**Week 5 – SRS Final Project**

Cody Angeline

The University of Arizona Global Campus

CST 499: Capstone for Computer Software Technology

Dr. Amr Elchouemi

January 23, 2023

Table of Contents

[1 Introduction 3](#_Toc125213015)

[1.1 Document Identifier 3](#_Toc125213016)

[1.2 Scope 3](#_Toc125213017)

[1.3 References 4](#_Toc125213018)

[1.4 System overview and key features 4](#_Toc125213019)

[1.4.1 Homepage Functionality 5](#_Toc125213020)

[1.4.2 Account Creation Page Functionality 5](#_Toc125213021)

[1.4.3 Search Page Functionality 6](#_Toc125213022)

[1.4.4 General Functionality 7](#_Toc125213023)

[1.5 Overall Description 8](#_Toc125213024)

[1.5.1 Product Perspective 8](#_Toc125213025)

[1.5.2 Product Functions 9](#_Toc125213026)

[1.5.3 User Classes and Characteristics 11](#_Toc125213027)

[1.6 Design and Implementation Constraints 12](#_Toc125213028)

[2 General 13](#_Toc125213029)

[2.1 Nonfunctional requirements 13](#_Toc125213030)

[2.1.1 Safety requirements 14](#_Toc125213031)

[2.1.2 Security Requirements 14](#_Toc125213032)

[2.1.3 Software Quality Attributes 14](#_Toc125213033)

[2.1.4 Business Rules 15](#_Toc125213034)

[References 16](#_Toc125213035)

# Introduction

This document intends to outline the software requirements of the University of Arizona’s Online Global Campus (UAGC) online course registration website, version 1.1. This document should be used to aid in the refinement of the final deliverable and should be reviewed and updated periodically to ensure its completeness. This artifact illustrates the system’s primary application and the required functionality specified by the stakeholder.

## Document Identifier

**The University of Arizona Global Campus Online Course Registration Website**

**Version 1.1 approved**

**Prepared by: Cody Angeline**

**1/23/2023**

## Scope

The UAGC, course CST 499, and Professor Amr Elchouemi have requested an online course registration website to schedule classes for new and current students. This capability is intended to give students more control over their enrollments. This is desirable because the university anticipates significant cost savings associated with personnel overtime; the system will offload a large part of the work. Furthermore, UAGC anticipates that a more significant return will be generated by the expeditious service offered to new and returning students. The online course registration website should feature the following:

* Provide a new user registration feature that supports account and profile creation
* Support user login at any point during the registration process.
* Associate user IDs with passwords and prevent duplicate account creations.
* Support profiles with vital user information such as name, phone, email, etc.
* Provide the user the ability to view classes hosted at the university during different

semesters of the calendar year.

* Ensure a maximal number of student enrollments for any course.
* Allow users to add themselves to a waiting list for a class that becomes full.
* Provide a user with the ability to opt out of any course they have enrolled in.

## References

Cockburn, A. (1999). *Writing effective use cases*. Pearson Education. ISB 0-201-70225-8

Purdue. (n.d.). *Software requirements specification.* SC 30700. Retrieved October 18, 2022, from

<https://eee.cs.purdue.edu/homes/bxd/307>

Richards, M., Ford, N. (2020). *Fundamentals of software architecture*. O’Reilly Media Inc. <http://oreilly.com/catalog/errata.csp?isbn=9781492043454>

Sommerville, I. (2018). Software engineering (10th Edition). Pearson.

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

## System overview and key features

Requirements specification is the phase of development in which stakeholders meet with design teams to write down user and system requirements. User requirements are typically written in natural language with accompanying diagrams and models (Sommerville, 20018). The following sections expand each requirement completely and set a priority level for development of each key functionality. As a high-level summary, the customer specified that the desired the UAGConline course registration website will incorporate a welcome, login, course registration, account review, and class removal page.

### Homepage Functionality

***Welcome page***:

Priority – High

The user interface will incorporate a welcome page. Present within this page will be:

* A login button for username and password entry.
* An account creation button.
* A course search button.

