

**VIETNAM NATION UNIVERSITY HO CHI MINH CITY
UNIVERSITY OF INFORMATION TECHNOLOGY**

Software engineering faculty



Introduction to Software Engineering

Student management website

Lecturer: **Ms Thai Thuy Han Uyen**

Student:

Pham Van Hieu – 21520857

TABLE OF CONTENTS

CHAPTER 1: OVERVIEW	4
1.1 Current situation	4
1.2 Purpose.....	4
1.3 Design direction	5
CHAPTER 2: REQUIREMENT	6
2.1 Funtion requirement.....	6
2.1.1 List of requirements	6
2.1.2 Requirements specification	6
2.2 Non-funntional requirement.....	10
CHAPTER 3: USE CASE MODEL.....	11
3.1 General use case diagram	11
3.2.1 Login use case diagram.....	12
3.2.2 Course registration use case diagram.....	12
3.3.3 Schedule / test schedule use case diagram	13
3.3.4 Attendance.....	13
3.3 List of actor	14
3.4 List of use case	14
3.5 Use case decription.....	14
3.5.1 Use case login.....	14
3.5.2 Use case Course registration	15
3.5.3 Use case Lecturer update grade	16
3.5.4 Use case Search a student	16
CHAPTER 4: MODEL SPECIFICATION	18
3.4 Class diagram.....	18
4.1.1 List of class	18
4.1.2 Class diagram.....	19
4.2 Activity diagram	20
4.2.1 Student register the course activity diagram.....	20
4.2.2 Lecturer update student's grade.....	21
4.2.3 Admin user generates a grade report for a class	22
4.2.4 Admin user search for a student	23
CHAPTER 5: DATABASE DESIGN	24

5.1 Requirement.....	24
5.1.1 List of entities	24
5.1.2 Relationships between entities	24
5.1.3 Integrity Constraint	25
5.2 ERD model.....	26
5.3 EER model	27
5.4 Details of tables and their attributes	27
5.5 Database Permissions	29
CHAPTER 6: INTERFACE DESIGN.....	31
6.1 Screen layout diagram	31
6.2 List of screens and functions.....	31
6.2.1 Login screen	31
6.2.2 Forgot password screen	32
6.2.3 Change password	32
6.2.4 Student dashboard	33
6.2.5 Lecturer dashboard.....	37
6.2.6 Amin user dashboard	38
6.2.7 Admin dashboard.....	41
CHAPTER 7: INSTALLATION	44
7.1 Technology Usage	44
7.2 Design pattern.....	45
CHAPTER 8: TESTING.....	46
Appendix.....	50
1. Score distribution histogram chart template	50
2. Pie-char template	51
3. Score data table template.....	51
4. Excel file template	52
5. Source code	54

CHAPTER 1: OVERVIEW

1.1 Current situation

The current situation of student management websites in universities can be described as a mixed bag. While many universities have implemented modern and efficient student management systems, there are still some institutions that struggle with outdated and inadequate platforms.

In general, modern student management systems have proven to be highly effective in streamlining administrative tasks and improving communication between students, faculty, and staff. These platforms typically offer a wide range of features, such as online registration, grade tracking, and course scheduling, that allow students and administrators to access important information quickly and easily.

However, there are still some universities that rely on outdated systems that can be slow, cumbersome, and prone to errors. These systems may not be optimized for mobile devices or may require users to navigate through a complex series of menus and submenus to access the information they need. This can be frustrating for students who are accustomed to the streamlined user experiences provided by modern websites and applications.

Overall, it is clear that there is a wide range of student management website platforms being used by universities today. While some institutions have embraced modern technology and implemented highly effective systems, others continue to struggle with outdated platforms that are difficult to use and maintain. As the importance of technology in education continues to grow, it is likely that universities will increasingly turn to modern, efficient student management systems to meet the needs of their students and staff.

1.2 Purpose

The purpose of a student management website in a university is to provide a centralized platform for the efficient management of student-related tasks. It aims to streamline administrative tasks and reduce the time and effort required for managing student records, attendance, timetables, assignments, and communication.

The website enables different stakeholders in a university, including administrators, teachers, students, to collaborate and communicate effectively. It promotes a collaborative learning environment by providing a user-friendly interface for students to access course materials, track their progress, and interact with teachers and peers.

In summary, the purpose of a student management website in a university is to improve communication and collaboration, streamline administrative tasks, and enhance the learning experience for students.

1.3 Design direction

The design direction will follow the V-model, which is a software development approach that emphasizes a sequential and iterative process. This means that the project will progress through a series of phases, with each phase being completed before moving on to the next one. The V-model emphasizes the importance of testing and verification at each stage of development, ensuring a high-quality end product.

To start with, a use case model and use case diagram will be created to identify the various functionalities that the website should have. This will help to ensure that the website meets the requirements of its users, such as students, professors, and administrators.

A class diagram will also be created to represent the classes, objects, and relationships within the system. This diagram will help to organize the website's data and logic, making it easier to understand and maintain.

An activity diagram will then be created to illustrate the flow of activities in the system, from user interactions to system responses. This diagram will help to ensure that the website is user-friendly and efficient.

For the database design, MySQL will be used to create a logical data model that represents the entities, relationships, and attributes of the system. This model will serve as the foundation for the database schema and will ensure that the data is stored and retrieved correctly.

Finally, for the coding and design aspects, HTML and CSS will be used to create the website's layout and styling, while Python will be used for the website's back-end development. Canva will be used for designing logos, icons, and other graphical elements.

Overall, this design direction will ensure that the student management website project is well-organized, user-friendly, and efficient, while also being easy to maintain and update in the future.

CHAPTER 2: REQUIREMENT

2.1 Function requirement

2.1.1 List of requirements

ID	Name of requirement	Priority	Diagram	Test case
1. Account				
1.1	Login the system	High	3.2.1	
1.2	Create account	High		
1.3	Reset password	High		
1.4	Change password	High		
2. Course				
2.1	Register	Medium	3.2.2,4.2.1	
2.2	Add	High		
3. Grade				
3.1	View	Medium		
3.2	Update	High	4.2.2	
	Search	Medium		
3.3	Generate reports	High	4.2.3	
4. User record				
4.1	Manage	High		
4.2	Search	Medium	4.2.4	
4.3	Generate reports	High		
4.4	View	Medium		
4.5	Update	High		
5. Schedule				
5.1	View	Medium		
5.2	Manage	Medium		
6. Test Schedule				
6.1	View	Medium	3.2.3	
6.2	Manage	Medium	3.2.3	
7. Other				
7.1	Tuition fee	Medium		
7.2	Attendance	Medium	3.2.4	
7.3	Back up, restore	High		

2.1.2 Requirements specification

1. Account

1.1 Login

-If the user enters the wrong password or username, display a message indicating that the user entered the wrong password or username.

-If the user enters the wrong password five times, their account will be blocked for five minutes.

1.2 Create

-An account should be automatically created and sent to the user through email when the admin inserts the user's record:

+Account: ID (student, lecturer, staff...)

+Password: 10 random characters

1.3 Forget password

-The website should have a "forgot password" function:

+Prompt users to enter their username and email. If the username and email are valid, send an email to the user's email address containing a randomly generated password with 10 characters. If the username or email are invalid, display an error message.

1.4 Change password

-The website should have "change password" function:

+Prompts users to enter their old password, new password, and repeat the new password. If the old password is correct, the user should be automatically logged out and a message should be displayed.

2. Course

2.1 Register

-The website should allow students to register for courses each semester, by providing a list of available courses for the student to choose from. Once the student confirms their chosen courses, the selected courses will be displayed.

2.2 Add

-The website should enable admin to add new courses and delete courses if meet the conditions and necessary.

3. Grade

3.1 View

-The website should provide students with access to view their personal grades and display their grade point average (GPA).

3.2 Update

-The website should allow lecturers to update the grades of students in their classes for the current semester.

3.3 Generate report

-The website should enable lecturers to generate reports for their classes for the current semester or previous semesters. Users should have the option to choose and download the generated file.

+Excel file for each class.

+PDF file with a bar chart to show the distribution of grades and a pie chart to show the distribution of pass/fail percentages for each class.

-The website should enable students to generate reports of their progress learning:

+PDF file with a bar chart to show the distribution of grades and a pie chart to show the distribution of pass/fail percentages of courses.

-The website should enable admin users to generate reports of student grades by class, semester, academic year, department, name... by search function.

+Excel file.

+PDF file with a bar chart to show the distribution of grades and a pie chart to show the distribution of pass/fail percentages.

4. User record

4.1 Manage

-The website should enable admin to add and delete user records.

+When admin add a user record, an email should be sent to user with account information.

4.2 Search

-The website should include a search function:

+Allows admin users to search for students record by name, ID number, or other criteria and return a list of students that meet the specified conditions.

+Allow the admin to search for user records by name, ID number, or other criteria and return a list of users that meet the specified conditions.

4.3 Generate report

-The website should enable admin users to generate reports of student records by class, semester, academic year, and department. Users should have the option to choose and download the generated file.

+Excel file.

4.4 View

-The website should enable users to view their personal information.

4.5 Update

-The website should enable users to edit their personal information.

5. Schedule

5.1 View

-The website should enable students to view their personal schedules..

5.2 Manage

-The website should enable admin users to manage schedules, allowing them to add, delete courses as necessary.

+New schedule should include: course and lecturer with the same faculty, room, date, time, semester.

-The website should enable admin users to add student to the current schedule and delete student from current schedule.

-The website should enable admin users to download the schedule as an Excel file.

6. Test schedule

6.1 View

-The website should enable students to view their personal test schedules..

6.2 Manage

-The website should enable admin users to manage test schedules, allowing them to add, delete courses as necessary.

+Test schedules must be added based on the current schedule and should include the following information: current schedule (course info, student list), room, lecturer (as an examiner), date, and time.

7. Other

7.1 Tuition fee

-The website should enable students to view their personal tuition fee information.

-Tuition fee should be auto update based on course register information and price per credit.

7.2 Attendance

-The website should have an attendance function, which allows the lecturer to check if the students attend the class. Students can only see the attendance form if the lecturer enables it.

-The website should enable lecturers to download attendance report when necessary. After downloading, attendance data should be deleted from the database.

7.3 Back up

-The website should allow the admin to perform system backup and recovery operations to ensure the safety and integrity of data.

+When the admin clicks on the backup option, a backup file should be sent to the admin's email.

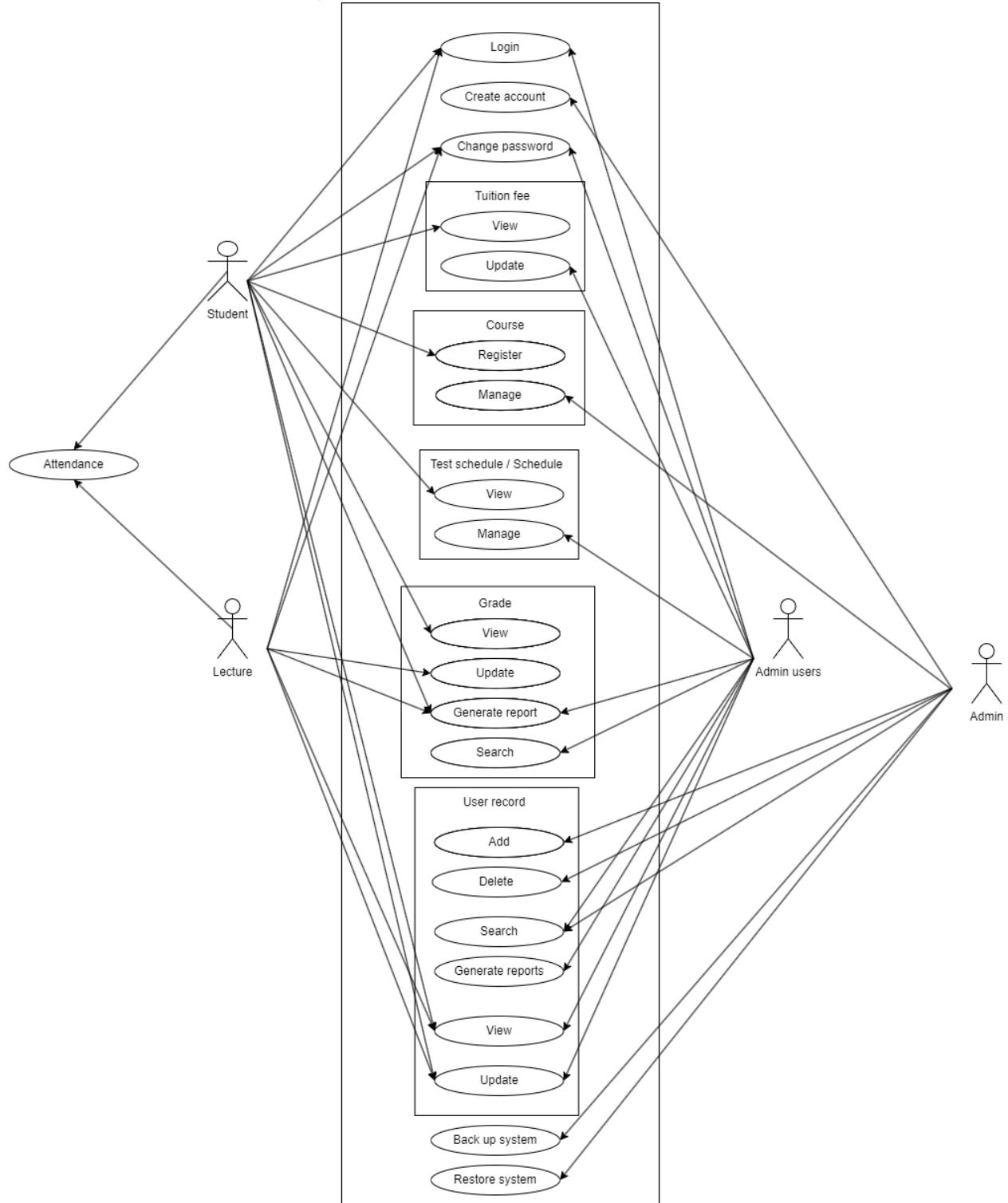
+In case of an incident, the admin needs to be able to restore the database system from a backup file.

