Software Requirements Specification

for

Student Registration Portal – CST499 Computer Software Technology Capstone

Version 1.0 - Approved

Prepared by Kendric Garmon

University of Arizona Global Campus

May 2023

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 Product Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Functions 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. External Interface Requirements 3

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

4. System Features 4

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

5. Other Nonfunctional Requirements 4

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

6. Other Requirements 5

Appendix A: Glossary 5

Appendix B: Analysis Models 5

Appendix C: To Be Determined List 6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Kendric Garmon | 4/17/2023 | Initial Draft | 0.1 |
| Kendric Garmon | 5/13/2023 | Published as final | 1.0 |

# Introduction

## Purpose

The purpose of this SRS document is to outline the requirements for the development of a new user registration and online course enrollment system.

## Document Conventions

This SRS document follows the IEEE standard for software requirements specification. Requirements are organized into sections based on their functionality. Each requirement statement has a unique identifier number for easy reference.

## Intended Audience and Reading Suggestions

This SRS document is intended for developers, project managers, testers, and other stakeholders involved in the development of the student portal system. Developers will find detailed information on the functional and non-functional requirements of the system, as well as technical specifications and constraints. Project managers will be able to use this document as a reference to track project progress and ensure that requirements are met. Testers will use this document to ensure that all requirements are met during testing.

The SRS document contains an overview section that introduces the system, its purpose, and features. The requirements section is divided into functional and non-functional requirements. The functional requirements describe the system's behavior in response to inputs and include use case scenarios, input/output formats, and processing rules. The non-functional requirements specify the system's quality attributes, including performance, reliability, usability, and security.

The suggested reading sequence for this document is as follows:

1. Introduction - Provides an overview of the document and the system
2. Scope - Defines the scope of the system and identifies stakeholders
3. Requirements - Provides detailed functional and non-functional requirements for the system
4. Use Case Scenarios - Describes the system's behavior in response to user interactions
5. User Interface - Specifies the design and layout of the user interface
6. Performance Requirements - Describes the system's performance characteristics and constraints
7. Security Requirements - Describes the system's security features and constraints
8. Constraints - Identifies any constraints or limitations on the system design and implementation.

## Product Scope

The system will allow new users to register for an account and create a profile. It will also facilitate online course enrollment for three semesters per year, with the ability to add courses to a waiting list if they are full. Additionally, users will have the ability to cancel course enrollments, and the system will notify the next user in the waiting list if a spot becomes available.

## References

Wiegers, K. E. (1999). Software Requirements. Microsoft Press.

# Overall Description

## Product Perspective

The student portal system is a new, self-contained product intended to provide students with an easy and convenient way to access academic resources and information. The system will be designed to be used in conjunction with the existing university website, with the portal serving as a gateway to specific academic functions and resources.

## Product Functions

The student portal system will be designed to perform the following major functions:

* Allow students to view their course schedules, including class times, locations, and instructors
* Enable students to access and download course materials, such as syllabi, assignments, and lecture notes
* Provide a platform for online submission of assignments, including file uploads and text submissions
* Allow students to track their grades and view feedback from instructors
* Enable students to communicate with their instructors and peers through online forums and messaging systems
* Provide access to important academic resources, such as the university library, tutoring services, and academic advising
* Allow students to view and manage their personal information, such as contact information and academic records.

A top-level data flow diagram depicting the major functional groups and how they relate is shown below:

**[Insert diagram here]**

## User Classes and Characteristics

This product is intended for two user classes: students and administrators.

Students are the primary users of the product and are characterized as follows:

* Technical expertise: Students are expected to have basic computer skills and be familiar with web-based applications.
* Educational level: Users are expected to be current students of the institution and have a basic understanding of the registration process.
* Experience: Students may have varying degrees of experience with online registration systems, but are expected to be able to navigate the product with minimal guidance.

Administrators are the secondary users of the product and are characterized as follows:

* Technical expertise: Administrators are expected to have advanced computer skills and be familiar with web-based applications.
* Educational level: Administrators are expected to have a higher education degree or equivalent experience in managing student registration processes.
* Experience: Administrators are expected to have significant experience with managing student registration processes and be able to navigate the product with ease.

Operating Environment

The student portal system will operate in a web-based environment and will be accessed through popular web browsers such as Google Chrome, Mozilla Firefox, and Microsoft Edge. The system will be hosted on a web server running a Linux operating system with PHP version 7.4. The server will also have a MySQL database management system installed for data storage and retrieval.

## Design and Implementation Constraints

* The student registration portal must be implemented using PHP as the backend programming language and MySQL as the database management system.
* The system must comply with all relevant security and privacy regulations, including the protection of personal data of users.
* The system must be compatible with modern web browsers such as Chrome, Firefox, and Safari.
* The portal must have a responsive design that works well on various devices with different screen sizes, including desktops, laptops, tablets, and smartphones.

## User Documentation

The following user documentation components will be delivered along with the software:

* User manual that describes how to use the portal, including how to register, log in, and perform various tasks such as course registration, grade viewing, and fee payment.
* On-line help that provides context-sensitive information and guidance on using the portal.
* Tutorials that walk users through common tasks and procedures on the portal.
* The user documentation will be delivered in HTML format and will be accessible through the portal itself.

## Assumptions and Dependencies

Assumptions:

* The development team assumes that the client has provided accurate and complete information regarding the student registration portal requirements.
* It is assumed that the client will be available to clarify requirements or provide additional information as needed throughout the development process.
* It is assumed that the development team has access to the necessary hardware and software resources required for development and testing.