### Account Creation Page Functionality

***Account creation page***:

Priority – High

The system must possess functionality that allows the user to create a profile. The university database will use the information from this page when a student registers for a course. The following fields must be captured within the account creation page:

* Email
* Password
* First Name
* Last Name
* Address
* Phone Number
* Salary
* Social Security Number

### Search Page Functionality

***Course Search Page***:

Priority – High

A “course” search button shall be present on each web page and available to the user while browsing. Pages exempt from this requirement are pages whose content already relates to course information. The course search page that is presented to a user upon interacting with a course search button will include the following:

* An accordion-style drop-down menu that lists the available courses.
* A textbox that displays course information.
* A button for adding courses to a list.
* A list that holds selected courses
* A finalize button that submits the users’ choices to the university.

***Course Selection Options***:

Priority – High

When a course has been selected from the accordion menu, the textbox that displays the course information must possess the following information:

* Semester
* Date Start
* Date End
* Number of Seats Left

If a class has been filled up, the user must be able to add the course to their list, with the caveat that a prompt alert them of the available seat status. If the user chooses to continue, they will be notified that they will be put into the class if a seat becomes available.

***Confirmation Email***:

Priority – High

Once an account has been created or modified, or an authenticated user has completed their course registration, an email shall be sent to the user. This email shall summarize their actions: account creation, account changes, or class registration.

### General Functionality

***User Information***:

Priority – High

A user information button shall be present on each web page to users whom, are logged in, and available to that user while browsing. Pages exempt from this requirement are pages whose content already relates to user information. The user information page shall allow a user to update or review any of the following:

* Name
* Phone Number
* Email
* Registered Classes

## Overall Description

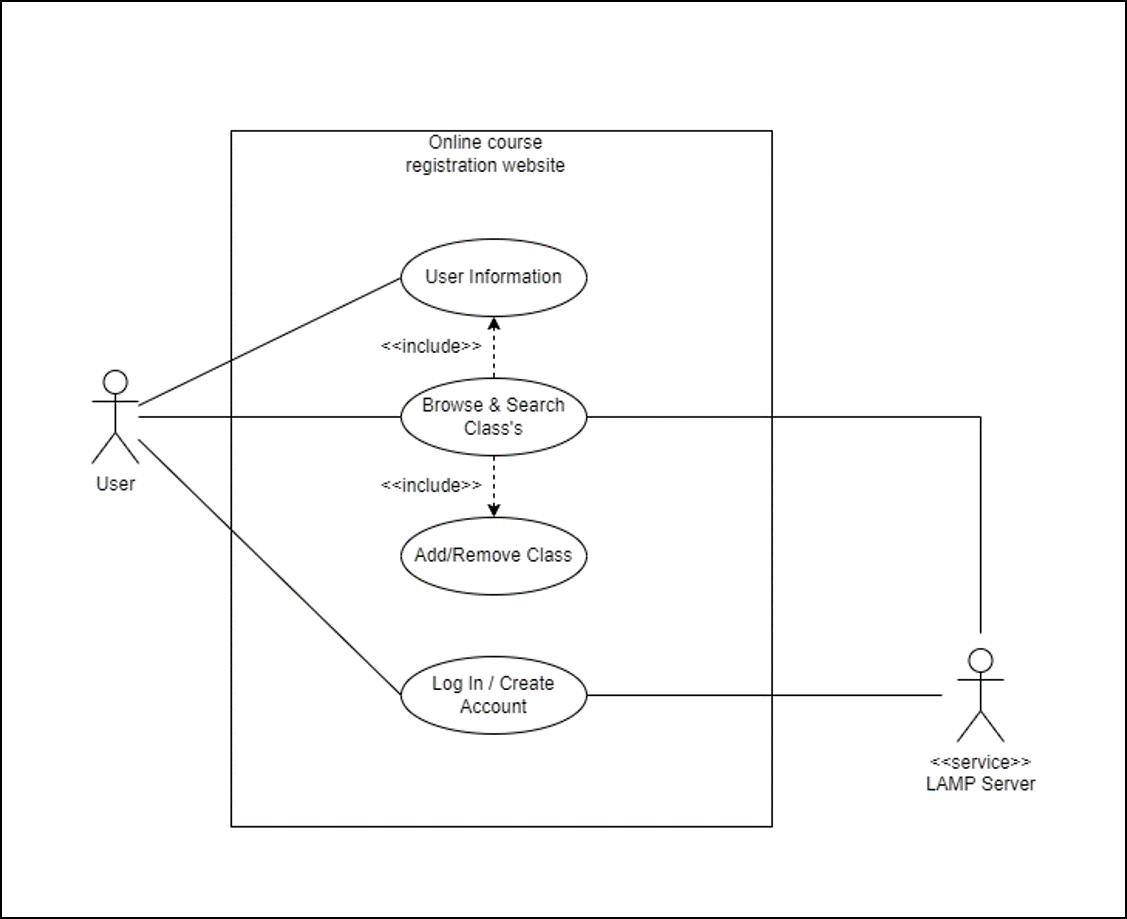
System models are intended to present the functionality of a system via abstraction. These high-level models are used during the requirements engineering phase to help specify the detailed requirements of a system. Models help to clarify what existing systems do or what new systems are meant to do (Sommerville, 2018).

