**Week 1 - Software Requirements Specification**

Reginald Tremor

The University of Arizona Global Campus

Course Code: CST313: Software Testing

Charmelia Butler

August 10, 2025

Software Requirements Specification

For

Online Course Registration System

Version 1.0 approved

Prepared by Reginald Tremor

UAGC

August 9, 2025

**Table of Contents**

[1. Revision History iii](#_Toc205627915)

[2. Introduction 1](#_Toc205627916)

[2.1 Purpose 1](#_Toc205627917)

[2.2 Document Conventions 1](#_Toc205627918)

[2.3 Intended Audience and Reading Suggestions 1](#_Toc205627919)

[2.4 Product Scope 2](#_Toc205627920)

[3. Overall Description 2](#_Toc205627921)

[3.1 Product Perspective 2](#_Toc205627922)

[3.2 Product Functions 2](#_Toc205627923)

[3.3 Operating Environment 3](#_Toc205627924)

[3.4 Design and Implementation Constraints 3](#_Toc205627925)

[4. System Features 4](#_Toc205627926)

[4.1 User Registrations and Login 4](#_Toc205627927)

[4.2 Profile Management 4](#_Toc205627928)

[4.3 Course Management 4](#_Toc205627929)

[4.4 Waiting List and Cancellation 5](#_Toc205627930)

[5. External Interfaces 5](#_Toc205627931)

[5.1 User Interfaces 5](#_Toc205627932)

[5.2 Software Interfaces 5](#_Toc205627933)

[6. Nonfunctional Requirements 6](#_Toc205627934)

[6.1 Performance 6](#_Toc205627935)

[6.2 Security 6](#_Toc205627936)

[6.3 Quality Attributes 6](#_Toc205627937)

[6.4 References 6](#_Toc205627938)

[7. Appendix A: Glossary 7](#_Toc205627939)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Reginald Tremor | 8/8/2025 | Initial draft | 1.0 |

# Introduction

## Purpose

This Software Requirements Specification (SRS) document details features and scope of the Online Class Registration System (OCRS) that allows students to register and maintain their profiles securely and register for academic courses by semester. This SRS incorporates feature specifications, including database design specifications, system feature specifications, security specifications, and user needs-based specifications. The application will utilize XAMPP (Apache server, MySQL, PHP) designed and developed for a consistent, accessible, web-based application.

## Document Conventions

Requirements will be labelled using tags (e.g. REQ-#). The main intention for users is provided by prioritizing High, Medium, or Low depending on importance and implementation sequence. Technical terms are defined in Appendix A. Predefined sections labelled as TBD represent where the specification can be refined in the future specification.

## Intended Audience and Reading Suggestions

This document is directed towards software developers, system architects, testers, project stakeholders and academic supervisors..

## Product Scope

The Online Course Registration System supports functions for a new user registering and creating an account and profile, with the ability to view courses and register based on the semester (time period) for the courses, and capacity limits. The system guarantees all user IDs are unique and that users can wait list courses (for limited capacity) and cancel courses if they desire. The system will securely values data in MySQL databases.

# Overall Description

## Product Perspective

The main goal is a stand-alone web-based application that will be designed to replace manual registration processes or fragmented registration processes. XAMPP will host the software locally during development and thus be extendable for deployment on remote hosting architecture. The software will interface internally with MySQL database and external interface through web-based front end.

## Product Functions

The most important functions provided by the system are:

* **Primary**
  + New user registration with validation of unique ID
  + Login/logout function
  + Profile management (name, email, phone, etc.)
  + Course offerings by semester
  + Enrollment limits and wait list function
  + Retract course and notify waitlist user function

## Operating Environment

The framework within which the system includes:

* **Hardware Requirements**
  + No less than 32GB RAM, and SSD as well as an 8-core CPU.
* **Software Requirements**
  + XAMPP Control Panel (latest version)
  + Apache Web Server
  + MySQL Database
  + PHP 7.4 and later
  + Supported browsers: Chrome, Firefox, and Edge

## Design and Implementation Constraints

The project will have the following design constraints and requirements:

* Fundamentally comply with web application security best practices (i.e., OWASP Top Ten recommendations)
* Relational database design using MySQL schema recommendations
* Support scalability and maintainability standards

# System Features

## User Registrations and Login

Description: New users register with credentials that must be unique. Priority: High Requirements:

**Stimulus/Response:**

* User fills in required fields → System validates users input → System stores record
* User submits non-unique ID → System displays error on duplicate entry "ID must be unique"

**Requirements**

These are the conditions which, once met, will achieve effective system performance.

