**Software Requirements Specification**

**for**

**Coarse Enrollment System**

**Version 1.0 approved**

**Prepared by Ben Merritt**

**The University of Arizona Global Campus**

**CST 499: Capstone for Computer Software Technology**

**Dr. Charmelia Butler**

**Date Created December 11, 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 2

1.5 References 2

**2.** **Overall Description 3**

2.1 Product Perspective 3

2.2 Product Functions 4

2.3 User Classes and Characteristics 5

2.4 Operating Environment 5

2.5 Design and Implementation Constraints 6

2.6 User Documentation 6

2.7 Assumptions and Dependencies 7

**3.** **External Interface Requirements 7**

3.1 User Interfaces 7

3.2 Hardware Interfaces 8

3.3 Software Interfaces 8

3.4 Communications Interfaces 8

**4.** **System Features 9**

4.1 Initial Home Page 9

4.2 Sample Course Showcase 9

4.3 New User Registration and Profile Creation 10

4.4 User Login 11

4.5 User Home Page 12

4.6 Announcements and Notifications 13

4.7 Course Enrollment 14

4.8 Enrollment Cancellation 15

4.9 Waiting List Management 16

4.10 User Logout 17

4.11 Session Timeout 17

**5.** **Other Nonfunctional Requirements 18**

5.1 Performance Requirements 18

5.2 Safety Requirements 18

5.3 Security Requirements 19

5.4 Software Quality Attributes 19

5.5 Business Rules 21

**6.** **Other Requirements 24**

**Appendix A: Glossary 25**

**Appendix B: Analysis Models 26**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |

1. **Introduction**
   1. **Purpose**

This document is a software requirements specification (SRS) for an online course enrollment system that aims to provide a convenient and user-friendly platform for students to register for courses, view course offerings, enroll in courses, cancel enrollments, and join waiting lists. The system should also ensure the security and privacy of the user data and prevent unauthorized access or duplication of user IDs. It defines the functional and non-functional requirements for building the course enrollmentsoftware (Chikh & Alajmi 2014). This document is for version 1.0 of the course enrollment system. The SRS covers all the aspects of the initial development, such as the External Interface Requirements, the system features in section 4, and other non-functional requirements in section 5.

* 1. **Document Conventions**

This SRS document has followed certain typographical conventions to enhance readability and comprehension (Chikh & Alajmi 2014). Standard fonts such as Times Roman have been used for the main text, while section headings are in bold for easy navigation. Priority levels for requirements have been indicated using High, Medium, and Low priority. It is assumed that higher-level requirements' priorities are inherited by their detailed requirements, ensuring consistency in prioritization throughout the document.

* 1. **Intended Audience and Reading Suggestions**

This document is intended for developers, project managers, designers, architects, testers, and other stakeholders involved in developing the course enrollment platform (Chikh & Alajmi 2014). It is recommended to begin with the "Introduction" section and proceed to the areas most relevant to the reader's role.

* 1. **Product Scope**

The course enrollment system is a comprehensive online platform designed to address the challenges associated with students enrolling in online courses. The primary objective of this system is to streamline the enrollment process by providing new user registration with unique identifiers and password protection. Each user's profile, containing essential information such as name, phone, and email, facilitates efficient communication and user management. Post-registration, users can log in using their unique ID and password. The system supports scheduling online courses across three semesters (spring, summer, and fall), allowing students to view and select available classes each term. To manage course capacities, the system establishes maximum enrollments for each course and provides waitlists when courses are full. Additionally, users can cancel enrollments, triggering notifications to the next student on the waitlist. This Student Enrollment System aligns with organizational objectives by enhancing the efficiency of the enrollment process, improving user experience, and supporting strategic goals related to accessible and streamlined online education services.

* 1. **References**

Ada-compliance (2010, September 15). *2010 ADA Standards ADA Regulations for Accessible Design*. Ada-Compliance.com. Retrieved from <https://www.ada-compliance.com/ada-compliance/2010-ada-standards>

Altexsoft. (25 July 2022). Non-functional Requirements: Examples, Types, How to Approach. Retrieved from <https://www.altexsoft.com/blog/non-functional-requirements/>

Chikh, A., & Alajmi, H. (2014, January 17–19). [*Towards a dynamic software requirements specification.*](https://go.openathens.net/redirector/ashford.edu?url=https%3a%2f%2fieeexplore.ieee.org%2fdocument%2f6916656%3farnumber%3d6916656%26SID%3dEBSCO%3aedseee)[Paper presentation]. 2014 World Congress on Computer Applications and Information Systems (WCCAIS), Hammamet, Tunisia. <https://doi.org/10.1109/WCCAIS.2014.6916656>

Lan, T., & Li, M. (2022). Design of college digital enrollment System based on big data platform in digital communication era. *2022 6th International Conference on Wireless Communications and Applications (ICWCAPP), Wireless Communications and Applications (ICWCAPP), 2022 6th International Conference on, ICWCAPP*, 196–200. <https://doi.org/10.1109/ICWCAPP57292.2022.00055>

Leadingagile. (2012). *Agile Budgeting: How to Calculate for Agile Teams*. Leadingagile.com. Retrieved from <https://www.leadingagile.com/2012/11/calculate-budgets-agile-team/>

Spillner, A., Linz, T., & Schaefer, H. (2014). [*Software testing foundations: A study guide for the certified tester exam* (4th ed.)](https://uagc.instructure.com/courses/121431/modules/items/6179936). Rocky Nook.

