

# System & User Requirements

## Online Course Registration System

Nour Ashour

October 27, 2025

### Part 1: User Requirements (User Stories)

- User Requirements define \*what\* a user can **do** with the system.
- They are often written as "User Stories" to provide context.
- The format is:

*"As a [User Role], I want to [Action], so that [Goal]."*

- We will review the core "I want to..." actions for each role.

#### User Requirements: The Student

- ... securely log in to the system.
- ... browse and search all available courses.
- ... view details for each course (prerequisites, credit hours).
- ... register (add) a course.
- ... drop (remove) a course.
- ... view my currently registered courses.
- ... view my academic history and grades.

#### User Requirements: The Instructor

- ... log in to the system.
- ... view a list of courses assigned to me.
- ... view and print the class roster (list of students) for my courses.

#### User Requirements: The Administrator

- ... manage all user accounts (create, edit, delete).
- ... add a new "Course" and define its prerequisites.
- ... manually manage student enrollments (add/drop).
- ... manage academic records (e.g., post final grades).

### Part 2: System Requirements

System requirements define the detailed \*how\* the system will be built.

# 1 1.0 User Management Requirements

## 1.1 User Registration

- **1.1.1** The system **shall** allow users (students and instructors) to register using an academic email and password.
- **Pre-condition:** User must provide a validly formatted academic email and a password.

## 1.2 User Authentication

- **1.2.1** The system **shall** provide functionality for users to log in and log out.
- **Pre-condition:** User must provide an existing academic email and its corresponding correct password.
- **Post-condition (Success):** A secure session is created and the user is redirected to their dashboard.
- **1.2.2 (Security):** The system **must** store user passwords securely encrypted by hashing.

## 1.3 Account Recovery

- **1.3.1** The system **shall** implement password reset functionality.
- **Pre-condition:** The user must provide an email address that exists in the system's database.
- **Post-condition:** A secure reset link is sent to the user's academic email.

## 1.4 User Profile Management

- **1.4.1** The system **shall** allow users to update their personal information and change their password.
- **Pre-condition:** The user must be authenticated (logged in) to access their profile.

# 2 2.0 Course Catalog Requirements

## 2.1 Course Browsing

- **2.1.1** The system **shall** allow students to view a list of all available courses.
- **2.1.2** The system **shall** categorize courses by department or academic level.
- **Condition (Usability):** The platform and its categorization must be easy to navigate.

## 2.2 Course Details & Admin

- **2.2.1** The system **shall** provide a dedicated page for each course (showing description, instructor, credits).
- **2.2.2** The system **shall** allow Admins to add, update, or delete courses.
- **Pre-condition:** The user attempting the action must be authenticated and possess 'Admin' privileges.

# 3 3.0 Registration & Enrollment Requirements

## 3.1 Registration Cart & Checkout

- **3.1.1** The system **shall** allow students to add/remove courses to/from a temporary cart.
- **3.1.2** The system **shall** require students to review their selected courses during checkout.
- **Pre-condition:** Student must have at least one course in their cart.
- **Post-condition:** The student's enrollment status is updated.

### **3.2 Enrollment Management (Admin)**

- **3.2.1** The system **shall** allow Admins to manage student accounts and enrollments.
- **Pre-condition:** The user must be authenticated and have 'Admin' privileges.

## **4 4.0 General System-Wide Constraints**

### **4.1 Performance & Scalability**

- **4.1.1** The system **shall** ensure website pages load in less than 5 seconds.
- **Condition:** This metric is based on average internet connections.
- **4.1.2** The system **must** handle increased traffic without performance degradation.
- **Condition (Peak Time):** This scalability is critical during peak registration periods.

### **4.2 Availability**

- **4.2.1** The system **must** be available 99.99% of the time (high uptime).
- **Condition:** This uptime requirement excludes pre-scheduled and communicated maintenance windows.

### **4.3 Security**

- **4.3.1** The system **must** protect against common web vulnerabilities.
- **Condition:** Protections must specifically include defenses against SQL Injection & CSRF.

### **4.4 Compatibility & Maintainability**

- **4.4.1** The website **must** be fully responsive and functional.
- **Condition:** Must be functional across major browsers and devices (PC, tablet, mobile).
- **4.4.2** The code **should** follow clean architecture principles.
- **Condition (Rationale):** This is required to allow for easy future maintenance and updates.

## **Non-Functional Requirements (Prioritized)**

1. Availability
2. Performance
3. Security
4. Scalability
5. Usability
6. Compatibility
7. Maintainability