2.2 Non-functional requirement

-Passwords stored in the database need to be encrypted with SHA256

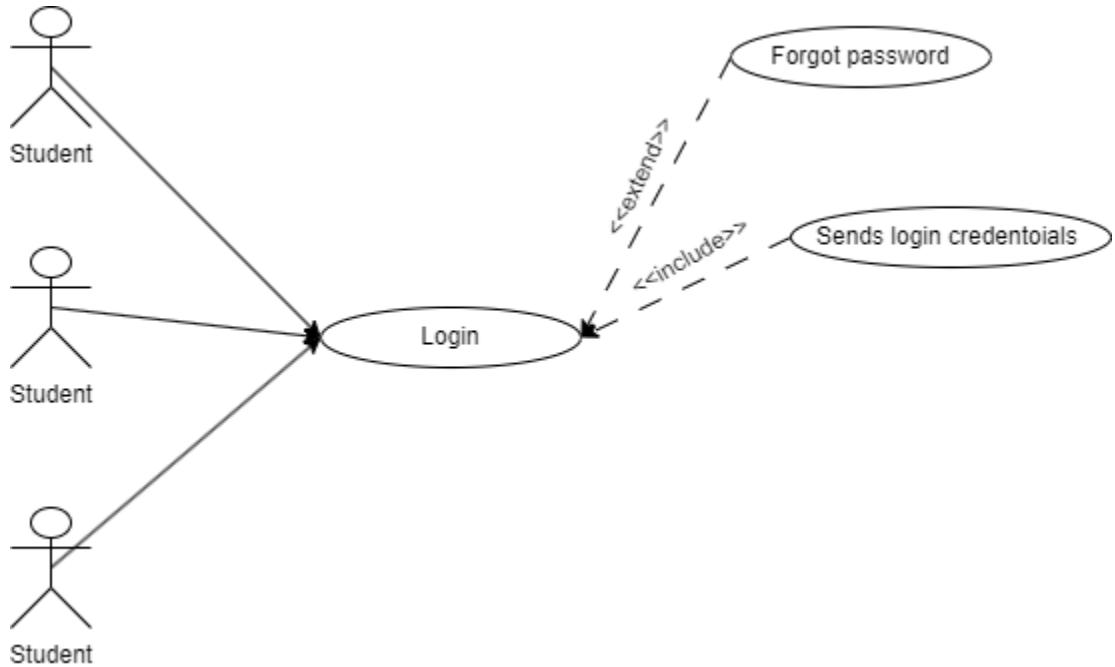
CHAPTER 3: USE CASE MODEL

3.1 General use case diagram

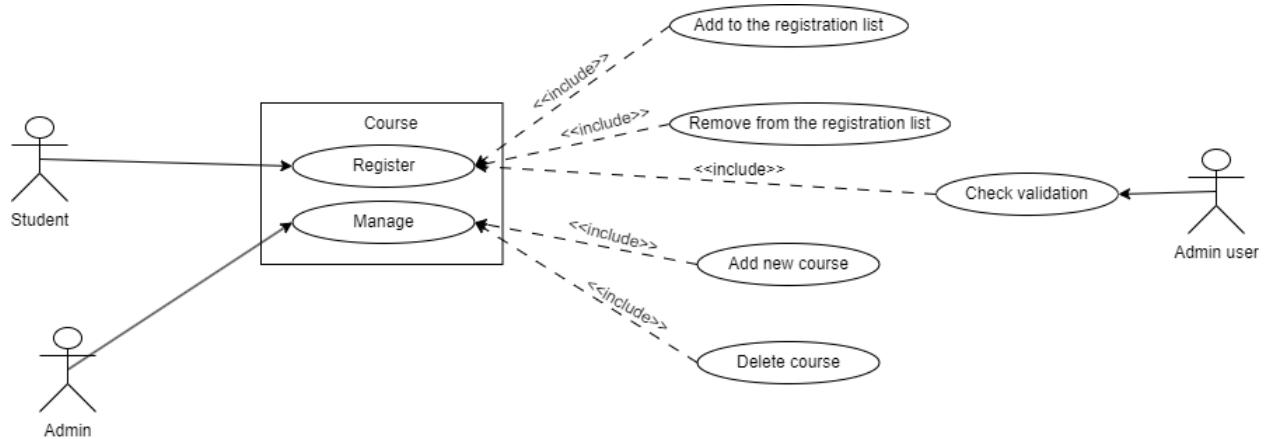


3.2 Specific use case diagram

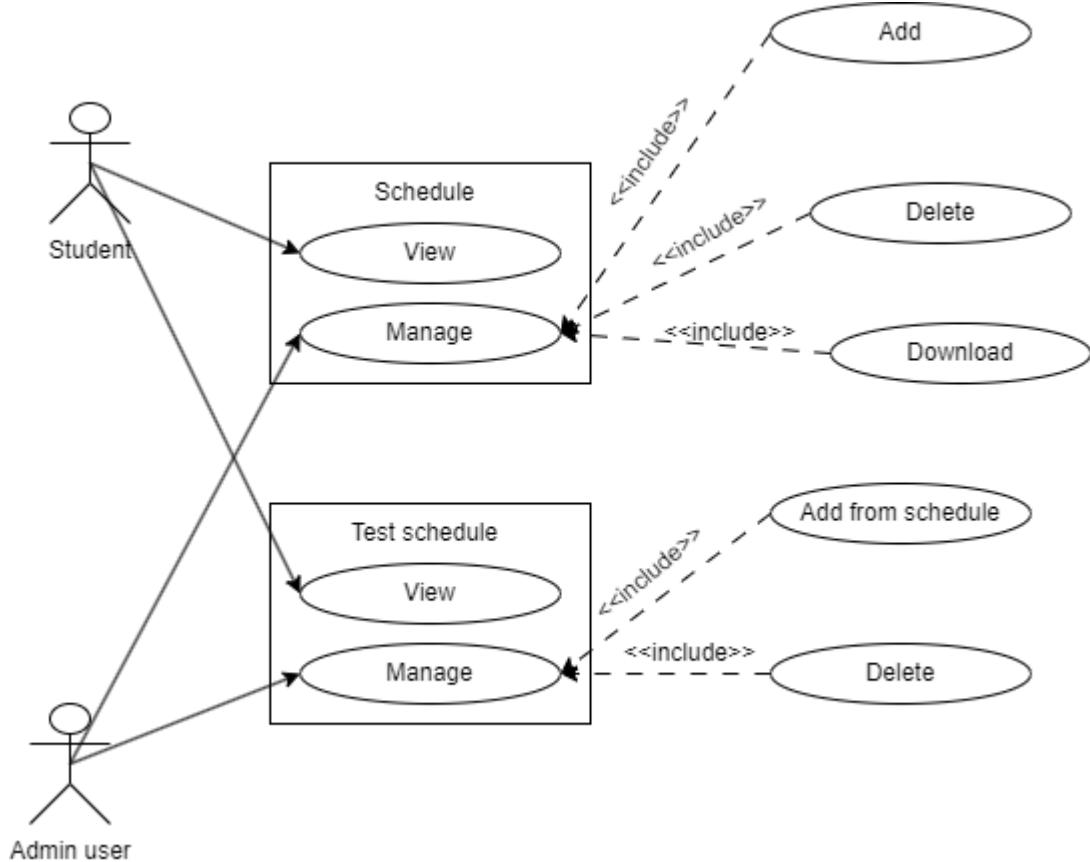
3.2.1 Login use case diagram



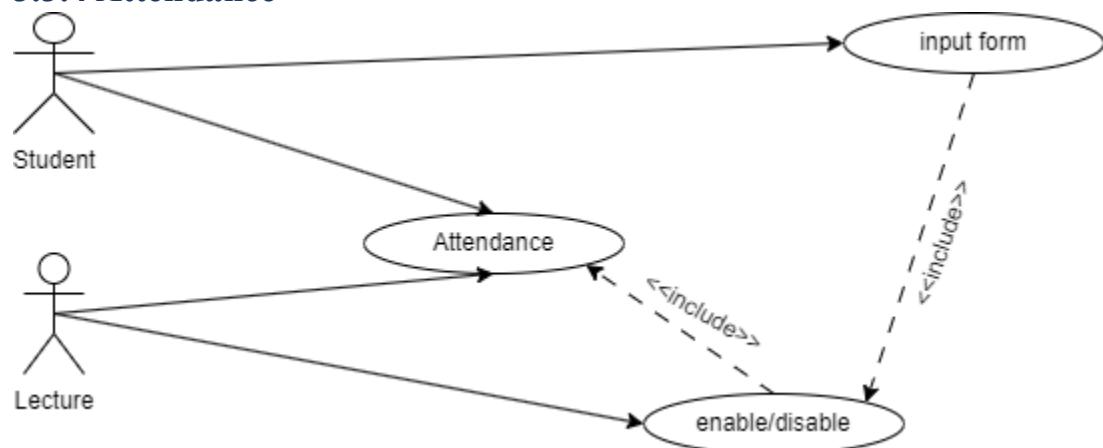
3.2.2 Course registration use case diagram



3.3.3 Schedule / test schedule use case diagram



3.3.4 Attendance



3.3 List of actor

No	Actor name	Meaning
1	Student	Full-time student of the school
2	Lecturer	College, university lecturers
3	Admin users	Staff, Expert
4	Admin	Responsible for the operation of the website

3.4 List of use case

No	Use case	Meaning
1	Login	Login to the system (all user)
2	Create account	Create new account for user (admin)
3	Forget password	User reset password when forgot (all user)
4	Change password	Change current password (all user)
5	View	View +tuition fee (student) +test schedule (student) +schedule (student) +grade (student) +user record (student, lecturer, admin user)
6	Update	Update +user record (student, lecturer, admin user) +tuition fee (admin user) +grade (lecturer)
7	Manage	Manage +add/delete user record (admin) +add/delete schedule/test schedule (admin user) +add/delete student to/from schedule (admin user)
8	Generate report	Generate report: +grade: excel and pdf file (student, lecturer, admin user) +user record: excel (admin user) +schedule student list: excel (admin user)
9	Search	User record: +Search user by name, ID...(admin user, admin)
10	Back up	Backup system (admin)
11	Restore	Restore system (admin)

3.5 Use case description

3.5.1 Use case login

No	Section	Content/explanation
1	Name	Login to the system

2	Authors	Hieu
3	Priority	High
4	Person responsible	Admin
5	Description	Describes the process of user logging into the student management system.
6	Actors	Student, lecturer, admin user
7	Pre-conditions	The user must have a valid account and password and they have access to a device with internet connectivity
8	Post-conditions	The user has successfully logged in and has access to other function
9	Main scenario	<ol style="list-style-type: none"> 1. The user enters their username and password. 2. The system checks the validity of the username and password. 3. If the login information is valid, the system allows the user to access the system
10	Exception scenario	If the login information is not valid, the system displays an error message and prompt the user to re-enter their login information

3.5.2 Use case Course registration

No	Section	Content/explanation
1	Name	Course registration
2	Authors	Hieu
3	Priority	High
4	Person responsible	Admin, admin user
5	Description	Describes the process of a student registering for courses using the student management system.
6	Actors	Student, admin user
7	Pre-conditions	The student has an account with the system and is eligible to register for courses. The course registration period must be open.
8	Post-conditions	The student has successfully registered for their desired courses and their registered course list has been updated accordingly.
9	Main scenario	<ol style="list-style-type: none"> 1. The student logs into the system and navigates to the course registration page. 2. The student selects the courses they wish to register for and submits their registration request. 3. The system checks the availability of the selected courses and any prerequisites or restrictions. 4. If all requirements are met, the system registers the student for the selected courses and updates their schedule accordingly.

10	Exception scenario	If a selected course is full or the student does not meet the prerequisites or restrictions, the system displays an error message and prompts the student to select a different course.
----	--------------------	---

3.5.3 Use case Lecturer update grade

No	Section	Content/explanation
1	Name	Lecturer update grade
2	Authors	Hieu
3	Priority	Medium to high
4	Person responsible	Lecturer, admin user
5	Description	Allows lecturers to update and record a student's grade for a specified subject that they are teaching in the system. The system checks the validity of the entered grade and any applicable grading policies or restrictions before updating the student's record.
6	Actors	Student, lecturer
7	Pre-conditions	The lecturer must be logged into the system and have permission to update grades for the specified subject.
8	Post-conditions	The student's grade for the specified subject has been updated and recorded successfully in the system.
9	Main scenario	<ol style="list-style-type: none"> 1. The lecturer logs to the system and navigates to the grade update page for the specified subject. 2. The lecturer selects the students and enters the updated grade. 3. The system checks the validity of the entered grade and any applicable grading policies or restrictions. 4. If all requirements are met, the system updates the student's grade for the specified subject.
10	Exception scenario	If the entered grade is not valid or does not meet applicable grading policies or restrictions, the system displays an error message and prompts the lecturer to enter a valid grade.

3.5.4 Use case Search a student

No	Section	Content/explanation
1	Name	Search a student
2	Authors	Hieu
3	Priority	Medium to high
4	Person responsible	Admin user, admin
5	Description	Allows an admin user to search for students in the system based on specific conditions or criteria. The system returns a list of students that meet the specified conditions.
6	Actors	Admin user
7	Pre-conditions	The admin user must be logged into the system and have permission to search for student records.

8	Post-conditions	The system displays a list of students that meet the specified search conditions.
9	Main scenario	<ol style="list-style-type: none"> 1. The admin user logs into the system and navigates to the student search page. 2. The admin user select the search conditions or criteria and enter what they want to search and submits the request. 3. The system searches for students that meet the specified conditions and displays the results.
10	Exception scenario	If no students are found that meet the specified conditions, the system display a message indicating that no results were found.