Tsui, F., Karam, O., & Bernal, B. (2018). [*Essentials of software engineering*](https://uagc.instructure.com/courses/124945/modules/items/6353220)(4th ed.). Jones & Bartlett Learning.

Wiegers, K. E. (1999). Software requirement specifications for <project>. [Template]. <https://web.cs.dal.ca/~hawkey/3130/srs_template-ieee.doc>

1. **Overall Description**
   1. **Product Perspective**

The course enrollment system is a new, standalone product developed to streamline the online course enrollment process. It does not replace existing systems but functions independently within the broader context of online education services. The system consists of critical modules, including user registration, course management, and enrollment processing, working together to provide a user-friendly interface for registration and enrollment. As an integral part of the larger online education ecosystem, the system interfaces with external components, such as user databases and course availability databases, to ensure efficient data exchange and system functionality (Lan & Li, 2022). The primary goal is to enhance the user experience by simplifying and optimizing the enrollment process for online courses.

* 1. **Product Functions**

The significant functions the course enrollment platform must perform or let the user perform are:

* New user registration that includes account and profile creation.
* Each new user should have a unique ID associated with a password. The system should guard against two users using the same ID for registration.
* Profiles must include some key information about the applicant including name, phone, email, and any other information you may see necessary.
* Post registration, users can log in to the system at any time using the ID and the password created during the registration process.
* Online courses run through three semesters per year (spring/summer/fall), and students can list the courses that will be offered during any semester, as not all courses will be offered in every semester.
* Each course should have a maximum number of enrollments that may be different depending on the course.
* If a user wants to enroll in a course and the course is full, the student can add themselves to a waiting list.
* A user can also cancel the enrollment from any course they are enrolled in, and the system should inform the first on the waiting list (if any) that they can enroll in the class.
  1. **User Classes and Characteristics**

The course enrollment system anticipates several user classes, including students who are the primary users engaging in regular course registration, instructors responsible for course management, system administrators overseeing maintenance, and waitlist managers addressing enrollment peaks. Students with varied technical proficiency are the most critical users, followed by instructors and administrators with higher technical expertise. Addressing the distinct needs of each user class is essential for the system's successful functionality, prioritizing efficient course management and user-friendly experiences during enrollment processes (Lan & Li, 2022).

* 1. **Operating Environment**

The course enrollment system necessitates cross-platform compatibility and robust mobile responsiveness. The platform's functionality will seamlessly extend across various devices, encompassing desktop computers, laptops, tablets, and smartphones. The platform must ensure compatibility with prevalent web browsers while adapting to diverse screen dimensions. Specifically, the website's operational integrity must encompass a spectrum of leading web browsers, comprising Chrome, Internet Explorer, Edge, Bing, Yahoo, and Safari. Furthermore, its performance shall be optimized across distinct operating systems including Mac, Linux, and Windows, as well as on iOS and Android platforms. This adaptability encompasses any auxiliary applications necessary for seamless operation (Tsui et al., 2018).

* 1. **Design and Implementation Constraints**

The development process of the course enrollment system will be guided by industry-leading principles in user experience design, security protocols, and optimal performance standards. The stipulated timeframe for the development endeavor is a 3-week span, necessitating the utilization of a relational database to underpin the system's architecture. Furthermore, stringent encryption measures will be mandated for all user account passwords to bolster security (Tsui et al., 2018).

In alignment with corporate directives, the website must align with the ADA web accessibility guidelines (Ada-compliance, 2010) to guarantee inclusivity for all users. Additionally, the customer will undertake the system's ongoing maintenance, emphasizing the significance of a design and construction supporting adaptability, sustainability, and verifiability through comprehensive testing processes.

* 1. **User Documentation**

The platform will furnish comprehensive user documentation comprising crafted user manuals and context-sensitive on-screen tooltips, meticulously designed to offer users expert guidance through diverse features and functionalities. In parallel with the specified functionalities of the course enrollment system, the platform will encompass an array of informative resources to aid users in troubleshooting and establishing a user account. Moreover, the platform will boast an integrated chatbot functionality, seamlessly facilitating users to request assistance whenever required (Lan & Li, 2022).

* 1. **Assumptions and Dependencies**

The platform assumes a stable internet connection for users to access and interact with the system. It depends on the availability of accurate product data and timely updates to reflect product availability. Constraints for the project include a low budget, a timeline of 3 weeks, and a project team of no more than 10, which includes designers, architects, Scrum Master, database administrators, developers, testers, and project managers (Leadingagile, 2012).