### Product Perspective

A use case model is a form of interaction model, and is used to provide a simple overview of the interaction of an actor with a system (Sommerville, 2018).

**Figure 1.**

Shows a use case diagram of the UAGC online course registration website.



Note. This use case model shows the online course registration website being interacted with by a user. In this instance, the user can log in or browse as a guest. In either instance, the user has access to the registration functionality of the website and can alter their status at any time before registration is completed.

### Product Functions

Use cases describe the interaction between a user and a system through text diagrams. Use cases identify the actors involved in an exchange and the system’s desired state after the interaction. They portray actors who use the system and the process they must take to accomplish some goal (Tsui et al., 2018). The following use cases were generated as part of the requirements gathering process.

|  |  |
| --- | --- |
| Use Case #1 | Account Creation |
| Goal in Context | Create a new user account. |
| Scope & Level | Basic functionality. |
| Preconditions | The user has prompted the system to make a new account and is on the account creation page. |
| Success End Condition | All information is filled in & user strikes the “Submit: button. |
| Trigger | The user strikes the “Create Account” button. |
| Description | The user utilizes the submit button once all information has been entered. |

|  |  |
| --- | --- |
| Use Case #2 | Account Login |
| Goal in Context | A textbox serves to receive user credentials for login. |
| Scope & Level | Basic functionality. |
| Preconditions | The user has entered a username and password. |
| Success End Condition | The user account exists, all information is entered & user strikes the “Login” button. |
| Trigger | The user selects the “Login” button. |
| Description | The user enters their login credentials, is validated, and is logged into their account. |

|  |  |
| --- | --- |
| Use Case #3 | Course Search |
| Goal in Context | A drop-down menu that shows all of the available courses. |
| Scope & Level | Basic functionality. |
| Preconditions | The user has initiated the “Course Search” dropdown menu. |
| Success End Condition | The user selects a single course & user strikes the “Submit” button. |
| Trigger | The user has initiated the “Course Search” dropdown menu. |
| Description | The user activates a dropdown menu that displays the university’s available courses. |

|  |  |
| --- | --- |
| Use Case #4 | Couse Search List Box |
| Goal in Context | A list box that displays the course information of the selection made by the user during a “course search.” |
| Scope & Level | Basic functionality. |
| Preconditions | Use Case #3. |
| Success End Condition | None. |
| Trigger | None. |
| Description | The user selects a course in the accordion menu. All of the information about the course is displayed in the list box. |

|  |  |
| --- | --- |
| Use Case #5 | Course Full Notification |
| Goal in Context | An alert that notifies a user that a course is full. |
| Scope & Level | Basic functionality. |
| Preconditions | Use Case #4. |
| Success End Condition | The user acknowledges the notification. |
| Trigger | The user selects a course from Use Case #4. |
| Description | The user selects a course shown to have “full “status. A prompt alerts users that they can be added to a waitlist. |

|  |  |
| --- | --- |
| Use Case #6 | User Information |
| Goal in Context | Outputs the user’s information and allows editing. |
| Scope & Level | Basic functionality. |
| Preconditions | The user is on the “About User” page. |
| Success End Condition | None. |
| Trigger | None. |
| Description | Users can review their account information and edit it, including scheduled courses. |