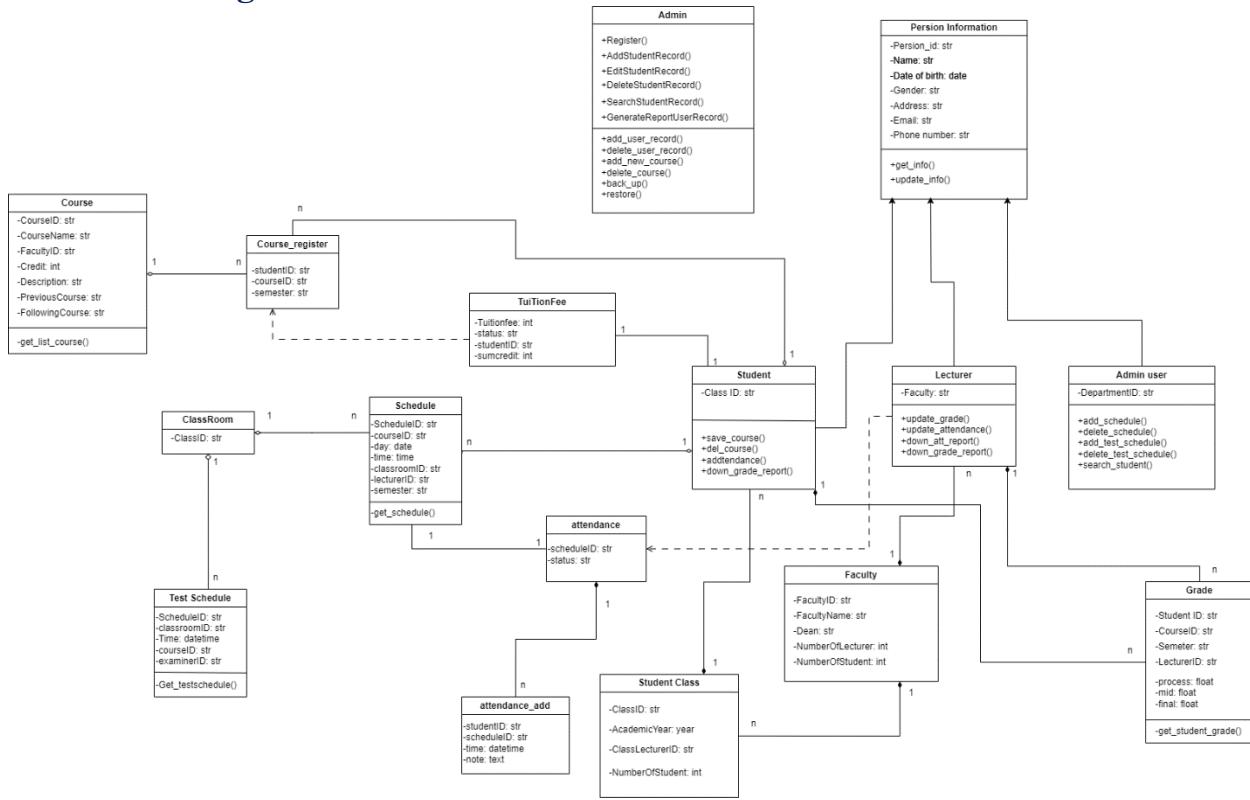
CHAPTER 4: MODEL SPECIFICATION

3.4 Class diagram

4.1.1 List of class

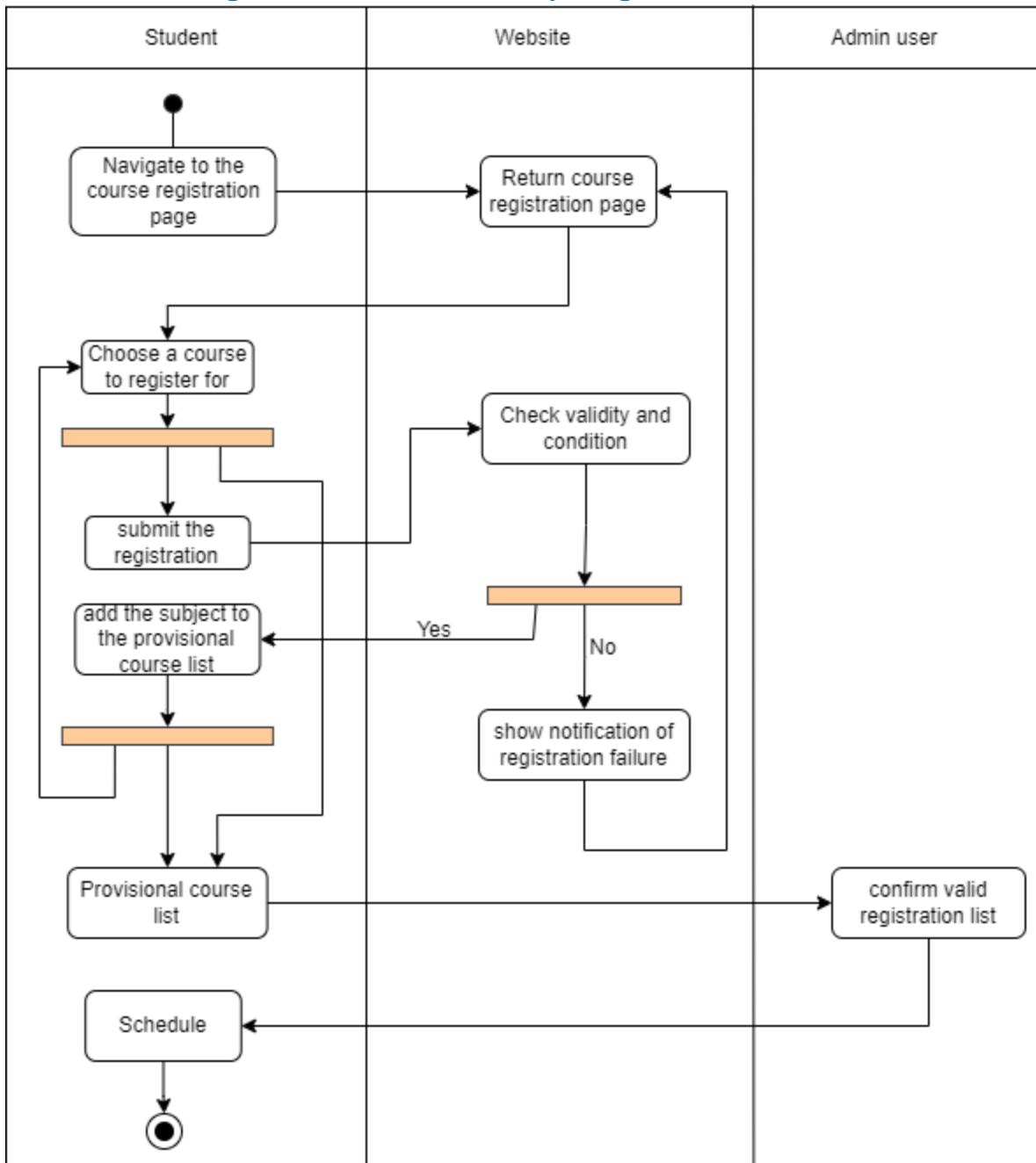
No	Name	Description
1	Personal information	Store user's Personal information
2	Student	Store student information, inheritance from Personal information
3	Lecturer	Store lecturer information, inheritance from Personal information
4	Admin user	Store admin user information, inheritance from Personal information
5	Admin	User admin
6	Grade	Store student's grade of each course
7	Faculty	Store faculty information
8	Student class	Store class information
9	Attendance	Store status of course attendance in schedule
10	Attendance_add	Store student attendance information
11	Schedule	Store schedule of each course
12	Tuitionfee	Store tuitionfee information of each student
13	Test schedule	Store test schedule of each course
14	Classroom	Store classroom list
15	Course register	Store registration course information of each student
16	Course	Store course information

4.1.2 Class diagram

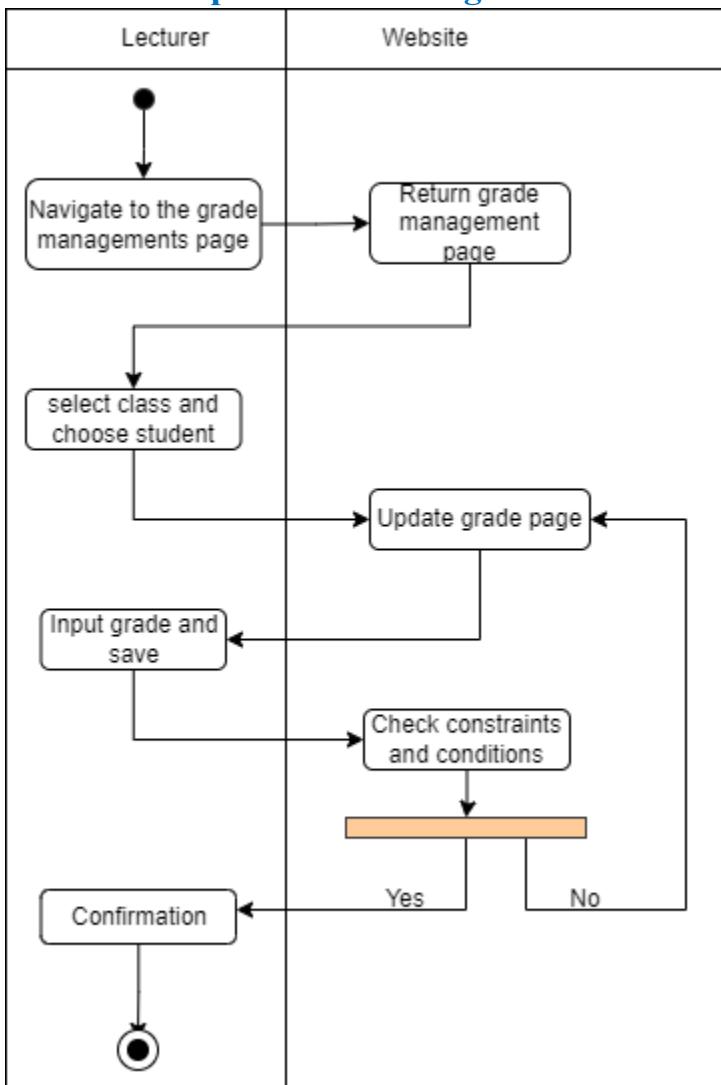


4.2 Activity diagram

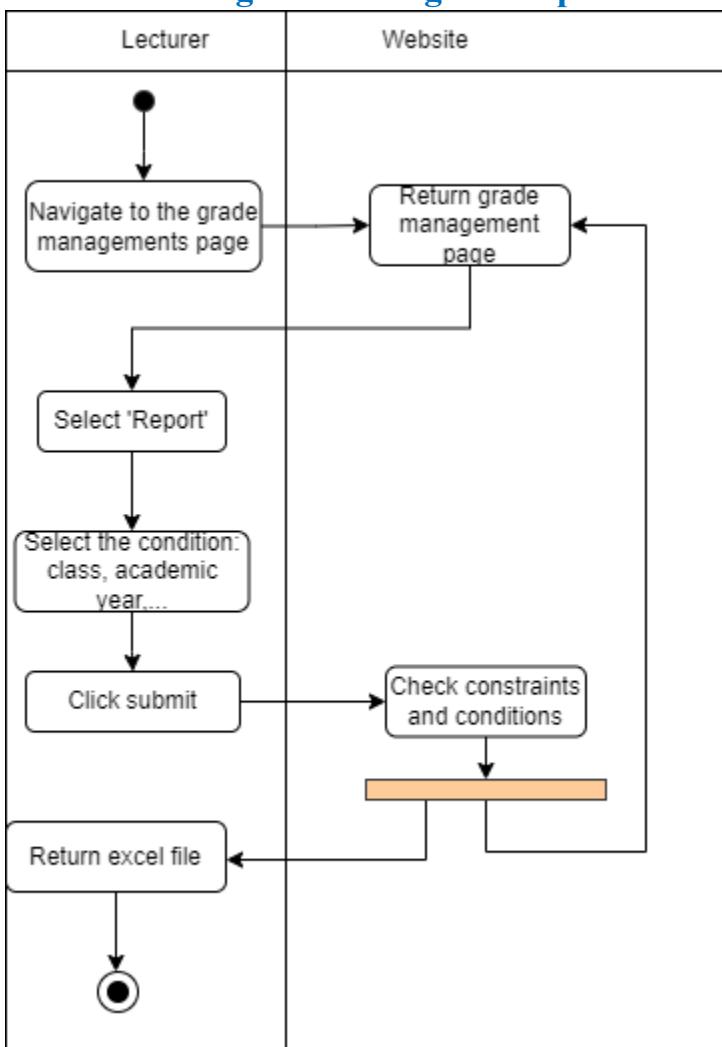
4.2.1 Student register the course activity diagram



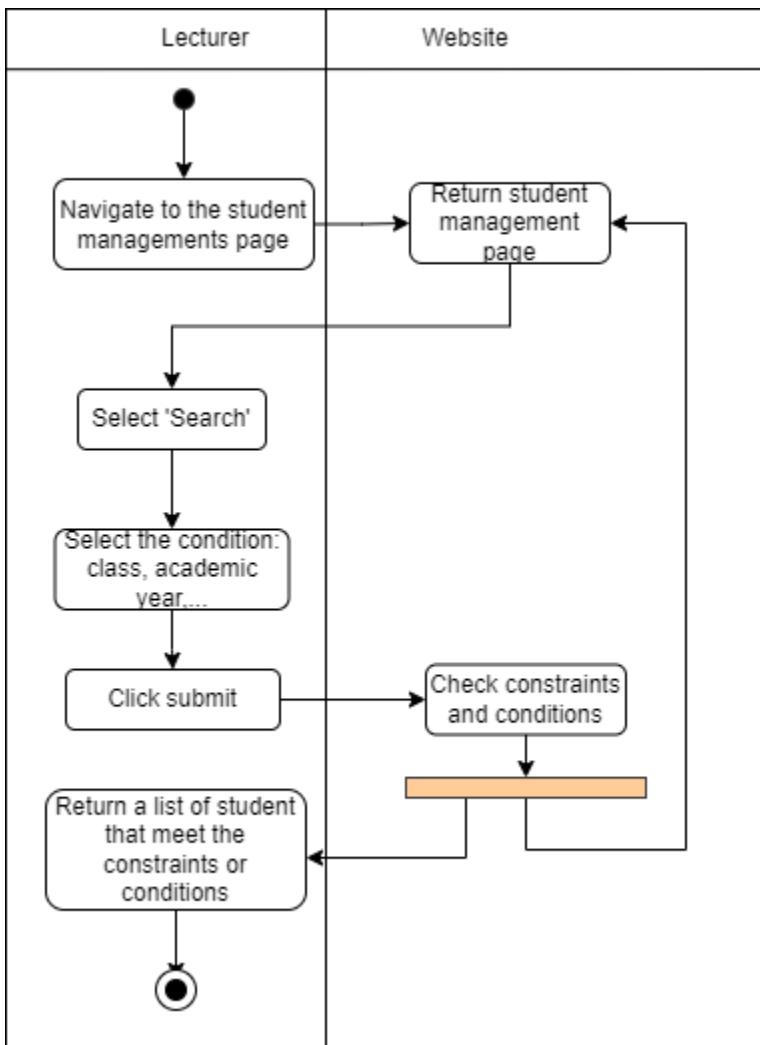
4.2.2 Lecturer update student's grade



4.2.3 Admin user generates a grade report for a class



4.2.4 Admin user search for a student



CHAPTER 5: DATABASE DESIGN

5.1 Requirement

5.1.1 List of entities

No	Name	Meaning	Attributes
1	Student	Store personal information	ID, name, date of birth, gender, address, email, phone number
2	Lecturer		classID
3	Admin_user		facultyID
4	Grade	Store student's grade of each course	departmentID studentID, courseID, semester, lecturerID, process, mid, final
5	Course	Store each course's information	courseID, course Name, facultyID, credit, description, previous course, following course
6	Faculty	Store faculty information	facultyID, faculty name, dean, number of lecturer, number of student
7	Test schedule	Store test schedule information	scheduleID, classroomID, Time, courseID, lecturerID
8	Schedule	Store general schedule information	scheduleID, courseID, studentID, day, time, classroomID, lecturerID, semeter
9	Attendance	Store attendance status of each course in schedule	scheduleID, status
10	Attendance_add	Store list of student that submit attendance form	studentID, scheduleID, time, note
11	Classroom	Store list of classroom	ClassroomID
12	Student_class	Store student class information	ClassID, LecturerID, number of student, academic year
13	Tuitionfee	Store tuitionfee information of each student	studentID, tuitionfee, status, sum credit
14	Account	Store account for each user	Username, password

5.1.2 Relationships between entities

-Student:

- Many-to-One relationship with StudentClass: Many students belong to one student class.
- Many-to-Many relationship with Schedule: Students can be enrolled in multiple schedules, and a schedule can have multiple students.