1. **External Interface Requirements**
   1. **User Interfaces**

The user interface of the course enrollment system prioritizes consistency, employing uniform headers, footers, and navbars across all pages. Font Awesome exclusively provides icons for cohesive design, while a standardized back arrow ensures seamless navigation. Accessibility compliance is ensured through meticulously crafted alt tags for images, facilitating users with impaired vision. A keyboard-navigable interface allows users to tab through each page effortlessly. Attention to detail extends to color selection for both visual appeal and heightened usability, with font choices prioritizing readability. Adherence to these principles ensures unobtrusive advertisements, maintaining a consistent layout for a harmonized user experience. The home button's prominent placement on every page facilitates swift navigation, and forms provide comprehensive feedback for error correction. By embracing these guidelines, the course enrollment system aims to exemplify professionalism, usability, and aesthetic refinement.

* 1. **Hardware Interfaces**

The platform will function on standard hardware devices with web browsing capabilities. The site will use up-to-date hardware and software to make requests to transfer resources from the server to the client. The website should also support screen readers and other ADA-compliant devices.

* 1. **Software Interfaces**

The platform will establish seamless connectivity with databases housing comprehensive product information, employing Near Real Time Indexing and Search (NRT) mechanisms to facilitate agile updates. The website must attain compatibility across diverse operating systems, encompassing iOS, Android, Mac, and Windows environments. Essential to this endeavor is integrating an API that effectively communicates with the relational database, enabling efficient storage and retrieval of pivotal user, course, and transactional data (Lan & Li, 2022). Furthermore, the future platform could connect to a payment system for tuition fees and a grading system.

* 1. **Communications Interfaces**

The platform will establish communication via conventional internet protocols, ensuring robust and dependable data interchange between users and the application. Users will be able to receive email notifications upon confirmation, along with the option to subscribe to newsletters and messages tailored to their preferences and history. Furthermore, it is paramount that all personally identifiable information (PII) remains safeguarded, and the encryption of all passwords is imperative to uphold stringent security measures (Tsui et al., 2018).

1. **System Features**
   1. **Initial Home Page**

4.1.1 Description and Priority

This feature establishes the system's initial home page, providing users with a welcoming and informative landing space before logging in. It is of Medium priority to offer potential users a glimpse of the platform's features and benefits.

4.1.2 Stimulus Sequences

The user accesses the platform without logging in.

4.1.3 Response Sequences

The system displays an initial home page with general information and key features.

4.1.4 Functional Requirements

REQ-1: The initial home page shall provide an overview of the platform's purpose and benefits.

REQ-2: A brief registration/login section shall be available on the initial home page.

REQ-3: Users can access sample course offerings and program highlights without logging in.

REQ-4: The system shall include a call-to-action for users to register or log in for a more personalized experience (Lan & Li, 2022).

* 1. **Sample Course Showcase**

4.2.1 Description and Priority

This feature complements the initial home page by showcasing a selection of sample courses and offering a preview of the educational opportunities available on the platform. It is of Medium priority to attract potential users and generate interest in the course offerings.

4.2.2 Stimulus Sequences

The user explores the initial home page.

4.2.3 Response Sequences

The system displays a curated selection of sample courses with brief descriptions.

4.2.4 Functional Requirements

REQ-1: The system shall present a visually appealing section on the initial home page featuring sample courses.

REQ-2: Each sample course shall include a title, brief description, and key highlights.

REQ-3: Users can click on a sample course to view more detailed information.

REQ-4: The system shall provide a seamless transition for users to the full course catalog and registration upon clicking a sample course (Lan & Li, 2022).

* 1. **New User Registration and Profile Creation**

4.3.1 Description and Priority

This feature enables new users to register on the platform, creating a unique account with a password. It is of High priority as it establishes the foundation for user interaction within the system.

4.3.2 Stimulus Sequences

The user accesses the registration page.

The user provides the necessary information for registration.

The user submits the registration form.

4.3.3 Response Sequences

The system validates the provided information.

If valid, the system creates a unique user ID and password.

The user receives a confirmation message.

4.3.4 Functional Requirements

REQ-1: The system shall verify that the entered user ID is unique.

REQ-2: User profiles must include fields for name, phone, email, and any other necessary information.

REQ-3: Upon successful registration, the system shall generate a unique user ID and securely store the password.

REQ-4: The system shall provide error messages for invalid inputs during registration (Lan & Li, 2022).

* 1. **User Login**

4.4.1 Description and Priority

This feature allows registered users to log in to the system using their unique ID and password. It is of High priority to ensure secure and personalized access to the platform.

4.4.2 Stimulus Sequences

The user enters their unique ID and password.

The user submits the login form.

4.4.3 Response Sequences

The system verifies the credentials.

If valid, the user gains access to their account.

If invalid, the system displays an error message.

4.4.4 Functional Requirements

REQ-1: The system shall authenticate users based on their unique ID and password.

REQ-2: After successful login, the system shall redirect the user to their profile/dashboard.

REQ-3: The system shall lock the account temporarily after a specified number of unsuccessful login attempts.

