



## **Assignment 1**

**Muhammad Usman**

**Sp23701**

**Numl-S23-26737**

**5<sup>th</sup> Semester**

**SCD**

**16 May-2025**

**Submitted to: Ms. Jaweria Kanwal**

**National University of Modern Languages**

**Islamabad**

---

# **Software Requirement Specification**

**for**

**<Online Course Enrollment System>**

**Version 1.0 Approved**

**Prepared By <Muhammad Usman>**

**<15 May 2025>**

# Functionalities:

## 1. User login/signup:

- **Description:** Allows new users to sign up as students, instructors, or admins and existing users to log in securely.
- **Purpose:** Provides authentication and personalized access to the system based on user roles.

## 2. Role-Based Dashboards:

- **Description:** After login, users are redirected to their specific dashboard (student, instructor, admin).
- **Purpose:** Organizes and limits access to relevant features based on user role to maintain security and usability.

## 3. View Course Catalog:

- **Description:** Students can browse and search through a list of available courses with filters (e.g., subject, level, schedule).
- **Purpose:** Helps students easily discover and select relevant courses.

## 4. Course Enrollments:

- **Description:** Students can enroll in selected courses by checking prerequisites and seat availability.
- **Purpose:** Ensures proper access to courses while maintaining academic integrity and class limits.

## 5. Course prerequisite verification:

- **Description:** Before enrolling, the system checks if the student has completed required courses.
- **Purpose:** Maintains course structure and prevents students from enrolling without foundational knowledge.

## 6. Real-Time Seat Availability:

- **Description:** The system updates seat counts dynamically as students enroll or drop courses.
- **Purpose:** Prevents over-enrollment and ensures fairness in course registration.

## 7. Drop/Withdraw Course:

- **Description:** Students can drop a course before a specified deadline.
- **Purpose:** Allows flexibility in course planning and helps manage academic load.

## 8. Instructor Course Management:

- **Description:** Instructors can add, edit, or remove courses including course title, description, schedule, prerequisites, and seat capacity.
- **Purpose:** Gives instructors full control over course offerings and updates.

## **9. Schedule Viewer:**

- **Description:** Students and instructors can view their class schedules in a calendar or table format.
- **Purpose:** Helps users manage their academic time and avoid schedule conflicts.

## **10. Admin User & Course Management:**

- **Description:** Admin can approve/reject instructor accounts, manage course data, delete users, and monitor activity logs.
- **Purpose:** Maintains system integrity, verifies instructors, and provides administrative oversight.

# **Functional Requirements:**

## **1. User Registration & Login:**

- The system shall allow users to register as a student, instructor, or admin.
- The system shall authenticate users based on username and password during login.

## **2. Role-Based Dashboards:**

- The system shall redirect users to their respective dashboards after successful login.
- The dashboard shall display features based on the user's role.

## **3. View Course Catalog:**

- The system shall display a list of available courses to students.
- The system shall allow students to filter courses by department, level, and instructor.

## **4. Course Enrollment:**

- The system shall allow students to enroll in available courses.
- The system shall ensure that the number of enrolled students does not exceed the course capacity.

## **5. Course Prerequisite Verification:**

- The system shall check and validate that students meet the prerequisites before enrollment.

## **6. Real-Time Seat Availability:**

- The system shall update the number of available seats in real-time after each enrollment or drop.

## **7. Drop/Withdraw Course:**

- The system shall allow students to drop a course before a predefined deadline.

## **8. Instructor Course Management:**

- The system shall allow instructors to create, update, or delete course offerings.
- Instructors shall be able to define prerequisites, schedules, and seat limits.

## **9. Schedule Viewer:**

- The system shall provide a schedule view for students and instructors showing enrolled or created classes respectively.

## **10. Admin User & Course Management**

- The system shall allow the admin to approve or reject instructor registration requests.
- The admin shall be able to manage user accounts and remove inappropriate content or users.

## **Non-Functional Requirements:**

### **1. Usability:**

- The system shall have a user-friendly interface with intuitive navigation and help prompts.