Dependencies:

* The student registration portal will depend on the availability and stability of the internet and network connections for both the users and the server hosting the portal.
* The portal may depend on third-party software libraries or tools to implement certain features or functionalities, and their availability and compatibility may impact the development timeline.
* The portal may need to integrate with other existing systems or databases within the organization, and the availability and accessibility of those systems will impact the development and deployment of the portal.

# External Interface Requirements

## User Interfaces

### New User Registration

*The system shall allow new users to register for an account and create a profile. The system shall generate a unique ID for each new user, which must be associated with a password. The system shall ensure that two users cannot register with the same ID. The user profile shall include key information such as name, email, and any other necessary information.*

### User Login

*The system shall allow registered users to login at any time using their ID and password.*

### Online Course Enrollment

*The system shall facilitate online course enrollment for three semesters per year (spring/summer/fall). Users shall be able to view the courses offered during each semester. Each course shall have a maximum number of enrollment, which may vary depending on the course. If a course is full, users shall have the option to add themselves to a waiting list. The system shall notify the next user in the waiting list if a spot becomes available.*

### Course Enrollment Cancellation

*Users shall be able to cancel their course enrollments. If a user cancels a course enrollment, the system shall notify the next user in the waiting list (if any) that they can enroll in the course.*

## Hardware Interfaces

As the student registration portal is a web-based application, it does not have any specific hardware interfaces. However, it will require standard hardware components such as a computer or mobile device with a modern web browser and internet connectivity to access the application. The web browser should be compatible with modern web standards and support JavaScript and cookies. Additionally, the application should be able to handle the expected traffic load and the server should have sufficient processing power, memory, and storage to handle the expected number of concurrent users.

## Software Interfaces

* Database system: The software product must interface with a database management system to store and retrieve user information. The specific database system to be used will be determined during the implementation phase.
* Email services: The software product must interface with external email services to send confirmation and reminder emails to users. The specific email service to be used will be determined during the implementation phase.
* Operating system: The software product must be compatible with the operating system(s) specified in the product perspective section (2.1).
* Web browsers: The software product must be compatible with commonly used web browsers, such as Google Chrome, Mozilla Firefox, and Microsoft Edge.
* Development tools: The software product may interface with development tools used during the development process, such as integrated development environments (IDEs), version control systems, and testing frameworks.
* Libraries and frameworks: The software product may interface with external libraries and frameworks used during development, such as front-end frameworks, back-end frameworks, and database connectors.
* APIs: The software product may use APIs to interact with other software components, such as payment gateways or third-party authentication services. The specific APIs to be used will be determined during the implementation phase.

## Communications Interfaces

N/A

# System Features

## Registration System

The system shall provide a new user registration process that includes account and profile creation. Each new user shall have a unique ID associated with a password. The system shall prevent two users from using the same ID for registration. Profiles shall include some key information about the applicant including name, phone, email, and any other information deemed necessary.

## Login System

The system shall allow registered users to log in using their unique ID and password at any time.

## Course Listing

The system shall allow students to view the courses that will be offered during any semester. Each course shall have a maximum number of enrollment that may be different depending on the course.

## Course Enrollment

The system shall allow users to enroll in courses that have not reached maximum enrollment. If a user wants to enroll into a course and the course is full, the student shall be added onto a waiting list. The system shall inform the first user in the waiting list (if any) that they can enroll into the class upon a cancellation by a current enrollee.

## Course Cancellation

The system shall allow users to cancel their enrollment from any course that they are enrolled in. The system shall inform the first user in the waiting list (if any) that they can enroll into the class upon cancellation by a current enrollee.

# Other Nonfunctional Requirements

## Performance Requirements

N/A

## Safety Requirements

N/A

## Security Requirements

N/A

## Software Quality Attributes

N/A

## Business Rules

N/A

# Other Requirements

N/A

Appendix A: Glossary

Terms:

1. SRS - Software Requirements Specification: A document that describes the functional and non-functional requirements of a software system.
2. User ID: A unique identifier assigned to each user for authentication and identification purposes.
3. Profile: Information about a user that includes personal details, such as name, phone number, and email address.
4. Registration: The process of creating a new user account in the system.
5. Password: A secret code used to authenticate a user's identity during login.
6. Online Course: A course offered through the student portal system that can be accessed and completed remotely via the internet.
7. Semester: A division of an academic year, typically consisting of a specific time frame during which courses are offered.
8. Enrollment: The act of registering or signing up for a course.
9. Waiting List: A list of users who are interested in enrolling in a course that is currently full, with a request to be notified if a spot becomes available.
10. Cancellation: The action of withdrawing or removing oneself from an enrolled course.
11. Notification: A message or alert sent to a user to provide information or updates regarding their enrollment status.

Acronyms and Abbreviations:

1. IEEE - Institute of Electrical and Electronics Engineers: A professional association that develops and publishes technical standards in various industries, including software engineering.
2. PHP - Hypertext Preprocessor: A server-side scripting language used for web development.
3. MySQL - My Structured Query Language: An open-source relational database management system.
4. IDE - Integrated Development Environment: A software application that provides comprehensive tools for software development.
5. API - Application Programming Interface: A set of rules and protocols that allow different software applications to communicate and interact with each other.
6. HTML - Hypertext Markup Language: The standard markup language used for creating web pages.
7. CSS - Cascading Style Sheets: A style sheet language used for describing the presentation and formatting of a document written in HTML.