-Lecturer:

- One-to-Many relationship with Schedule: One lecturer can have multiple schedules.
- One-to-Many relationship with Grade: One lecturer can assign grades to multiple students.

-Course:

- One-to-Many relationship with Schedule: One course can have multiple schedules.
- One-to-Many relationship with Grade: One course can have multiple grades assigned to different students.
- One-to-One relationship with Course (previousCourse and followingCourse): A course can have a previous course and a following course.

-Grade:

- Many-to-One relationship with Student: Many grades belong to one student.
- Many-to-One relationship with Course: Many grades belong to one course.
- Many-to-One relationship with Lecturer: Many grades are assigned by one lecturer.

-Faculty:

- One-to-Many relationship with Lecturer: One faculty can have multiple lecturers.
- One-to-Many relationship with Course: One faculty can offer multiple courses.
- One-to-Many relationship with StudentClass: One faculty can have multiple student classes.

-Schedule:

- Many-to-One relationship with Course: Many schedules belong to one course.
- Many-to-One relationship with Lecturer: Many schedules are assigned to one lecturer.
- One-to-One relationship with TestSchedule: Each schedule can have a corresponding test schedule.
- One-to-One relationship with Classroom: Each schedule is associated with one classroom.

-TestSchedule:

- One-to-One relationship with Schedule: Each test schedule corresponds to one regular schedule.
- One-to-One relationship with Classroom: Each test schedule is associated with one classroom.
- One-to-One relationship with Lecturer: Each test schedule is assigned to one lecturer.
- One-to-One relationship with Course: Each test schedule is related to one course.

-StudentClass:

- One-to-One relationship with Lecturer: One student classes are led by one lecturer.
- Many-to-One relationship with Faculty: Many student class belongs to one faculty.

-TuitionFee:

- One-to-One relationship with Student: One tuition fee records belong to one student.

5.1.3 Integrity Constraint

-The insert_attendance_trigger trigger automatically inserts a new record into the attendance table with a status of 'close' when a new schedule is inserted.

-The update_numberOfLecturer trigger updates the numberOfLecturer column in the Faculty table when a new lecturer is inserted.

-The update_numberOfStudent trigger updates the numberOfStudent column in the StudentClass table when a new student is inserted.

-The update _numberOfStudent _1 trigger updates the numberOfStudent column in the Faculty table when the numberOfStudent value in the StudentClass table is updated.

-The check _grade _range trigger checks the grade values before inserting a new record into the Grade table and throws an error if any grade is out of the valid range (0-10).

-The insert _course _register trigger updates the TuitionFee table when a new course registration is inserted, adding the course tuition fee to the existing fee for the student or creating a new tuition fee record if it doesn't exist.

-The delete _course _register trigger updates the TuitionFee table when a course registration is deleted, subtracting the course tuition fee from the existing fee for the student.

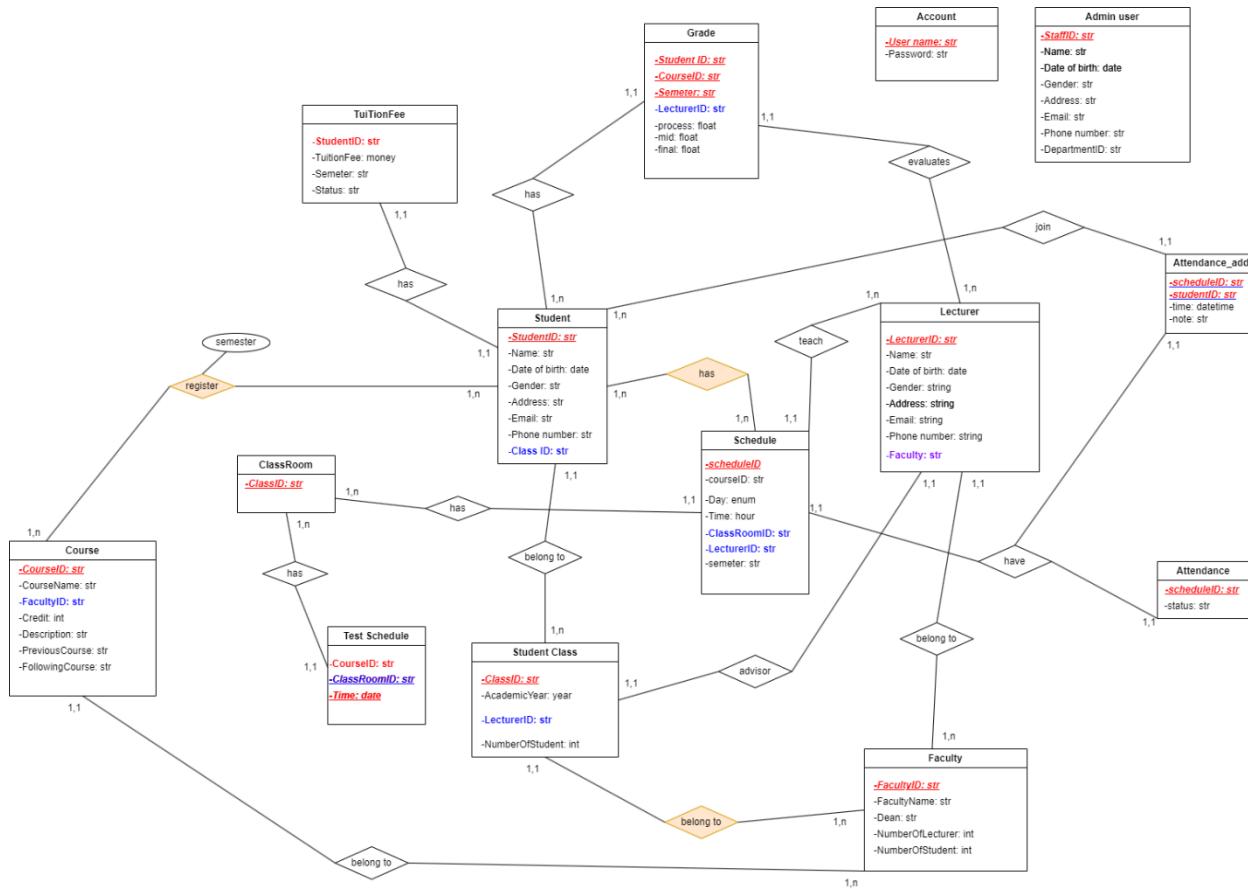
-The check _previous _course trigger checks if the previous course has been completed before allowing a new course registration to be inserted, based on the previousCourse column in the Course table and the student's grade records.

5.2 ERD model

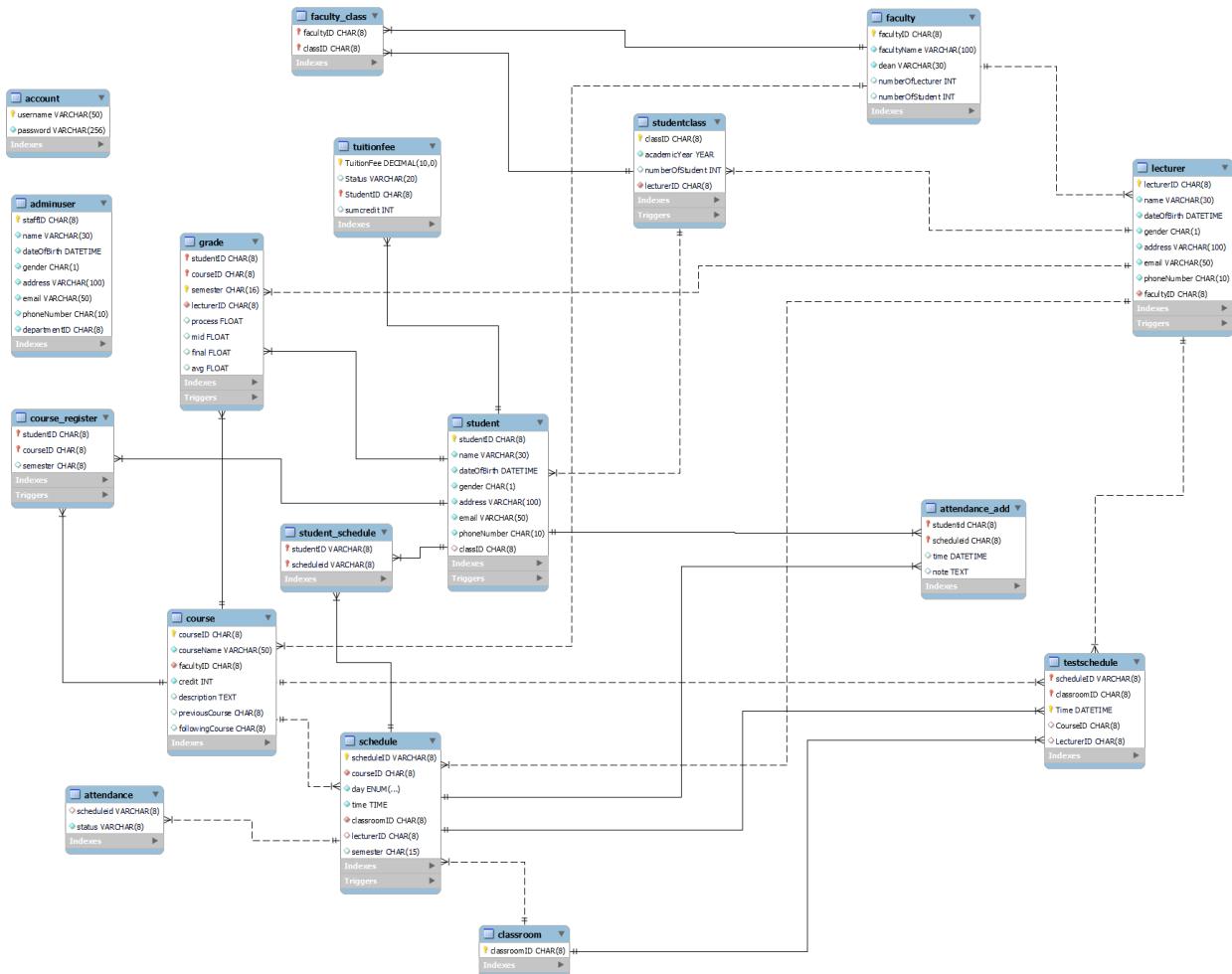
+Red with underline: Primary key

+Blue: Foreign key

+Colored relationship entity: creates a new relationship



5.3 EER model



5.4 Details of tables and their attributes

No	Relation	Attributes	Data type	Interpretation
1	Student	<u>studentID</u> -name -dateOfBirth -gender -address -email -phone number <u>classID</u>	Char(8) Nvarchar(30) Datetime Char(1) Nvarchar(100) Nvarchar(50) Char(10) Char(8)	PRK NOT NULL NOT NULL NOT NULL NOT NULL NOT NULL NOT NULL FK1, ref student_class
2	Lecturer	<u>lecturerID</u> -name -dateOfBirth -gender -address -email	Char(8) Nvarchar(30) Datetime Char(1) Nvarchar(100) Nvarchar(50)	PRK NOT NULL NOT NULL NOT NULL NOT NULL NOT NULL

		-phone number -facultyID	Char(10) Char(8)	NOT NULL FK1, ref Faculty
3	Admin_user	staffID -name -dateOfBirth -gender -address -email -phone number -departmentID	Char(8) Nvarchar(30) Datetime Char(1) Nvarchar(100) Nvarchar(50) Char(10) Char(8)	PRK NOT NULL NOT NULL NOT NULL NOT NULL NOT NULL NOT NULL NOT NULL
4	Grade	studentID courseID semeter lecturerID -process -mid -final	Char(8) Char(8) Char(16) Char(8) Float Float Float	PRK, ref Student PRK, ref Course PRK FK, ref lecturer NULL NULL NULL
5	Course	courseID -courseName facultyID -credit -description -previousCourse -followingCourse	Char(8) Nvarchar(50) Char(8) Int Text Char(8) Char(8)	PRK NOT NULL FK1, ref Faculty NOT NULL NOT NULL NOT NULL NOT NULL
6	Course_register	studentID courseID -semester	Char(8) Char(8) Char(15)	PRK, ref student PRK, ref course
7	Faculty	facultyID -facultyName -dean -numberOfLecturer -numberOfStudent	Char(8) Nvarchar(100) Nvarchar(30) int int	PRK NOT NULL NOT NULL NOT NULL NOT NULL
8	Test_Schedule	scheduleid classroomID Time CourseID lecturerID	Varchar(8) char(8) datetime char(8) char(8)	PRK, ref Schedule PRK, ref Classroom PRK FK1, ref Course FK2, ref Lecturer
9	Schedule	scheduleID courseID -day -time classroomID lecturerID	Char(8) Char(8) Enum(...) Time char(8) char(8)	PRK FK1, ref Course NOT NULL NOT NULL FK2, ref classroom FK3, ref lecturer

		-semeter	char(15)	NOT NULL
10	Student_schedule	<u>studentID</u> <u>scheduleID</u>	Char(8) Char(8)	PRK, ref student PRK, ref schedule
11	Attendance	<u>scheduleID</u> -status	Char(8) Char(8)	PRK, ref schedule NOT NULL
12	Attendance_add	<u>studentID</u> <u>scheduleID</u> -time -note	Char(8) Char(8) Datetime Text	PRK, ref student PRK, ref schedule
13	Classroom	<u>classroomID</u>	Char(8)	PRK
14	Student_class	<u>classID</u> -academic Year -numberOfStudent <u>lecturerID</u>	Nvarchar(8) year int char(8)	PRK NOT NULL NOT NULL FK1, ref Lecturer
15	Faculty_class	<u>facultyID</u> <u>classID</u>	Char(8) Char(8)	PRK, ref faculty PRK, ref student_class
16	Tuition_Fee	<u>studentID</u> <u>TuitionFee</u> -status -sumcredit	Char(8) Numeric Varchar(20) int	PRK, ref student PRK NOT NULL NOT NULL
17	Account	<u>username</u> -password	Varchar(50) Varchar(256)	PRK NOT NULL, sha256

5.5 Database Permissions

We need to have separate login permissions for 4 types of users to ensure the security of the database.

