OnlineCourse

Software Requirement Specification

– Hanoi, November 2023 –

# Record of changes

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **A\* M D** | **In charge** | **Change description** |
| 17/9/2023 . | A | Thanh | Add (I.1), (I.2), (I.3) |
|  |  |  |  |

\*A - Added M - Modified D - Deleted

# Overview

## Introduction

### [Content part 1]

The new product we are developing is a software system designed to manage the student progress for university class. The primary users of this system will be the lecturer, administrators, and students who are enrolled in class that are managed by the system.

The system will need to operate within the existing infrastructure of the university, which may present certain constraints and dependencies. For example, the system will need to integrate with existing student record-keeping systems, and it may need to comply with certain regulations and data privacy laws.

In terms of assumptions, we assume that users will have a basic level of computer literacy and familiarity with common software systems. Additionally, we assume that the university will provide the necessary hardware and network infrastructure to support the system's operation.

### [Content part 2]

The system can be represented using a context diagram, which illustrates the boundary of the system and its connections to other external entities. In this case, the system boundary encompasses the management of class, subject, teacher, and student progress, while external entities include lecturer’s, administrators, and students who interact with the system.

Data flows between these entities and the system, with lecturer’s update lecturer’s profile, show all the class and the subject they teach in the class, students list all the subject in their class, create change class request ,and administrators monitoring student progress and managing request response.

A diagram of a course

Description automatically generated

Fig\_1: Data flow diagram

2.System function

### Screen Flow

A diagram of a student

Description automatically generated

*Screen flow diagram*

### Screen Details

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **Screen** | **Description** |
| 1 | Login | Login Screen | Login screen for admin, teacher and student, system will filter user and go to direct homepage for each role |
| 2 | Home | Student home | Homepage for students. Can see all message when do any action |
| 3 | Home | Teacher home | Homepage for teacher. Can see all message when do any action |
| 4 | Home | Admin home | Homepage for admin. Can see all message when do any action |
| 5 | Profile | Student profile | Page for student. Can display student information and provide form for change information |
| 6 | List Subject | View Subject | Page for student. Can display subject information that student enrolls in |
| 7 | Request | Create change class request | Page for student. Display old class information and form for create change class request |
| 8 | Profile | Teacher profile | Page for teacher. Can display student information and provide form for change information |
| 9 | List Subject | View Subject | Page for teacher. Can display subject information that teacher teach |
| 10 | Profile | Admin Profile | Page for Admin. Can display student information and provide form for change information |
| 11 | Request | Response student request | Page for admin. Display request information and provide form for change request status |

### User Authorization

[Provide the system roles authorization to the system features (down to screens, and event to the screen activities if applicable) in the table form as below – replace Role1, Role2,… with the specific system user role names]

*Table\_2: User Authorization*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **#** | **Screen** | **Student** | **Lecturer** | **Admin** |
| 1 | Login | x | x | X |
| 2 | Student home | x |  |  |
| 3 | Update Profile | x | x | X |
| 4 | View Subject | x | X |  |
| 5 | Create request of change class | x |  |  |
| 6 | Reply student request |  |  | X |
| 7 | Teacher home |  | X |  |
| 8 | Admin home |  |  | x |

### Non-Screen Functions

*Table\_3: Non-Screen Functions*

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Feature** | **System Function** | **Description** |
| 1 | User Authentication | Handle user authentication | Implement logic for validating login credentials, verifying user accounts, and managing password reset functionality. |
| 2 | Encrypt password | Handle encode password for user | Implement library for encode password to database. |

## Entity Relationship Diagram