

TRƯỜNG ĐẠI HỌC BÁCH KHOA - ĐHQG TP. HỒ CHÍ MINH

KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



ĐỒ ÁN TỔNG HỢP  
HƯỚNG CÔNG NGHỆ PHẦN MỀM  
HỌC KỲ: 241

**CHỦ ĐỀ:**

**HỆ THỐNG QUẢN LÝ VÀ TRA CỨU ĐIỂM  
MÔN HỌC**

GV hướng dẫn: Lê Đình Thuận  
Lớp L07 - Nhóm con gà

Sinh viên:

Nguyễn Hữu Khánh - 2211521  
Nguyễn Hà Thùy Linh - 2211856  
Nguyễn Ngọc Quý - 2212870  
Nguyễn Minh Tân - 2213057  
Lê Trường Thịnh - 2213282  
Trần Huy Đức - 2210812  
Nguyễn Công Thành - 2213132

TP. Hồ Chí Minh, Tháng 12/2024



## Mục lục

|          |  |          |
|----------|--|----------|
| <b>1</b> | <b>Danh sách thành viên và phân công nhiệm vụ</b>                | <b>2</b> |
| <b>2</b> | <b>Giới thiệu chung về dự án</b>                                 | <b>3</b> |
| 2.1      | Bối cảnh và lý do thực hiện đề tài . . . . .                     | 3        |
| 2.2      | Các bên liên quan (Stakeholders) và nhu cầu của họ . . . . .     | 3        |
| 2.3      | Lợi ích của dự án . . . . .                                      | 4        |
| <b>3</b> | <b>Phân Tích Các Yêu Cầu Của Dự Án</b>                           | <b>5</b> |
| 3.1      | Các yêu cầu chức năng (Functional requirements) . . . . .        | 5        |
| 3.2      | Các yêu cầu phi chức năng (Nonfunctional requirements) . . . . . | 5        |
| <b>4</b> | <b>Mô hình hệ thống (system modeling)</b>                        | <b>6</b> |
| 4.1      | Use-case diagram . . . . .                                       | 6        |
| 4.2      | Use Case Specification . . . . .                                 | 7        |
| 4.2.1    | Quản lý lớp học . . . . .  | 7        |
| 4.2.2    | Tra cứu điểm . . . . .   | 8        |
| 4.2.3    | Quản lý tài khoản . . . . .                                      | 9        |
| 4.3      | Activity diagram . . . . .                                       | 10       |
| 4.3.1    | Đăng nhập và quản lý lớp học . . . . .                           | 10       |
| 4.3.2    | Tra cứu điểm . . . . .   | 11       |
| 4.3.3    | Quản lý tài khoản . . . . .                                      | 12       |
| 4.3.4    | Quản lý môn học . . . . .  | 13       |
| 4.4      | Sequence diagram . . . . .                                       | 14       |
| 4.4.1    | Manage grade (lecturer) . . . . .                                | 14       |
| 4.4.2    | View student grades (lecturer) . . . . .                         | 15       |
| 4.4.3    | Search grade . . . . .   | 16       |
| 4.4.4    | View grade . . . . .   | 17       |
| 4.4.5    | Login . . . . .  | 18       |
| 4.4.6    | Register . . . . .   | 19       |
| 4.4.7    | Generate report . . . . .  | 20       |
| 4.4.8    | Manage account . . . . .   | 21       |
| 4.4.9    | Manage subject . . . . .   | 23       |
| 4.5      | Manage class . . . . .   | 24       |
| 4.5.1    | View student grade . . . . .                                     | 25       |
| 4.6      | Class diagram . . . . .  | 27       |
| 4.7      | Entity-Relationship diagram . . . . .                            | 28       |



## 1 Danh sách thành viên và phân công nhiệm vụ

| Họ và tên           | MSSV    | Nhiệm vụ | Phần trăm hoàn thành |
|---------------------|---------|----------|----------------------|
| Nguyễn Hữu Khánh    | 2211521 | PO       | 100%                 |
| Nguyễn Hà Thùy Linh | 2211856 | DEV      | 100%                 |
| Nguyễn Ngọc Quý     | 2212870 | DEV      | 100%                 |
| Nguyễn Minh Tân     | 2213057 | DEV      | 100%                 |
| Lê Trường Thịnh     | 2213282 | DEV      | 100%                 |
| Trần Huy Đức        | 2210812 | DEV      | 100%                 |
| Nguyễn Công Thành   | 2213132 | DEV      | 100%                 |

## 2 Giới thiệu chung về dự án

### 2.1 Bối cảnh và lý do thực hiện đề tài

Trong những năm gần đây, khi số lượng sinh viên theo học tại Đại học Bách Khoa - Đại học Quốc gia TP.HCM đang không ngừng gia tăng. Điều đó khiến việc quản lý thông tin học tập và điểm số của sinh viên trở thành một thách thức ngày càng lớn. Cùng với đó, chương trình đào tạo tại trường liên tục đổi mới, cập nhật để bắt kịp với sự phát triển nhanh chóng của công nghệ và khoa học toàn cầu, đặt ra yêu cầu cấp bách về việc xây dựng các hệ thống hỗ trợ quản lý và tra cứu thông tin học tập một cách hiệu quả. Trong số đó, một hệ thống cho phép sinh viên có thể tra cứu điểm số trong quá trình học tập là vô cùng quan trọng, không chỉ giúp sinh viên có thể theo dõi kết quả học tập của mình mà còn tạo điều kiện thuận lợi cho giảng viên trong việc quản lý lớp học và các dữ liệu liên quan đến điểm số.

Hiện nay, tại nhiều trường đại học trên thế giới, việc áp dụng các hệ thống quản lý điểm số thông minh đã trở thành xu hướng phổ biến, nhằm giúp cả sinh viên lẫn giảng viên dễ dàng truy cập, quản lý và cập nhật thông tin về kết quả học tập. Đại học Bách Khoa cũng không phải là ngoại lệ, với nhu cầu cấp thiết cho một hệ thống quản lý điểm số khi số lượng sinh viên ngày càng tăng, đặc biệt là trong những môn học có đông sinh viên tham gia. Việc áp dụng một hệ thống giúp sinh viên có thể tra cứu điểm số, theo dõi quá trình học tập và cập nhật kết quả của mình sẽ giúp giảm bớt áp lực cho cả giảng viên và sinh viên trong việc xử lý các thông tin về điểm số một cách thủ công, đồng thời giúp nâng cao tính minh bạch và chính xác trong việc công bố kết quả học tập.

Chính vì lẽ đó, dự án thiết kế hệ thống quản lý điểm trực tuyến này đã được thực hiện nhằm hướng đến mục tiêu hỗ trợ sinh viên và giảng viên tại Đại học Bách Khoa theo dõi, tra cứu điểm các môn học một cách thuận tiện hơn. Hệ thống này sẽ là một giải pháp hoàn hảo giúp cải thiện chất lượng học tập và giảng dạy cho sinh viên và giảng viên trong trường.

Sử dụng hệ thống này, sinh viên có thể kiểm tra, tra cứu điểm dễ dàng, tiết kiệm thời gian hơn trước. Điều này giúp cho các sinh viên có thể theo dõi và tự điều chỉnh quá trình học tập của mình nhằm phù hợp hơn với môi trường Bách Khoa. Về phần giảng viên, các thầy cô sẽ được hỗ trợ tính năng đính kèm bảng điểm dưới dạng tập tin CSV hoặc Excel, khiến cho việc nhập và theo dõi bảng điểm các lớp không còn phức tạp và mất thời gian như trước.

Một tính năng nổi bật khác của hệ thống, tuy không quá quan trọng nhưng lại vô cùng ý nghĩa, chính là tính năng bảng vinh danh (Hall of fame). Tính năng này sẽ động viên tinh thần sinh viên, giúp cho những sinh viên đã nỗ lực xứng đáng cảm thấy được trân trọng và khích lệ những sinh viên khác hãy cố gắng để đạt được thành tích tốt hơn. Từ đó tạo nên một môi trường học tập lành mạnh, nâng cao chất lượng học tập và giảng dạy.

Chung quy lại, nếu thành công, hệ thống này được dự báo sẽ trở thành công cụ hỗ trợ đắc lực cho sinh viên và giảng viên tại trường đại học Bách Khoa. Góp phần hiện đại hóa hệ thống giáo dục, nâng cao chất lượng và danh tiếng của trường.

### 2.2 Các bên liên quan (Stakeholders) và nhu cầu của họ

- **Sinh viên:** Sinh viên mong muốn một hệ thống hỗ trợ việc tra cứu điểm số nhanh chóng, minh bạch và dễ dàng sử dụng. Họ cần hệ thống tự động cập nhật điểm số một cách thường xuyên và đảm bảo thông tin chính xác khi có thay đổi. Ngoài ra, sinh viên kỳ vọng hệ thống có giao diện rõ ràng, thân thiện và đảm bảo mức độ bảo mật cao để bảo vệ quyền lợi cá nhân.
- **Giảng viên và nhân viên nhà trường:** Đối với giảng viên, một hệ thống quản lý điểm số hiệu quả là rất cần thiết. Hệ thống cần có khả năng xử lý khối lượng dữ liệu lớn, chẳng

hạn như danh sách điểm của sinh viên, và hỗ trợ các thao tác cập nhật điểm số một cách chính xác. Giảng viên cần tính năng tải và cập nhật dữ liệu từ các tệp như CSV, Excel một cách linh hoạt, đồng thời hệ thống phải tự động phản hồi khi có bản cập nhật mới.

- **Quản trị viên hệ thống (Admin):** Quản trị viên có nhiệm vụ quản lý tài khoản người dùng và phân quyền truy cập cho giảng viên. Do đó, họ cần một công cụ quản lý đơn giản, tiện lợi để giám sát, phân quyền, và bảo vệ dữ liệu người dùng. Ngoài ra, hệ thống cần duy trì tính ổn định trong quá trình vận hành, đặc biệt khi xử lý và cập nhật dữ liệu lớn.

## 2.3 Lợi ích của dự án

### Sinh viên (Student)

- **Dễ dàng tra cứu và cập nhật thông tin:** Sinh viên có thể nhanh chóng truy cập hệ thống để kiểm tra điểm số một cách chính xác và cập nhật kịp thời. Điều này giúp họ tiết kiệm thời gian và công sức so với các phương pháp truyền thống.
- **Theo dõi kết quả học tập minh bạch:** Hệ thống cung cấp thông tin rõ ràng và minh bạch, hỗ trợ sinh viên điều chỉnh chiến lược học tập phù hợp dựa trên kết quả thực tế.
- **Tăng tính chủ động:** Việc tra cứu bảng điểm trực tuyến giúp sinh viên tự quản lý quá trình học tập của mình hiệu quả hơn.
- **Tạo động lực học tập:** Tính năng Bảng Vinh Danh (Hall of Fame) khuyến khích sinh viên phấn đấu đạt thành tích cao và được ghi nhận trong cộng đồng.