Student User:

-The user 'student' is granted the following privileges:

UPDATE privilege on the 'student' table .

SELECT privilege on all tables.

INSERT and DELETE privileges on the 'course_register' table.

INSERT and UPDATE privileges on the 'attendance_add' table.

Lecturer User:

-The user 'lecturer' is granted the following privileges:

SELECT privilege on all tables

UPDATE privilege on the 'lecturer' table.

INSERT privilege on the 'GRADE' table.

UPDATE privilege on the 'GRADE' table.

UPDATE privilege on the 'attendance' table.

INSERT, UPDATE, and DELETE privileges on the 'attendance_add' table.

Admin User:

SELECT privilege on all tables.

UPDATE, INSERT, and DELETE privileges on the 'schedule' table.

UPDATE, INSERT, and DELETE privileges on the 'student_schedule' table.

UPDATE, INSERT, and DELETE privileges on the 'testschedule' table.

UPDATE privilege on the 'account' table.

Admin:

UPDATE, INSERT, and DELETE privileges on the 'student' table.

UPDATE, INSERT, and DELETE privileges on the 'lecturer' table.

UPDATE, INSERT, and DELETE privileges on the 'adminuser' table.

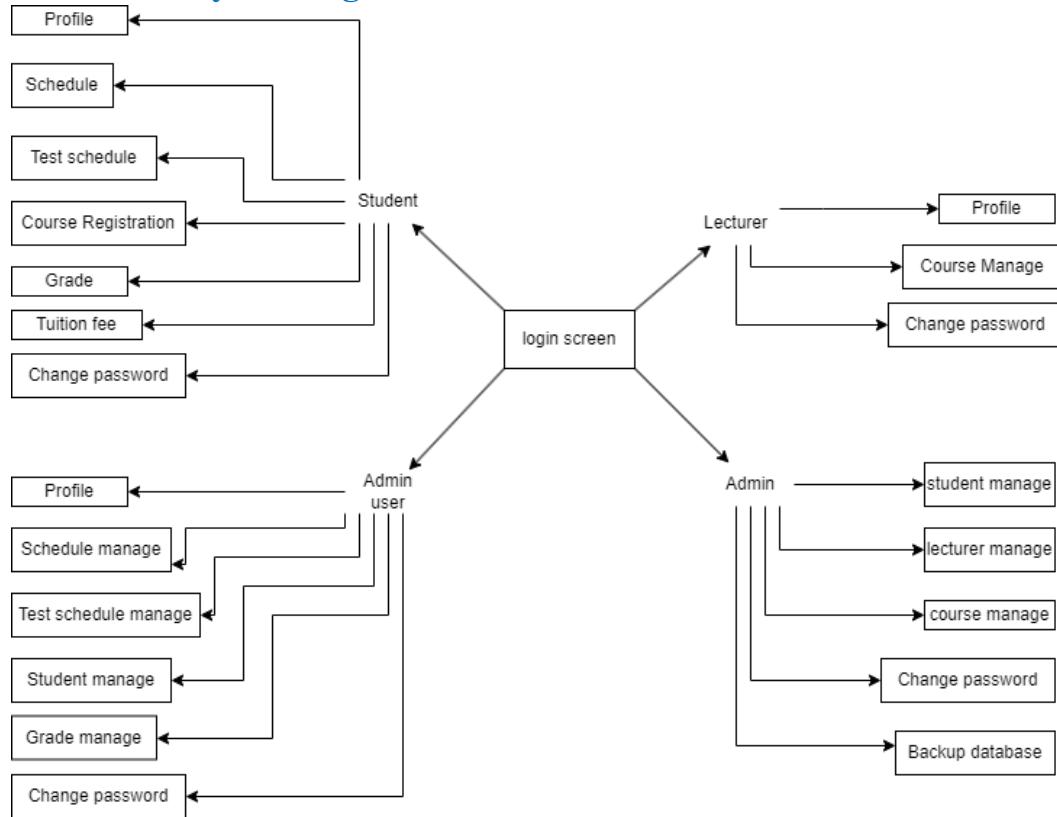
UPDATE, INSERT, and DELETE privileges on the 'account' table.

SELECT privilege on all tables.

Additional privileges like INSERT, UPDATE, DELETE on the 'course' table, and broader privileges on other databases.

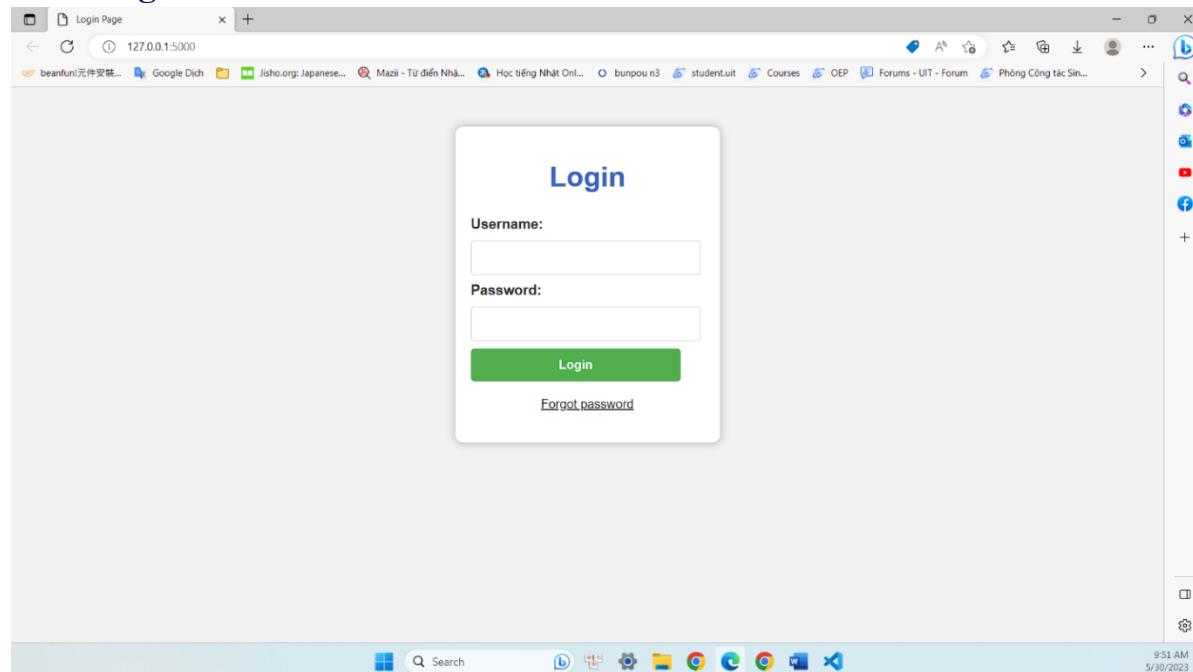
CHAPTER 6: INTERFACE DESIGN

6.1 Screen layout diagram

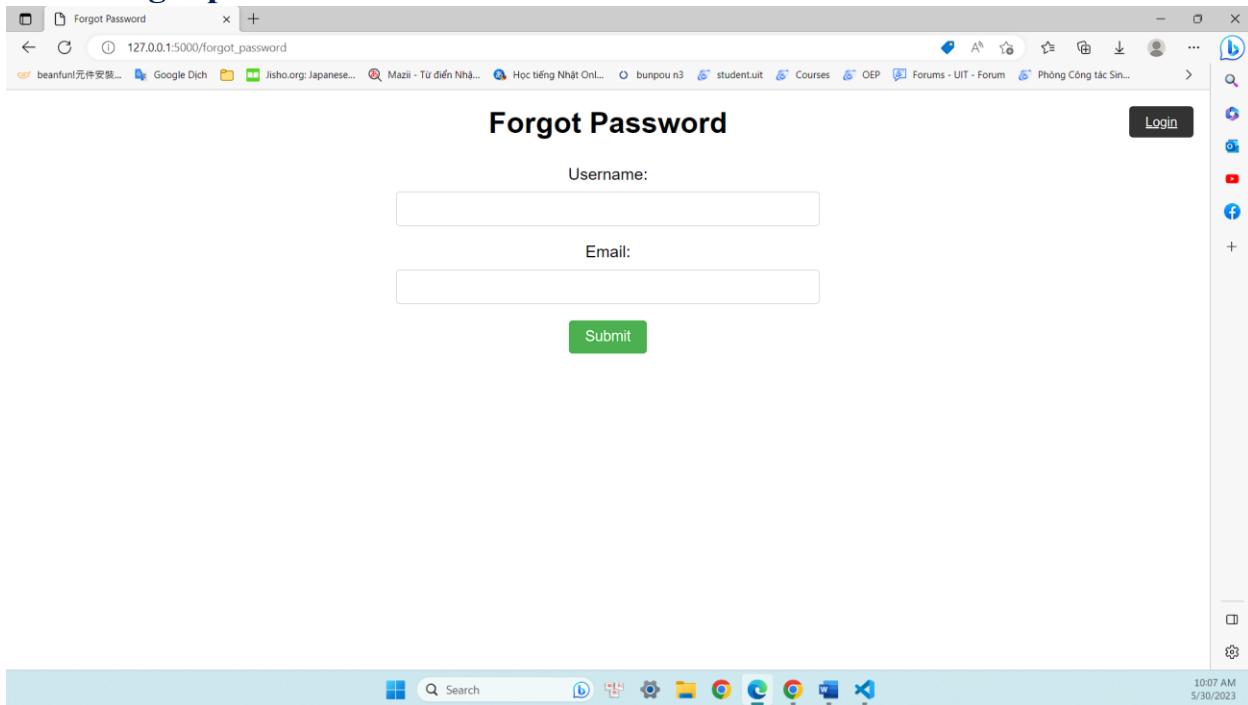


6.2 List of screens and functions

6.2.1 Login screen

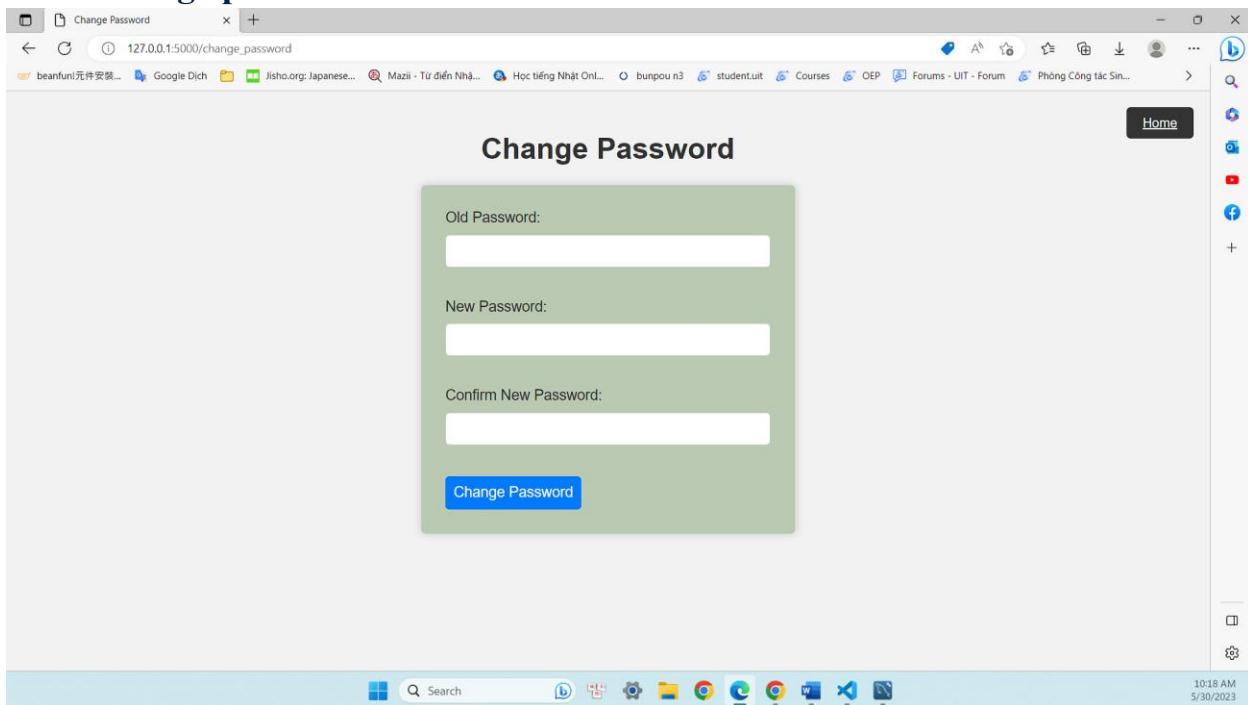


6.2.2 Forgot password screen



The screenshot shows a web browser window titled "Forgot Password". The URL in the address bar is "127.0.0.1:5000/forgot_password". The page content is titled "Forgot Password" and contains two input fields: "Username:" and "Email:", both represented by white input boxes. Below these fields is a green "Submit" button. In the top right corner of the browser window, there is a "Login" button.

6.2.3 Change password



The screenshot shows a web browser window titled "Change Password". The URL in the address bar is "127.0.0.1:5000/change_password". The page content is titled "Change Password" and contains three input fields: "Old Password:", "New Password:", and "Confirm New Password:", all represented by white input boxes. Below these fields is a blue "Change Password" button. In the top right corner of the browser window, there is a "Home" button.

6.2.4 Student dashboard

The screenshot shows a web browser window titled "Trang chủ - Quản lý sinh viên". The address bar displays "127.0.0.1:5000/login". The sidebar on the left contains several buttons: PROFILE, SCHEDULE, TEST SCHEDULE, COURSE REGISTRATION, GRADE, TUITION FEE, CHANGE PASSWORD, and LOGOUT. The main content area displays the message "Hello student". The bottom right corner of the screen shows the Windows taskbar with various pinned icons and the date/time "10:08 AM 5/30/2023".