REQ-4: Password recovery/reset functionality shall be available (Lan & Li, 2022).

* 1. **User Home Page**

4.5.1 Description and Priority

Upon logging in, this feature provides users with a personalized home page, offering a central hub for accessing essential information and actions. Enhancing user engagement and navigation is of medium priority.

4.5.2 Stimulus Sequences

The user successfully logs into the system.

4.5.3 Response Sequences

The system redirects the user to their personalized home page.

4.5.4 Functional Requirements

REQ-1: The home page shall display a welcome message with the user's name.

REQ-2: The system shall present a summary of enrolled courses and their current status.

REQ-3: Users can easily navigate to the course enrollment section from the home page.

REQ-4: Important announcements or notifications shall be displayed on the home page.

REQ-5: The system shall provide quick links to profile settings and password change functionality (Lan & Li, 2022).

* 1. **Announcements and Notifications**

4.6.1 Description and Priority

This feature enables the system to broadcast announcements and notifications to users, ensuring timely communication of important information. It is of High priority to keep users informed and engaged.

4.6.2 Stimulus Sequences

System administrators create an announcement.

Events trigger notifications (e.g., enrollment status changes).

4.6.3 Response Sequences

Users receive announcements on their home page.

Notifications are displayed promptly based on relevant events.

4.6.4 Functional Requirements

REQ-1: The system shall allow administrators to create and publish announcements.

REQ-2: Users shall receive notifications for significant events (e.g., enrollment changes, waiting list, system updates).

REQ-3: The home page shall prominently display the latest announcements.

REQ-4: Users can customize notification preferences (Lan & Li, 2022).

* 1. **Course Enrollment**

4.7.1 Description and Priority

This feature allows users to enroll in courses offered during the three semesters. It is of High priority as it directly impacts the system's core functionality.

4.7.2 Stimulus Sequences

The user selects a semester and views available courses.

The user clicks on a course to enroll.

4.7.3 Response Sequences

The system checks for available slots.

If available, the user is enrolled in the course.

If full, the user is prompted to join a waiting list.

4.7.4 Functional Requirements

REQ-1: The system shall display a list of courses available in each semester.

REQ-2: Each course shall have a maximum enrollment limit.

REQ-3: If a course is full, the user can opt to join a waiting list.

REQ-4: Users can view and manage their enrolled courses.

REQ-5: The system shall notify the first user on the waiting list when a slot becomes available (Lan & Li, 2022).

* 1. **Enrollment Cancellation**

4.8.1 Description and Priority

This feature allows users to cancel their enrollment in a course and triggers notifications to the first person on the waiting list (if any). It is of High priority to provide flexibility and maintain an efficient enrollment process.

4.8.2 Stimulus Sequences

The user accesses the list of enrolled courses.

The user selects a course to cancel enrollment.

4.8.3 Response Sequences

The system confirms the cancellation.

If a waiting list exists, the first person is notified of the available slot.

4.8.4 Functional Requirements

REQ-1: The system shall allow users to view a list of courses in which they are currently enrolled.

REQ-2: Users can select a course for cancellation.

REQ-3: Upon cancellation, the system updates the enrollment status for the user.

REQ-4: If a waiting list exists for the course, the system shall notify the first person on the waiting list.

REQ-5: The system shall provide confirmation messages for successful and unsuccessful cancellation attempts (Lan & Li, 2022).

* 1. **Waiting List Management**

4.9.1 Description and Priority

This feature efficiently manages waiting lists and ensures prompt notifications and enrollment updates. It is of Medium priority to enhance the overall user experience.

4.9.2 Stimulus Sequences

The system detects an available slot due to an enrollment cancellation.

The first person on the waiting list is notified.

4.9.3 Response Sequences

If the notified user confirms, the system enrolls them in the course.

If not, the system notifies the next person on the waiting list.

4.9.4 Functional Requirements

REQ-1: The system shall maintain a waiting list for courses that are currently full.

REQ-2: The first person on the waiting list shall be notified when a slot becomes available.

REQ-3: The system shall update enrollment status based on user responses from the waiting list notifications.

REQ-4: If the notified user does not confirm within a specified time, the system shall notify the next person on the waiting list (Lan & Li, 2022).

* 1. **User Logout**

4.10.1 Description and Priority

This feature allows users to log out of the system, terminating their current session for security and privacy purposes. It is of Medium priority to ensure user control over their active sessions.

4.10.2 Stimulus Sequences

The user clicks on the logout button.

4.10.3 Response Sequences

The system terminates the user's session and redirects them to the login or initial home page.

4.10.4 Functional Requirements

REQ-1: The system shall provide a visible and accessible logout button on all user pages.

REQ-2: The system shall clear the user's session data upon logout.

REQ-3: Users shall be redirected to the login or initial home pages after logging out.

REQ-4: The system shall display a confirmation message upon successful logout (Lan & Li, 2022).

* 1. **Session Timeout**

4.11.1 Description and Priority

This feature ensures the security of user sessions by automatically logging out users after a specified period of inactivity. It is of High priority to prevent unauthorized access to active sessions.