### Giảng viên (Lecture)

- **Hỗ trợ quản lý điểm số hiệu quả:** Giảng viên có thể dễ dàng tải lên và cập nhật điểm số từ các tệp CSV hoặc Excel, giúp giảm thiểu khối lượng công việc thủ công.
- **Tăng tính chính xác và minh bạch:** Hệ thống tự động xử lý và cập nhật điểm số, hạn chế sai sót trong quá trình nhập liệu và công bố kết quả.
- **Tiết kiệm thời gian:** Các tính năng tự động hóa giảm thiểu thời gian xử lý dữ liệu, cho phép giảng viên tập trung vào chuyên môn giảng dạy.

### Quản trị viên hệ thống (Admin)

- **Quản lý người dùng và phân quyền đơn giản:** Admin có thể dễ dàng quản lý tài khoản sinh viên, giảng viên và thiết lập các quyền truy cập phù hợp với từng vai trò.
- **Đảm bảo bảo mật thông tin:** Hệ thống chú trọng đến tính bảo mật, giúp bảo vệ dữ liệu cá nhân của sinh viên và giảng viên khỏi các rủi ro truy cập trái phép.
- **Giám sát và vận hành hiệu quả:** Admin có thể theo dõi toàn bộ hoạt động của hệ thống, đảm bảo tính ổn định trong suốt quá trình sử dụng và xử lý các sự cố nhanh chóng, kịp thời.

## 3 Phân Tích Các Yêu Cầu Của Dự Án

### 3.1 Các yêu cầu chức năng (Functional requirements)

Đối với sinh viên:

- Sinh viên có tạo tài khoản, đăng nhập vào hệ thống, xem và chỉnh sửa thông tin của mình
- Sinh viên có thể đăng ký môn học, xem lớp học, xem điểm, xem bảng vinh danh trong lớp mình

Đối với giảng viên:

- Giảng viên có thể đăng nhập vào hệ thống bằng tài khoản được cấp bởi admin
- Giảng viên có quyền tạo lớp học, thêm sửa xóa thông tin lớp học, thêm tài liệu học tập vào lớp học, chỉnh sửa thông tin hiển thị bảng vinh danh
- Giảng viên có thể thêm điểm cho sinh viên, chỉnh sửa cũng như xóa điểm

Đối với admin:

- Admin có quyền tạo, xóa, cập nhật tài khoản cho giảng viên
- Admin có quyền tạo môn học mới, chỉnh sửa cũng như xóa môn học
- Admin có quyền xem tất cả thông tin của hệ thống

### 3.2 Các yêu cầu phi chức năng (Nonfunctional requirements)

Về hiệu năng:

- Hệ thống phải xử lý ít nhất 1.000 yêu cầu tra cứu đồng thời mà không ảnh hưởng đến hiệu suất
- Thời gian phản hồi khi tra cứu điểm không quá 3 giây

Về tính ổn định:

- Hệ thống cần hoạt động liên tục với thời gian hoạt động tối thiểu 99,9% mỗi năm.
- Đảm bảo dữ liệu không bị mất khi xảy ra sự cố.

Về tính bảo mật:

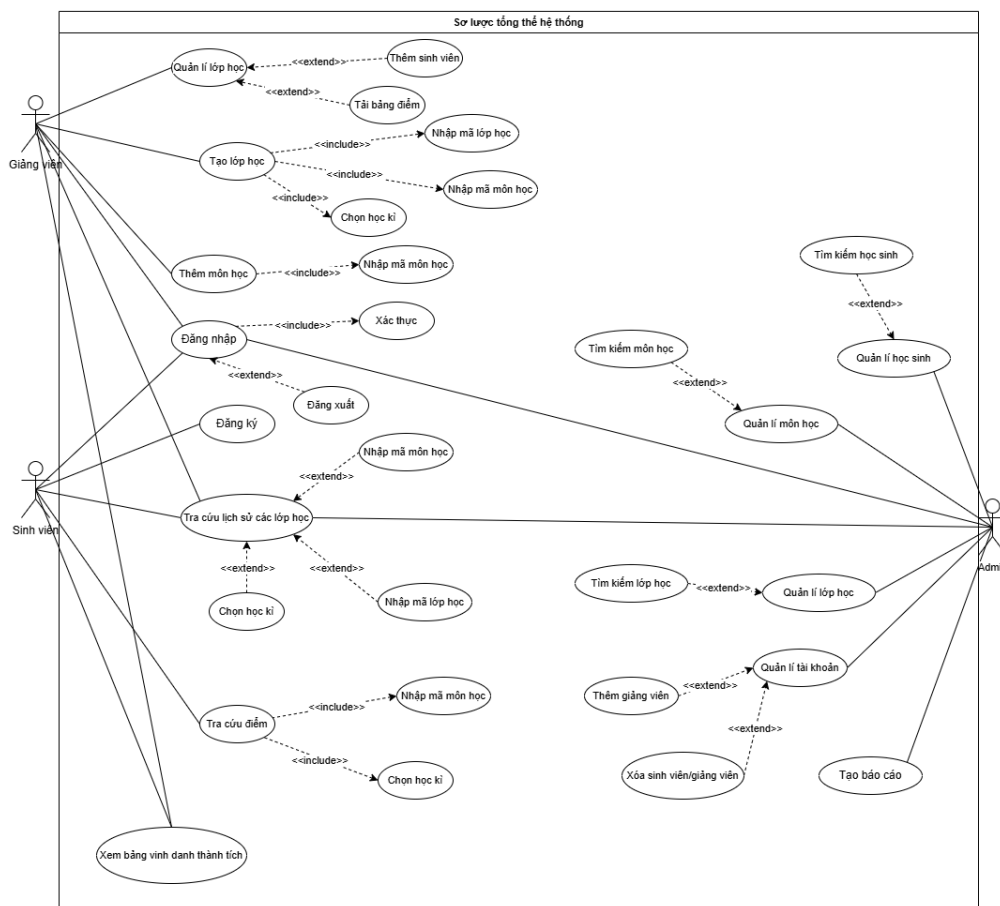
- Thông tin cá nhân và điểm số của sinh viên phải được mã hóa trong quá trình lưu trữ và truyền tải.
- Chỉ người dùng có quyền hợp lệ mới có thể truy cập hoặc chỉnh sửa dữ liệu.

Các tính năng khác:

- Hệ thống phải có khả năng mở rộng để xử lý dữ liệu của hơn 50.000 sinh viên trong tương lai.
- Giao diện phải thân thiện với người dùng và dễ dàng điều hướng trên cả thiết bị di động và máy tính.
- Hệ thống phải được thiết kế để dễ dàng sửa lỗi và nâng cấp tính năng mới.
- Các thay đổi trong dữ liệu phải được ghi lại trong lịch sử hệ thống và dễ dàng kiểm tra.

## 4 Mô hình hệ thống (system modeling)

### 4.1 Use-case diagram

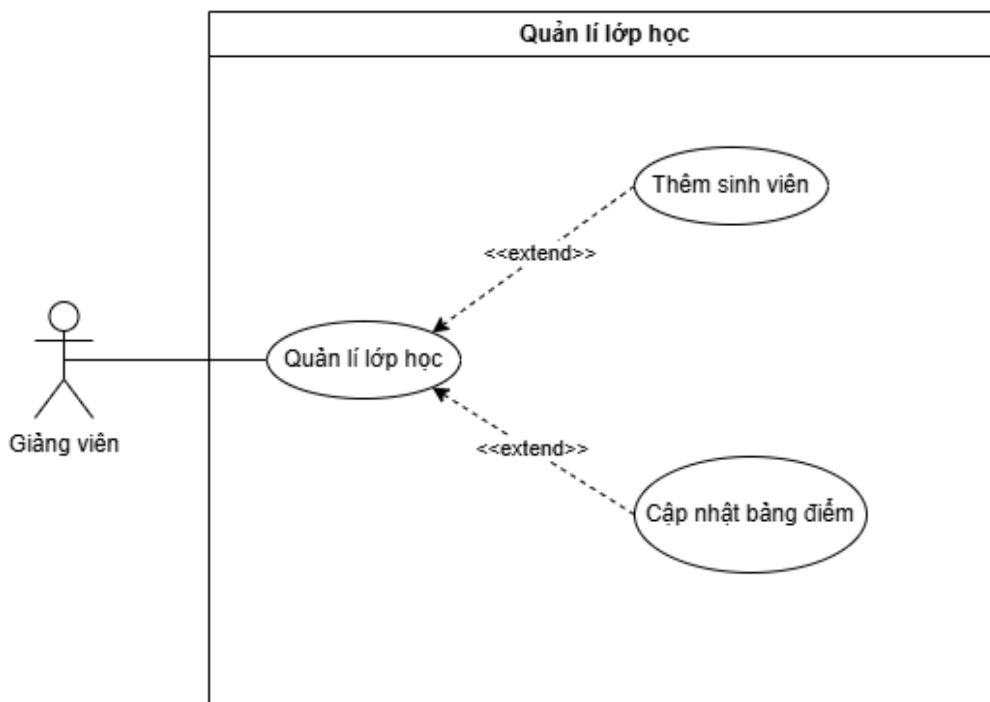


Hình 1: Sơ đồ use-case cho toàn bộ hệ thống

Như mọi người có thể thấy, sơ đồ use-case trên mô tả sơ lược toàn bộ các tác nhân và các hành động mà họ có thể thực hiện khi tương tác với hệ thống. Các tác nhân này gồm có sinh viên, giảng viên và quản trị viên. Đối với sinh viên, hệ thống cung cấp các chức năng như đăng ký tài khoản, tra cứu điểm môn học, tra cứu lịch sử các lớp học, xem bảng vinh danh thành tích. Còn với giảng viên, hệ thống sẽ cung cấp các chức năng quản lý lớp học, tạo lớp học, thêm môn học, tra cứu lịch sử các lớp học và xem bảng vinh danh. Tác nhân giảng viên không có chức năng đăng ký tài khoản vì tài khoản giảng viên sẽ được cung cấp bởi quản trị viên. Quản trị viên là người quản lý hệ thống. Họ sẽ có quyền quản lý tài khoản của giảng viên và học sinh, quản lý các lớp học, quản lý môn học, cũng như là tạo báo cáo.