6.2.4.1 Profile

The screenshot shows a web browser window titled "Student Profile". The address bar displays "127.0.0.1:5000/profile". The main content area displays a table of student information:

STUDENT ID	ST000001
NAME	Nguyễn Văn A
DATE OF BIRTH	2000-01-01 00:00:00
GENDER	M
ADDRESS	Hà Nội
EMAIL	nguyenvana@gmail.com
PHONE	0123456789
CLASS ID	CNTT2021

Below the table, there is a section titled "Student Information" with input fields for Name, Email, Phone, and Address. At the bottom, there is a "Update information" button. The bottom right corner of the screen shows the Windows taskbar with various pinned icons and the date/time "10:08 AM 5/30/2023".

6.2.4.2 Schedule

Student Schedule

COURSE NAME	DAY	TIME	CLASSROOM	LECTURER NAME	ATTENDANCE
Data Mining	mon	7:00:00	C013	Tran Thi Y	
Big Data Analytics	tue	10:00:00	C007	Tran Thi Y	
Các giao thức mạng	wed	14:00:00	C003	Nguyen Van X	
Quản lý dự án phần mềm	thu	9:00:00	C002	Nguyen Van X	
Mạng máy tính truyền thông	fri	11:00:00	C004	Nguyen Van X	

6.2.4.3 Test schedule

Test Schedule

Course ID	Course Name	Classroom ID	Time
CN101	Các giao thức mạng	C001	2023-08-06 08:00:00
CS115	Quản lý dự án phần mềm	C001	2023-08-07 09:00:00
DC114	Big Data Analytics	C001	2023-06-06 09:00:00

6.2.4.5 Course registration

Course registered

Course ID	Course name	Credit	Select
CN100	Giới thiệu về Mạng máy tính	3	<input type="checkbox"/>
CN101	Các giao thức mạng	3	<input type="checkbox"/>
IT119	Đồ án tốt nghiệp	4	<input type="checkbox"/>

[Delete selected courses](#)

Select	Course ID	Course Name	Faculty ID	Credit	Description	Previous Course	Following Course
<input type="checkbox"/>	CN100	Giới thiệu về Mạng máy tính	ITD03	3	Khóa học này cung cấp một giới thiệu về mạng máy tính.	None	CN101
<input type="checkbox"/>	CN101	Các giao thức mạng	ITD03	3	Khóa học này bao gồm các nguyên tắc và thực hành của các giao thức mạng.	None	CN102
<input type="checkbox"/>	CN102	Bảo mật mạng	ITD03	3	Khóa học này bao gồm các nguyên tắc và thực hành của bảo mật mạng.	None	CN103
<input type="checkbox"/>	CN103	Thiết kế mạng	ITD03	3	Khóa học này cung cấp kiến thức về thiết kế mạng.	None	CN104
<input type="checkbox"/>	CN104	Quản trị mạng	ITD03	3	Khóa học này cung cấp kiến thức về quản trị mạng.	None	CN105
<input type="checkbox"/>	CN105	Mạng không dây	ITD03	3	Khóa học này cung cấp kiến thức về mạng không dây.	CN101	CN106

10:16 AM
5/30/2023

<input type="checkbox"/>	IT113	Phân tích và thiết kế hệ thống thông tin	ITD01	3	None	IT102	IT114
<input type="checkbox"/>	IT114	Lập trình web	ITD01	3	None	IT113	IT107
<input type="checkbox"/>	IT115	Học máy (Machine Learning)	ITD01	3	None	IT104	IT116
<input type="checkbox"/>	IT116	Phân tích dữ liệu (Data Analytics)	ITD01	3	None	IT105	IT117
<input type="checkbox"/>	IT117	Mạng xã hội và phân tích mạng xã hội	ITD01	3	None	IT106	IT118
<input type="checkbox"/>	IT118	Kỹ thuật phần mềm	ITD01	3	None	IT107	IT119
<input type="checkbox"/>	IT119	Đồ án tốt nghiệp	ITD01	4	None	IT108	None

Save Selected Courses

10:16 AM
5/30/2023

6.2.4.6 Grade

Course Grades

GPA: 6.93

COURSE_ID	SEMESTER	LECTURER_ID	CREDIT	PROCESS	MID	FINAL	AVG
IT100	2021.1	LT000001	3	8.5	6.75	9.25	8.35
IT101	2021.1	LT000002	4	4.25	7.5	8.0	7.1
IT102	2021.1	LT000003	4	9.0	3.25	5.5	5.525
IT103	2021.1	LT000001	3	6.75	8.25	7.0	7.325
IT104	2021.1	LT000002	3	3.5	4.75	6.25	5.25
IT105	2021.2	LT000002	3	4.25	7.5	8.0	7.1
IT106	2021.2	LT000013	3	9.0	3.25	5.5	5.525
IT107	2021.2	LT000020	3	6.75	8.25	7.0	7.325
IT108	2021.2	LT000021	3	3.5	4.75	6.25	5.25
IT109	2021.2	LT000021	3	6.5	5.75	6.25	6.15
IT110	2022.1	LT000023	3	5.75	6.5	7.75	6.975
IT111	2022.1	LT000024	3	9.0	7.25	8.0	8.275

6.2.4.7 Tuition fee

Tuition Fee

TUITION FEE	STATUS	STUDENT ID	CREDIT
4000000 VND	unpaid	st000001	10

6.2.5 Lecturer dashboard

The screenshot shows a web browser window with the URL 127.0.0.1:5000/lecturer_home. The page has a teal background and displays the message "Hello lecturer". On the left side, there are four buttons: "PROFILE" (highlighted in a white box), "COURSE MANAGE", "CHANGE PASSWORD", and "LOGOUT". The browser's address bar shows the same URL. The taskbar at the bottom includes icons for various applications like File Explorer, Google Chrome, and Microsoft Word.

6.2.5.1 Profile

The screenshot shows a web browser window with the URL 127.0.0.1:5000/lecturer_profile. The page title is "Lecturer Profile". It contains a table with the following data:

LECTURER ID	LT000001
NAME	Nguyen Van X
DATE OF BIRTH	1990-01-01 00:00:00
GENDER	M
ADDRESS	123 Nguyen Van Cu, Quan 5, TP. Ho Chi Minh
EMAIL	nx@itd.edu.vn
PHONE	0123456789
FACULTY ID	ITD01

Below the table, there is a section titled "Lecturer Information" with input fields for Name, Email, Phone, and Address, each containing the values from the table. A "Update Information" button is located at the bottom of this section. The browser's address bar shows the URL and the taskbar at the bottom includes various application icons.

6.2.5.1 Course manage

The screenshot shows a web application titled "Update Grade" for the year "2021.1". On the left, there is a table titled "Class" listing three courses:

Day	Time	Classroom ID	Course ID	Course Name	Semester	Select	Attendance	Attendance report
thu	9:00:00	C002	CS115	Quản lý dự án phần mềm	2023.2	Select	Open	
wed	14:00:00	C003	CN101	Các giao thức mạng	2023.2	Select	Open	
fri	11:00:00	C004	CN112	Mạng máy tính truyền thông	2023.2	Select	Open	

On the right, there is a sidebar with a table showing course details:

Course ID	Course Name	Excel	Report pdf
IT100	Cấu trúc dữ liệu và giải thuật	Excel	Report pdf
IT103	Lập trình Python	Excel	Report pdf

The browser address bar shows "127.0.0.1:5000/update_grade". The system status bar at the bottom indicates "10:22 AM 5/30/2023".

6.2.6 Amin user dashboard

The screenshot shows the "Trang chủ - Quản lý sinh viên" (Admin Dashboard) for the URL "127.0.0.1:5000/login". The sidebar on the left lists several management options:

- PROFILE
- SCHEDULE MANAGE
- TEST SCHEDULE MANAGE
- STUDENT MANAGE
- GRADE MANAGE
- CHANGE PASSWORD
- LOGOUT

The main area displays the message "Hello". The browser address bar shows "127.0.0.1:5000/login". The system status bar at the bottom indicates "10:22 AM 5/30/2023".

6.2.6.1 Profile

The screenshot shows a web browser window with the URL 127.0.0.1:5000/ad_user_profile. The page title is "Profile". The content displays a table of user information:

STAFF ID	AD000001
NAME	Nguyễn Văn A
DATE OF BIRTH	1980-01-01 00:00:00
GENDER	M
ADDRESS	Số 10, Đường 3/2, Quận 10, TP. Hồ Chí Minh
EMAIL	nguyenvana@gmail.com
PHONE	0901234567
DEPARTMENT ID	DPT001

Below the table, there is a section titled "Lecturer Information" with input fields for Name, Email, Phone, and Address, followed by a "Update Information" button. The browser's address bar shows the same URL, and the status bar indicates the time as 10:23 AM and the date as 5/30/2023.

6.2.6.2 Schedule manage

The screenshot shows a web browser window with the URL 127.0.0.1:5000/schedule_manage. The page title is "Schedule". A green header bar contains the text "Add Schedule". Below it, a search bar allows selecting a department and semester, with dropdown menus for Room, Date, Time, and Semester. A "Confirm" button is also present. A table lists existing schedules:

Schedule ID	Course ID	Day	Time	Classroom ID	Lecturer ID	Semester	Select
SC10	DC114	tue	10:00:00	C007	LT000002	2023.2	[More]
SC6	CS115	thu	9:00:00	C002	LT000001	2023.2	[More]
SC7	CN101	wed	14:00:00	C003	LT000001	2023.2	[More]
SC8	CN112	fri	11:00:00	C004	LT000001	2023.2	[More]
SC9	DC113	mon	7:00:00	C013	LT000002	2023.2	[More]

The browser's address bar shows the same URL, and the status bar indicates the time as 10:23 AM and the date as 5/30/2023.

6.2.6.3 Test schedule manage

The screenshot shows a web browser window titled "Test List" at the URL 127.0.0.1:5000/test_schedule_manage. The page has a header with "Add Test" and "Home" buttons. Below the header is a search bar with fields for "Choose a test:" (CN112 - fr - 11:00:00), "Room:" (C001), "Lecturer:" (LT000001 - Nguyen Van X), "Date:" (mm/dd/yyyy), "Time:" (--:--:--), and "Confirm". A table lists three scheduled tests:

Schedule ID	Course name	Course ID	Time	Classroom ID	Select	Select	Student ID	Student Name
SC7	Các giao thíc mang	CN101	2023-08-06 08:00:00	C001	<button>More</button>	<button>Delete</button>		
SC6	Quản lý dự án phần mềm	CS115	2023-08-07 09:00:00	C001	<button>More</button>	<button>Delete</button>		
SC10	Big Data Analytics	DC114	2023-06-06 09:00:00	C001	<button>More</button>	<button>Delete</button>		

The browser's address bar shows the URL 127.0.0.1:5000/test_schedule_manage. The taskbar at the bottom includes icons for Search, Home, and various applications.

6.2.6.4 Student manage

The screenshot shows a web browser window titled "Search Student" at the URL 127.0.0.1:5000/search_student. The page has a header with "Search Student" and "Home" buttons. Below the header is a search form with fields for "Student ID:", "Student Name:", "Date of Birth:", "Gender:", "Address:", "Email:", "Phone Number:", "Class:", "Faculty:", and a "Search" button. There is also a "Download" button. A table below the form lists student information:

ID	Name	Date of birth	Gender	Address	Email	Phone Number	Class ID	Faculty ID
ST000018	Nguyễn Văn An	2003-01-01 00:00:00	M	Số 1, Đường A, Quận B, Hà Nội	hs1@example.com	123456789	CNTT2020	ITD01
ST000019	Trần Thị Bình	2003-01-02 00:00:00	F	Số 2, Đường X, Quận Y, Hồ Chí Minh	hs2@example.com	234567890	CNTT2020	ITD01
ST000020	Lê Văn Cường	2003-01-03 00:00:00	M	Số 3, Đường Z, Quận T, Đà Nẵng	hs3@example.com	345678901	CNTT2020	ITD01
ST000021	Phạm Thị Đào	2003-01-04 00:00:00	F	Số 4, Đường M, Quận N, Hải Phòng	hs4@example.com	456789012	CNTT2020	ITD01
ST000022	Hoàng Văn Em	2003-01-05 00:00:00	M	Số 5, Đường E, Quận F, Cần Thơ	hs5@example.com	567890123	CNTT2020	ITD01
ST000023	Vũ Thị F	2003-01-06 00:00:00	F	Số 6, Đường G, Quận H, Đồng Nai	hs6@example.com	678901234	CNTT2020	ITD01
ST000024	Nguyễn Văn G	2003-01-07 00:00:00	M	Số 7, Đường I, Quận J, Thành Hóa	hs7@example.com	789012345	CNTT2020	ITD01
ST000025	Trần Thị H	2003-01-08 00:00:00	F	Số 8, Đường K, Quận L, Nghệ An	hs8@example.com	890123456	CNTT2020	ITD01
ST000026	Lê Văn I	2003-01-09 00:00:00	M	Số 9, Đường O, Quận P, Hà Tĩnh	hs9@example.com	901234567	CNTT2020	ITD01
ST000027	Phạm Thị J	2003-01-10 00:00:00	F	Số 10, Đường Q, Quận R, Bình Định	hs10@example.com	012345678	CNTT2020	ITD01
ST000028	Hoàng Văn K	2003-01-11 00:00:00	M	Số 11, Đường S, Quận T, Đồng Tháp	hs11@example.com	123456789	CNTT2020	ITD01

The browser's address bar shows the URL 127.0.0.1:5000/search_student. The taskbar at the bottom includes icons for Search, Home, and various applications.

6.2.6.5 Grade manage

The screenshot shows a web browser window titled "Search grade". The address bar displays the URL "127.0.0.1:5000/search_student_grade". The main content area is titled "Search Student" and contains a search form with fields for "Student ID", "Student Name", "Course ID", "Semester", "LecturerID", and "Class". There are dropdown menus for "Faculty" and a "Search" button. Below the form are two download buttons: "Download Excel" and "Download report pdf". The title "List of Student" is displayed above a table. The table has columns: ID, Name, Class ID, Faculty ID, Course ID, Semester, Lecturer ID, Process, Mid, Final, and Avg. The data in the table shows multiple entries for "Nguyễn Văn A" across different courses and semesters.