4.11.2 Stimulus Sequences

The user remains inactive for the defined session timeout period.

4.11.3 Response Sequences

The system automatically logs out the user and redirects them to the login or initial home page.

4.11.4 Functional Requirements

REQ-1: The system shall implement a session timeout mechanism.

REQ-2: The session timeout period shall be configurable by administrators.

REQ-3: Users shall receive a warning before the session timeout occurs.

REQ-4: The system shall securely handle session data and clear it upon timeout (Lan & Li, 2022).

1. **Other Non-functional Requirements**
   1. **Performance Requirements**

The platform shall handle concurrent user requests, ensuring a smooth user experience during peak traffic. The website should load quickly, with a response time of less than 3 seconds for each page (Altexsoft, 2022).

* The system should handle thousands of concurrent users during peak periods.
  1. **Safety Requirements**
* User input and interactions shall be treated with confidentiality and privacy (Tsui et al., 2018).
* Payment information and personal data shall be encrypted and securely transmitted.
  1. **Security Requirements**
* The platform shall implement user authentication mechanisms to prevent unauthorized access (Tsui et al., 2018).
* Users shall be notified of secure connections (HTTPS) to ensure safe data transmission.
* Data transmission and payment processing should be secured using the latest SSL encryption (Lan & Li, 2022).
* User data should be stored securely with proper encryption mechanisms.
* The platform should comply with PCI DSS standards for payment security (Altexsoft, 2022).
  1. **Software Quality Attributes**
* Usability:
* Quality Characteristic: The ease of use and user satisfaction while interacting with the product.
* Preference: High usability is a top priority, focusing on intuitive navigation, clear layouts, and a minimal learning curve.
* Reliability:
* Quality Characteristic: The ability of the product to perform consistently and without failures.
* Preference: High reliability ensures users can access and manage enrollment without disruptions or errors.
* Maintainability:
* Quality Characteristic: The ease of making modifications or updates to the product.
* Preference: The product should be designed for easy maintenance, allowing efficient updates, bug fixes, and enhancements.
* Portability:
* Quality Characteristic: The product can run on different platforms and environments.
* Preference: Moderate portability is essential, ensuring compatibility across major operating systems and browsers.
* Robustness:
* Quality Characteristic: The product can handle unexpected inputs or situations without crashing.
* Preference: High robustness is desired to prevent system failures due to invalid inputs or unexpected events.
* Interoperability:
* Quality Characteristic: The ability of the product to work seamlessly with external systems and interfaces.
* Preference: Moderate interoperability is necessary, with a focus on integrating with payment gateways and other relevant APIs.
* Testability:
* Quality Characteristic: The ease of testing the product's functionality, performance, and security.
* Preference: High testability ensures thorough testing, identifies issues early, and provides a reliable product.
* Adaptability:
* Quality Characteristic: The product's capability to accommodate changes in requirements or technology.
* Preference: Moderate adaptability is needed to ensure the product remains relevant as business needs evolve.
* Availability:
* Quality Characteristic: The product's accessibility and uptime for users.
* Preference: High availability is crucial to prevent downtime and ensure users can manage inventory whenever needed.
* Correctness:
* Quality Characteristic: The accuracy and correctness of the product's functionality and outputs.
* Preference: High correctness is essential for accurate course enrollment management and reliable information.
* The platform shall be designed for high usability, ensuring ease of use and navigation for users of varying technical expertise.
* The platform shall be maintainable, allowing for efficient updates and enhancements as needed (Spillner et al., 2014).
  1. **Business Rules**
* The platform shall correctly identify user search queries and their associated categories to enhance search precision.
* Search suggestions shall include category information to guide users toward relevant search results.
* Role-based Access Control:
* Operating Principle: Only registered Students, Teachers, Administrators, and Resource personnel with authenticated accounts can perform enrollment management functions.
* Implication for Functional Requirements: Implement user authentication and authorization mechanisms to allow access to functions based on user roles.
* Profile and Course Ownership:
* Operating Principle: Each user can only manage and modify their own profile and courses.
* Implication for Functional Requirements: Implement a user-specific session context to ensure that users can only manipulate profiles and courses associated with their accounts.
* Quantity Limits and Thresholds:
* Operating Principle: Users cannot set course quantities beyond predefined limits or below specific thresholds.
* Implication for Functional Requirements: Implement validation checks during quantity updates to ensure adherence to defined limits and thresholds.
* Locking during Updates:
* Operating Principle: When a user updates information, the system should temporarily lock to prevent concurrent updates.
* Implication for Functional Requirements: Implement a locking mechanism to prevent multiple users from editing the same information simultaneously, ensuring data consistency.
* Notification Triggers:
* Operating Principle: Users receive notifications when course quantities reach specified minimum thresholds.
* Implication for Functional Requirements: Develop a notification system that triggers alerts when quantities fall below defined levels, allowing users to take prompt action.
* Error Handling and Recovery:
* Operating Principle: If an error occurs during enrollment operations, the system should provide clear error messages and suggestions for resolution.
* Implication for Functional Requirements: Develop comprehensive error handling mechanisms, including informative error messages and user troubleshooting steps (Tsui et al., 2018).
* Audit Trail and Logging:
* Operating Principle: All actions and changes should be logged for accountability and audit purposes.
* Implication for Functional Requirements: Implement logging features that record details of inventory-related actions performed by users, facilitating accountability and tracking changes.
* Time-sensitive Updates:
* Operating Principle: Changes made by users, such as course additions, deletions, or updates, should be reflected in real-time or near real-time.
* Implication for Functional Requirements: Utilize Near Real Time Indexing and Search (NRT) capabilities to ensure that updates are immediately visible and consistent across the platform (Lan & Li, 2022).