## 4.2 Use Case Specification

### 4.2.1 Quản lí lớp học

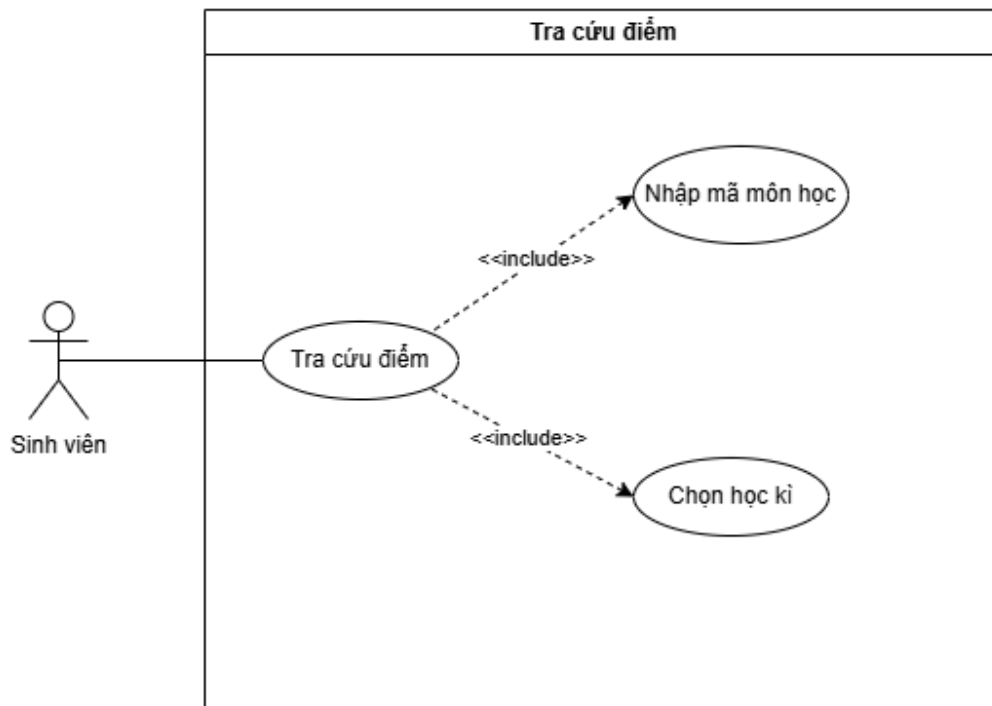


Hình 2: Usecase quản lí lớp học

| Thành phần          | Mô tả   |
|---------------------|---|
| Tác nhân chính      | Giảng viên.   |
| Tiền điều kiện      | Giảng viên đăng nhập vào hệ thống thành công.   |
| Hậu điều kiện       | Danh sách sinh viên cùng điểm được cập nhật thành công.   |
| Điều kiện kích hoạt | Giảng viên muốn quản lí lớp học.  |
| Luồng chính         | <ol style="list-style-type: none"> <li>1. Giảng viên truy cập chức năng "Quản lý lớp học" và chọn một lớp học do giảng viên phụ trách.</li> <li>2. Chọn thêm sinh viên hoặc cập nhật bảng điểm.</li> <li>3. Thêm những thông tin cần thiết.</li> <li>4. Hệ thống lưu thay đổi và thông báo thành công.</li> </ol> |
| Luồng phụ           | Giảng viên có thể quay lại chỉnh sửa thông tin trước khi lưu thay đổi.  |
| Luồng ngoại lệ      | Không có.   |



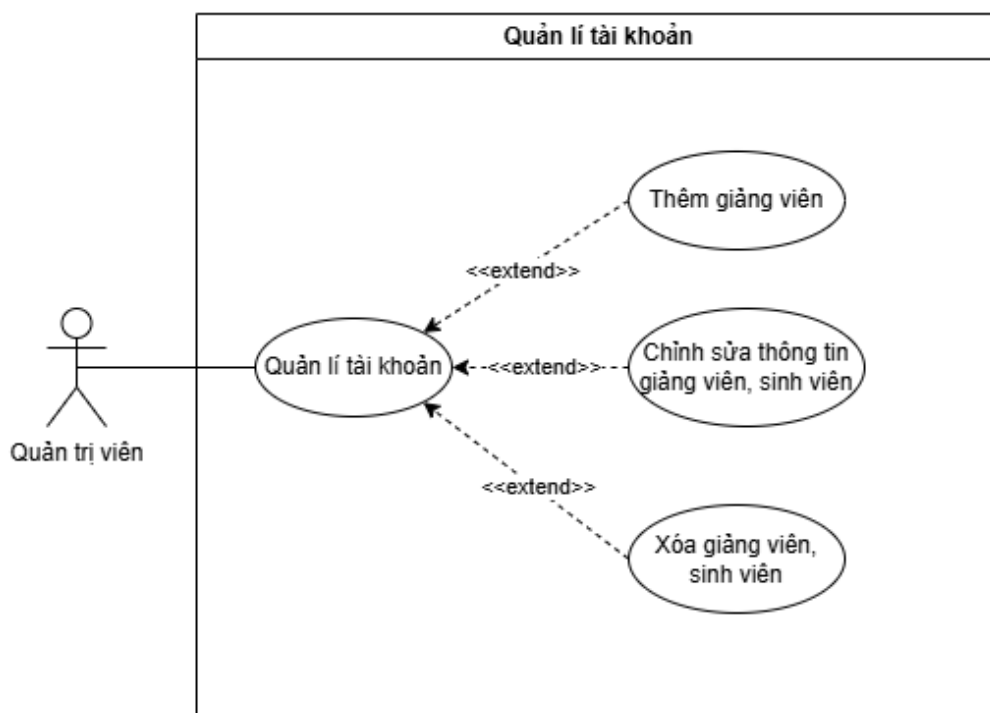
#### 4.2.2 Tra cứu điểm



Hình 3: Usecase tra cứu điểm

| Thành phần          | Mô tả  |
|---------------------|--|
| Tác nhân chính      | <i>Sinh viên.</i>  |
| Tiền điều kiện      | <i>Sinh viên đăng nhập vào hệ thống thành công.</i>  |
| Hậu điều kiện       | <i>Điểm của sinh viên được hiển thị theo mã môn học và học kỳ đã chọn.</i>   |
| Điều kiện kích hoạt | <i>Sinh viên muốn tra cứu điểm của mình.</i>   |
| Luồng chính         | <i>1. Sinh viên truy cập chức năng "Tra cứu điểm".<br/>2. Nhập mã môn học và chọn học kỳ.<br/>3. Hệ thống hiển thị kết quả điểm của sinh viên.</i> |
| Luồng phụ           | <i>Sinh viên có thể quay lại chỉnh sửa thông tin trước khi gửi yêu cầu.</i>  |
| Luồng ngoại lệ      | <i>Nếu sai mã môn học hoặc không có kết quả của sinh viên theo những thông tin được nhập thì thông báo lỗi.</i>                                    |

### 4.2.3 Quản lí tài khoản



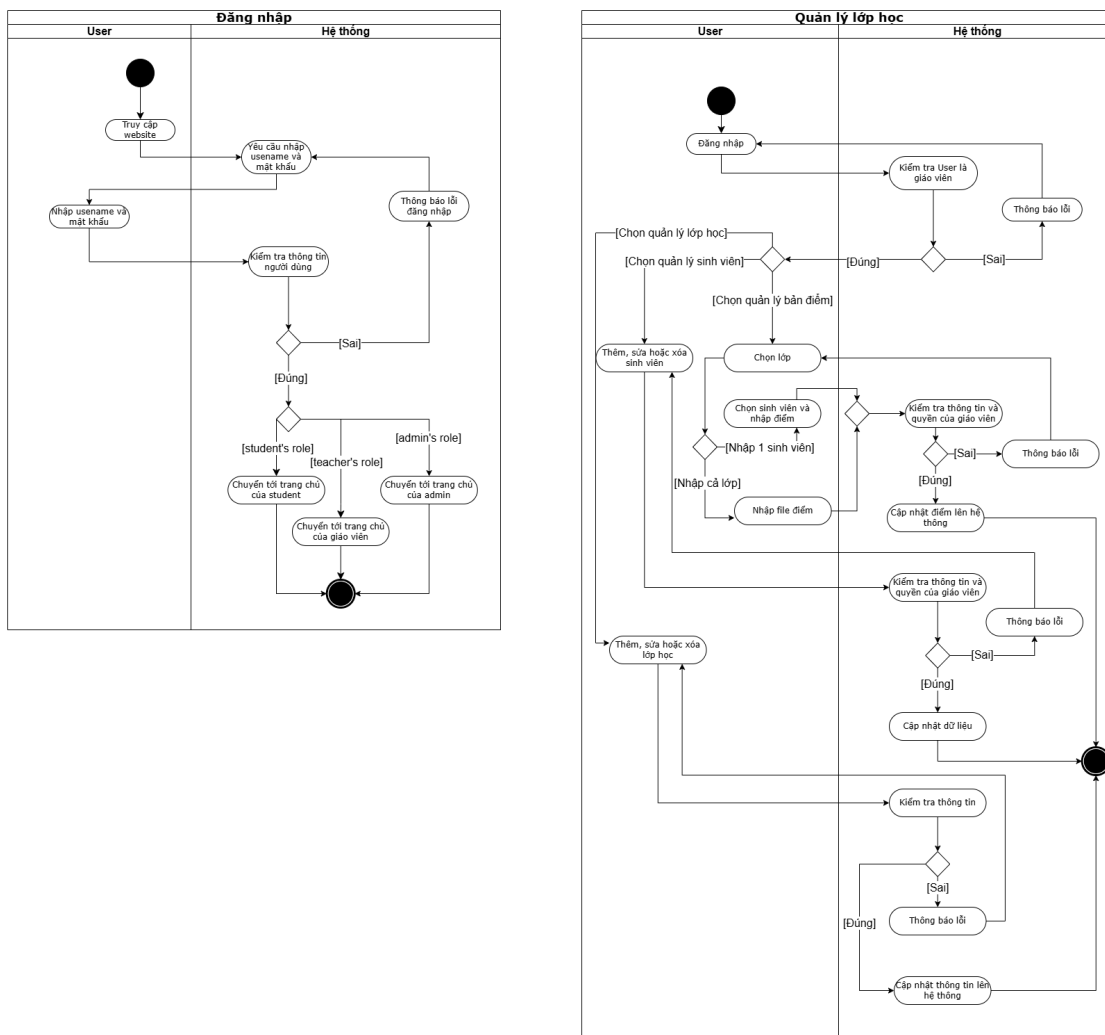
Hình 4: Usecase quản lí tài khoản

| Thành phần          | Mô tả   |
|---------------------|---|
| Tác nhân chính      | Quản trị viên.  |
| Tiền điều kiện      | Quản trị viên đăng nhập vào hệ thống thành công.  |
| Hậu điều kiện       | Danh sách tài khoản của giảng viên, sinh viên được cập nhật theo yêu cầu.   |
| Điều kiện kích hoạt | Quản trị viên muốn thực hiện các thao tác quản lí tài khoản.  |
| Luồng chính         | 1. Quản trị viên truy cập chức năng "Quản lí tài khoản".<br>2. Chọn một trong các hành động: thêm tài khoản giảng viên, chỉnh sửa thông tin tài khoản giảng viên và sinh viên, xóa tài khoản giảng viên và sinh viên.<br>3. Nhập thông tin cần thiết (nếu có).<br>4. Hệ thống cập nhật danh sách tài khoản. |
| Luồng phụ           | Quản trị viên có thể hủy thao tác và quay lại trang chính trước khi xác nhận gửi yêu cầu.   |
| Luồng ngoại lệ      | Nếu mã giảng viên hoặc mã sinh viên được nhập không hợp lệ thì thông báo lỗi.   |