### **2. Security:**

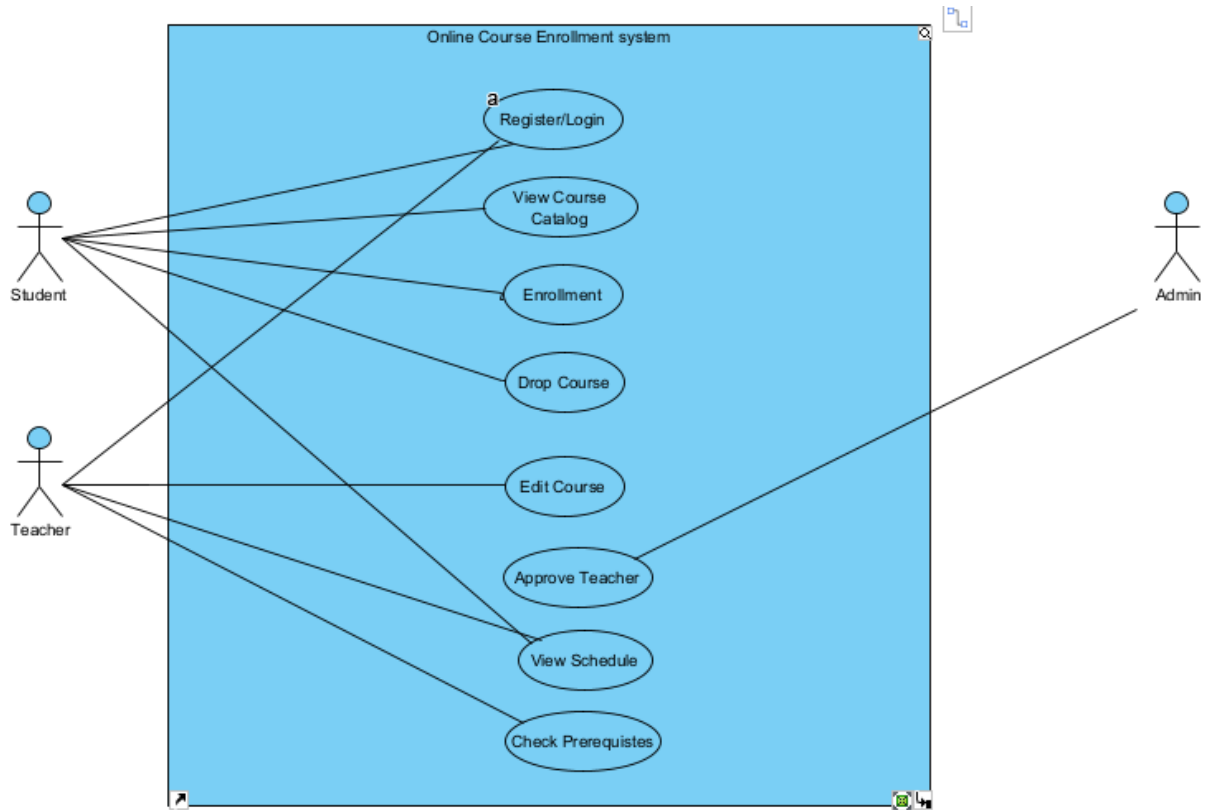
- The system shall securely hash and store passwords.
- Only authorized users shall access specific functionalities based on roles.