1. **Other Requirements**

* The platform shall provide a mechanism for administrators to manage and update course and profile information, including adding and removing data.
* The platform shall offer a feedback mechanism for users to report inaccuracies or issues with information.
* Usability:
* The user interface should be intuitive, with easy navigation and clear course categorization.
* Users should be able to complete an enrollment with a maximum of 5 steps in the registration process.
* Reliability:
* The platform should have high uptime, with minimal downtime for maintenance or updates.
* The system should be able to recover quickly in case of any failures.
* Scalability:
* The software should be designed to handle increased user traffic as the platform grows.
* Scalability should be achieved through cloud-based infrastructure.
* Accessibility:
* The platform should adhere to WCAG guidelines to ensure accessibility for disabled users.
* Data Backup and Recovery:
* Regular automated data backups should be performed to ensure data integrity.
* The system should have a robust disaster recovery plan in case of data loss (Spillner et al., 2014).
* Compatibility:
* The software should be compatible with major browsers (Chrome, Firefox, Safari, Edge) and different devices (desktops, smartphones, tablets).
* Maintenance and Updates:
* Regular updates, bug fixes, and security patches should be provided to maintain optimal performance (Spillner et al., 2014).
* Updates should not disrupt user experience.
* Integration with External Systems:
* The integration with the education management system should be seamless and real-time.
* SEO Optimization:
* The platform should implement effective SEO strategies to improve search engine visibility and organic traffic.
* Customer Support:
* The customer support channels should be responsive, with email inquiries answered within 24 hours and live chat available during business hours.

**Appendix A: Glossary**

**Near Real Time Indexing and Search (NRT):** capabilities to ensure that updates are immediately visible and consistent across the platform.

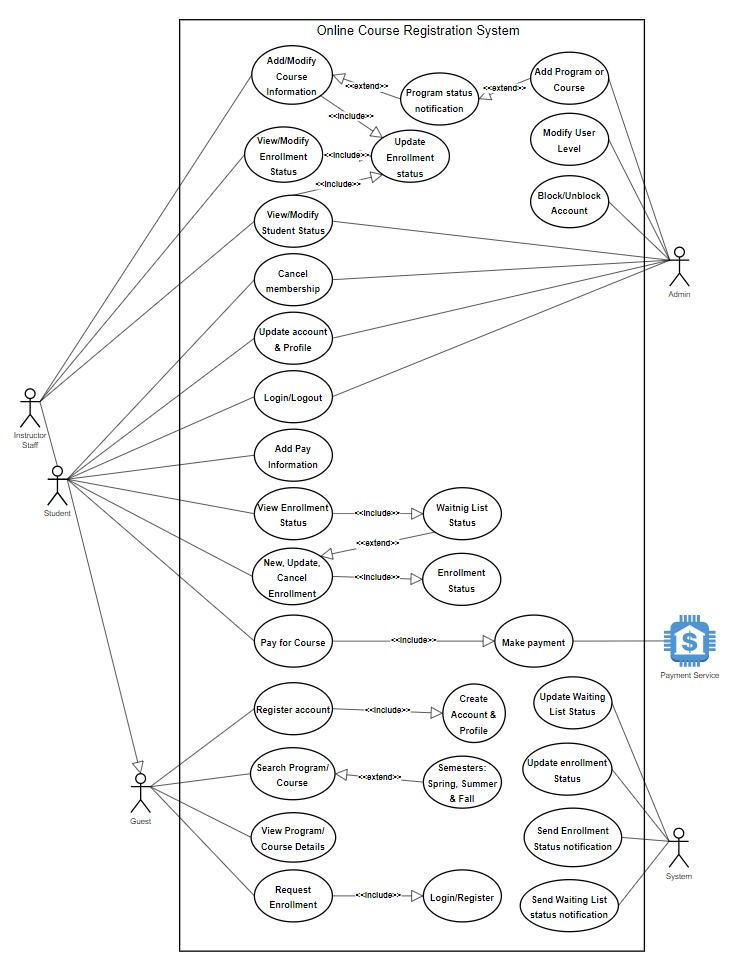
**Search Engine Optimization (SEO):** involves optimizing a website's technical configuration, content relevance, and link popularity to enhance findability, relevance, and popularity in response to user search queries, ultimately leading to improved search engine rankings.