### 4.3 Activity diagram

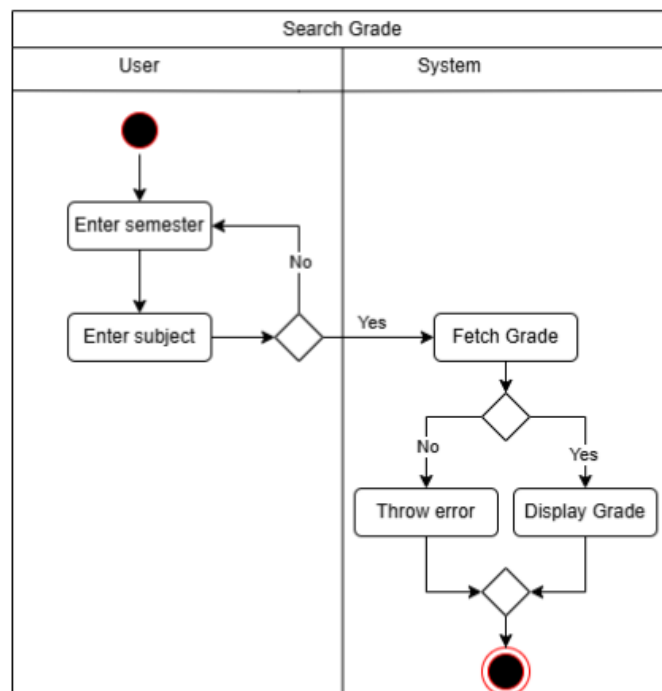
Để dễ dàng hơn trong việc xác định luồng điều khiển và thiết kế hệ thống, nhóm chúng em đã vẽ một số activity diagram như sau:

#### 4.3.1 Đăng nhập và quản lý lớp học



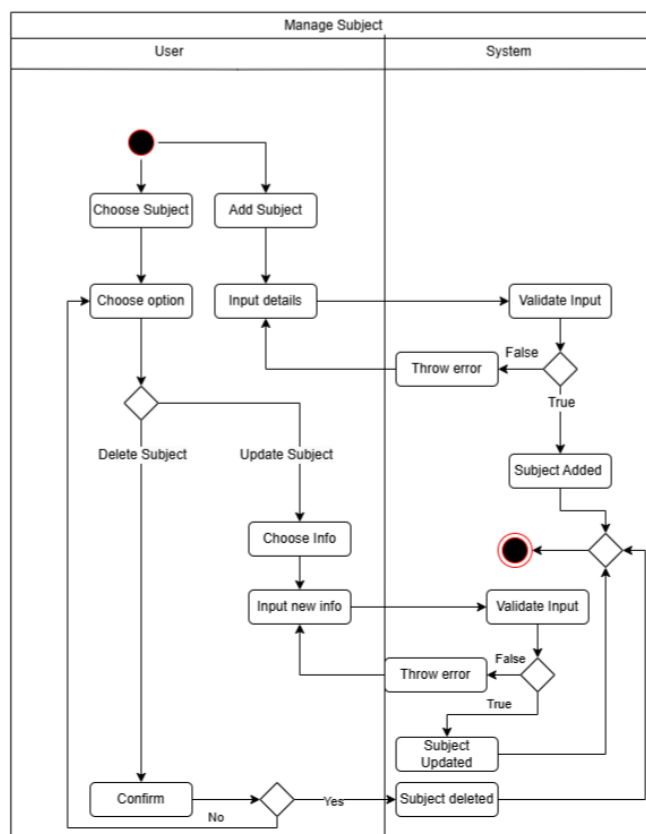
Hình 5: Sơ đồ hoạt động của hành động đăng nhập và quản lý lớp học

#### 4.3.2 Tra cứu điểm



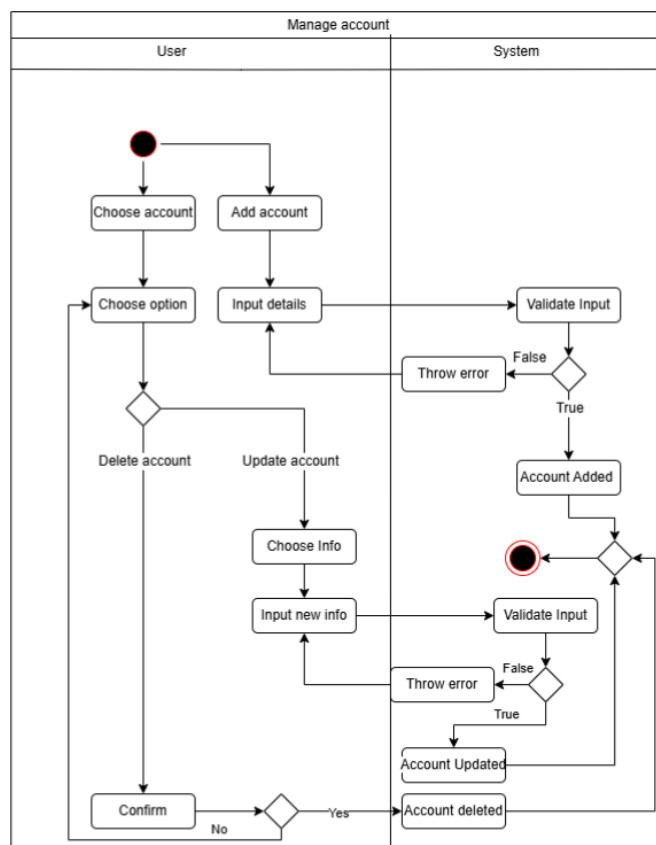
Hình 6: Sơ đồ hoạt động của hành động tra cứu điểm

### 4.3.3 Quản lý tài khoản



Hình 7: Sơ đồ hoạt động của hành động quản lý tài khoản

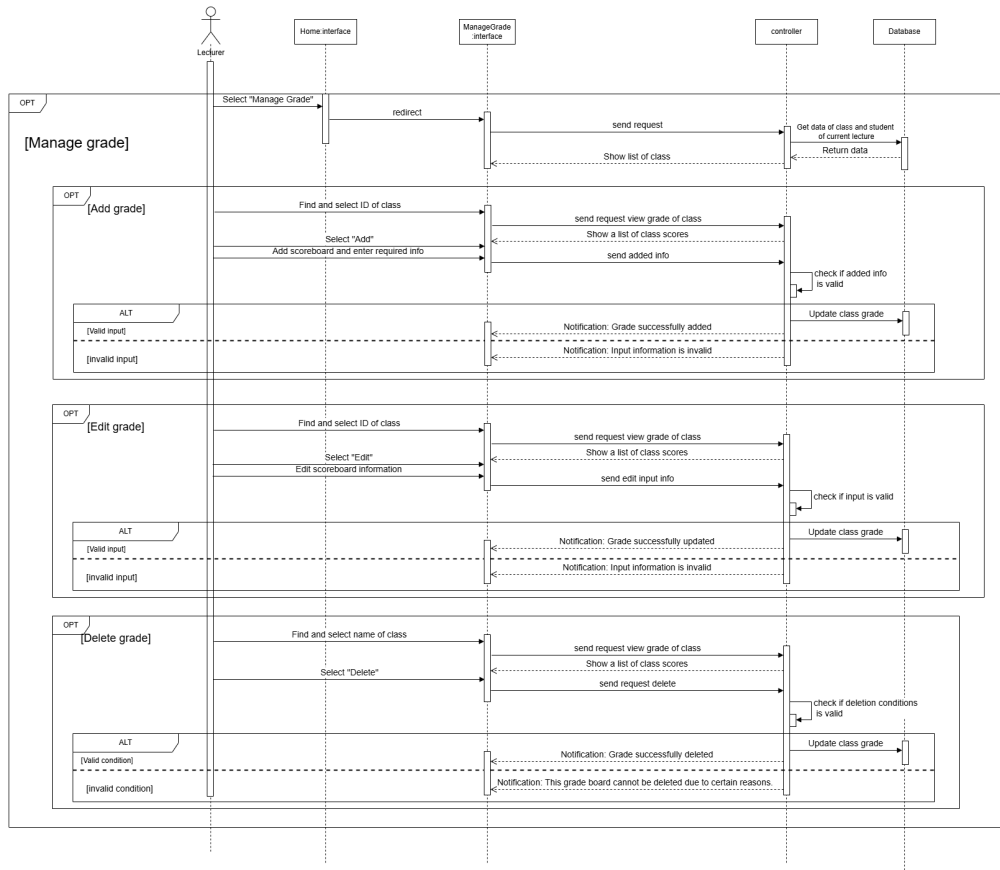
#### 4.3.4 Quản lý môn học



Hình 8: Sơ đồ hoạt động của hành động quản lý môn học

## 4.4 Sequence diagram

### 4.4.1 Manage grade (lecturer)

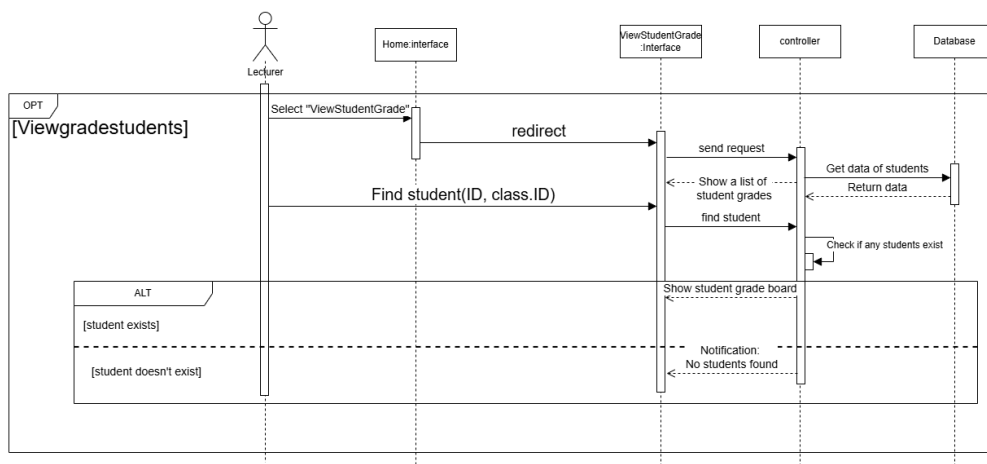


Hình 9: Sơ đồ tuần tự của hành động quản lý điểm

Mô tả: Hoạt động quản lý điểm của lớp học của giảng viên. các Object liên quan theo mô hình MVC. Các UI - Controller - Database.