ID	Name	Class ID	Faculty ID	Course ID	Semester	Lecturer ID	Process	Mid	Final	Avg
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT100	2021.1	LT000001	8.5	6.75	9.25	8.35
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT101	2021.1	LT000002	4.25	7.5	8.0	7.1
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT102	2021.1	LT000003	9.0	3.25	5.5	5.525
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT103	2021.1	LT000001	6.75	8.25	7.0	7.325
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT104	2021.1	LT000002	3.5	4.75	6.25	5.25
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT105	2021.2	LT000002	4.25	7.5	8.0	7.1
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT106	2021.2	LT000013	9.0	3.25	5.5	5.525
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT107	2021.2	LT000020	6.75	8.25	7.0	7.325
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT108	2021.2	LT000021	3.5	4.75	6.25	5.25
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT109	2021.2	LT000021	6.5	5.75	6.25	6.15
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT110	2022.1	LT000023	5.75	6.5	7.75	6.975

6.2.7 Admin dashboard

The screenshot shows a web browser window titled "Trang chủ - Quản lý sinh viên". The address bar displays the URL "127.0.0.1:5000/login". The main content area displays the message "Hello". On the left side, there is a vertical navigation menu with buttons for "STUDENT MANAGE", "LECTURER MANAGE", "COURSE MANAGE", "CHANGE PASSWORD", "BACK UP DATABASE", and "LOGOUT". The bottom of the screen shows the Windows taskbar with various pinned icons.

6.2.7.1 Student manage

The screenshot shows a web browser window titled "Search Student" at the URL 127.0.0.1:5000/search_student_record. The page has fields for Student ID, Student Name, Date of Birth, Gender, Address, Phone Number, Class, Faculty, and a Search button. Below this is a table titled "List of Student" with columns: ID, Name, Date of birth, Gender, Address, Email, Phone Number, Class ID, and Faculty ID. The table contains 18 rows of student data. At the bottom right of the page, there are "Add student" and "Delete student" buttons.

ID	Name	Date of birth	Gender	Address	Email	Phone Number	Class ID	Faculty ID
ST000001	Nguyễn Văn A	2000-01-01 00:00:00	M	Hà Nội	nguyenvana@gmail.com	0123456789	CNTT2021	ITD01
ST000003	Lê Văn C	2000-03-03 00:00:00	M	Hồ Chí Minh	levanc@gmail.com	0123456789	CNTT2021	ITD01
ST000005	Hoàng Văn E	2000-05-05 00:00:00	M	Hà Tĩnh	hoangvane@gmail.com	0123456789	CNTT2021	ITD01
ST000007	Trần Văn G	2000-07-07 00:00:00	M	Hải Phòng	tranvang@gmail.com	0123456789	CNTT2021	ITD01
ST000009	Phạm Văn I	2000-09-09 00:00:00	M	Đà Nẵng	phanvani@gmail.com	0123456789	CNTT2021	ITD01
ST000011	Nguyễn Văn L	2000-11-11 00:00:00	M	Hà Nội	nguyenvanl@gmail.com	0123456789	CNTT2021	ITD01
ST000013	Lê Văn N	2001-01-01 00:00:00	M	Hồ Chí Minh	levann@gmail.com	0123456789	CNTT2021	ITD01
ST000015	Hoàng Văn P	2001-03-03 00:00:00	M	Hà Tĩnh	hoangvnp@gmail.com	0123456789	CNTT2021	ITD01
ST000017	Trần Văn R	2001-05-05 00:00:00	M	Hải Phòng	tranvanr@gmail.com	0123456789	CNTT2021	ITD01
ST000018	Nguyễn Văn An	2003-01-01 00:00:00	M	Số 1, Đường A, Quận B, Hà Nội	hs1@example.com	123456789	CNTT2020	ITD01
ST000020	Lê Văn Cường	2003-01-03 00:00:00	M	Số 3, Đường Z, Quận T, Đà Nẵng	hs3@example.com	345678901	CNTT2020	ITD01

6.2.7.2 Lecturer manage

The screenshot shows a web browser window titled "Search Lecturer" at the URL 127.0.0.1:5000/search_lecturer_record. The page has fields for Lecturer ID, Lecturer Name, Date of Birth, Gender, Address, Phone Number, Faculty, and a Search button. Below this is a table titled "List of Lecturer" with columns: ID, Name, Date of birth, Gender, Address, Email, Phone Number, and Faculty ID. The table contains 24 rows of lecturer data. At the bottom right of the page, there are "Add lecturer" and "Delete lecturer" buttons.

ID	Name	Date of birth	Gender	Address	Email	Phone Number	Faculty ID
LT000001	Nguyen Van X	1990-01-01 00:00:00	M	123 Nguyen Van Cu, Quan 5, TP. Ho Chi Minh	nx@itd.edu.vn	0123456789	ITD01
LT000003	Pham Van Z	1977-07-15 00:00:00	M	789 Pham Van Dong, Quan 12, TP. Ho Chi Minh	pz@itd.edu.vn	0111222333	ITD02
LT000005	Le Van U	1980-12-31 00:00:00	M	555 Ly Thuong Kiet, Quan 10, TP. Ho Chi Minh	lvu@itd.edu.vn	0123456789	ITD03
LT000007	Tran Van W	1983-09-30 00:00:00	M	777 Le Hong Phong, Quan 10, TP. Ho Chi Minh	tvw@itd.edu.vn	0111222333	ITD03
LT000009	Hoang Van Y	1984-01-10 00:00:00	M	123 Pham Ngu Lao, Quan 1, TP. Ho Chi Minh	hvy@itd.edu.vn	0123456789	ITD04
LT000011	Nguyen Van A	1975-01-01 00:00:00	M	123 Nguyen Van Cu, Quan 5, TP. Ho Chi Minh	vana@itd.edu.vn	0111222333	ITD03
LT000013	John Doe	1980-01-01 00:00:00	M	123 Main St., Anytown, USA	johndoe@example.com	1234567890	ITD01
LT000015	Peter Parker	1989-07-01 00:00:00	M	789 5th Ave., New York, NY	peterparker@example.com	3456789012	ITD03
LT000020	John Doe	1980-01-01 00:00:00	M	123 Đường Chinh, Thành phố HCM	johndoe@example.com	0123456789	ITD01
LT000022	Nguyễn Văn A	1975-12-31 00:00:00	M	789 Đường Chinh, Thành phố Đà Nẵng	vana@example.com	0123456789	ITD01
LT000024	Lê Văn C	1990-03-20 00:00:00	M	654 Đường Chinh, Thành phố Cần Thơ	levanc@example.com	0123456789	ITD01

6.2.7.3 Course manage

Course ID	Course Name	Faculty ID	Credit	Description	Previous Course	Following Course
IT100	Cấu trúc dữ liệu và giải thuật	ITD01	3	None	None	IT101
IT101	Lập trình C/C++	ITD01	4	None	None	IT102
IT102	Lập trình Java	ITD01	4	None	None	IT103
IT103	Lập trình Python	ITD01	3	None	None	IT104
IT104	Hệ điều hành	ITD01	3	None	None	IT105
IT105	Mạng máy tính	ITD01	3	None	IT102	IT106
IT106	Cơ sở dữ liệu	ITD01	3	None	IT101	IT107
IT107	Quản lý dự án phần mềm	ITD01	3	None	IT103	IT108
IT108	Hệ thống thông tin	ITD01	3	None	IT105	IT109
IT109	An toàn thông tin	ITD01	3	None	IT104	IT110

6.2.7.4 Backup database

Username:

Password:

Backup

CHAPTER 7: INSTALLATION

7.1 Technology Usage

-The project uses the following technologies:

- +Draw.io is used for diagrams related specifications.
- +MySQL is used for database design.
- +HTML, CSS, and JavaScript for interface design.
- +Python is used for programming.
- +Flask module is used for main programming.

-In addition to these technologies, there are other Python modules used to support various functionalities such as login, report generation, and more.

-Here's a breakdown of the technologies used in the project:

-Draw.io: This tool is used to create diagrams and visual representations of specifications related to the project. It helps in visualizing the system architecture, data flow, and other important aspects of the project.

-MySQL: The project utilizes MySQL as the database management system. It is used to design and manage the project's database. MySQL allows for efficient storage and retrieval of data, making it a popular choice for web applications.

-HTML, CSS, and JavaScript: These web technologies are used for interface design. HTML provides the structure of web pages, CSS is used for styling and layout, and JavaScript adds interactivity and enhances the user experience.

-Python: Python is the programming language used for the project. It is a versatile language with a wide range of libraries and frameworks available for web development. Python's simplicity and readability make it a popular choice for developing web applications.

-Flask: Flask is a lightweight web framework for Python. It provides tools and libraries for building web applications, handling routing, managing sessions, and more. Flask follows the Model-View-Controller (MVC) architectural pattern, making it easy to organize and develop web applications.

-Other Python modules: The project likely utilizes additional Python modules for specific functionalities. For example, the project includes modules for generating reports, handling email functionality with Flask-Mail, integrating with external libraries like Matplotlib for data visualization, and more. These modules enhance the capabilities of the application and provide additional features.

-Overall, the project combines various technologies and libraries to create a web application using Flask as the main framework. The chosen technologies support different aspects of the application, including database management, interface design, programming logic, and additional functionalities required for the project.

7.2 Design pattern

-The three-tier architecture is a commonly used software architectural pattern in software development. It divides a system into three main layers: the presentation layer, the application layer, and the data layer.

+Presentation Layer: This is the user interface layer that interacts with the users. It includes components such as web pages, mobile applications, or graphical user interfaces. Each actor (student, lecturer, admin user, admin) will have their own interface to interact with the system.

+Application Layer: This layer contains the business rules and application logic of the system. It processes user requests from the presentation layer and interacts with the data layer to retrieve or update student and lecturer information, as well as administrative accounts. The application layer is responsible for implementing the specific functionalities of the system, such as user authentication, data validation, and business operations.

+Data Layer: This is the storage layer of the system. It can utilize a database management system MySQL to store student and lecturer information, as well as administrative data. The data layer handles data retrieval, storage, and management, ensuring the persistence and integrity of the system's data.

CHAPTER 8: TESTING

-To validate the website's functionality and usability, three testing approaches were employed:

+Manual Testing: This approach involved human testers interacting with the website's user interface, performing various actions, and verifying the expected outcomes. It allowed for real-world simulation and identification of user experience issues, usability flaws, and design inconsistencies.

+Black-Box Testing: By treating the website as a "black box," the testers focused on inputs and outputs without knowledge of the internal implementation. Test cases were designed to evaluate the website's functionality based on specified requirements, ensuring that it produced correct outputs for different inputs.

+White-Box Testing: Testers accessed the source code and database structure to design test cases targeting specific code sections, data flows, and algorithms. This approach helped uncover vulnerabilities, logical flaws, and performance bottlenecks by analyzing the internal structure of the website.

-Results and Conclusion:

+Through thorough testing, the student management website successfully met the specified requirements and demonstrated its functionality, reliability, and usability. The combination of manual testing, black-box testing, and white-box testing ensured that the website was robust, secure, and efficient.

+In conclusion, the development and testing of the student management website using Flask and MySQL proved to be a valuable project. The combination of appropriate development methodologies and testing approaches ensured the creation of a reliable and user-friendly platform for managing student-related information.

NO	Function	Completion	Note
1	Login	100%	<ul style="list-style-type: none">-If the username or password is not correct, or the password is less than 8 characters, a message will be displayed.-If the user enters the wrong password 5 times, the user will be blocked for 5 minutes.
2	Create account	100%	<ul style="list-style-type: none">-An account will be automatically created and sent to the user when the admin adds a new user record.
3	Reset password	100%	<ul style="list-style-type: none">-If the user inputs the username and email correctly, an email with a new password will be automatically sent to the user.
4	Change password	100%	<ul style="list-style-type: none">-If the user inputs the old password correctly, and the new password and

			confirm new password are the same, the password will be changed and the user will need to log in again.
Student			
5	View tuitionfee	100%	-The tuition fee is automatically updated whenever the user adds or deletes a course on the course registration page..
6	View profile	100%	
7	Update profile	100%	-Successfully update new email or phone number or address.
8	Schedule	100%	-Students can only see the attendance form whenever the lecturer opens it. -The attendance form was successfully saved to the database.
9	View test schedule	100%	
10	Course registration	100%	-Successfully adding a new course to the registered list, or deleting an existing course from the list.
11	Grade	100%	-Report pdf file was download successfully.
Lecturer			
12	View profile	100%	
13	Update profile	100%	-Successfully update new email or phone number or address.
14	Course manage	100%	-Successfully update grade for any student that belong to lecturer's course. -Successfully generate report for previous semester and current semester with both excel and pdf file with chart for each class. -Students can only see the attendance form if the lecturer opens it. -The lecturer can download the attendance data form if any student's form exists. The data will be automatically deleted after the lecturer downloads it.
Admin user			
15	View profile	100%	
16	Update profile	100%	-Successfully update new email or phone number or address.
17	Schedule manage	100%	-Successfully adding new schedule and deleting an existing schedule. -Successfully adding new student to existing schedule and delete from schedule. -Admin user can download schedule as an excel file.