**WCAG guidelines:** ensure accessibility for users with disabilities.

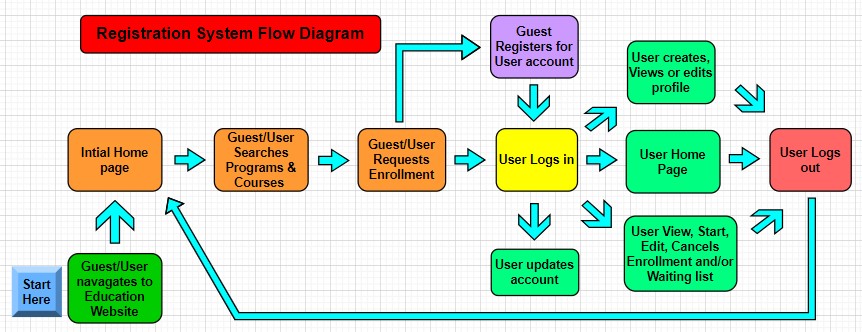
**Appendix B: Analysis Models**

**UML Design Diagrams for the Course Registration System**

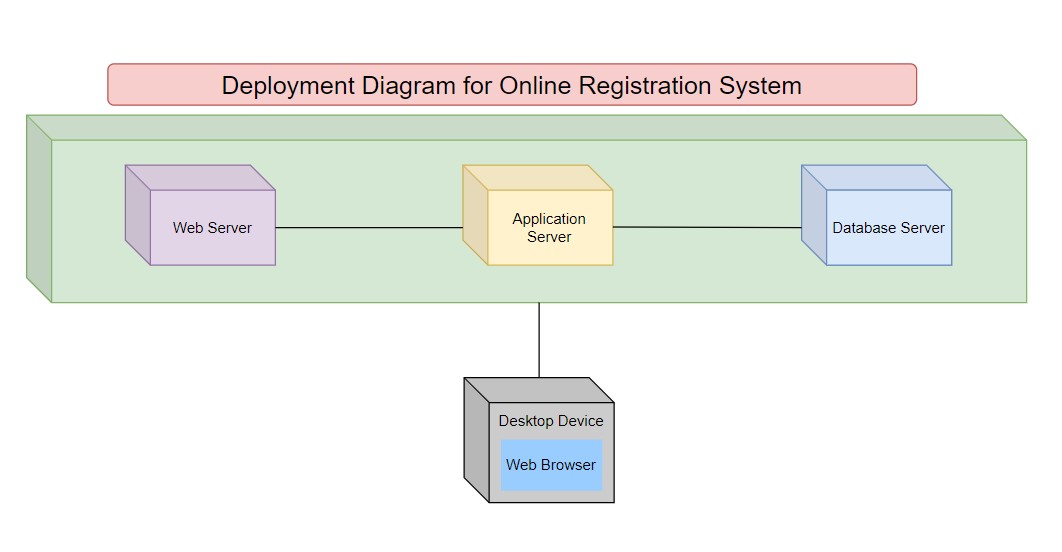
**Figure 1***.* Online Course Registration System Use Case Diagram.



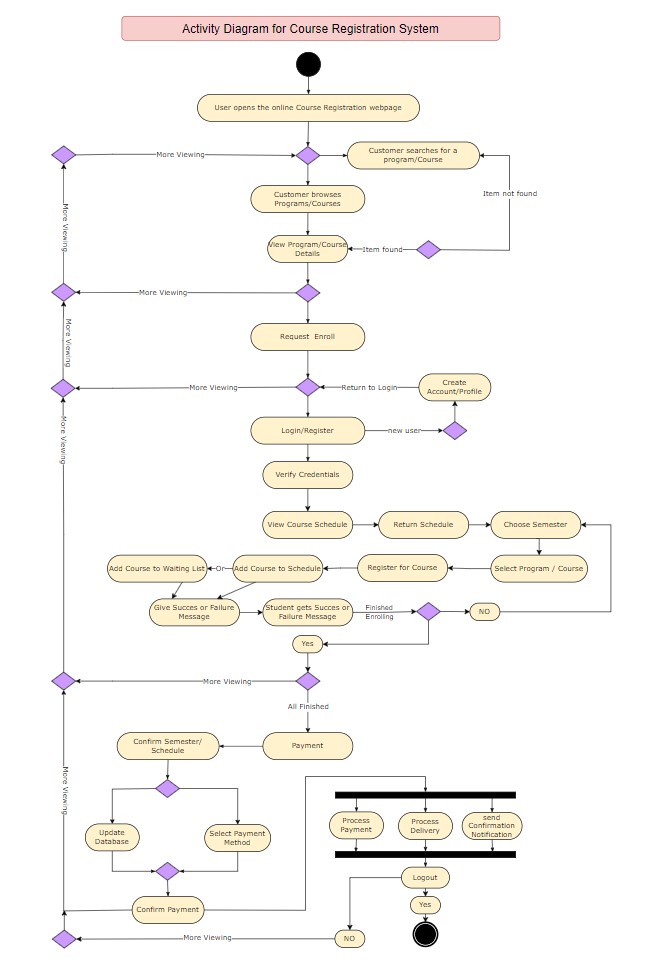
**Figure 2.** Online Course Registration System Flow Diagram.