- Lecturer: Actor chính tương tác với hệ thống để quản lý lớp học, quản lý điểm,...
- Home:interface : Giao diện khi đăng nhập vào của giảng viên có thể thực hiện các thao tác để chuyển đổi sang các chức năng khác.
- ManageGrade:interface : Giao diện để giáo viên quản lý điểm, thực hiện các thao tác tạo, thêm, xóa,... điểm.
- Controller: Chịu trách nhiệm xử lý các yêu cầu từ giao diện người dùng như kiểm tra input, các điều kiện cần để giảng viên có thể thêm, xóa, sửa,.. và để lấy dữ liệu hoặc nhận dữ liệu từ Database,...
- Database: lưu trữ dữ liệu của hệ thống.

#### 4.4.2 View student grades (lecturer)



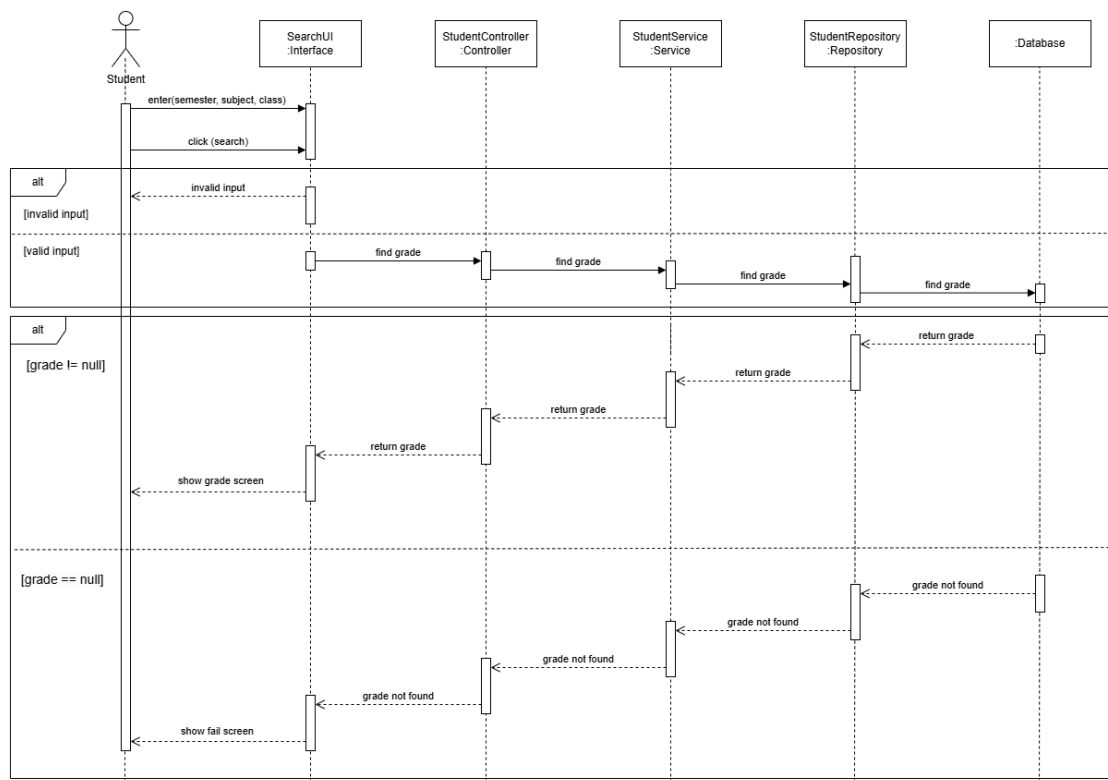
Hình 10: Sơ đồ tuần tự của hành động xem điểm sinh viên do giảng viên thực hiện

Mô tả: Các hoạt động bao gồm xem điểm sinh viên của giảng viên. các Object liên quan theo mô hình MVC. Các UI - Controller - Database.

- Lecturer: Actor chính tương tác với hệ thống để quản lý lớp học, quản lý điểm,...
- Home:interface : Giao diện khi đăng nhập vào của giảng viên có thể thực hiện các thao tác để chuyển đổi sang các chức năng khác.
- ViewStudentGrade:interface : Giao diện để giáo viên có thể xem được bảng điểm của từng sinh viên.
- Controller: Chịu trách nhiệm xử lý các yêu cầu từ giao diện người dùng như kiểm tra input, các điều kiện cần để giảng viên có thể thêm, xóa, sửa,.. và để lấy dữ liệu hoặc nhận dữ liệu từ Database,...
- Database: lưu trữ dữ liệu của hệ thống.



#### 4.4.3 Search grade

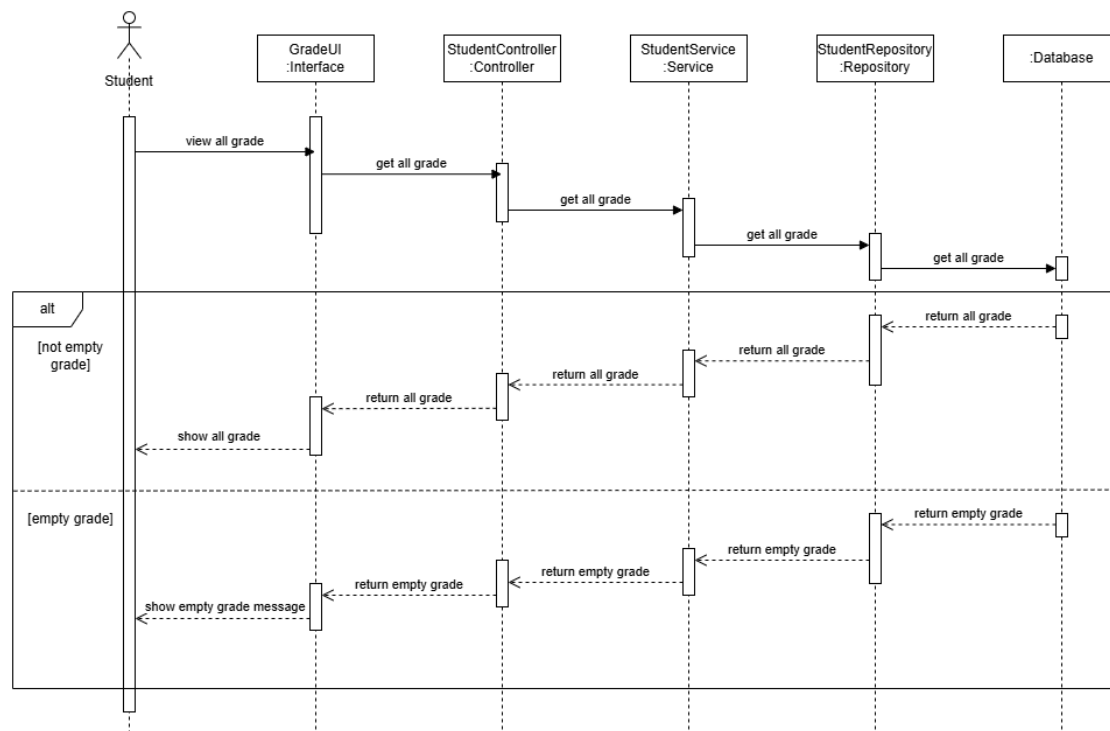


Hình 11: Sơ đồ tuần tự của hành động tra cứu điểm

Mô tả: Sơ đồ thể hiện hành động tra cứu điểm của sinh viên, các Object liên quan theo mô hình MVC. Các UI - Controller - Database.

- Student: Actor chính tương tác với hệ thống để tra cứu điểm.
- StudentUI:interface : Giao diện khi đăng nhập vào của sinh viên có thể thực hiện các thao tác để chuyển đổi sang các chức năng khác.
- StudentController:controller : Chịu trách nhiệm xử lý các yêu cầu từ phía người dùng là sinh viên bao gồm tra cứu điểm, xem điểm,...
- StudentService:service : Chịu trách nhiệm xử lý logic cho các yêu cầu từ phía người dùng là sinh viên.
- StudentRepository:repository : Chịu trách nhiệm tương tác với dữ liệu từ database.
- Database: lưu trữ dữ liệu của hệ thống.

#### 4.4.4 View grade

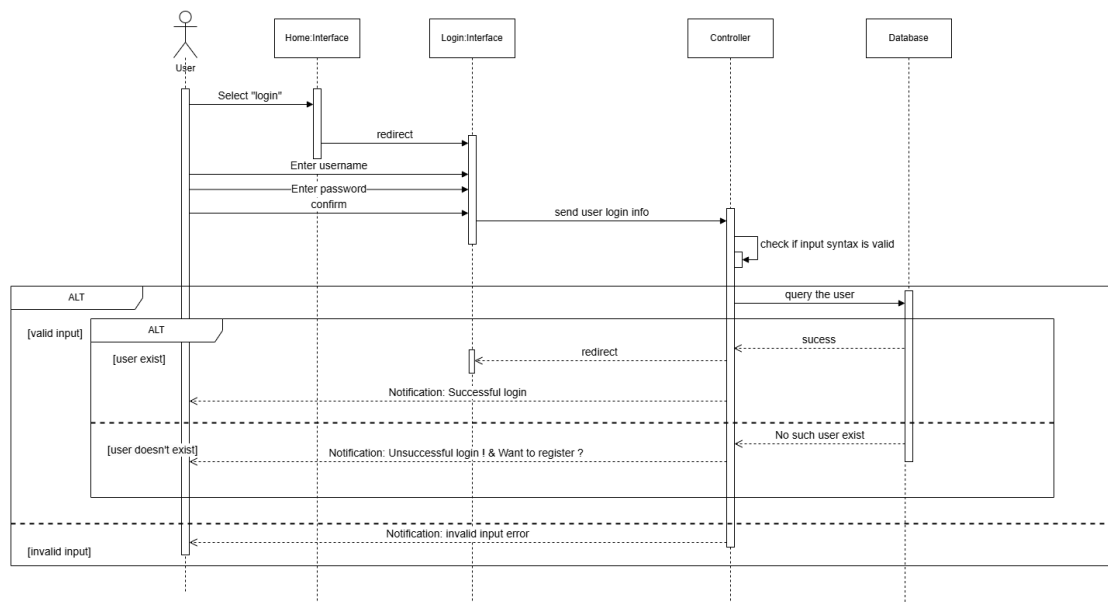


Hình 12: Sơ đồ tuần tự của hành động xem điểm

Mô tả: Sơ đồ thể hiện hành động xem tất cả điểm của sinh viên, các Object liên quan theo mô hình MVC. Các UI - Controller - Database.

- Student: Actor chính tương tác với hệ thống để xem điểm.
- StudentUI:interface : Giao diện khi đăng nhập vào của sinh viên có thể thực hiện các thao tác để chuyển đổi sang các chức năng khác.
- StudentController:controller : Chịu trách nhiệm xử lý các yêu cầu từ phía người dùng là sinh viên bao gồm tra cứu điểm, xem điểm,...
- StudentService:service : Chịu trách nhiệm xử lý logic cho các yêu cầu từ phía người dùng là sinh viên.
- StudentRepository:repository : Chịu trách nhiệm tương tác với dữ liệu từ database.
- Database: lưu trữ dữ liệu của hệ thống.

