Appendix C: Issues List 6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Betina Driggers | 9/19/2025 | Requirements Added | 1.0 |

# Introduction

## Purpose

The purpose of this SRS is to define the requirements for an Online Course Registration System that enables students to register, manage (including withdraw from courses), and cancel courses across three academic semesters. The system also supports administrators and faculty in managing courses, enrollment limits, and waiting lists. The requirements are defined in alignment with established software engineering best practices and IEEE SRS standards (Pressman & Maxim, 2020).

## Document Conventions

Requirements are uniquely labeled as REQ-n for easy reference, and each includes a defined priority level (H, M, L) as recommended by structured requirement documentation methods (Sommerville, 2016).

## Intended Audience and Reading Suggestions

This document is intended for developers, project managers, testers, and end users (students, faculty, administrators). Developers should focus on Sections 3–5 for functional and non-functional requirements, while testers should use the same sections for validation criteria.

## Project Scope

The Online Course Registration System will allow students to register for courses offered per semester, manage their enrollments (including withdrawing from a course), and join waiting lists for full courses. Faculty and administrators will manage course offerings, enrollment limits, and cancellations. The system aims to simplify enrollment processes, enforce data integrity, and ensure fair and transparent course allocation.

## References

Pressman, R. S., & Maxim, B. R. (2020). Software Engineering: A Practitioner’s Approach (9th ed.). McGraw-Hill Education.

Sommerville, I. (2016). Software Engineering (10th ed.). Pearson.

Tsui, F., Karam, O., & Bernal, B. (2018). [Essentials of software engineering](https://uagc.instructure.com/courses/150851/modules/items/7693353) (4th ed.). Jones & Bartlett Learning.

# Overall Description

## Product Perspective

This system will be a standalone web application/user interface accessible via standard browsers on any device that can connect to the internet. It will integrate with the institution’s existing student information database.

## Product Features

* New user registration with profile creation.
* Unique ID and password authentication.
* Login/logout functionality.
* Course listing by semester.
* Enrollment with waitlist management.
* Course cancellation with automatic waitlist notification.

## User Classes and Characteristics

* **Students:** Register, enroll, cancel/withdraw from courses.
* **Faculty:** View/manage courses.
* **Administrators:** Manage course availability, maximum enrollment, and waiting lists.

## Operating Environment

* Web-based applications/user interface (Chrome, Firefox, Edge, Safari).
* Backend database (MySQL or PostgreSQL).
* Server: Windows/Linux with Apache or Nginx

## Design and Implementation Constraints

* Compliance with FERPA regulations.
* Secure password management (hashed and salted storage).
* Course enrollment must enforce maximum limits.

## User Documentation

* Online help/user manual
* Step-by-step user guide.
* FAQs

## Assumptions and Dependencies

* Users have stable internet access.
* The institution provides an updated course catalog each semester with an accurate count of open seats for each available course.
* Integration with the institutional email system for notifications.

# System Features

## User Registration and Profile Creation

**Description and Priority:** Enables new users to create accounts (High priority).  
**Stimulus:** User provides name, phone, institutional email address, desired user ID/username, and password.  
**Response Sequence:** System validates → Creates profile with unique ID.  
   
**Functional Requirements:**

* REQ-1: System must prevent registrations that contain duplicate user ID/username, user first and last name, and user institutional email address.
* REQ-2: System must store user details securely.

## Login/Authentication

**Description and Priority:** Provides secure login with session timeout controls, enforcing security best practices (Tsui et al., 2018).

**Functional Requirements:**

* REQ-3: Users can log in using a unique ID/username and password.
* REQ-4: Session management must automatically log out users who have not interacted within the last 10 minutes during the current session.

## Course Listing and Enrollment

**Description and Priority:** Provides secure access (High priority).  
  
**Functional Requirements:**

* REQ-5: System displays courses by upcoming semester, organized by course category, and then in alphabetical order.
* REQ-6: System prevents enrollment if the maximum limit is reached.
* REQ-7: Students may join the waitlist if the course is full.

## Course Cancellation and Waitlist Notification

**Description and Priority:** Supports course drop (cancellation) and waitlist management (Medium priority).  
  
**Functional Requirements:**

* REQ-8: Students can cancel enrollment/withdraw from a course.
* REQ-9: System notifies the first student on the waitlist when a seat opens via institutional email.

# External Interface Requirements

## User Interfaces

* Responsive web forms for registration, login, course browsing, course cancellation, and course waitlist.
* Notifications via institutional email.

## Hardware Interfaces

* Standard client devices (PCs, laptops, tablets, cell phones).

## Software Interfaces

* Integration with the institutional database and email system.

## Communications Interfaces

* HTTPS for secure data transmission.

# Other Nonfunctional Requirements

## Performance Requirements

* The system must support at least 500 concurrent users.
* Response must take less than two seconds for transactions/interactions.

## Safety Requirements

* Regular database backups to prevent data loss.

## Security Requirements

* Role-based access (student, faculty, admin).
* Encrypted passwords and secure login.

## Software Quality Attributes

* Availability: 99% uptime SLA guarantee.
* Usability: Simple, clean, visually appealing, and intuitive design.
* Maintainability: Modular design for updates and adaptability for new technology integration.

# Other Requirements

* The system must support automated notifications for course enrollment changes.
* Must be scalable for future semester/course expansions.