(Cockburn, 1999)

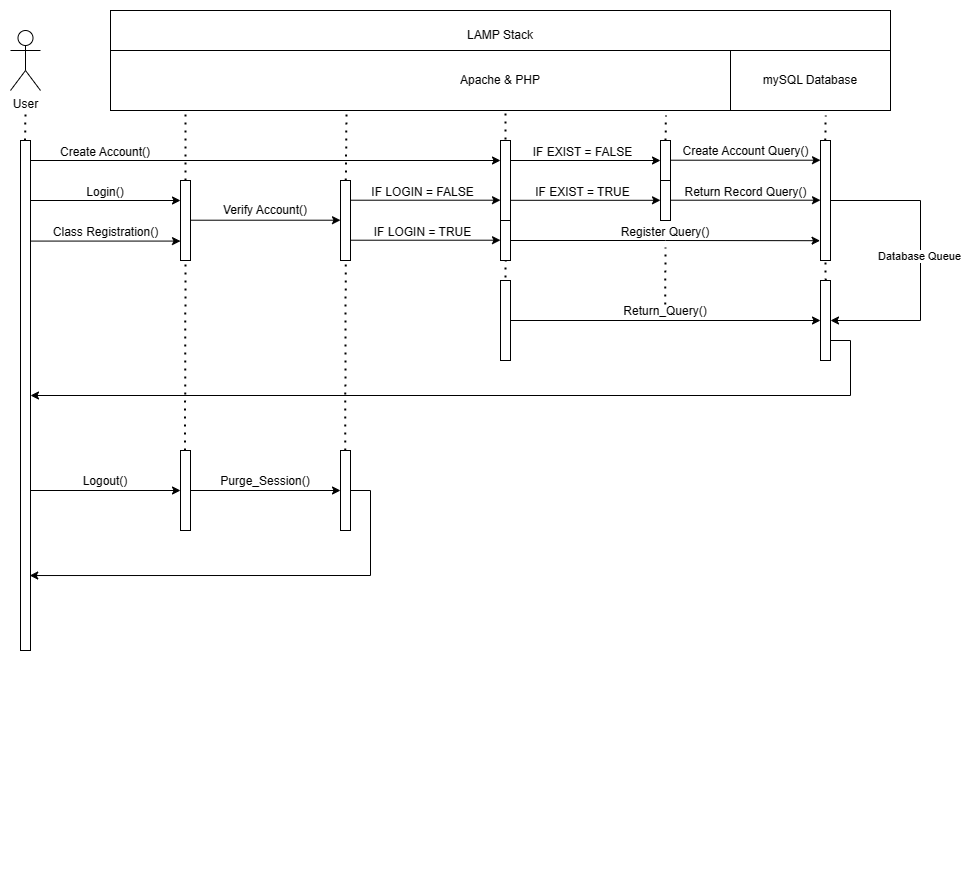
These use cases should be used during the integration testing phase of software development. This is because they represent subsystem level functionality. These use cases represent contracts that each subsystem evaluates; where the “Preconditions” and “Success End Conditions” are the contracts themselves (Richards & Ford, 2020).

### User Classes and Characteristics

Sequence diagrams are a form of interaction model, and are primarily used to portray the interactions of actors and system objects, as well as interactions between the objects themselves. These diagrams show, from left to right, an actor’s initiating action, followed by a potential number of class interactions within the system (Sommerville, 2018).

**Figure 2.**

*Shows a sequence diagram of the UAGC online course registration website.*

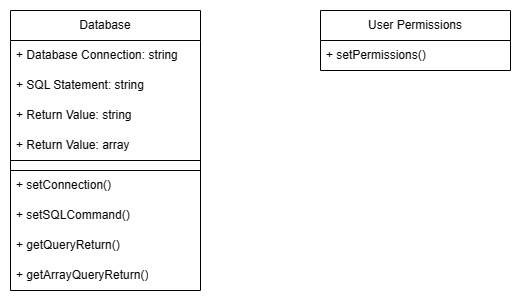


Note. This sequence diagram shows the methods that provide the functionality of the online course registration website.

Class Diagrams are known as structural models, and used to portray the object-oriented classes that exist within a system and their interactions with one another. Associations between classes are shown as link lines between either parameters or methods.