#### 4.4.5 Login

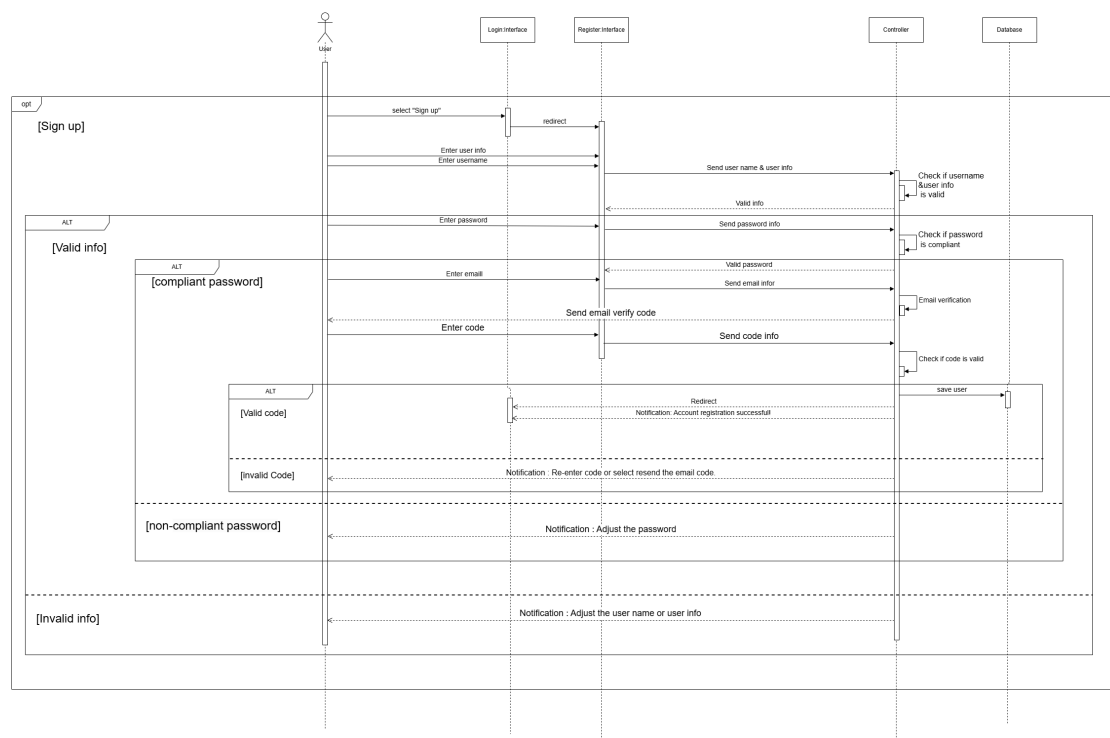


Hình 13: Sơ đồ tuần tự của hành động đăng nhập

Mô tả: Sơ đồ thể hiện hành động đăng nhập của người dùng, các Object liên quan theo mô hình MVC. Các UI - Controller - Database.

- User: Actor chính tương tác với hệ thống để tra cứu điểm.
- Home:interface : Giao diện mặc định của hệ thống khi chưa đăng nhập.
- Login:interface : Giao diện đăng nhập người dùng.
- Controller: Chịu trách nhiệm xử lý các yêu cầu từ phía người dùng.
- Database: lưu trữ dữ liệu của hệ thống.

#### 4.4.6 Register

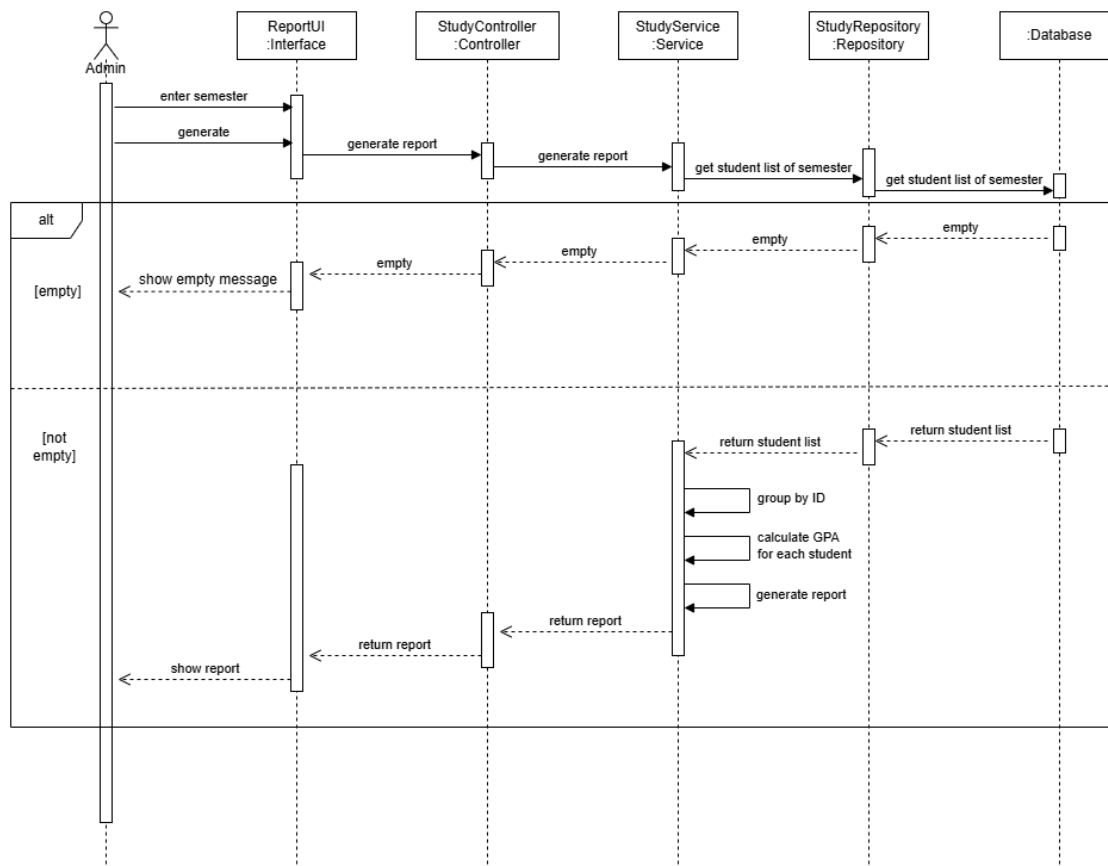


Hình 14: Sơ đồ tuần tự của hành động đăng ký tài khoản

Mô tả: Sơ đồ thể hiện hành động đăng ký của người dùng, các Object liên quan theo mô hình MVC. Các UI - Controller - Database.

- User: Actor chính tương tác với hệ thống để tra cứu điểm.
- Login:interface : Giao diện đăng nhập người dùng.
- Register:interface : Giao diện đăng ký người dùng.
- Controller: Chịu trách nhiệm xử lý các yêu cầu từ phía người dùng.
- Database: lưu trữ dữ liệu của hệ thống.

#### 4.4.7 Generate report

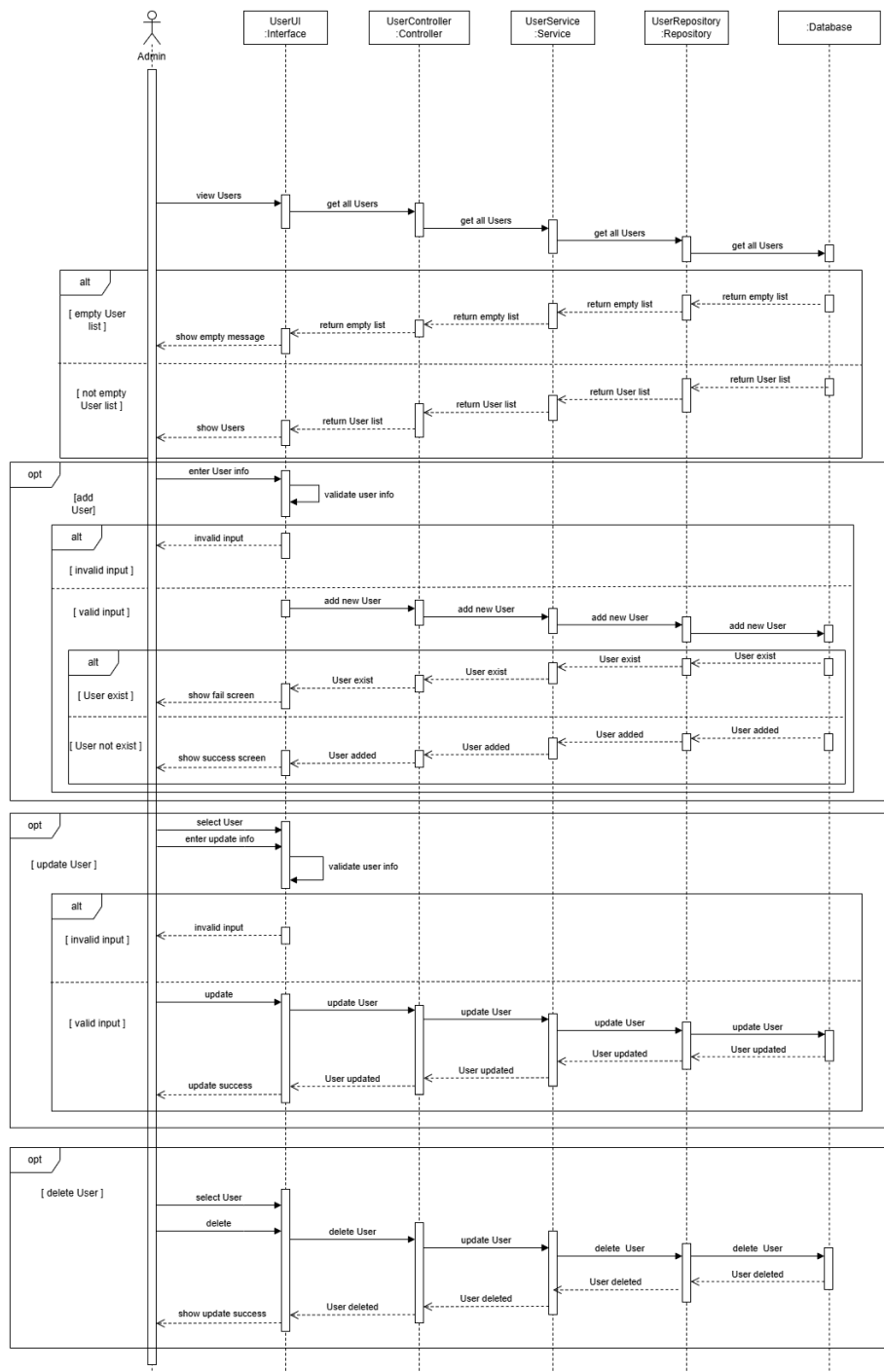


Hình 15: Sơ đồ tuần tự của hành động tạo báo cáo

Mô tả: Sơ đồ thể hiện hành động tạo báo cáo của admin, các Object liên quan theo mô hình MVC. Các UI - Controller - Database.

