# Student Management System

Version 1.0

Date: 25/10/2025

## I. Overall System Specification

### 1. Introduction

* 1.1 Purpose

The Student Management System (SMS) is designed to support academic administration activities in universities or colleges. Its main goals are to automate student, lecturer, course, class, grade, and tuition management; minimize human errors; and enhance transparency and efficiency in academic processes.

* 1.2 Scope

The system is used internally within the institution and serves all departments and individuals involved in teaching and academic management, including: Administrator (Admin), Academic Officer (Academic Officer), Lecturer, and Student.

### 2. System Actors

|  |  |
| --- | --- |
| Actor | Description |
| Admin | The highest-level manager, responsible for user accounts, permissions, and overall monitoring of the system. |
| Academic Officer | Academic staff who handle student information, courses, and class scheduling. |
| Lecturer | Instructor responsible for teaching, grading, and managing assigned classes. |
| Student | End user who views schedules, registers for courses, checks grades, and pays tuition fees. |

### 3. Functional Requirements

* 3.1 Common Features

- Login and logout using personal accounts.  
- Change password to ensure account security.

* 3.2 Administrator Functions

- Manage user accounts (add, update, delete Academic Officers, Lecturers, and Students).  
- Assign user roles and permissions.  
- Manage tuition fee records and confirm payments.  
- Generate reports and system statistics.

* 3.3 Academic Officer (Academic Officer) Functions

- Manage student, lecturer, class, and course information.  
- Manage academic schedules and timetables.  
- Create initial accounts for lecturers and students.  
- Assign lecturers to courses.  
- Handle course registration, class changes, and grade updates.  
- Approve student registration requests and resolve scheduling conflicts.

* 3.4 Lecturer Functions

- Manage class lists and student attendance.  
- Enter and update student grades.  
- Create assignments and upload course materials.  
- View personal teaching schedules and progress.

* 3.5 Student Functions

- Register for courses each semester.  
- View schedules and academic results.  
- Track tuition details and payment status.  
- Make online payments.  
- View academic announcements and lecturer information.

### 4. Non-Functional Requirements

|  |  |
| --- | --- |
| Category | Description |
| Performance | Response time < 3 seconds for standard actions. |
| Security | Passwords must be encrypted; only authenticated users can access the system. |
| Availability | System uptime ≥ 99%. |
| Scalability | Easily expandable for new faculties, subjects, or modules. |
| Usability | User-friendly interface compatible with browsers and mobile devices. |
| Data Backup | Regular database backups and recovery mechanism in case of failure. |

### 7. Conclusion

The Student Management System digitizes the entire academic management process, improving accuracy, transparency, and productivity. By clearly defining roles and privileges, the system becomes easy to maintain, extend, and secure.

## II. Task Assignment

|  |  |  |  |
| --- | --- | --- | --- |
| Member | Student ID | Responsibility | Notes |
| Phan Văn Thái Khang |  | Requirement analysis, Use Case design, Class Diagram |  |
| Ngô Khắc Phát |  | Database design (ERD), data constraints |  |
| Lê Xuân Vỹ |  | Interface and UX design | Leader |
| Nguyễn Thái Bảo |  | Documentation and final report writing |  |