**Figure 3.**

*Shows a class diagram of the UAGC online course registration website.*



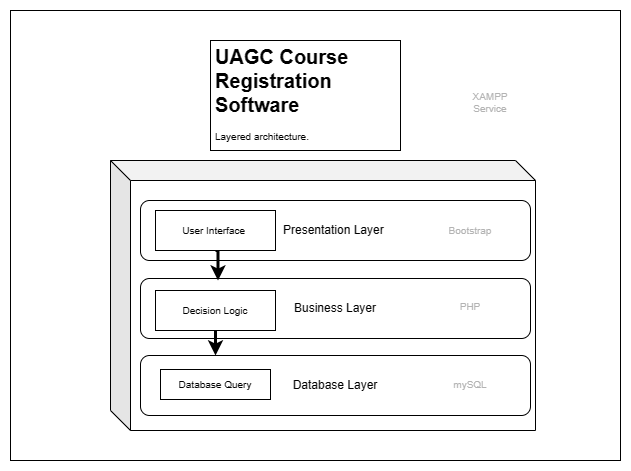
Note. This class diagram shows that in the initial phase of development, the classes used to create system functionality will have no association. It is desirable that eventually a full object-oriented approach to development be taken, but version 1.1 of the software will not implement this feature. Rather the design will take advantage of the individual classes in a very procedural way.

## Design and Implementation Constraints

The architecture of the system will be the layered architectural approach, and be deployed on a Microsoft Windows 10 machine utilizing the Cross-platform, Apache, MySQL, PHP, and Perl (XAMPP) server hosting software.

**Figure 3.**

*Shows the architectural diagram of the UAGC online course registration website.*

**

Note. Although XAMPP is not generally used in deployment environments, it will be used in the initial phases of development because of its ease of operation.

# General

## Nonfunctional requirements

Non-functional requirements describe how a system will work. These requirements are not directly concerned with a specific service that a user interacts with. Non-functional requirements specify or constrain the system’s operation (Sommerville, 2018).

### Safety requirements

Because the UAGC software application represents an educational institution that requires students to be a certain age, and meet educational pre-requisites the system will need safe guards to ensure that these features are implemented. The system should also ensure that all information presented to the user is accurate and understandable.

### Security Requirements

The information that is stored on UAGC servers, as well as transmitted a crossed the internet poses security concerns and liabilities to both the UAGC and the clients utilizing the software. Therefore, the system shall maintain a level of no less than SHA-256 encryption at all times when transmitting information to and from the recipient of the service. All personally identifiable information present within the server-side application shall be hashed or encrypted.

### Software Quality Attributes

Software quality attributes are features that provide measurable metrics of the performance of a software product (Richards, 2020). The software quality attributes of the system are:

**Reliability**: The system should be available whenever a user attempts to access it.

**Usability**: The system must be easy enough for someone with an intermediate level of technological understanding.

**Maintainability**: The system must be updatable and programmed with modern acceptable coding practices.

### Business Rules

Business rules are logical statements that define the behavior and operation of a business. They outline the rules, not the implementation (Tsui et al., 2018). The business rules to be applied to the system are:

* The system should not allow a user to register for a class twice.
* User profiles must include Name, Phone Number, and Email.
* Users must be able to log in at any point during the registration process.
* Students must be able to see courses offered during any semester within the year.
* Courses must have a maximum number of openings. They must not allow more than the maximum number of students to register for a class.
* If a user desires to register for a full class, they may choose to be placed on a waiting list.
* A user may cancel their enrollment in any class they are registered for.
* The system must alert students on a waiting list in descending order of course availability.

References

Cockburn, A. (1999). *Writing effective use cases*. Pearson Education. ISB 0-201-70225-8

Purdue. (n.d.). *Software requirements specification.* SC 30700. Retrieved October 18, 2022, from

<https://eee.cs.purdue.edu/homes/bxd/307>

Richards, M., Ford, N. (2020). *Fundamentals of software architecture*. O’Reilly Media Inc. <http://oreilly.com/catalog/errata.csp?isbn=9781492043454>

Sommerville, I. (2018). Software engineering (10th Edition). Pearson.

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