- Admin: Actor chính tương tác với hệ thống để tra cứu điểm.
- ReportUI:interface : Giao diện tạo báo cáo của admin.
- StudyController:controller : Chịu trách nhiệm xử lý các yêu cầu về thông tin học tập của sinh viên.
- StudyService:service : Chịu trách nhiệm xử lý logic cho các yêu cầu về thông tin học tập của sinh viên.
- StudyRepository:repository : Chịu trách nhiệm tương tác với dữ liệu từ database.
- Database: lưu trữ dữ liệu của hệ thống.

#### 4.4.8 Manage account



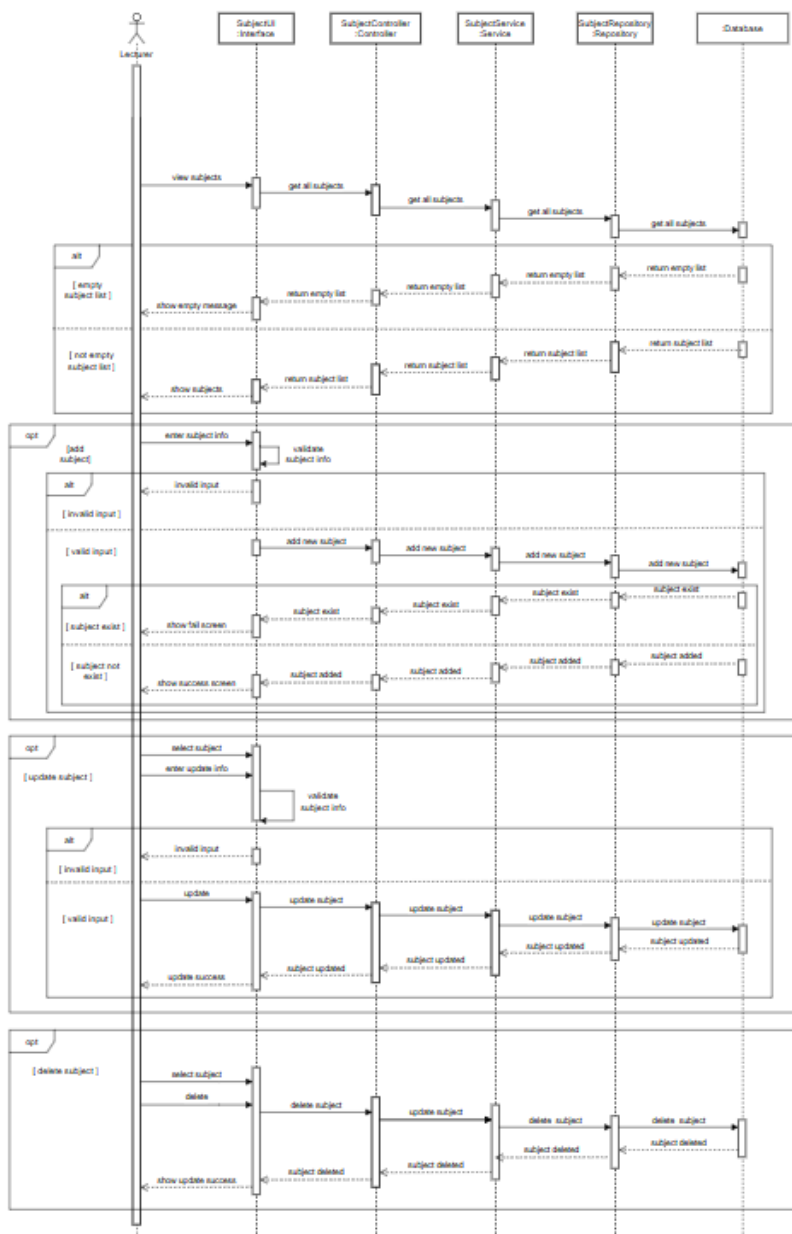
Hình 16: Sơ đồ tuần tự của hành động quản lí tài khoản



Mô tả: Sơ đồ thể hiện hành động quản lý tài khoản của admin, các Object liên quan theo mô hình MVC. Các UI - Controller - Database.

- Admin: Actor chính tương tác với hệ thống để quản lý tài khoản người dùng.
- UserUI:interface : Giao diện quản lý tài khoản người dùng của admin.
- UserController:controller : Chịu trách nhiệm xử lý các yêu cầu về thông tin người dùng.
- UserService:service : Chịu trách nhiệm xử lý logic cho các yêu cầu về thông tin người dùng.
- UserRepository:repository : Chịu trách nhiệm tương tác với dữ liệu từ database.
- Database: lưu trữ dữ liệu của hệ thống.

#### 4.4.9 Manage subject



Hình 17: Sơ đồ tuần tự của hành động quản lý môn học

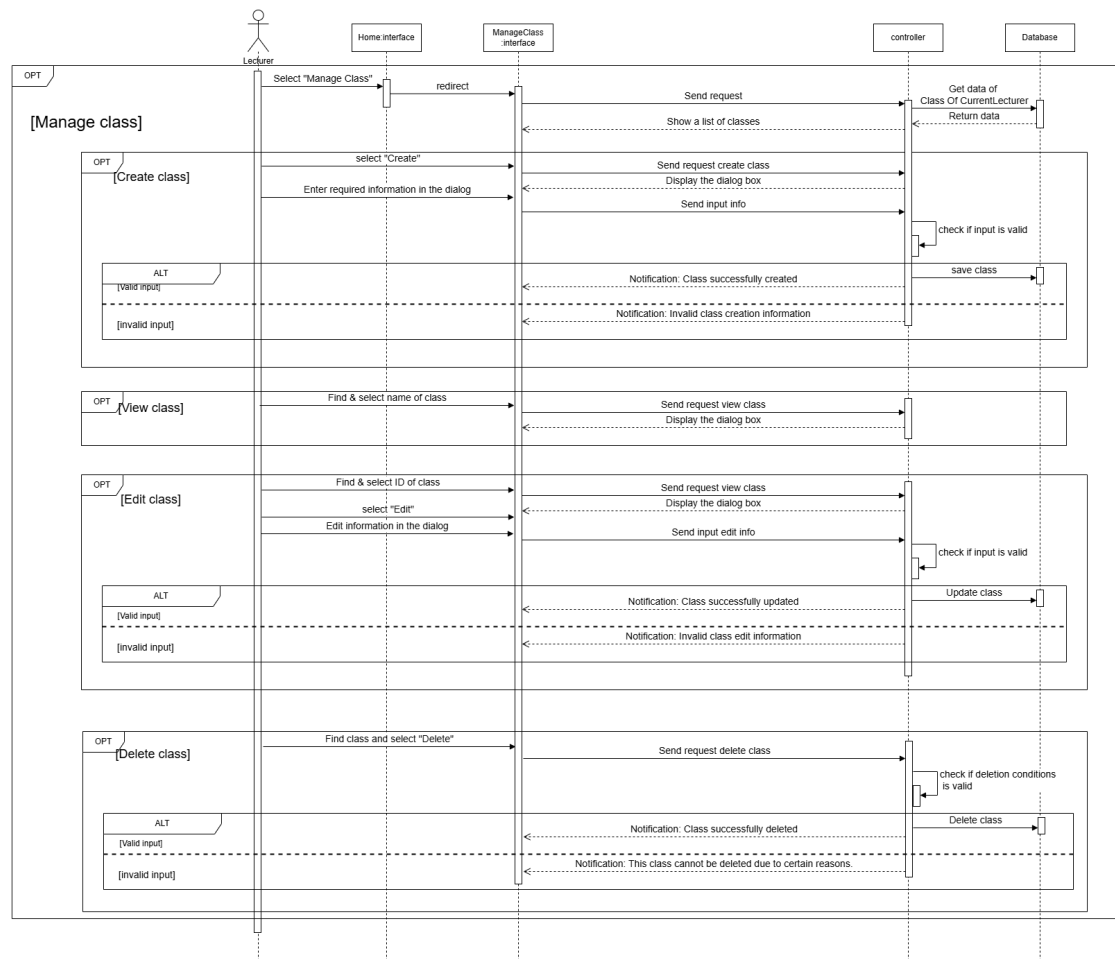
Mô tả: Sơ đồ thể hiện hành động quản lý môn học của giảng viên, các Object liên quan theo mô hình MVC. Các UI - Controller - Database.

- Lecturer: Actor chính tương tác với hệ thống để quản lý môn học.



- SubjectUI:interface : Giao diện quản lý môn học của giảng viên.
- SubjectController:controller : Chịu trách nhiệm xử lý các yêu cầu về thông tin môn học của giảng viên.
- SubjectService:service : Chịu trách nhiệm xử lý logic cho các yêu cầu về thông tin môn học của giảng viên.
- SubjectRepository:repository : Chịu trách nhiệm tương tác với dữ liệu từ database.
- Database: lưu trữ dữ liệu của hệ thống.

## 4.5 Manage class

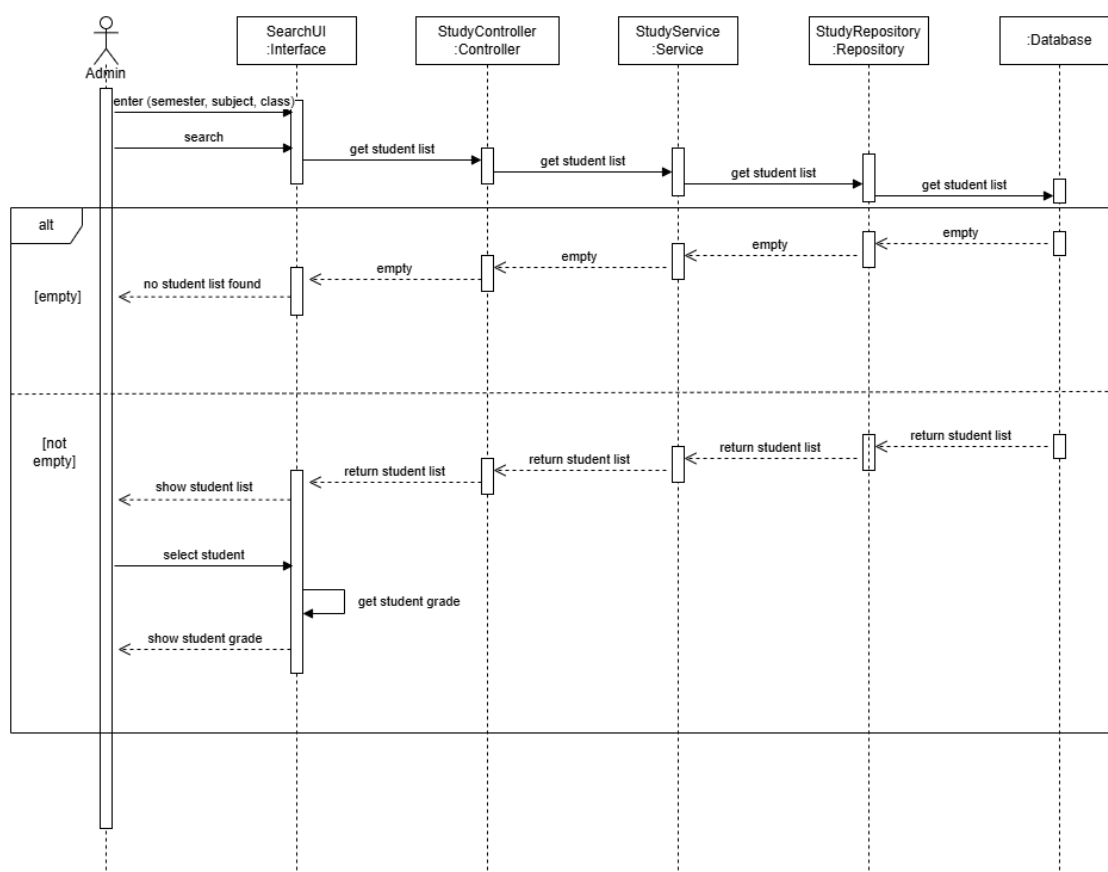


Hình 18: Sơ đồ tuần tự của hành động quản lý lớp học

Mô tả: Sơ đồ thể hiện hành động quản lý lớp học của giảng viên, các Object liên quan theo mô hình MVC. Các UI - Controller - Database.