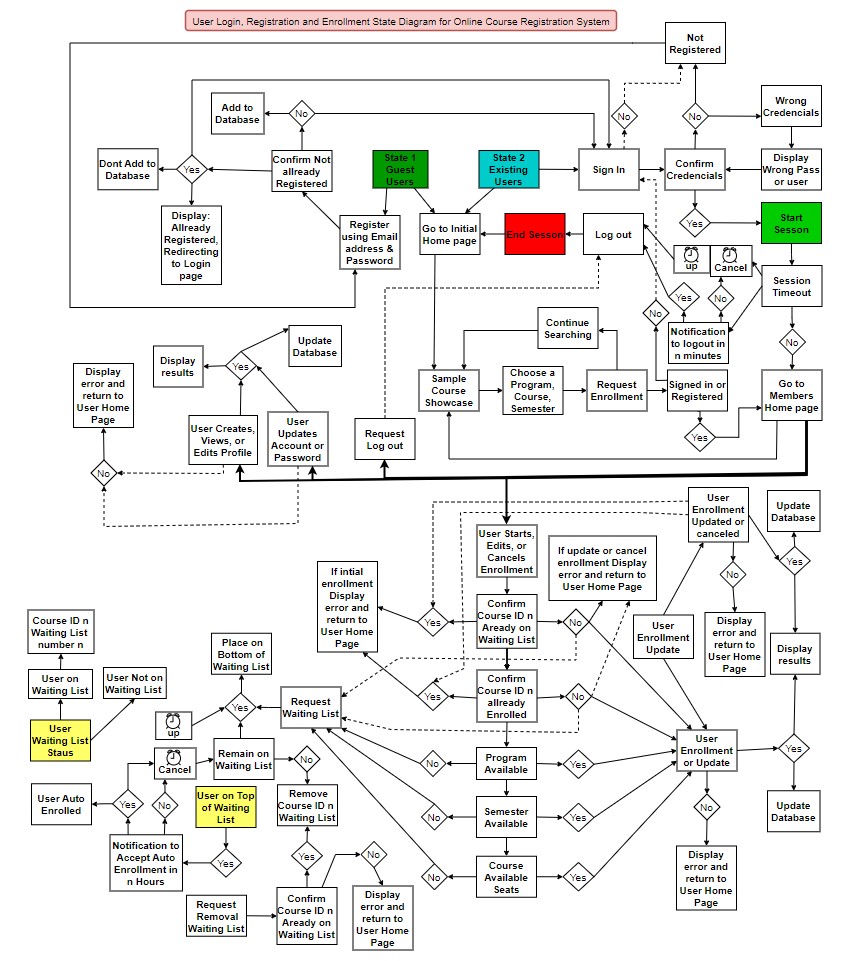
**Figure 3.** Online Course Registration System Deployment Diagram.



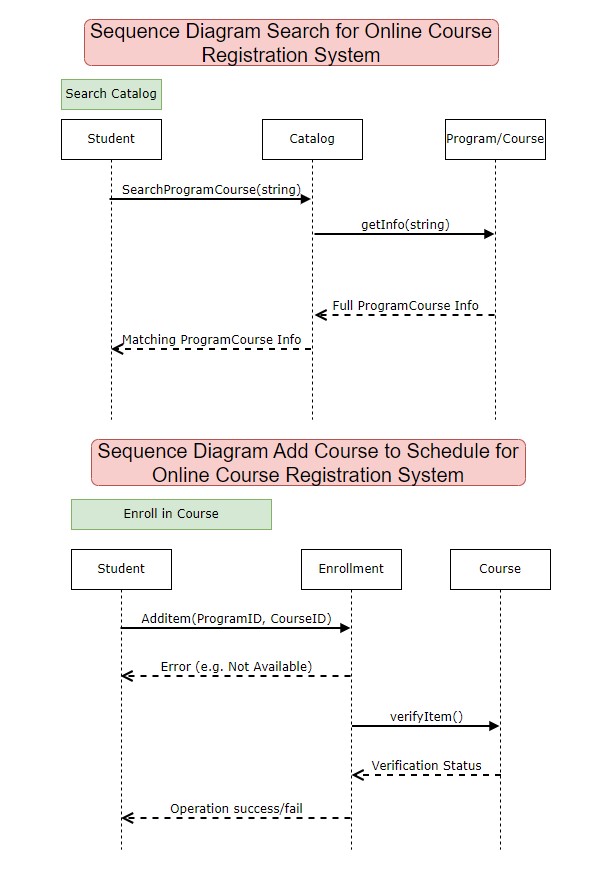
**Figure 4.** Online Course Registration System Activity Diagram.



**Figure 5.** Online Course Registration System State Diagram.



**Figure 6.** Online Course Registration System Sequence Diagrams.



**Figure 7.** Online Course Registration System Class Diagram.

