Software Requirements Specification

for

<Week 1 - Assignment>

Version 1.0 approved

Prepared by <Gino De Paoli>

<Capstone for Computer Software Technology >

<9/26/2022>

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** |
| Week 1 | 8/29 | First Submission | V1 |
|  |  |  |  |

# Introduction

## Purpose

<The purpose of this software is a multi-functional tool that allow new students to register, old students to manage their classes, allow students to enroll into new classes, and more. This tool will also be able to give information to students about classes they are wanting to take and join waitlists if the class is full .>

## Document Conventions

<In this case each of the requirements that are inherited from the higher level requirements will have similar priority. Each of the detailed requirements are inline with the business goal and needs to be implemented in the software at the time of deployment..>

## Intended Audience and Reading Suggestions

<This document is targeted for the stakeholders of a university, and other higher executives in the university management as it will interact with their data. This will also need to be a user friendly system that makes it easy for students to manage their courses while enrolling or currently enrolled in courses..>

## Product Scope

<This software product will have multiple functions that will allow students to better manage their classes. Students will be able to enroll into the university. Waitlist themselves in courses that are already full. Students will also be able to cancel enrollment which the system updates the first person on the waitlist. .>

## References

<.>

# Overall Description

## Product Perspective

<This system will be a newly designed system that helps students manage their classes. This system will be interacting with a database that holds tables on class information, and student information. The front end development will be done in PHP and HTML. The backend will connect to a MYSQL database using PHP connections. Users will be using their unique information like a user ID and password to log into the system..>

## Product Functions

<

* New user registration that includes account and profile creation
* Website creates a user and gets back a student ID and stores the students information and password
* Students will be able to see a list of courses available to sign up for
* Students can waitlist themselves for a class that is full
* If class that is full has a student drop out then the next person in line will be notified

>

## User Classes and Characteristics

<.>

## Operating Environment

<This system will be first available in any modern web browser. Later after a successful launch and bug fixes the next phase will include an phone application..>

## Design and Implementation Constraints

< >

## User Documentation

<A sharepoint will be created to host any FAQ’s/Tips and Tricks to help with common questions students have when starting out with the new software..>

## Assumptions and Dependencies

<>

# External Interface Requirements

## User Interfaces

<The webpage design will be very simple and straight to the point. Students will be able to see simple forms to fill out for information they want to see like classes available.>

## Hardware Interfaces

<Users that are trying to interact with the system will be able to do so through a device with a web browser. If you are on a desktop, you will use both the keyboard and mouse as input. If you are using a web browser through a mobile phone then you will be using touch to interact with the system for both clicking and typing..>

## Software Interfaces

<The software web application will be written in HTMl and PHP. The web application will be pulling and pushing data to a database server through MySql connection queries in PHP. The HTML will mainly be used as the layout structure for the forms being used. When data needs to be pushed to a database securely the Post Method will be used in PHP.>

## Communications Interfaces

<>

# System Features

<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

## User Registration

<>

4.1.1 Description and Priority

<In order to get add new students to the system they must be able to register to the school and get added into the database. This is going to be a high priority as it is important to be able to register students before they can get into the managing of their classes.

4.1.2 Stimulus/Response Sequences

<Student will first start and the landing page of the website and will have a few options to choose from. In order to register as a new student you will click the link to register. The system will re-direct you to a new page to fill out a form that will record your information. Once this has been filled out a button at the end of the form can be clicked to push data to a database..>

4.1.3 Functional Requirements

<TBD>

<REQ-1.>

## 4.2 Unique User Info

<>

4.2.1 Description and Priority

<Once a new student has filled out the registration form the system will create a unique ID and store their password. This unique ID is also then shared to the user.

4.2.2 Stimulus/Response Sequences

<Once the form has been submitted the webpage will display a confirmation status that the submission was successful. The user will then be given their student ID’s so they can use to log into the system. .>

4.2.3 Functional Requirements

<TBD>

<REQ-2.>

## 4.3 List Courses

<>

4.3.1 Description and Priority

<Once a user is able to login as themselves they will then want to start managing their classes. In order to enroll in new classes they need to be able to get a list of available classes, the capacity, and the status. This can be printed in table format making it easy to read for the student.>

4.3.2 Stimulus/Response Sequences

<Once the user has an idea of what classes they want to see if it is available the student would just have to select the course they are interested in. The application will the query the information about the class like capacity, and the current status. >

4.3.3 Functional Requirements

<TBD>

<REQ-3.>

## 4.4 Add to Waitlist

<>

4.4.1 Description and Priority

<Once a user selects a course and they realize that the class desired is full they can waitlist themselves. This will add them to a list of students also wanting to take this course. If this course becomes available if a student drops the course or room changes were made then the first person on the waitlist will be notified..>

4.4.2 Stimulus/Response Sequences

<Once the student gets the class details they will then be able to add themselves to a waitlist. This will give them a chance to sign up for something if just in case it becomes available.. >

4.4.3 Functional Requirements

<TBD>

<REQ-4.>

# Other Nonfunctional Requirements

## Performance Requirements

<>

## Safety Requirements

<>

## Security Requirements

<This system will be reaching out to a database and storing personal information about the student. The security concern would be for any data breaches that occur. We will be using best practices and implementing complexity to passwords and unique ID’s given. Users will be needing more than one way to authenticate to gain access to their account.>

## Software Quality Attributes

<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>

## Business Rules

<>

# Other Requirements

<

**Database Requirements**

* Each ID will be unique
* The system should guard against two users with same ID
* Database will store student information like name, phone, and personal email

>

Appendix A: Glossary

< >

Appendix B: Analysis Models

< >

Appendix C: To Be Determined List

< >

**Github Link -** [gdepaoli17/ComputerSoftwareTechnology: Computer Software Technology Capstone Site (github.com)](https://github.com/gdepaoli17/ComputerSoftwareTechnology)

**A screenshot of a computer

Description automatically generated with medium confidence**