* REQ-1: The system will reject duplicate IDs when registering.
* REQ-2: Passwords will reflect complex standards defined for this study.
* REQ-3: User's must securely login with their defined credentials.

## Profile Management

Description: User's need to maintain their personal and academic information. Priority: Medium Requirements:

* REQ-4: User's can enter name, email, phone no. and other relevant entries.
* REQ-5: All data will be stored securely in MySQL.

## Course Management

Description: Students will view and enroll in allocated courses every semester. Priority: High Requirements:

* REQ-6: Admin's will set defined course offerings each semester.
* REQ-7: There will be a maximum number of students for each course.
* REQ-8: The system must disallow enrollment in full courses.

## Waiting List and Cancellation

Description: Students that no longer want their course can drop it and waitlisted students can be notified that their name has come to the top of the bus. Priority: High Requirements:

* REQ-9: Dropping a course will trigger automated waitlist processing.
* REQ-10: The first waitlisted student will be notified that a position has opened.

# External Interfaces

## User Interfaces

The web-based interface must be easy, intuitive and retain usability across platforms. Usability design shall follow best practices as described by Nielsen (1994), and shall use responsive design.

## Software Interfaces

The system will support data interchange through the utilization of the following protocols:

* Apache Server for all HTTP communications.
* MySQL Database for storing data.
* PHP will provide server-side logic.

# Nonfunctional Requirements

## Performance

Description and Priority

* Page load times (per page) must be no more than 2 seconds in standard (normal) load conditions.
* Concurrent users must be up to 500.

## Security

* User authentication will include password`s that are encrypted at a minimum using SHA-256.
* Role-based accessed designed to include access controls.

## Quality Attributes

The system must meet the following performance:

* **Reliability**: The system would have a 99.9% uptime.
* **Maintainability**: Code would follow modular standards that facilitates future upgrades.
* **Usability**: User navigation would be intuitive and easy to navigate.

## References

* Sommerville, I. (2015). Software Engineering (10th ed.). Pearson.
* Wiegers, K. E. (2003). Software Requirements. Microsoft Press.
* Nielsen, J. (1994). Usability Engineering. Academic Press.
* Tsui, F., Karam, O., & Bernal, B. (2018). Essentials of software engineering (4th ed.). Jones & Bartlett Learning.

# Appendix A: Glossary

This section defines important terms for the SRS.

* **OCRS:** Online Course Registration System.
* **XAMPP:** Local development suite combining Apache, MySQL, PHP.
* **FERPA:** Family Educational Rights and Privacy Act.
* **SMTP:** Simple Mail Transfer Protocol, used for sending emails.
* **WCAG**: Web Content Accessibility Guidelines

**References**

Sommerville, I. (2015). Software engineering (10th ed.). Pearson Education. [Software Engineering, 10th GLOBAL Edition](https://dn790001.ca.archive.org/0/items/bme-vik-konyvek/Software%20Engineering%20-%20Ian%20Sommerville.pdf)

Wiegers, K. E. (2003). Software requirements (2nd ed.). Microsoft Press.

[Software requirements : practical techniques for gathering and managing requirements throughout the product development cycle : Wiegers, Karl Eugene, 1953- : Free Download, Borrow, and Streaming : Internet Archive](https://archive.org/details/softwarerequirem0002wieg)

OWASP Foundation. (2022). OWASP top ten web application security risks.

<https://owasp.org/www-project-top-ten/>

Nielsen, J. (1994). Usability engineering. Morgan Kaufmann.

[Usability Engineering: | Guide books | ACM Digital Library](https://dl.acm.org/doi/book/10.5555/2821575)

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