- Lecturer: Actor chính tương tác với hệ thống để quản lý lớp học, quản lý điểm,...
- Home:interface : Giao diện khi đăng nhập vào của giảng viên có thể thực hiện các thao tác để chuyển đổi sang các chức năng khác.
- ManageClass:interface : Giao diện để giáo viên quản lý lớp, thực hiện các thao tác tạo, thêm, xóa,... lớp.
- Controller: Chịu trách nhiệm xử lý các yêu cầu từ giao diện người dùng như kiểm tra input, các điều kiện cần để giảng viên có thể thêm, xóa, sửa,.. và để lấy dữ liệu hoặc nhận dữ liệu từ Database,...
- Database: lưu trữ dữ liệu của hệ thống.

#### 4.5.1 View student grade



Hình 19: Sơ đồ tuần tự của hành động xem điểm sinh viên của admin

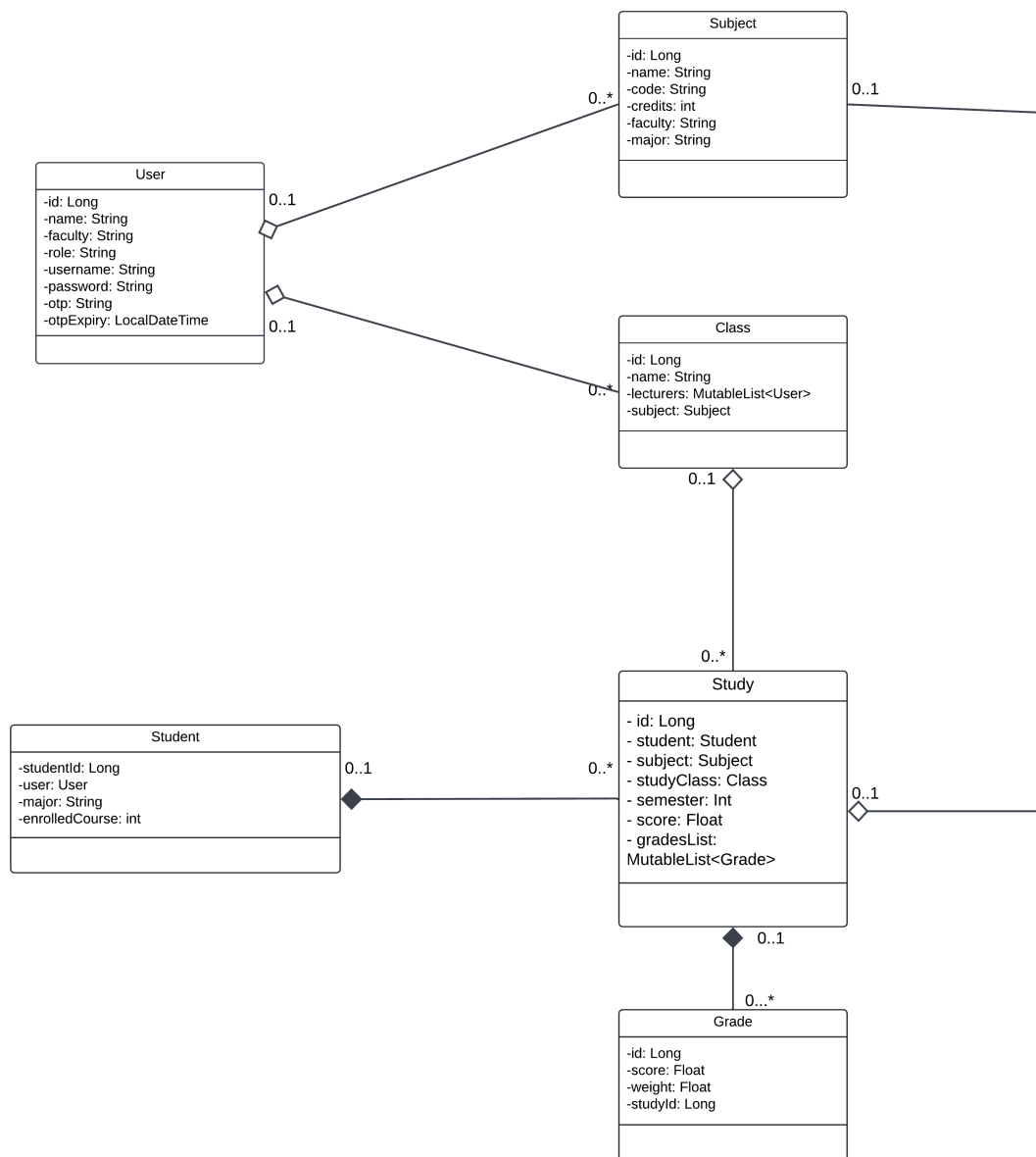
Mô tả: Sơ đồ thể hiện hành động quản lý môn học của giảng viên, các Object liên quan theo mô hình MVC. Các UI - Controller - Database.

- Admin: Actor chính tương tác với hệ thống để quản lý môn học.



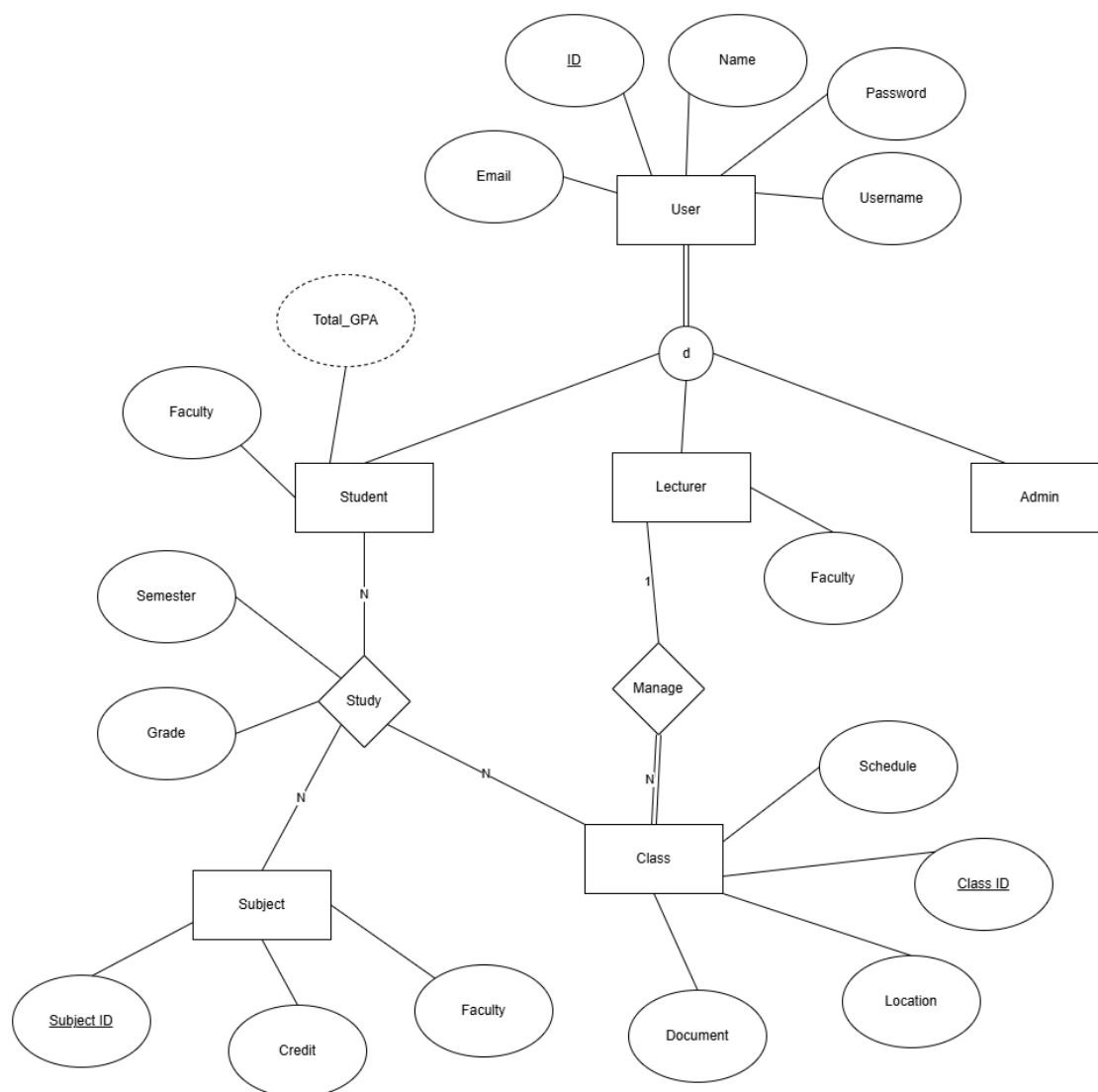
- SearchUI:interface: Giao diện tra cứu điểm của admin.
- StudyController:controller: Chịu trách nhiệm xử lý các yêu cầu về thông tin học tập của giảng viên.
- StudyService:service: Chịu trách nhiệm xử lý logic cho các yêu cầu về thông tin học tập của giảng viên.
- StudyRepository:repository: Chịu trách nhiệm tương tác với dữ liệu từ database.
- Database: lưu trữ dữ liệu của hệ thống.

## 4.6 Class diagram



Hình 20: Sơ đồ các lớp của hệ thống

## 4.7 Entity-Relationship diagram



Hình 21: Sơ đồ quan hệ thực thể của hệ thống