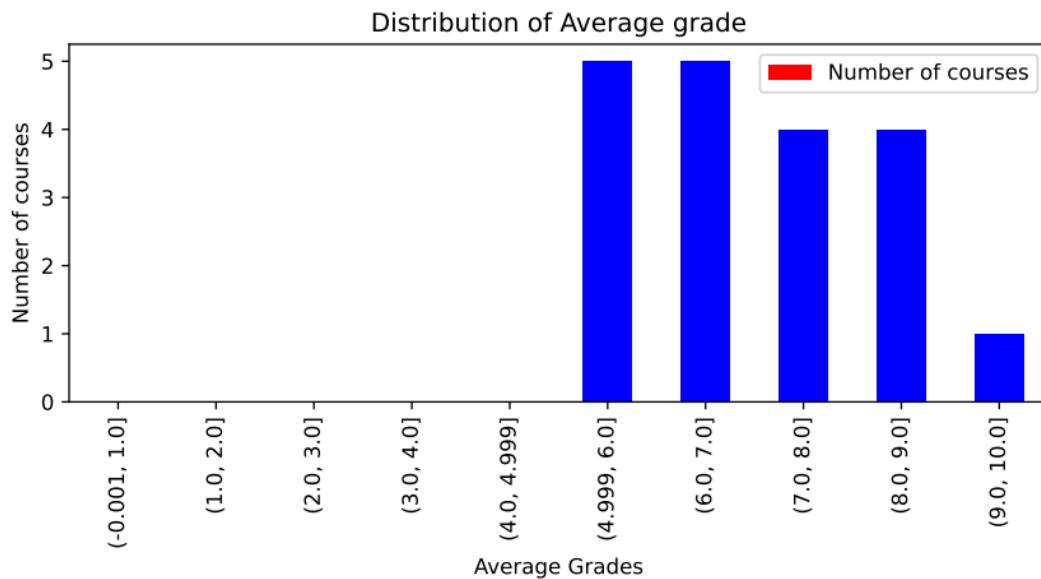
18	Test schedule manage	100%	-Successfully adding and deleting test schedule base on current schedule.
19	Student manage	100%	-Successfully search for students with: student ID, student name, date of birth, gender, address, email, phone number, class, faculty. Results will include approximate matches for the input fields. -User can download search data as an Excel file.
20	Grade report	100%	-Successfully search for students with: student ID, student name, course ID, semester, lecturer ID, class, faculty. Results will include approximate matches for the input fields. -User can download search data as an Excel file and an report pdf file.
Admin			
21	Student manage	100%	-Successfully search for students with: student ID, student name, date of birth, gender, address, email, phone number, class, faculty. Results will include approximate matches for the input fields. -Successfully adding new student. Account will be automatically created and send to user through email. -Delete the student successfully if the conditions are met.
22	Lecturer manage	100%	-Successfully search for lecturers with: student ID, lecturer name, date of birth, gender, address, email, phone number, faculty. Results will include approximate matches for the input fields. -Successfully adding new lecturer. Account will be automatically created and send to user through email. -Delete the lecturer successfully if the conditions are met.
23	Course manage	100%	-Successfully search for courses with: course ID, course name, faculty. Results will include approximate matches for the input fields. -Sucessfully adding new course or deleting existing course.
24	Back up database	100%	-Database was backup successfully and send to admin email.
25	Restore database	0%	

26	End semester	0%	-Delete the old schedule if all grades have been entered by the lecturer and update the new schedule from the course registration list.
27	Update tuitionfee	0%	-Update tuition fee if student paid.

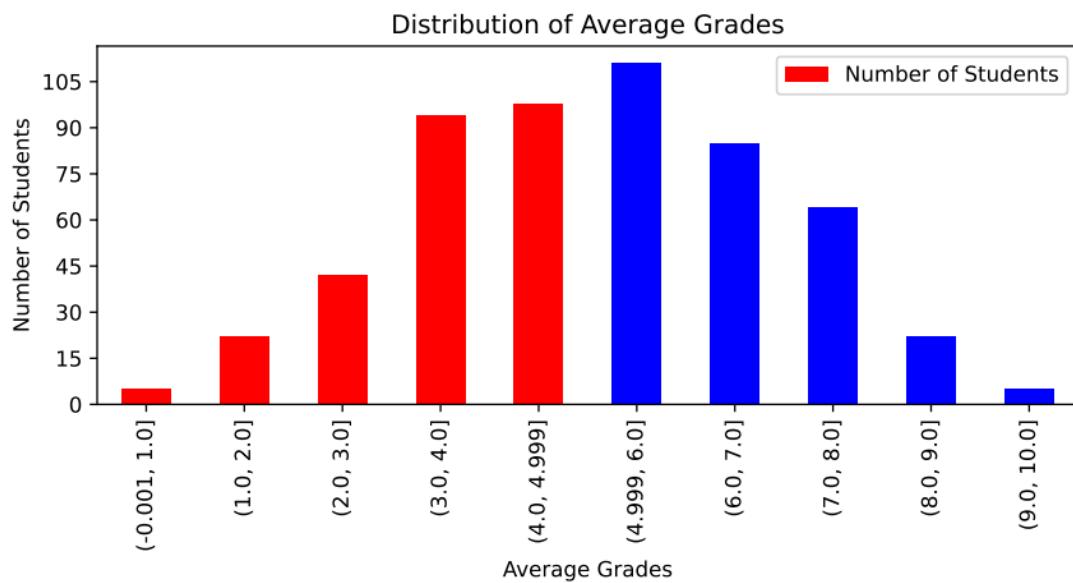
Appendix

1. Score distribution histogram chart template

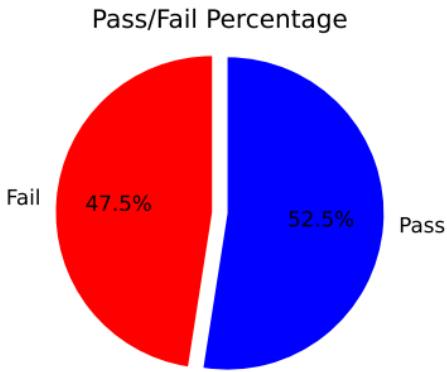
a. For students



b. For lecturers, admin users



2. Pie-char template



3. Score data table template

a. For students

GRADE REPORT								
GPA:	6.93							
courseID	semester	lecturerID	process	mid	final	avg	credit	
IT100	2021.1	LT000001	8.5	6.75	9.25	8.35	3	
IT101	2021.1	LT000002	4.25	7.5	8.0	7.1	4	
IT102	2021.1	LT000003	9.0	3.25	5.5	5.525	4	
IT103	2021.1	LT000001	6.75	8.25	7.0	7.325	3	
IT104	2021.1	LT000002	3.5	4.75	6.25	5.25	3	
IT105	2021.2	LT000002	4.25	7.5	8.0	7.1	3	
IT106	2021.2	LT000013	9.0	3.25	5.5	5.525	3	
IT107	2021.2	LT000020	6.75	8.25	7.0	7.325	3	
IT108	2021.2	LT000021	3.5	4.75	6.25	5.25	3	
IT109	2021.2	LT000021	6.5	5.75	6.25	6.15	3	
IT110	2022.1	LT000023	5.75	6.5	7.75	6.975	3	
IT111	2022.1	LT000024	8.0	7.25	9.0	8.275	3	
IT112	2022.1	LT000025	6.25	5.5	8.0	6.9	3	
IT113	2022.1	LT000026	4.75	6.75	7.5	6.725	3	
IT114	2022.1	LT000027	9.25	8.0	9.75	9.125	3	
IT115	2022.2	LT000028	7.0	7.75	6.5	6.975	3	
IT116	2022.2	LT000029	5.5	4.25	6.75	5.75	3	

- b. For lecturers, admin users

GRADE REPORT											
StudentID	Name	ClassID	FacultyID	CourseID	Semester	LecturerID	process	mid	final	avg	
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT101	2021.1	LT000002	4.25	7.5	8.0	7.1	
ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT101	2022.2	LT000023	1.4	2.7	2.6	2.39	
ST000002	Trần Thị B	CNTT2021	ITD01	IT101	2021.2	LT000025	4.5	9.7	7.1	7.36	
ST000004	Phạm Thị D	CNTT2021	ITD01	IT101	2022.2	LT000024	3.6	2.4	8.1	5.49	
ST000006	Nguyễn Thị F	CNTT2021	ITD01	IT101	2022.2	LT000027	6.2	8.2	6.9	7.15	
ST000007	Trần Văn G	CNTT2021	ITD01	IT101	2022.2	LT000022	1.1	8.3	8.0	6.71	
ST000008	Lê Thị H	CNTT2021	ITD01	IT101	2022.2	LT000029	7.3	7.7	0.2	3.87	
ST000011	Nguyễn Văn L	CNTT2021	ITD01	IT101	2021.2	LT000022	9.7	0.2	1.2	2.6	
ST000012	Trần Thị M	CNTT2021	ITD01	IT101	2022.2	LT000025	2.9	3.4	9.1	6.15	
ST000015	Hoàng Văn P	CNTT2021	ITD01	IT101	2022.2	LT000025	7.1	4.1	6.8	6.05	
ST000016	Nguyễn Thị Q	CNTT2021	ITD01	IT101	2022.2	LT000001	7.5	1.3	2.9	3.34	
ST000017	Trần Văn R	CNTT2021	ITD01	IT101	2021.1	LT000001	4.8	4.6	4.0	4.34	
ST000019	Trần Thị Bình	CNTT2020	ITD01	IT101	2021.1	LT000025	4.2	8.5	1.7	4.24	
ST000020	Lê Văn Cường	CNTT2020	ITD01	IT101	2021.2	LT000020	0.8	2.2	9.5	5.57	
ST000022	Hoàng Văn Em	CNTT2020	ITD01	IT101	2022.2	LT000027	0.3	3.0	3.4	2.66	
ST000023	Vũ Thị F	CNTT2020	ITD01	IT101	2022.2	LT000024	9.4	0.5	6.9	5.48	
ST000025	Trần Thị H	CNTT2020	ITD01	IT101	2021.1	LT000021	9.2	5.7	7.3	7.2	
ST000025	Trần Thị H	CNTT2020	ITD01	IT101	2022.2	LT000023	5.9	5.4	3.9	4.75	
ST000028	Hoàng Văn K	CNTT2020	ITD01	IT101	2021.2	LT000028	9.7	8.1	6.5	7.62	
ST000029	Vũ Thị L	CNTT2020	ITD01	IT101	2022.2	LT000013	1.2	0.2	0.9	0.75	
ST000031	Trần Thị N	CNTT2020	ITD01	IT101	2022.2	LT000023	3.8	3.8	0.9	2.35	
ST000032	Lê Văn O	CNTT2020	ITD01	IT101	2021.2	LT000001	1.1	4.3	0.4	1.71	
ST000039	Phạm Thị V	CNTT2020	ITD01	IT101	2022.2	LT000023	3.2	6.5	0.3	2.74	

4. Excel file template

- a. For lecturers

A	B	C	D	E	F
1	Student ID	Full name	Process	Mid	Final
2	ST000001	Nguyễn Văn A	8.5	6.75	9.25
3	ST000028	Hoàng Văn K	3.4	7.1	2.3
4					
5					

- b. For admin users

-Schedule information

A	B	C	D
Key	Value	Student ID	Name
2 Course ID	DC114		
3 Course Name	Big Data Analytics		
4 Day	tue		
5 Time	0		
6 Classroom ID	C007		
7 Lecturer Name	Tran Thi Y		
8 Semester	2023.2		
9		ST000001	Nguyễn Văn A
10		ST000002	Trần Thị B
11		ST000003	Lê Văn C
12			
13			
14			
15			

-List of students

A	B	C	D	E	F	G	H	I	
1	studentID	name	dateOfBirth	gender	address	email	phoneNumber	classID	facultyID
2	ST000018	Nguyễn Văn An	2003-01-01 00:00:00	M	Số 1, Đường A, Quận B, Hà Nội	hs1@example.com	123456789	CNTT2020	ITD01
3	ST000019	Trần Thị Bình	2003-01-02 00:00:00	F	Số 2, Đường X, Quận Y, Hồ Chí Minh	hs2@example.com	234567890	CNTT2020	ITD01
4	ST000020	Lê Văn Cường	2003-01-03 00:00:00	M	Số 3, Đường Z, Quận T, Đà Nẵng	hs3@example.com	345678901	CNTT2020	ITD01
5	ST000021	Phạm Thị Đào	2003-01-04 00:00:00	F	Số 4, Đường M, Quận N, Hải Phòng	hs4@example.com	456789012	CNTT2020	ITD01
6	ST000022	Hoàng Văn Em	2003-01-05 00:00:00	M	Số 5, Đường E, Quận F, Cần Thơ	hs5@example.com	567890123	CNTT2020	ITD01
7	ST000023	Vũ Thị F	2003-01-06 00:00:00	F	Số 6, Đường G, Quận H, Đồng Nai	hs6@example.com	678901234	CNTT2020	ITD01
8	ST000024	Nguyễn Văn G	2003-01-07 00:00:00	M	Số 7, Đường I, Quận J, Thanh Hóa	hs7@example.com	789012345	CNTT2020	ITD01
9	ST000025	Trần Thị H	2003-01-08 00:00:00	F	Số 8, Đường K, Quận L, Nghệ An	hs8@example.com	890123456	CNTT2020	ITD01
10	ST000026	Lê Văn I	2003-01-09 00:00:00	M	Số 9, Đường O, Quận P, Hà Tĩnh	hs9@example.com	901234567	CNTT2020	ITD01
11	ST000027	Phạm Thị J	2003-01-10 00:00:00	F	Số 10, Đường Q, Quận R, Bình Định	hs10@example.com	012345678	CNTT2020	ITD01
12	ST000028	Hoàng Văn K	2003-01-11 00:00:00	M	Số 11, Đường S, Quận T, Đồng Tháp	hs11@example.com	123456789	CNTT2020	ITD01
13	ST000029	Vũ Thị L	2003-01-12 00:00:00	F	Số 12, Đường U, Quận V, Long An	hs12@example.com	234567890	CNTT2020	ITD01
14	ST000030	Nguyễn Văn M	2003-01-13 00:00:00	M	Số 13, Đường W, Quận X, An Giang	hs13@example.com	345678901	CNTT2020	ITD01

-List of students with grade

A	B	C	D	E	F	G	H	I	J	K	
1	StudentID	Name	ClassID	FacultyID	CourseID	Semester	LecturerID	process	mid	final	avg
2	ST000001	Nguyễn Văn A	CNTT2021	ITD01	CS115	2023.2	lt000001	2	3	4	3.3
3	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT100	2021.1	LT000001	8.5	6.75	9.25	8.35
4	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT101	2021.1	LT000002	4.25	7.5	8	7.1
5	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT101	2022.2	LT000023	1.4	2.7	2.6	2.39
6	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT102	2021.1	LT000003	9	3.25	5.5	5.525
7	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT102	2022.2	LT000028	8.3	5.5	1.3	3.96
8	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT103	2021.1	LT000001	6.75	8.25	7	7.325
9	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT103	2022.2	LT000023	4.7	6.4	7.9	6.81
10	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT104	2021.1	LT000002	3.5	4.75	6.25	5.25
11	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT105	2021.2	LT000002	4.25	7.5	8	7.1
12	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT105	2022.2	LT000002	2.9	8.2	0.9	3.49
13	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT106	2021.2	LT000013	9	3.25	5.5	5.525
14	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT106	2022.2	LT000020	6.9	5.1	3.8	4.81
15	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT107	2021.2	LT000020	6.75	8.25	7	7.325
16	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT108	2021.2	LT000021	3.5	4.75	6.25	5.25
17	ST000001	Nguyễn Văn A	CNTT2021	ITD01	IT109	2021.1	LT000002	1.4	0.1	1.6	1.11

5. Source code

<https://github.com/hieupham12345/Student-management-website-using-Python-Flask>