### **3. Performance:**

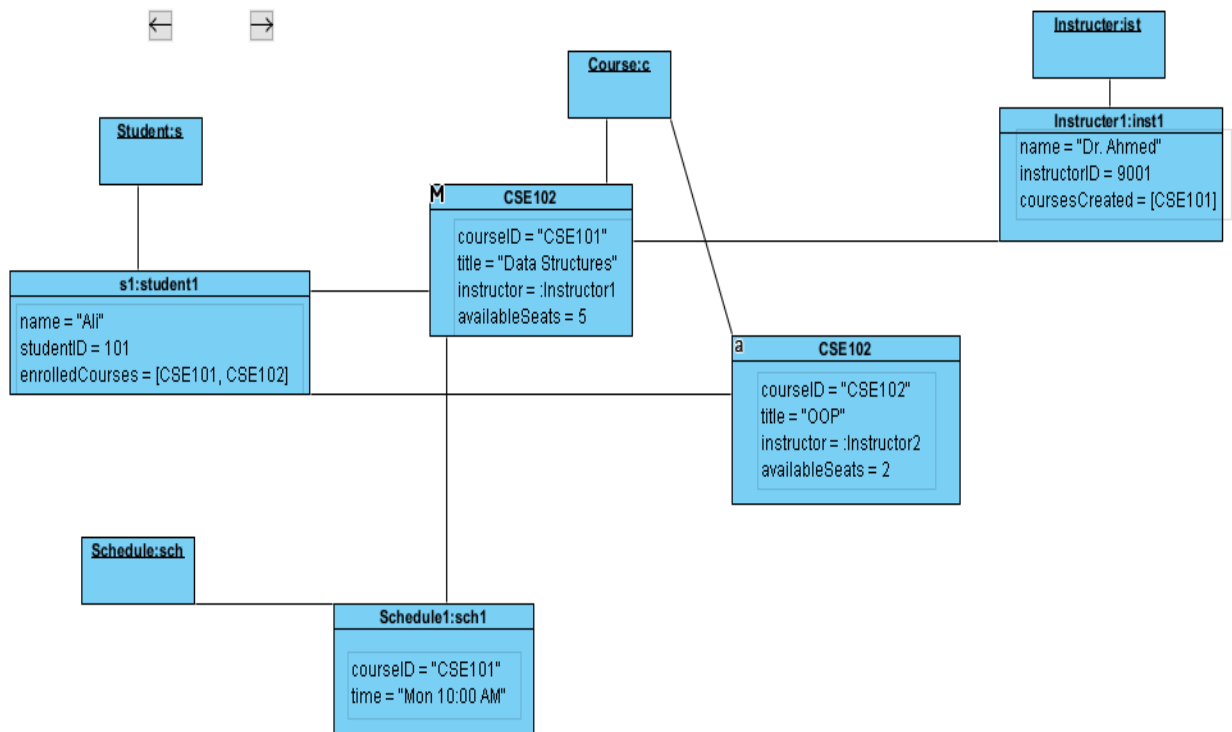
- The system shall process login, enrollment, and updates within 2 seconds.
4. **Reliability:**
    - The system shall ensure that data (e.g., course capacity, user session) remains consistent and accurate during concurrent access.
  5. **Scalability:**
    - The system shall support at least 1000 concurrent users without performance degradation.
  6. **Availability:**
    - The system shall be available 99.5% of the time during working hours (excluding maintenance windows).
  7. **Maintainability:**
    - The system shall be modular, and code shall be well-commented to facilitate future maintenance and updates.
  8. **Compatibility:**
    - The system shall be accessible through modern browsers (Chrome, Firefox, Edge).
  9. **Auditability:**
    - All critical operations (e.g., login, enrollments, drops) shall be logged for audit purposes.
  10. **Accessibility:**
    - The system shall be usable by people with visual impairments via screen readers.

## 2. Detailed Design documents:

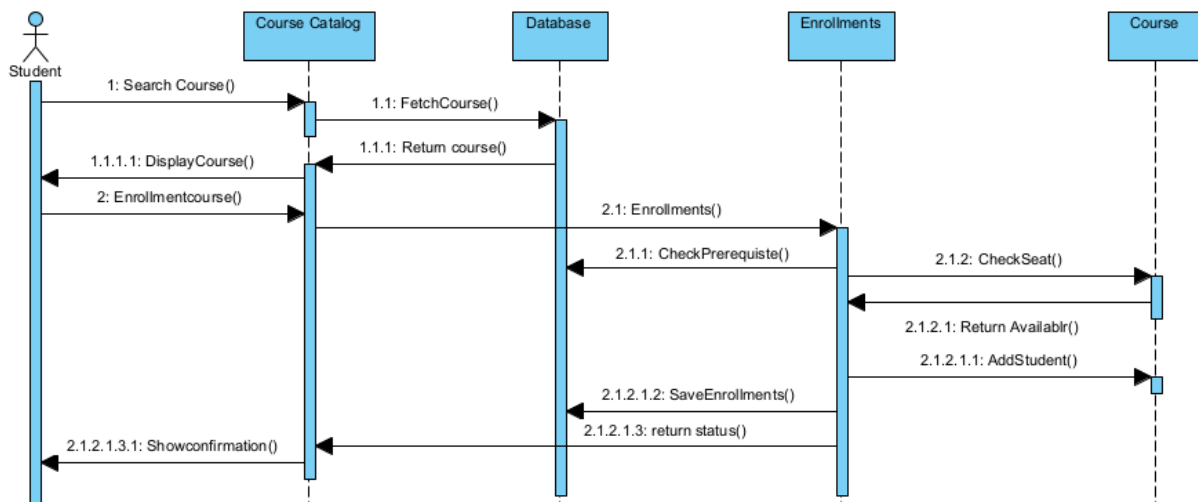
## Use Case Diagram:



## Object Diagram:



## Sequence Diagram:





# Communication Diagram:

