

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**

**BÁO CÁO TỔNG KẾT**

**ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN**  
**THAM GIA XÉT GIẢI THƯỞNG..**

**XÂY DỰNG HỆ THỐNG QUẢN LÝ CỦA LÃNH ĐẠO**  
**KHOA CNTT**

**UTEHY.S.2022.85**

Thuộc nhóm ngành học: Công nghệ thông tin

**Hưng Yên, 05/2022**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**

**BÁO CÁO TỔNG KẾT**

**ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN**  
**THAM GIA XÉT GIẢI THƯỞNG...**

**XÂY DỰNG HỆ THỐNG QUẢN LÝ CỦA LÃNH ĐẠO**  
**KHOA CNTT**  
**UTEHY.S.2022.85**

Thuộc nhóm ngành khoa học: Công nghệ thông tin

Sinh viên thực hiện: Mai Hoàng Thái Bảo Nam, Nữ: Nam

Dân tộc: Kinh

Lớp, Khoa: Công nghệ thông tin Năm thứ: 3 /Số năm đào tạo: 4

Ngành học: Kỹ thuật phần mềm

Người hướng dẫn: ThS. Trịnh Thị Nhị

**Hưng Yên, 05/2022**

# MỤC LỤC

MỤC CÁC BẢNG.....	4
DANH MỤC HÌNH VẼ.....	5
DANH MỤC CÁC TỪ VIẾT TẮT.....	7
THÔNG TIN KẾT QUẢ NGHIÊN CỨU CỦA ĐỀ TÀI.....	8
THÔNG TIN VỀ SINH VIÊN.....	10
CHỊU TRÁCH NHIỆM CHÍNH THỰC HIỆN ĐỀ TÀI.....	10
LỜI NÓI ĐẦU.....	11
CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI.....	12
1.1. Lý do chọn đề tài	12
1.2. Mục tiêu của đề tài	12
1.2.1 Mục tiêu tổng quát	12
1.2.2 Mục tiêu cụ thể	12
1.3. Giới hạn và phạm vi của đề tài	12
1.3.1 Đối tượng nghiên cứu	12
1.3.2 Phạm vi nghiên cứu	13
1.4. Nội dung thực hiện	13
1.5. Phương pháp tiếp cận	13
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	15
2.1. Quy trình phát triển phần mềm	15
2.1.1. Giải pháp, yêu cầu	15
2.1.2. Thiết kế	15
2.1.3. Lập trình	15
2.1.4. Kiểm thử	15
2.1.5. Triển khai	15
2.2 Thiết kế phần mềm	16
2.3 Lập trình phần mềm	17
2.4 Kiểm thử phần mềm	17
2.5 Triển khai và bảo trì phần mềm	18
2.6 Phương pháp phân tích thiết kế hướng đối tượng	18
2.7. Giới thiệu các xu hướng lập trình web mới	22
2.8 Tổng quan về Angular	24
2.9    Lập trình Web API NET CORE	25
CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG.....	27
3.1    Phát biểu bài toán	27

3.2	Đặc tả yêu cầu phần mềm	31
3.2.1	Các yêu cầu chức năng	31
3.2.1.1	Bảng chức năng phân hệ trang quản trị	31
3.2.1.2	Biểu đồ usecase	33
3.2.1.3	Luồng sự kiện	36
3.2.1.4	Biểu đồ lớp thực thể	38
3.2.2	Các yêu cầu phi chức năng	40
3.2	Thiết kế hệ thống	41
3.2.1	Thiết kế cơ sở dữ liệu	41
3.2.2	Thiết kế lớp đối tượng	49
3.3	Thiết kế giao diện	65
KẾT LUẬN.....		68
TÀI LIỆU THAM KHẢO.....		69

## MỤC CÁC BẢNG

Bảng 1 Các yêu cầu chức năng của quản lý chương trình đào tạo	31
Bảng 2 Các yêu cầu chức năng của quản lý danh sách giáo viên chủ nhiệm	32
Bảng 3 Quản lý hướng dẫn đồ án	33
Bảng 4 Các thuộc tính của bảng khung chương trình đào tạo	38
Bảng 5 Các thuộc tính của bảng học phần	39
Bảng 6 Các thuộc tính của bảng bộ môn	39
Bảng 7 Mô tả cấu trúc bảng cơ sở dữ liệu giảng viên (TBL_teacher_ref)	41
Bảng 8 Mô tả cấu trúc bảng cơ sở dữ liệu bộ môn (TBL_department_ref)	42
Bảng 9 Mô tả cấu trúc bảng cơ sở dữ liệu ngành đào tạo (TBL_major_ref)	43
Bảng 10 Mô tả cấu trúc bảng cơ sở dữ liệu chuyên ngành (TBL_specialization_ref)	43
Bảng 11 Mô tả cấu trúc bảng cơ sở dữ liệu hệ đào tạo (TNL_degree_ref)	44
Bảng 12 Mô tả cấu trúc bảng cơ sở dữ liệu lớp (TBL_class_ref)	45
Bảng 13 Mô tả cấu trúc bảng cơ sở dữ liệu khung CTĐT (TBL_Leader_Curriculum_Framework)	45
Bảng 14 Mô tả cấu trúc bảng cơ sở dữ liệu phân công số lượng đồ án (TBL_Leader_Teacher_Project)	46
Bảng 15 Mô tả cấu trúc bảng cơ sở dữ liệu danh mục chương trình đào tạo (TBL_Leader_Education_Curriculum)	47
Bảng 16 Mô tả cấu trúc bảng cơ sở dữ liệu học phần	48

# DANH MỤC HÌNH VẼ

Hình 1: Biểu đồ Use Case tổng quát	33
Hình 2: Biểu đồ phân rã Use Case chương trình đào tạo	33
Hình 3 Use case Quản lý danh sách giáo viên chủ nhiệm	34
Hình 4 Use case quản lý hướng dẫn đồ án	34
Hình 5: Biểu đồ lớp thực thể	39
Hình 6: Mô hình cơ sở dữ liệu quan hệ	40
Hình 7: Biểu đồ lớp VOPC thêm học phần	48
Hình 8: Biểu đồ lớp VOPC sửa học phần	49
Hình 9: Biểu đồ lớp VOPC xóa học phần	49
Hình 10: Biểu đồ lớp VOPC tìm kiếm học phần	50
Hình 11: Biểu đồ lớp VOPC thêm danh mục chương trình đào tạo	51
Hình 12: Biểu đồ lớp VOPC sửa danh mục chương trình đào tạo	51
Hình 13: Biểu đồ lớp VOPC xóa danh mục chương trình đào tạo	51
Hình 14: Biểu đồ lớp VOPC xem danh mục chương trình đào tạo	52
Hình 15: Biểu đồ lớp VOPC tìm kiếm danh mục chương trình đào tạo	53
Hình 16: Biểu đồ lớp VOPC in danh mục chương trình đào tạo	54
Hình 17: Biểu đồ lớp VOPC kết xuất danh sách danh mục chương trình đào tạo	55
Hình 18: Biểu đồ lớp VOPC thêm khung chương trình đào tạo	56
Hình 19: Biểu đồ lớp VOPC sửa khung chương trình đào tạo	56
Hình 20: Biểu đồ lớp VOPC xóa khung chương trình đào tạo	57
Hình 21: Biểu đồ lớp VOPC xem khung chương trình đào tạo	57
Hình 22: Biểu đồ lớp VOPC tìm kiếm khung chương trình đào tạo	58
Hình 23: Biểu đồ lớp VOPC in khung chương trình đào tạo	59
Hình 24: Biểu đồ lớp VOPC kết xuất khung chương trình đào tạo	59
Hình 25: Biểu đồ tuần tự thêm khung chương trình đào tạo	60
Hình 26: Biểu đồ tuần tự sửa khung chương trình đào tạo	60
Hình 27: Biểu đồ tuần tự sửa danh mục chương trình đào tạo	61
Hình 28 Biểu đồ tuần tự sửa danh mục hướng dẫn đồ án	61
Hình 29 Biểu đồ tuần tự sửa danh mục giáo viên chủ nhiệm	62
Hình 30: Biểu đồ lớp chi tiết thêm học phần	62
Hình 31: Biểu đồ lớp chi tiết thêm danh mục chương trình đào tạo	63
Hình 32: Biểu đồ lớp chi tiết thêm khung chương trình đào tạo	64
Hình 33 Giao diện quản lý chương trình đào tạo	64
Hình 34 Giao diện quản lý thêm học phần	65
Hình 35 Giao diện quản lý đồ án	65
Hình 36 Giao diện phân công	66
Hình 37 Giao diện quản lý giảng viên	66

DANH MỤC CÁC TỪ VIẾT TẮT

STT	Từ viết tắt	Cụm từ tiếng anh	Diễn giải
1	HTML	Hypertext Markup Language	Ngôn ngữ đánh dấu siêu văn bản
2	CSS	Cascading Style Sheet	Ngôn ngữ định dạng văn bản

## **THÔNG TIN KẾT QUẢ NGHIÊN CỨU CỦA ĐỀ TÀI**

### **1. Thông tin chung:**

- Tên đề tài: Nghiên cứu và xây dựng hệ thống quản lý của lãnh đạo Khoa CNTT
- Sinh viên thực hiện: Mai Hoàng Thái Bảo
- Lớp: 101191A      Khoa: Công nghệ thông tin      Năm thứ: 3      Số năm đào tạo: 4
- Sinh viên thực hiện: Lê Đình Minh
- Lớp: 101191A      Khoa: Công nghệ thông tin      Năm thứ: 3      Số năm đào tạo: 4
- Sinh viên thực hiện: Thái Bình An
- Lớp: 101191A      Khoa: Công nghệ thông tin      Năm thứ: 3      Số năm đào tạo: 4
- Người hướng dẫn: ThS Trịnh Thị Nhị

### **2. Mục tiêu đề tài:**

Nghiên cứu và thực hiện hệ thống quản lý của lãnh đạo khoa CNTT gồm các chức năng sau:

- Nghiệp vụ về quản lý chương trình đào tạo.
- Nghiệp vụ về quản lý danh sách giáo viên chủ nhiệm.
- Nghiệp vụ về giám sát quá trình hướng dẫn đồ án.

### **3. Tính mới và sáng tạo:**

- Tiến hành số hóa nhằm đẩy nhanh cải thiện, đẩy nhanh các công việc của khoa CNTT.
- Áp dụng công nghệ nhằm thống nhất các quy trình trong thực hiện việc của khoa.

### **4. Kết quả nghiên cứu:**

- Phân tích nghiệp vụ của hệ thống quản lý của lãnh đạo khoa CNTT.
- Hoàn thiện hệ thống quản lý lãnh đạo khoa CNTT.



**5. Đóng góp về mặt kinh tế - xã hội, giáo dục và đào tạo, an ninh, quốc phòng và khả năng áp dụng của đề tài:**

- Số hóa nghiệp vụ quản lý cho khoa CNTT.

Ngày 17 tháng 4 năm 2022

**Sinh viên chịu trách nhiệm chính**  
**thực hiện đề tài**  
*(ký, họ và tên)*

Mai Hoàng Thái Bảo

**Nhận xét của người hướng dẫn về những đóng góp khoa học của sinh viên thực hiện đề tài** *(phần này do người hướng dẫn ghi):*

**Xác nhận của trường đại học**  
*(ký tên và đóng dấu)*

Ngày 17 tháng 4 năm 2022

**Người hướng dẫn**  
*(ký, họ và tên)*

ThS Trịnh Thị Nhị

**THÔNG TIN VỀ SINH VIÊN**

**CHỊU TRÁCH NHIỆM CHÍNH THỰC HIỆN ĐỀ TÀI**

**I. SƠ LƯỢC VỀ SINH VIÊN:**

Họ và tên: Mai Hoàng Thái Bảo

Sinh ngày: 20 tháng 06 năm 2001

Nơi sinh: Tử Đông, Lý Thường Kiệt, Yên Mỹ, Hưng Yên

Lớp: 101191A

Khóa: 17

Khoa: Công nghệ thông tin

Địa chỉ liên hệ: Tử Đông, Lý Thường Kiệt, Yên Mỹ, Hưng Yên

Điện thoại: 0986501713

Email: maihoangthaibao01@gmail.com



**II. QUÁ TRÌNH HỌC TẬP** (kê khai thành tích của sinh viên từ năm thứ 1 đến năm đang học):

**\* Năm thứ 1:**

Ngành học: Công nghệ phần mềm

Khoa: Công nghệ thông tin

Kết quả xếp loại học tập: Giỏi

Sơ lược thành tích: Điểm TBC : 8.73

**\* Năm thứ 2:**

Ngành học: Công nghệ phần mềm

Khoa: Công nghệ thông tin

Kết quả xếp loại học tập: Giỏi

Sơ lược thành tích: Điểm TBC : 8.88

Ngày 17 tháng 4 năm 2022

**Xác nhận của trường đại học**

**Sinh viên chịu trách nhiệm chính**

*(ký tên và đóng dấu)*

**thực hiện đề tài**

*(ký, họ và tên)*

Mai Hoàng Thái Bảo

## **LỜI NÓI ĐẦU**

Hiện nay, số hóa đang là một xu thế sống còn không chỉ đối với các doanh nghiệp kinh doanh mà còn là đối với các cơ quan nhà nước. Việc số hóa không chỉ giúp cho việc xử lý nghiệp vụ được thuận tiện hơn mà còn giúp cho việc xử lý dữ liệu trở nên hiệu quả hơn qua đó nâng cao năng suất lao động. Do đó trong phạm vi đề tài này em chọn đề tài “Xây dựng hệ thống quản lý cho lãnh đạo Khoa Công nghệ thông tin”.

## **CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI**

### **1.1. Lý do chọn đề tài**

Trong xu hướng phát triển của thời đại công nghệ 4.0 việc ứng dụng công nghệ thông tin vào quy trình làm việc của mình để nâng cao hiệu quả hoạt động ngày càng cao. Do đó yêu cầu xây dựng phần mềm hỗ trợ cho lĩnh vực giáo dục là điều cần thiết. Trong khi tin học được chú trọng trong tất cả các khâu thì việc để có những phần mềm tốt nhất giúp cho nhà trường có thể quản lý công việc đạt hiệu quả cao nhất, giúp tiết kiệm thời gian, chi phí và đáp ứng được những yêu cầu cần thiết là vấn đề mà nhà trường quan tâm. Xuất phát từ thực tế đó, Khoa Công nghệ thông tin của Trường Đại học Sư Phạm Kỹ Thuật Hưng Yên đã quyết định xây dựng website quản lý các hoạt động của nhà trường nhằm đáp ứng những yêu cầu trên.

### **1.2. Mục tiêu của đề tài**

#### ***1.2.1 Mục tiêu tổng quát***

Tạo ra một website quản lý các hoạt động của nhà trường với đầy đủ các chức năng, nghiệp vụ đối với phân hệ quản trị nội dung thiết kế theo mô hình API sử dụng Angular.

#### ***1.2.2 Mục tiêu cụ thể***

Trên cơ sở mục đích nghiên cứu, website cần đạt được những mục tiêu sau:

- Quản lý danh sách giáo viên chủ nhiệm.
- Quản lý chương trình đào tạo.
- Quản lý hướng dẫn đồ án.

### **1.3. Giới hạn và phạm vi của đề tài**

#### ***1.3.1 Đối tượng nghiên cứu***

Tìm hiểu, nghiên cứu, lập trình website quản lý các hoạt động của nhà trường tích hợp cùng với vai trò của lãnh đạo trường, giáo viên, sinh viên.

### **1.3.2 Phạm vi nghiên cứu**

- Phạm vi không gian: tại Trường Đại học Sư phạm Kỹ thuật Hưng Yên
- Phạm vi thời gian: Các thông tin số liệu được thu thập trong khoảng thời gian từ 01/09/2021 đến 18/09/2021.
- Ý nghĩa khoa học và thực tiễn của đề tài:
  - Ý nghĩa khoa học: Giúp mọi người hiểu hơn về nghiệp vụ và công việc cần phải làm của một ứng dụng Web quản lý.
  - Ý nghĩa thực tiễn: Giúp thúc đẩy quá trình chuyển đổi số trong lĩnh vực giáo dục diễn ra nhanh và hiệu quả hơn.

### **1.4. Nội dung thực hiện**

Trong phạm vi của đề tài này, nội dung thực hiện theo năm chương cụ thể như sau:

- Chương 1: Trình bày tổng quan về đề tài. Chương này nêu ra lý do chọn đề tài, mục tiêu của đề tài, giới hạn và phạm vi của đề tài, nội dung thực hiện và phương pháp tiếp cận.
- Chương 2: Trình bày về cơ sở lý thuyết. Chương này trình bày những kiến thức cơ sở về quy trình phát triển phần mềm, phương pháp phân tích thiết kế hướng đối tượng, các xu hướng lập trình web mới, tổng quan về Angular và lập trình Web API Net Core.
- Chương 3: Trình bày những kiến thức cơ sở về phân tích và thiết kế hệ thống. Chương này sẽ đặc tả các yêu cầu phần mềm và thiết kế hệ thống.
  - Phần đặc tả yêu cầu phần mềm sẽ đặc tả các yêu cầu chức năng, xác định các thực thể và các yêu cầu phi chức năng.
  - Phần thiết kế hệ thống sẽ thiết kế lớp đối tượng, thiết kế cơ sở dữ liệu và thiết kế giao diện cho trang web.

### **1.5. Phương pháp tiếp cận**

- Đối với lý thuyết:
  - Tìm hiểu về lập trình website theo mô hình API.

- Tìm hiểu về Angular và cách xây dựng giao diện bằng Angular.
- Tìm hiểu về kỹ thuật lập trình, cách thức hoạt động và phương pháp hướng đối tượng trong TypeScript, C#.
- Tìm hiểu cách thức hoạt động của Client – Server.
- Tìm hiểu cách lưu dữ liệu của hệ quản trị cơ sở dữ liệu Microsoft SQL Server.
- Đối với lập trình:
  - Thiết kế theo mô hình API sử dụng ADO Dot Net, C# và Angular.
  - Sử dụng hệ quản trị cơ sở dữ liệu Microsoft SQL Server.
  - Xây dựng trang web theo đúng mô tả của hệ thống dựa trên các yêu cầu thực tế.

## **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

### **2.1. Quy trình phát triển phần mềm**

- Quy trình phát triển phần mềm là một cấu trúc bao gồm tập hợp các thao tác và các kết quả tương quan sử dụng trong việc phát triển để cung cấp ra một sản phẩm phần mềm. Nhìn chung, một quy trình phát triển phần mềm bao gồm các giai đoạn như sau:

#### **2.1.1. Giải pháp, yêu cầu**

- Thu thập yêu cầu của khách hàng
- Đầu ra của giai đoạn này là Tài liệu đặc tả yêu cầu

#### **2.1.2. Thiết kế**

- Thiết kế giao diện
- Thiết kế dữ liệu, thuật toán

#### **2.1.3. Lập trình**

- Lập trình viên thực hiện lập trình dựa trên tài liệu Giải pháp và Thiết kế đã được phê duyệt.
- Kết quả: Source code.

#### **2.1.4. Kiểm thử**

- Tạo kịch bản kiểm thử từ đó phát hiện ra những bug cần chỉnh sửa

#### **2.1.5. Triển khai**

- Triển khai sản phẩm với khách hàng

#### **2.1.1 Khảo sát, phân tích và đặc tả yêu cầu**

- Khảo sát các chức năng nghiệp vụ, yêu cầu của website.
- Tham khảo trang website của trường.



- Phân tích yêu cầu của website.
- Từ những yêu cầu của bài toán đặc tả thành yêu cầu của website.

## 2.2 Thiết kế phần mềm

- Thiết kế môi trường làm việc cục bộ:
  - Để tạo và tổ chức các tệp nguồn của một trang website, môi trường làm việc rất quan trọng.
  - Visual Studio Code là một trình soạn thảo code dành cho các lập trình viên:
    - Nhẹ, khởi động nhanh, tốn ít tài nguyên.
    - Nhiều tính năng hữu ích như chỉnh sửa tại nhiều vị trí cùng một lúc, soạn thảo toàn màn hình, soạn thảo với layout nhiều cột.
    - Hỗ trợ nhiều extension mạnh mẽ.
    - Giao diện đơn giản, mạnh mẽ, có sẵn và hỗ trợ nhiều theme.
- Phía front – end:
  - Sử dụng framework Angular.
  - Angular là một JavaScript framework dùng để viết giao diện web, được phát triển bởi Google để xây dựng các Single Page Application (SPA) bằng JavaScript, HTML, TypeScript.
  - TypeScript là một dự án mã nguồn mở được phát triển bởi Microsoft, nó có thể được coi là phiên bản nâng cao của JavaScript bởi nó bổ sung tùy chọn kiểu tĩnh và hướng đối tượng.
  - TypeScript có thể sử dụng để phát triển các ứng dụng chạy ở client – side (Angular) và server – side (NodeJS).
- Phía back – end:
  - Sử dụng công nghệ lập trình web API sử dụng ASP.Net Core.

- ASP.Net Core là gì? Nó là một open – source mới và framework đa nền tảng (cross – platform) cho việc xây dựng những ứng dụng hiện tại dựa trên kết nối đám mây.
- API được ví như “cánh tay phải” đắc lực không thể thiếu với lập trình website và thiết kế các phần mềm ứng dụng di động. Nó có tác dụng liên kết các tính năng của web, app và các cơ sở dữ liệu, tăng tương tác tối đa giữa người dùng và ứng dụng.
- Cơ sở dữ liệu:
  - Sử dụng hệ quản trị cơ sở dữ liệu Microsoft SQL Server.
  - SQL là một ngôn ngữ, là tập hợp các lệnh để tương tác với cơ sở dữ liệu. Dùng để lưu trữ, thao tác và truy xuất dữ liệu được lưu trữ trong một cơ sở dữ liệu quan hệ.
  - SQL sẽ giúp quản lý hiệu quả và truy vấn thông tin nhanh hơn, giúp bảo trì, bảo mật hệ thống dễ dàng hơn.

### **2.3 Lập trình phần mềm**

Dựa trên các yêu cầu chức năng của hệ thống ta sẽ tiến hành thiết kế website với các chức năng chính như sau:

- Quản lý danh sách giáo viên chủ nhiệm.
- Quản lý thông báo tới sinh viên, cựu sinh viên, giảng viên.
- Quản lý chương trình đào tạo.

Ngoài ra để có một website hoàn chỉnh sẽ sử dụng Angular, WEB API NET CORE, SQL.

### **2.4 Kiểm thử phần mềm**

- Lập kế hoạch kiểm thử.
- Thiết kế kiểm thử.

- Chuẩn bị dữ liệu kiểm thử.
- Thực thi kiểm thử:
  - Kiểm thử liên kết.
  - Kiểm thử Angular.
  - Kiểm thử HTML, CSS, JavaScript, TypeScript.
  - Kiểm tra điều hướng website.
  - Kiểm thử cơ sở dữ liệu.
- Đánh giá kết quả và báo cáo.

## **2.5 Triển khai và bảo trì phần mềm**

- Sau khi phần mềm được hoàn thành sẽ tiến hành bàn giao cho người dùng.
- Nếu như trong quá trình người dùng sử dụng phần mềm mà xảy ra lỗi thì sẽ tiến hành kiểm tra và sửa lỗi.

## **2.6 Phương pháp phân tích thiết kế hướng đối tượng**

- Ngôn ngữ UML:

Ngôn ngữ mô hình hóa thống nhất (Unified Modeling Language – UML) là một ngôn ngữ biểu diễn mô hình theo hướng đối tượng với chủ đích là:

- Mô hình hóa các hệ thống sử dụng các khái niệm hướng đối tượng.
- Thiết lập một kết nối từ nhận thức của con người đến các sự kiện cần mô hình hóa.
- Giải quyết vấn đề về mức độ thừa kế trong các hệ thống phức tạp, có nhiều ràng buộc.
- Tạo một ngôn ngữ mô hình hóa có thể sử dụng được bởi người và máy.
- UML trong phân tích và thiết kế hệ thống:

- Giai đoạn nghiên cứu sơ bộ (Preliminary Investigation): Use case đưa ra các khái niệm Use case để nắm bắt các yêu cầu của khách hàng (người sử dụng). UML sử dụng biểu đồ Use case để nêu bật mối quan hệ cũng như giao tiếp với hệ thống.
  - Giai đoạn phân tích (Analysis): Mục đích chính của giai đoạn này là trừu tượng hóa và các cơ cấu có trong phạm vi bài toán. Class Diagram trên bình diện các trừu tượng hóa các thực thể ngoài đời được sử dụng để làm rõ sự tồn tại cũng như mối quan hệ giữa chúng. Chỉ những lớp nằm trong bài toán mới đáng quan tâm.
  - Giai đoạn thiết kế (Design): Kết quả phần Analysis được phát triển thành giải pháp kỹ thuật. Các lớp được mô hình hóa chi tiết để cung cấp hạ tầng kỹ thuật như giao diện, nền tảng database, ... Kết quả phần này là các đặc tả chi tiết cho giai đoạn xây dựng phần mềm.
  - Giai đoạn xây dựng (Development): Mô hình Design được chuyển thành code. Programmer sử dụng các UML Diagram trong giai đoạn Design để hiểu vấn đề và tạo code.
  - Giai đoạn thử nghiệm (Testing): Sử dụng các UML Diagram trong các giai đoạn trước.
- Các quan hệ trong UML:

**Bảng 2.1: Các quan hệ trong UML**

Tác nhân	Biểu diễn các Actor (tác nhân), biểu diễn người hay hệ thống khác tương tác với hệ thống hiện tại.
----------	--

Ca sử dụng	Biểu diễn các Use Case (tình huống sử dụng) biểu diễn chuỗi hành động mà hệ thống thực hiện.
Quan hệ	Biểu diễn mối quan hệ giữa Actor (người dùng) và Use Case (chức năng hệ thống).
Quan hệ include	<p>Quan hệ &lt;&lt;Include&gt;&gt; thể hiện Use Case này chứa Use Case khác.</p> <p>Use Case A được gọi là Include B nếu trong xử lý của A có gọi đến B ít nhất một lần.</p>
Quan hệ Extend	<p>Quan hệ &lt;&lt;Extend&gt;&gt; thể hiện Use Case này mở rộng Use Case khác.</p> <p>Use Case A được gọi là Extend B nếu use case B được gọi bởi A nếu thỏa mãn điều kiện nào đó.</p>
Quan hệ tổng quát hóa	<p>Quan hệ tổng quát hóa (thừa kế):</p> <p>Use case A được gọi là Generalization Use Case B nếu Use Case B là một trường hợp riêng của Use Case A.</p>
Biểu đồ lớp	<p>Biểu đồ Lớp ClassDiagram:</p> <ul style="list-style-type: none"> <li>- Mô tả các đối tượng và mối quan hệ của chúng trong hệ thống.</li> <li>- Mô tả các thuộc tính và các hành vi (Behavior) của đối tượng.</li> </ul>

- Biểu đồ UML:

Thành phần mô hình chính trong UML là các biểu đồ:

- Biểu đồ Use Case: Biểu diễn sơ đồ chức năng của hệ thống. Từ tập yêu cầu của hệ thống, biểu đồ usecase sẽ phải chỉ ra hệ thống cần thực hiện điều gì để thỏa mãn các yêu cầu của người dung hệ thống đó. Đi kèm với biểu đồ usecase là các kịch bản.
- Biểu đồ lớp: Chỉ ra các lớp đối tượng trong hệ thống, các thuộc tính và phương thức của từng lớp và những mối quan hệ giữa các lớp đó.
- Biểu đồ trạng thái: Tương ứng với mỗi lớp sẽ chỉ ra các trạng thái mà đối tượng của lớp đó có thể có và sự chuyển tiếp giữa những trạng thái đó.
- Các biểu đồ tương tác biểu diễn mối quan hệ giữa các đối tượng trong hệ thống và giữa các đối tượng với các tác nhân bên ngoài. Có hai loại biểu đồ tương tác:
  - Biểu đồ tuần tự: Biểu diễn mối quan hệ giữa các đối tượng, giữa các đối tượng và tác nhân theo thời gian.
  - Biểu đồ cộng tác: Biểu diễn các quan hệ giữa các đối tượng, giữa các đối tượng và tác nhân nhưng nhấn mạnh đến vai trò của các đối tượng trong tương tác.
- Biểu đồ hoạt động: Biểu diễn các hoạt động và sự đồng bộ, chuyển tiếp các hoạt động, thường được sử dụng để biểu diễn các phương thức phức tạp của các lớp.
- Biểu đồ thành phần: Định nghĩa các thành phần của hệ thống và mối liên hệ giữa các thành phần đó.

- Biểu đồ triển khai: Mô tả hệ thống sẽ được triển khai như thế nào, thành phần nào được cài đặt ở đâu, các liên kết vật lý hoặc phương thức truyền thông được sử dụng.

## 2.7. Giới thiệu các xu hướng lập trình web mới

Sự phát triển của công nghệ luôn nằm ở mức thay đổi chóng mặt. Chính sự phát triển không ngừng ấy đòi hỏi phải có cái mới hình thành, tiên tiến hơn và thay thế cái cũ. Trong lĩnh vực công nghệ thông tin, các lập trình viên luôn phải cập nhật kiến thức không ngừng để bắt kịp xu hướng để vừa hiện đại vừa tránh bị đào thải. “XU HƯỚNG” là cụm từ luôn được đề cập sau mỗi lần chuyển đổi hoặc có sự ra đời mang tính chuyển biến của một ngôn ngữ hay công nghệ nào đó.

- Voice Search (Tìm kiếm bằng giọng nói)

Tương lai phát triển web dường như sẽ nhiều tiếng nói hơn và nó không chỉ nói về các trợ lý gia đình đa chức năng ảo như Siri, Google Home hay Amazon Alexa. Vào năm 2020, hơn một nửa trong số tất cả các thiết bị thông minh và IoT sẽ nghe người dùng và thực hiện các lệnh được đưa ra bằng giọng nói. Và hơn thế nữa - nó có thể nhận ra giọng nói của những người khác nhau và cung cấp trải nghiệm dựa trên AI được cá nhân hóa cho mọi người

- WebAssembly

Khi xây dựng một ứng dụng web, hiệu suất thường bị tổn hại. Các giới hạn JavaScript làm cho các phép tính nặng bị chậm và điều đó ảnh hưởng đáng kể đến trải nghiệm người dùng. Đây là lý do tại sao hầu hết các trò chơi phổ biến và các ứng dụng mạnh mẽ chỉ có sẵn dưới dạng một ứng dụng máy tính để bàn gốc. WebAssembly ra đời để thay đổi điều đó - nhằm mục đích cho hiệu suất giữa các ứng dụng web là như nhau trên mọi trình duyệt và nền tảng. Với WebAssembly, mã trong bất kỳ ngôn ngữ lập trình nào cũng có thể được biên dịch thành mã byte chạy trong trình duyệt.

- Cá nhân hóa nội dung



Trí tuệ nhân tạo, bao gồm cả học máy, ảnh hưởng đến các hoạt động hàng ngày của chúng ta trên internet mà chúng ta thậm chí không nhận ra. Đây là điểm chính của ML - cung cấp trải nghiệm được cải thiện nguyên bản. Học máy là khả năng của phần mềm để cải thiện hiệu suất mà không cần đầu vào trực tiếp từ các nhà phát triển. Về cơ bản, phần mềm phân tích dữ liệu đến, phát hiện các mẫu, đưa ra quyết định và cải thiện công việc của nó.

- Data Security (Bảo mật dữ liệu)

Càng nhiều dữ liệu mà một ứng dụng web xử lý, nó càng hấp dẫn đối với tội phạm mạng. Cách làm của chúng có thể thay đổi, nhưng luôn có mục đích làm hỏng dịch vụ của bạn và đánh cắp dữ liệu của người dùng hoặc thông tin nội bộ của công ty. Điều này có thể gây tổn hại lớn đến danh tiếng của bạn và khiến bạn tổn kém rất nhiều. Các mối đe dọa và hoạt động độc hại dễ dàng phát hiện hơn với phần mềm bảo mật do AI cung cấp. Khi sự đa dạng của phần mềm độc hại phát triển, nó trở nên mạnh hơn và nguy hiểm hơn. Vì vậy, các công ty lớn hiện đào tạo các hệ thống AI của họ để phân tích các mẫu hành vi trong mạng và phản ứng với mọi hoạt động đáng ngờ ngay lập tức.

- Progressive web Apps and AMP

Đây có thể không còn là xu hướng. Nó gần như là một tiêu chuẩn bởi vì... Google ưu tiên các ứng dụng web tải nhanh trên thiết bị di động là một điểm cộng cho xếp hạng. Đây là lý do tại sao nên xem xét triển khai PWA (Progressive Web Apps) hoặc AMP, đây là những công nghệ độc đáo giúp giảm thời gian tải trang web. Người dùng ngày càng không kiên nhẫn. Bạn chậm đồng nghĩa với việc tiền của khách hàng sẽ rơi vào túi đối thủ.

- Trí tuệ nhân tạo và Bots

Tương lai của phát triển web sẽ như thế nào? Điều đó được nhiều người tin rằng các con bot năm tới sẽ trở nên tự học hơn và sẽ dễ dàng phù hợp với nhu cầu và hành vi của người dùng cụ thể. Điều đó có nghĩa là các bot hoạt động 24/7 sẽ có thể thay thế các nhà quản lý hỗ trợ và tiết kiệm chi phí cho nhiều công ty. Sự gia tăng của Bot cũng sẽ ảnh hưởng đến xu hướng thiết kế web, đặc biệt là UX. Sự tiến bộ đáng kể của AI và công nghệ máy học dẫn đến nhu cầu thiết kế trợ lý ảo, không chỉ về trải nghiệm hình ảnh, hoặc thuật toán phục vụ sản phẩm, mà còn về thiết kế âm thanh hữu ích.

- Motion UI
- Automation testing (Kiểm thử tự động)
- JavaScript Frameworks
- Ứng dụng và kiến trúc không máy chủ
- Ứng dụng công nghệ blockchain
- Internet of thing - IOT
- GDPR và Cybersecurity
- Single page application - SPA

## 2.8 Tổng quan về Angular

Angular là một nền tảng web framework được phát triển bởi Google. Phiên bản đầu tiên của Angular được công bố vào tháng 10/2010 với tên gọi là AngularJS phát triển dựa trên nền tảng JavaScript với kiến trúc MVC (Model View Controller). Các phiên bản tiếp theo của Angular được đổi tên thành tên phiên bản thay vì sử dụng tên là AngularJS (Angular 2.0; Angular 4.0...) và được phát triển dựa trên TypeScript thay vì JavaScript (tính tới thời điểm của bài viết, phiên bản mới nhất của Angular là phiên bản Angular 12 được công bố tháng 5/2021). Chúng ta có thể coi TypeScript là phiên bản nâng cao của JavaScript vì nó bao gồm các phiên bản ES và bổ sung thêm các kiểu dữ liệu tĩnh, các lớp hướng đối tượng giúp người dùng dễ dàng viết các đoạn code hơn (code viết bằng TypeScript sẽ được biên dịch sang JavaScript).

Để dễ tiếp cận cách thức hoạt động của Angular, chúng ta sẽ tìm hiểu kiến trúc tổng quan của Angular 2 bao gồm 8 thành phần chính là Modules, Components, Templates, Metadata, Data binding, Directives, Services và Dependency injection [1] (Các phiên bản tiếp theo có bổ sung và thay đổi cách sắp xếp các thành phần chính của Angular nhưng nội dung, ý nghĩa và cách thức sử dụng cũng tương tự như Angular 2.

## 2.9 Lập trình Web API NET CORE

### ❖ ASP.NET Core là gì?

Nó là một open-source mới và framework đa nền tảng (cross-platform) cho việc xây dựng những ứng dụng hiện tại dựa trên kết nối đám mây, giống như web apps, IoT và backend cho mobile.

Ứng dụng ASP.NET Core có thể chạy trên .NET Core hoặc trên phiên bản đầy đủ của .NET Framework. Nó được thiết kế để cung cấp và tối ưu development framework cho những dụng cụ mà được triển khai trên đám mây (cloud) hoặc chạy on-premise.

Nó bao gồm các thành phần theo hướng module nhằm tối thiểu tài nguyên và chi phí phát triển, như vậy bạn giữ lại được sự mềm dẻo trong việc xây dựng giải pháp của bạn. Bạn có thể phát triển và chạy những ứng dụng ASP.NET Core đa nền tảng trên Windows, Mac và Linux.

Đồng thời nó đã trở thành một mã nguồn mở. Đây là một thay đổi rất lớn và theo mình là quan trọng nhất của ASP.NET Core. Điều mà trước đây khó có một lập trình viên nào có thể nghĩ đến. Có lẽ đó cũng là một xu thế mà các ngôn ngữ lập trình hiện nay đang hướng tới.

### ❖ Một số tính năng nổi bật của ASP .NET Core:

- Hợp nhất việc xây dựng web UI và web APIs
- Tích hợp những client-side frameworks hiện đại và những luồng phát triển
- Hệ thống cấu hình dựa trên môi trường đám mây thật sự

- Dependency injection được xây dựng sẵn
- HTTP request được tối ưu nhẹ hơn
- Có thể host trên IIS hoặc self-host trong process của riêng bạn
- Được xây dựng trên .NET Core, hỗ trợ thực sự app versioning
- Chuyển các thực thể, thành phần, module như những NuGet packages
- Những công cụ mới để đơn giản hóa quá trình phát triển web hiện đại
- Xây dựng và chạy đa nền tảng (Windows, Mac và Linux)
- Mã nguồn mở và tập trung vào cộng đồng

#### ❖ **Xây dựng web UI và web API**

- Có thể xây dựng HTTP services cái mà hỗ trợ nhiều định dạng và có đầy đủ hỗ trợ cho nội dung của dữ liệu trả về
- Model Binding tự động ánh xạ dữ liệu từ HTTP request tới tham số của method action
- Model Validation tự động thực hiện validate client và server

#### ❖ **Phát triển client-side**

ASP.NET Core được thiết kế để tích hợp một cách liên tục với nhiều client-side frameworks, bao gồm AngularJS, KnockoutJS và Bootstrap. Về phần này tôi sẽ giới thiệu với các bạn ở những bài viết sau

## CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

### 3.1 Phát biểu bài toán

Hệ thống công thông tin FITUTEHY – Lãnh đạo dành cho các lãnh đạo quản lý trong Khoa Công nghệ thông tin. Các yêu cầu của hệ thống gồm các yêu cầu chính sau:

Hệ thống được quản lý bằng tài khoản, mỗi giảng viên khi sử dụng hệ thống sẽ được cấp phát một tài khoản gồm tên đăng nhập, mật khẩu, quyền. Mỗi tài khoản sẽ có hai quyền đó là quyền là lãnh đạo Khoa, lãnh đạo bộ môn hoặc không có quyền gì cả.

#### a) Quản lý chương trình đào tạo

- Khi đào tạo một ngành mới khoa sẽ tiến hành mở ngành theo quy trình mở ngành với nhà trường, mã ngành đào tạo tuân theo hệ thống mã ngành của bộ Giáo dục và đào tạo. Một ngành mới mở có thể có nhiều chuyên ngành.
- Mỗi chuyên ngành đào tạo sẽ có một khung chương trình đào tạo. Một khung chương trình đào tạo, được chia thành 4 khối kiến thức gồm: Khối kiến thức đại cương, khối kiến thức cơ sở ngành, khối kiến thức chuyên ngành và khối kiến thức đồ án và thực tập. Tổng số tín chỉ của các khối kiến thức phải tuân theo quy định của bộ Giáo dục và Đào tạo (Hiện nay tổng số tín chỉ áp dụng cho chương trình đào tạo của Khoa CNTT là  $150TC + 11 TC$  giáo dục quốc phòng). Mỗi khung chương trình gồm nhiều môn học thuộc các khối kiến thức khác nhau được bố trí giảng dạy vào các học kỳ khác nhau (Hiện tại chương trình đào tạo của khoa CNTT gồm có 8 học kỳ). Một môn học tùy theo khối kiến thức có thể thuộc trong các khung chương trình đào tạo khác nhau và thuộc vào môn học bắt buộc hay tự chọn. Thông tin chi tiết về chương trình đào tạo, khung chương trình đào tạo, các môn học tham khảo tài liệu 3 -1.
- Các lớp cùng một chuyên ngành và cùng một khóa được học cùng một khung chương trình đào tạo. Khung chương trình đào tạo của các khóa khác nhau có

thể thay đổi các môn học và thời gian thực hiện các môn học khi khoa tiến hành chỉnh sửa chương trình. Khi chỉnh sửa khung chương trình có thể chỉnh sửa chương trình của một khóa đang đào tạo với các học kỳ chưa thực hiện. Việc chỉnh sửa khung chương trình đào tạo có thể là: loại bỏ một môn học nào đó; thay đổi số tín, thay đổi kỳ giảng dạy một môn học, ... Mỗi khung chương trình đào tạo chỉnh sửa được đánh version quản lý phân biệt nhau. Khi thay đổi khung chương trình đào tạo, các version khác nhau cũng như thời gian chỉnh sửa cần lưu lại để quản lý.

- Sau khi xây dựng hoặc chỉnh sửa khung chương trình, các khung chương trình được in ra và ký gửi phòng đào tạo. Để đảm bảo công tác lưu trữ và tra cứu về quá trình thay đổi chương trình, các bản cứng scan có chữ ký cần được lưu trữ lại để dễ dàng quản lý, đối chiếu và tra cứu sau này. Mẫu khung chương trình in ra xem Tài liệu 3-1.

**b) Quản lý danh sách giáo viên chủ nhiệm**

- Khoa công nghệ thông tin có 3 bộ môn chuyên môn khác nhau là Bộ môn Công nghệ phần mềm (CNPM), bộ môn Khoa học máy tính (KHMT), bộ môn Hệ thống thông tin (HTTT). Các giáo viên trong khoa tùy thuộc chuyên môn sẽ được quản lý bởi một trong 3 bộ môn. Trong quá trình công tác, giáo viên có thể được chuyển sang bộ môn khác theo điều động chuyên môn của lãnh đạo khoa.
- Mỗi một bộ môn sẽ quản lý một hoặc nhiều ngành đào tạo khác nhau (Hiện tại là quản lý 1 ngành. Bộ môn CNPM quản lý ngành KTPM, BM KHMT quản lý ngành KHMT, BM HTTT quản lý ngành CNTT).
- Khoa Công nghệ thông tin (CNTT) có nhiều ngành đào tạo khác nhau. Mỗi ngành đào tạo có thể có nhiều chuyên ngành khác nhau. Các ngành và chuyên ngành của khoa công nghệ thông tin đang đào tạo được mô tả trong tài liệu 3-1.

- Một chuyên ngành đào tạo có thể có một hoặc nhiều lớp. Quy tắc đánh mã lớp như sau: mỗi mã lớp gồm 6 ký tự trong đó 3 ký tự đầu là mã ngành, 2 ký tự sau là 2 số cuối năm vào, ký tự cuối cùng là thứ tự lớp. Mô tả chi tiết mã lớp được mô tả trong mục 3.1 của tài liệu 3-2.
  - Đầu mỗi năm học, khi có khóa mới vào, trưởng/phó các bộ môn bộ (T/P BM) môn sẽ tiến hành phân công các giáo viên chủ nhiệm cho các lớp bộ môn quản lý (Trong trường hợp thiếu giáo viên, bộ môn có thể nhờ các thầy cô ở ngoài bộ môn chủ nhiệm). Để đảm bảo cân đối công tác giáo viên chủ nhiệm khi phân công sẽ ưu tiên giáo viên có số lớp chủ nhiệm ít và đang chủ nhiệm các lớp năm cuối cũng như giáo viên chủ nhiệm lớp phù hợp với chuyên môn của giáo viên. Nếu thông tin về giáo viên chủ nhiệm có sai sót trong quá trình phân công, hoặc điều chỉnh trong quyết định của nhà trường T/P BM có thể điều chỉnh thay đổi, hoặc lớp bị hủy có thể xóa thông tin GVCN. Trong quá trình học tập có thể do lý do nào đó, giáo viên chủ nhiệm thôi không tiếp tục chủ nhiệm lớp được nữa, T/P BM có thể thực hiện thay đổi giáo viên chủ nhiệm cho lớp.
  - Ngoài ra để phục vụ cho việc tra cứu thông tin các giáo viên và lãnh đạo khoa có thể tra cứu thông tin về giáo viên chủ nhiệm theo lớp, theo khóa, theo chuyên ngành, theo ngành, theo bộ môn, theo môn học, ...
- c) Quản lý hướng dẫn đồ án.
- Việc giám sát quá trình HD đồ án là rất quan trọng, đảm bảo quá trình thực hiện của sinh viên diễn ra đúng quy định; nâng cao chất lượng đào tạo của Môn học.
  - Trước khi triển khai HD đồ án, Lãnh đạo BM sẽ phân công số lượng SV cho từng GV, theo từng lớp, từng loại đồ án (1-5), theo từng kỳ học, năm học. Đối với ĐATN sẽ phân số lượng dựa trên quy định của nhà trường (Tiền sĩ tối đa 10; Th.S tối đa 7), việc quy định số lượng HD ĐATN không theo lớp, mà theo kỳ học, năm học.

- Trong trường hợp SV không chọn được GV HD thì lãnh đạo bộ môn sẽ phân công.
- Trong quá trình giám sát việc thực hiện, các LĐ Bộ môn hoặc những GV được phân công (gọi chung là người giám sát) là những người thường xuyên kiểm tra xem SV có thực hiện đồ án theo đúng quy trình hay không (thông qua nội dung báo cáo, hoặc sản phẩm của đồ án, thông qua việc kiểm tra ngẫu nhiên theo tuần). Ngoài ra, kiểm tra xem GVHD có thực hiện đánh giá kết quả của SV theo từng tuần hay không?
- Người giám sát sẽ phân công GV kiểm tra tiến độ (Lần 1, Lần 2, Lần 3 (đối với ĐATN)), để dễ thực hiện sẽ phân công 1 GV sẽ tham gia kiểm tra tiến độ cả lần 1, 2 hoặc lần 3.
- Sau khi phân công GV đánh giá tiến độ (gọi chung là người đánh giá tiến độ), hệ thống cần thông báo đến người được phân công (có thể là thông báo trên hệ thống hoặc Email, ghi cụ thể được phân công đánh giá tiến độ nhóm nào, lớp nào, đồ án nào thời gian cần hoàn thành,...).
- Người đánh giá tiến độ, khi đăng nhập vào hệ thống sẽ lựa chọn chức năng tham gia đánh giá tiến độ. Chức năng này, sẽ hiển thị danh sách SV theo từng nhóm, từng loại đồ án. Người đánh giá sẽ xem được báo cáo (có thể cả sản phẩm) của từng sinh viên được phân công đánh giá tiến độ; sau đó đưa ra đánh giá về 3 tiêu chí: Tiến độ (Có hoàn thành đúng tiến độ hay không), Chất lượng (của báo cáo, sản phẩm; Đạt hay không đạt), Điểm (đánh giá theo thang điểm 10). Ngoài ra, để chứng cho kết quả đánh giá; người đánh giá tiến độ cần đưa ra những ưu nhược điểm của đồ án; SV có thể xem được những góp ý này để tiếp tục hoàn thiện đồ án tốt hơn.
- Người giám sát, hàng tuần hoặc một số tuần liên tiếp cần nhận được thống kê tình hình hướng dẫn và thực hiện đồ án của từng nhóm theo 03 danh sách:



- Đối với GVHD: Đưa ra DS các GVHD chưa hoàn thành viên đánh giá đồ án
- Đối với GV đánh giá tiến độ: Đưa ra DS những GV chưa hoàn thành việc đánh giá tiến độ theo quy định;
- Đối với SV: DS những SV không nộp báo cáo, sản phẩm; hoặc những SV chưa hoàn thành báo cáo theo tiến độ, hoặc sản phẩm chưa đạt yêu cầu.

Sau tuần thứ 15, cần thống kê DS những SV đủ điều kiện được bảo vệ (Điểm đánh giá tiến độ, điểm HD  $\geq 5$ ), những SV không đủ điều kiện bảo vệ.

### 3.2 Đặc tả yêu cầu phần mềm

#### 3.2.1 Các yêu cầu chức năng

##### 3.2.1.1 Bảng chức năng phân hệ trang quản trị

###### a) Quản lý chương trình đào tạo

**Bảng 1 Các yêu cầu chức năng của quản lý chương trình đào tạo**

STT	Chức năng	Mô tả
<b>I</b>	<b>Quản lý khung chương trình đào tạo</b>	<b>Là chức năng cho phép cập nhật các khung chương trình đào tạo</b>
1	Thêm khung chương trình đào tạo	Là chức năng cho phép thêm chương trình đào tạo.
2	Sửa khung chương trình đào tạo	Là chức năng cho phép sửa chương trình đào tạo.

3	Xóa khung chương trình đào tạo	Là chức năng cho phép xóa chương trình đào tạo.
4	Gán lớp	Là chức năng cho phép gán lớp học vào chương trình đào tạo đã chọn.
<b>II</b>	<b>Quản lý danh mục học phần</b>	<b>Là chức năng cho phép quản lý danh mục học phần</b>
1	Thêm học phần	Là chức năng cho phép thêm học phần vào chương trình đào tạo.
2	Sửa học phần	Là chức năng cho phép sửa học phần khi bị nhập sai.
3	Xóa học phần	Là chức năng cho phép người dùng xóa học phần khi học phần không còn tồn tại.

**b) Quản lý danh sách giáo viên chủ nhiệm**

***Bảng 2 Các yêu cầu chức năng của quản lý danh sách giáo viên chủ nhiệm***

STT	Chức năng	Mô tả
1	Phân công giáo viên chủ nhiệm theo bộ môn	Lãnh đạo khoa hoặc lãnh đạo bộ môn sẽ phân công giáo viên chủ nhiệm cho các lớp.
2	Tìm kiếm GVCN theo lớp	Là chức năng cho phép tìm kiếm GVCN theo lớp được nhập.
3	Tìm kiếm lớp theo GVCN	Là chức năng cho phép tìm kiếm các lớp mà một GV nào đó phụ trách chủ nhiệm

4	Kết xuất/ In danh sách giáo viên chủ nhiệm theo Khoa/ Bộ môn/ Ngành học	Là chức năng cho phép in danh sách giáo viên chủ nhiệm theo Khoa/ Bộ môn/ Ngành học dưới dạng bảng tính Excel.
---	---	--

c) Quản lý quá trình hướng dẫn đồ án.

***Bảng 3 Quản lý hướng dẫn đồ án***

STT	Chức năng	Mô tả
1	Tạo đồ án	Là chức năng cho phép tạo các đồ án là các danh mục quản lý chung các lớp đang thực hiện một đồ án nào đó.
2	Sửa đồ án	Sửa thông tin đồ án khi bị nhập sai.
3	Xóa đồ án	Lãnh đạo bộ môn hoặc lãnh đạo khoa sẽ thực hiện xóa đồ án khi đồ án đó không còn được thực hiện nữa.
4	Quản lý danh mục lớp thực hiện đồ án	Là chức năng cho phép người sử dụng quản lý danh mục các lớp đang thực hiện đồ án nào đó. Lãnh đạo hoặc lãnh đạo khoa có thể thêm, sửa xóa các lớp vào danh mục

***Biểu đồ usecase***

+

+

-

-

+

-

-

-

-

-

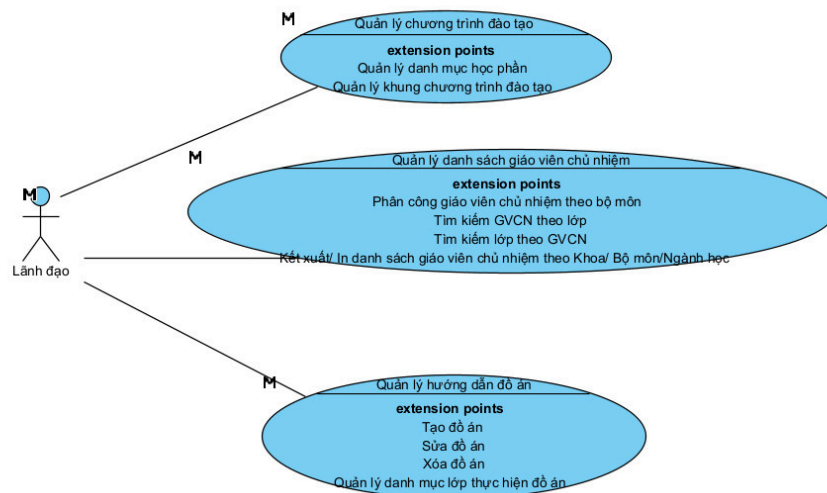
-

-

-

3.2.1.2

◆ **Biểu đồ Use Case tổng quát**

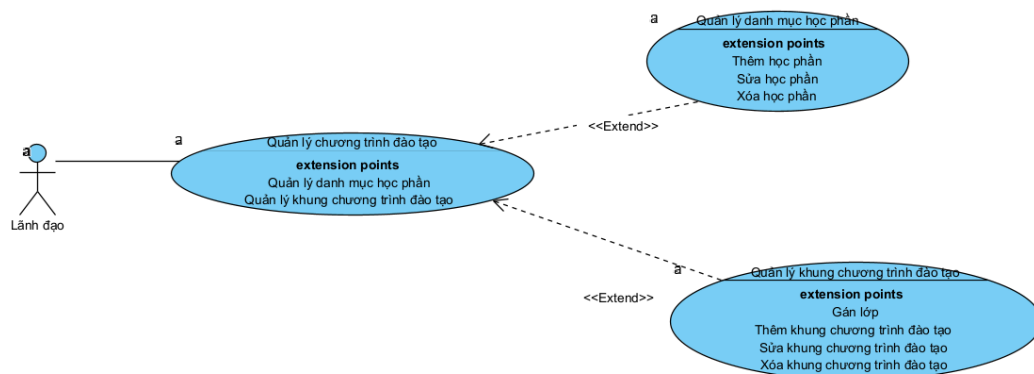


**Hình 1: Biểu đồ Use Case tổng quát**

## ◆ Biểu đồ Use Case phân rã

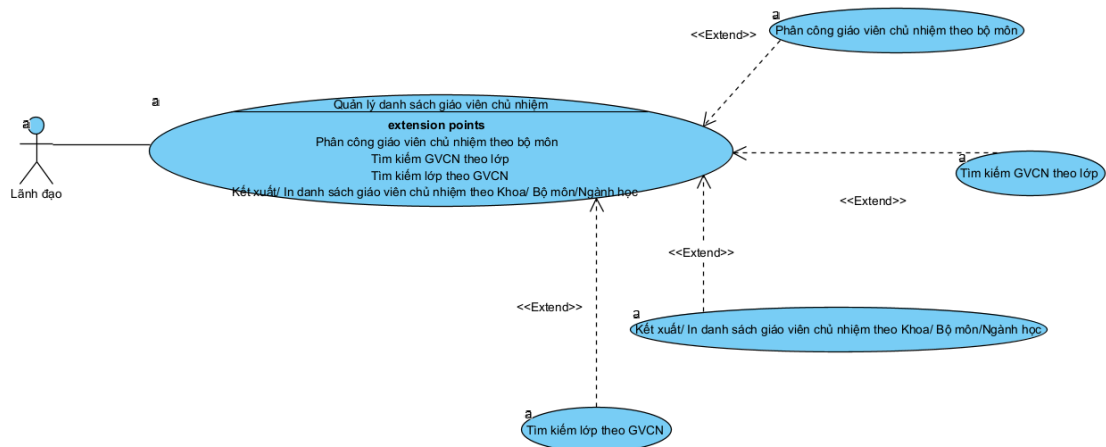
### ● Use Case quản lý chương trình đào tạo

uc [Chương trình đào tạo]



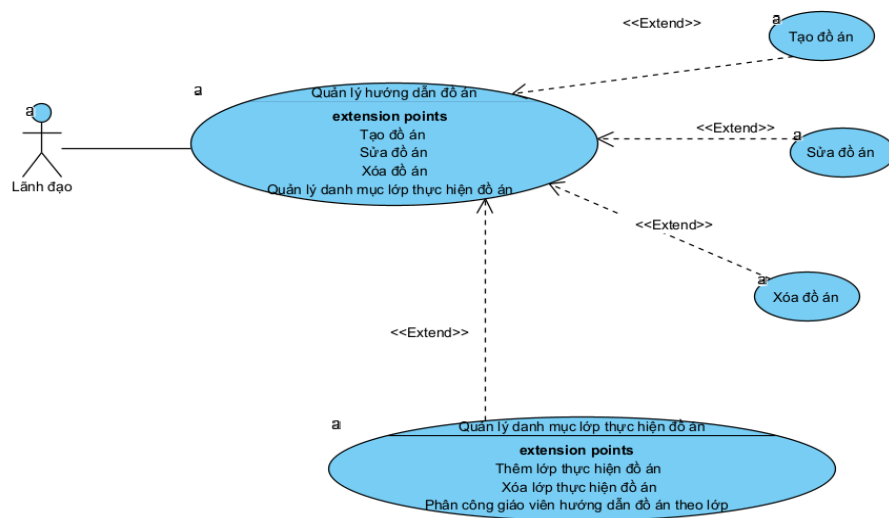
**Hình 2: Biểu đồ phân rã Use Case chương trình đào tạo**

- Use case Quản lý danh sách giáo viên chủ nhiệm.



**Hình 3 Use case Quản lý danh sách giáo viên chủ nhiệm**

- Use case quản lý hướng dẫn đồ án.



**Hình 4 Use case quản lý hướng dẫn đồ án**

### 3.2.1.3 Luồng sự kiện

- ❖ *Use Case quản lý học phần*
- ❖ *Mô tả luồng sự kiện của ca sử dụng thêm học phần.*

<p><b>Use case name:</b> Thêm học phần.</p> <p><b>Brief description:</b> Lãnh đạo thực hiện chức năng thêm học phần.</p> <p><b>Actors:</b> Lãnh đạo.</p> <p><b>Precondition:</b> Lãnh đạo đăng nhập vào hệ thống thành công</p> <p><b>Post – condition:</b> Nếu thêm học phần thành công, một học phần sẽ được lưu và hiển thị. Trong trường hợp khác, hệ thống yêu cầu hiển thị lỗi.</p> <p><b>Trigger:</b> Lãnh đạo yêu cầu chức năng thêm học phần.</p>
<p><b>Basic flow</b></p>
<p>1. Lãnh đạo chọn chức năng thêm học phần.</p>

<p>2. <b>SYSTEM</b> Hệ thống hiển thị form nhập thông tin học phần.</p> <p>3. Lãnh đạo nhập thông tin học phần.</p> <p>4. Lãnh đạo thực hiện chức năng lưu thông tin, nếu không lưu thì chuyển sang luồng phụ 4a, nếu lưu thì tiếp tục tới bước 5.</p> <p>5. <b>SYSTEM</b> Hệ thống kiểm tra thông tin học phần, nếu không hợp lệ chuyển sang luồng phụ 5a. nếu hợp lệ chuyển tới bước 6.</p> <p>6. <b>SYSTEM</b> Hệ thống lưu thông tin học phần vào hệ thống.</p>
<b>Alternate flows</b>
<p>4a. Lãnh đạo không lưu học phần.</p> <p>1. Lãnh đạo không thêm học phần mới, chọn nút hủy</p> <p>2. <b>SYSTEM</b> Thoát khỏi form nhập thông tin học phần.</p> <p>5a. Thông tin học phần không hợp lệ</p> <p>1. <b>SYSTEM</b> Hệ thống thông báo thông tin học phần cần thêm không hợp lệ, yêu cầu người dùng nhập lại thông tin quay về bước 6.</p>

- *Mô tả luồng sự kiện của ca sử dụng sửa học phần.*

<p><b>Use case name:</b> Sửa học phần.</p> <p><b>Brief description:</b> Lãnh đạo thực hiện chức năng sửa học phần.</p> <p><b>Actors:</b> Lãnh đạo.</p> <p><b>Precondition:</b> Lãnh đạo đăng nhập vào hệ thống thành công</p> <p><b>Post – condition:</b> Nếu sửa học phần thành công, một học phần sẽ được sửa và hiển thị. Trong trường hợp khác, hệ thống yêu cầu hiển thị lỗi.</p> <p><b>Trigger:</b> Lãnh đạo yêu cầu chức năng sửa học phần.</p>
<b>Basic flow</b>
<p>1. Lãnh đạo chọn chức năng sửa học phần.</p> <p>2. <b>SYSTEM</b> Hệ thống hiển thị form sửa thông tin học phần.</p> <p>3. Lãnh đạo sửa thông tin học phần cần sửa.</p>



4. Lãnh đạo thực hiện chức năng lưu thông tin, nếu không lưu thì chuyển sang luồng phụ 4a
5. <b>SYSTEM</b> Hệ thống kiểm tra thông tin học phần. Nếu thông tin không hợp lệ chuyển sang luồng phụ 5a, nếu hợp lệ chuyển sang bước 6.
6. <b>SYSTEM</b> Hệ thống lưu thông tin về sản phẩm vừa sửa vào hệ thống.
<b>Alternate flows</b>
4a. Lãnh đạo không lưu thông tin học phần. 1. Lãnh đạo không sửa học phần đó, chọn nút hủy 2. <b>SYSTEM</b> Thoát khỏi form sửa thông tin học phần. 5a. Thông tin học phần không hợp lệ 1. <b>SYSTEM</b> Hệ thống thông báo thông tin học phần cần sửa không hợp lệ, yêu cầu người dùng nhập lại thông tin và quay trở về bước 5.

- **Mô tả luồng sự kiện của ca sử dụng xóa học phần**

<b>Use case name:</b> Xóa học phần.
<b>Brief description:</b> Lãnh đạo thực hiện chức năng xóa học phần.
<b>Actors:</b> Lãnh đạo.
<b>Precondition:</b> Lãnh đạo đăng nhập vào hệ thống thành công.
<b>Post – condition:</b> Nếu xóa thành công, một học phần sẽ được xóa. Trong trường hợp khác, hệ thống yêu cầu hiển thị lỗi.
<b>Trigger:</b> Lãnh đạo yêu cầu chức năng xóa học phần.
<b>Basic flow</b>
1. Lãnh đạo chọn chức năng xóa học phần. 2. Lãnh đạo yêu cầu xóa học phần. 3. <b>SYSTEM</b> Hệ thống hiển thị thông báo xác nhận xóa bỏ, xóa học phần đó khỏi hệ thống. Nếu không xóa học phần sẽ chuyển sang mục 3a.
<b>Alternate flows</b>
3a. Người dùng không xóa học phần.

1. Lãnh đạo không xóa học phần đó, chọn nút hủy.
2. **SYSTEM** Thoát khỏi form xóa thông tin học phần.

#### 3.2.1.4 *Biểu đồ lớp thực thể*

Trong quá trình khảo sát hệ thống, em đã xác định được các thuật ngữ miền của hệ thống như sau:

#### ❖ **Khung chương trình đào tạo**

***Bảng 4 Các thuộc tính của bảng khung chương trình đào tạo***

STT	Tên thuộc tính	Mô tả
1	Mã chương trình đào tạo	Mã định danh của chương trình đào tạo
2	Tên chương trình đào tạo	Tên của chương trình đào tạo
3	Năm bắt đầu đào tạo	Năm bắt đầu đào tạo ngành đó
4	Loại chương trình đào tạo	Loại chương trình đào tạo
5	Tổng số tín chỉ	Tổng số tín chỉ của chương trình đào tạo

#### ❖ **Học phần**

***Bảng 5 Các thuộc tính của bảng học phần***

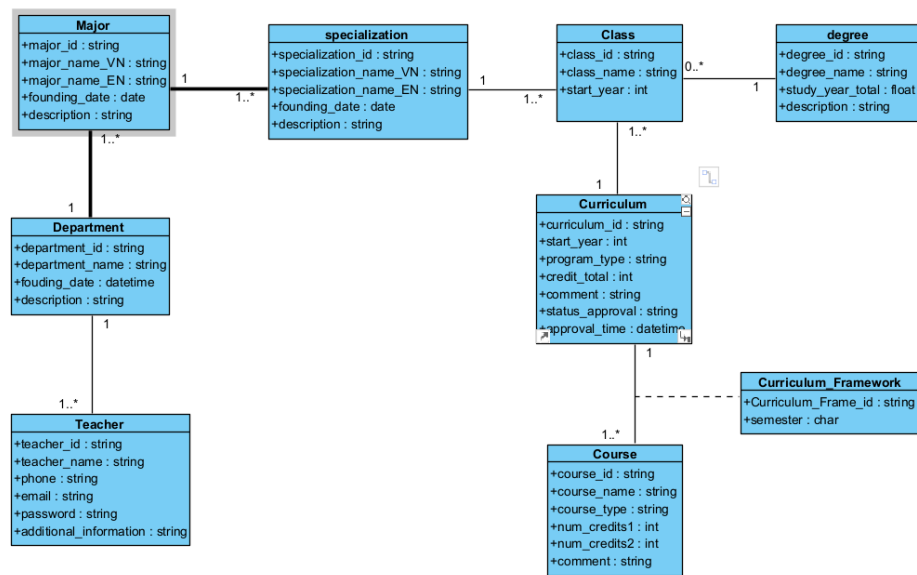
STT	Tên thuộc tính	Mô tả
1	Mã học phần	Mã học phần
2	Tên học phần	Tên học phần
3	Số tín chỉ lý thuyết	Số tín chỉ lý thuyết
4	Số tín chỉ thực hành	Số tín chỉ thực hành

❖ **Bộ môn**

*Bảng 6 Các thuộc tính của bảng bộ môn*

<b>STT</b>	<b>Tên thuộc tính</b>	<b>Mô tả</b>
1	Mã bộ môn	Mã định danh bộ môn
2	Tên bộ môn	Tên gọi của bộ môn
3	Ngày thành lập	Ngày thành lập
4	Mô tả	Mô tả về bộ môn

❖ **Biểu đồ lớp thực thể**



**Hình 5: Biểu đồ lớp thực thể**

### 3.2.2 Các yêu cầu phi chức năng

- Yêu cầu về hiệu quả sử dụng: phần mềm phải được thiết kế tăng hiệu quả làm việc, tiết kiệm thời gian cho các cấp quản lý chuyên môn.
- Tính dễ sử dụng: phần mềm phải thân thiện, dễ sử dụng và có đầy đủ tài liệu hướng dẫn, hướng dẫn cài đặt cho người dùng.
- Khả năng dễ thay đổi, cải tiến: xây dựng phần mềm có tính mở, dễ dàng cập nhật, bổ sung, cải tiến khi có các yêu cầu phát sinh.
- Yêu cầu về giao diện: giao diện người dùng phải được thiết kế rõ ràng, nhất quán, khoa học, đảm bảo hiệu suất giao tiếp người-máy.
- Tính khả chuyển: hệ thống chạy được trên nhiều môi trường khác nhau.



❖ Giảng viên (TBL\_teacher\_ref)

*Bảng 7 Mô tả cấu trúc bảng cơ sở dữ liệu giảng viên (TBL\_teacher\_ref)*

STT	Tên trường	Kiểu dữ liệu	Mô tả
1	teacher_id_rcd	varchar	Mã giảng viên
2	teacher_name	nvarchar	Tên giảng viên
3	manager_role_id1	varchar	Chức vụ quản lý của cán bộ giảng viên
4	manager_role_id2	varchar	Chức vụ đoàn thể của cán bộ giảng viên
5	department_id	varchar	Mã bộ môn, hoặc trung tâm
6	phone	varchar	Số điện thoại giảng viên
7	email	varchar	Email giảng viên
8	password	varchar	Mật khẩu
9	additional_information	nvarchar	Thông tin bổ sung.
10	created_by_user_id	uniqueidentifier	
11	created_date_time	datetime	
12	lu_updated	datetime	
13	lu_user_id	uniqueidentifier	
14	active_flag	int	

❖ Bộ môn (TBL\_department\_ref)

*Bảng 8 Mô tả cấu trúc bảng cơ sở dữ liệu bộ môn (TBL\_department\_ref)*

STT	Tên trường	Kiểu dữ liệu	Mô tả
-----	------------	--------------	-------

1	department_id_rcd	varchar	Mã bộ môn, hoặc trung tâm
2	department_name	nvarchar	Tên bộ môn hoặc trung tâm
3	teacher_id1	varchar	Mã trưởng bộ môn/ GD trung tâm
4	teacher_id2	varchar	Mã phó trưởng bộ môn/ Phó GD TT
5	founding_date	date	Ngày thành lập
6	description	nvarchar	Giới thiệu về bộ môn
7	created_by_user_id	uniqueidentifier	
8	created_date_time	datetime	
9	lu_updated	datetime	
10	lu_user_id	uniqueidentifier	
11	active_flag	int	

❖ **Ngành đào tạo (TBL\_major\_ref)**

*Bảng 9 Mô tả cấu trúc bảng cơ sở dữ liệu ngành đào tạo (TBL\_major\_ref)*

STT	Tên trường	Kiểu dữ liệu	Mô tả
1	major_id_rcd	varchar	Mã ngành đào tạo theo quy định của Bộ
2	major_name_VN	nvarchar	Tên ngành đào tạo bằng tiếng Việt
3	major_name_EN	nvarchar	Tên ngành đào tạo bằng tiếng anh
4	department_id	varchar	Mã bộ môn quản lý ngành đào tạo
5	founding_date	date	Thời gian mở ngành

6	description	nvarchar	Giới thiệu về ngành đào tạo
7	created_by_user_id	uniqueidentifier	
8	created_date_time	datetime	
9	lu_updated	datetime	
10	lu_user_id	uniqueidentifier	
11	active_flag	int	

❖ **Chuyên ngành (TBL\_specialization\_ref)**

***Bảng 10 Mô tả cấu trúc bảng cơ sở dữ liệu chuyên ngành  
(TBL\_specialization\_ref)***

STT	Tên trường	Kiểu dữ liệu	Mô tả
1	specialization_id_rcd	varchar	Mã chuyên ngành (Mã ngành + STT; VD 241 - chuyên ngành Trí tuệ nhân tạo và Khoa học dữ liệu)
2	specialization_name_VN	nvarchar	Tên chuyên ngành đào tạo, tiếng việt
3	specialization_name_EN	nvarchar	Tên chuyên ngành đào tạo, tiếng Anh
4	major_id	varchar	Thuộc Mã ngành đào tạo
5	founding_date	date	Thời gian xây dựng chuyên ngành
6	description	nvarchar	Giới thiệu về chuyên ngành đào tạo
7	created_by_user_id	uniqueidentifier	
8	created_date_time	datetime	
9	lu_updated	datetime	



10	lu_user_id	uniqueidentifier	
11	active_flag	int	

❖ **Hệ đào tạo (TBL\_degree\_ref)**

*Bảng 11 Mô tả cấu trúc bảng cơ sở dữ liệu hệ đào tạo (TNL\_degree\_ref)*

STT	Tên trường	Kiểu dữ liệu	Mô tả
1	degree_id_rcd	varchar	Mã các hệ đào tạo ('1'- Đại học CQ, '2'- Đại học liên thông, '5'-Đại học VLVH, 'V' - Văn bằng 2, 'H'- Cao học
2	degree_name	nvarchar	Tên hệ đào tạo
3	study_year_total	float	Tổng thời gian đào tạo
4	description	nvarchar	Mô tả về hệ đào tạo
5	created_by_user_id	uniqueidentifier	
6	created_date_time	datetime	
7	lu_updated	datetime	
8	lu_user_id	uniqueidentifier	
9	active_flag	int	

❖ **Lớp (TBL\_class\_ref)**

*Bảng 12 Mô tả cấu trúc bảng cơ sở dữ liệu lớp (TBL\_class\_ref)*

STT	Tên trường	Kiểu dữ liệu	Mô tả
1	degree_id_rcd	varchar	Mã các hệ đào tạo ('1'- Đại học CQ, '2'- Đại học liên thông,

			'5'-Đại học VLVH, 'V' - Văn bằng 2, 'H'- Cao học
2	degree_name	nvarchar	Tên hệ đào tạo
3	study_year_total	float	Tổng thời gian đào tạo
4	description	nvarchar	Mô tả về hệ đào tạo
5	created_by_user_id	uniqueidentifier	
6	created_date_time	datetime	
7	lu_updated	datetime	
8	lu_user_id	uniqueidentifier	
9	active_flag	int	

❖ **Khung CTĐT (TBL\_Leader\_Curriculum\_Framework)**

***Bảng 13 Mô tả cấu trúc bảng cơ sở dữ liệu khung CTĐT  
(TBL\_Leader\_Curriculum\_Framework)***

STT	Tên trường	Kiểu dữ liệu	Mô tả
1	Curriculum_Fram_id_rcd	varchar	Mã khung CTĐT
2	Curriculum_id_rcd	varchar	Mã CTĐT
3	Course_id_rcd	varchar	Mã học phần
4	Semester	char	Học kỳ (1-8,...)
5	active_flag	int	
6	created_by_user_id	uniqueidentifier	
7	created_date_time	datetime	
8	lu_updated	datetime	
9	lu_user_id	uniqueidentifier	

❖ **Phân công số lượng đồ án (TBL\_Leader\_Teacher\_Project)**

**Bảng 14 Mô tả cấu trúc bảng cơ sở dữ liệu phân công số lượng đồ án  
(TBL\_Leader\_Teacher\_Project)**

STT	Tên trường	Kiểu dữ liệu	Mô tả
1	Teacher_Pro_id_rcd	varchar	Mã phân công
2	academic_year	varchar	Năm học (2021-2022,..)
3	Semester	char	Học kỳ (1 hoặc 2)
4	Project_type	char	Loại đồ án (1-5) hoặc 6 là TTCN, 7 là ĐATN
5	class_id_rcd	varchar	Mã định danh của lớp(VD 124201,..), nếu là ĐATN thì không cần mã lớp
6	teacher_id	varchar	Mã giảng viên được phân công
7	Quantily	smallint	Số lượng tối đa được HD
8	Internship_id_rcd	varchar	Mã đợt thực tập nếu là TTCN, NULL trong trường hợp khác
9	active_flag	int	
10	created_by_user_id	uniqueidentifier	
11	created_date_time	datetime	
12	lu_updated	datetime	
13	lu_user_id	uniqueidentifier	

❖ **Danh mục chương trình đào tạo (TBL\_Leader\_Education\_Curriculum)**

**Bảng 15 Mô tả cấu trúc bảng cơ sở dữ liệu danh mục chương trình đào tạo  
(TBL\_Leader\_Education\_Curriculum)**

STT	Tên trường	Kiểu dữ liệu	Mô tả
-----	------------	--------------	-------

1	Curriculum_id_rcd	varchar	Mã CTĐT,
2	Degree_id	varchar	Mã các hệ đào tạo ('1'- Đại học CQ, '2'- Đại học liên thông, '5'-Đại học VLVH, 'V' - Đại học văn bằng 2, 'H'- Cao học
3	Major_id	varchar	Mã ngành đào tạo, theo quy định của bộ
4	Start_year	smallint	Năm bắt đầu đào tạo (dựa trên Degree_id, và Start_year suy ra khóa đào tạo K17, K19,...)
5	Program_type	nvarchar	Loại chương trình đào tạo: Đại trà, KSTN, 2+2,...
6	Credit_Total	smallint	Tổng số tín chỉ
7	Comment	nvarchar	Mô tả về CTĐT
8	Status_approval	nvarchar	Trạng thái phê duyệt: Đang chờ phê duyệt, Đã phê duyệt,..
9	Approval_time	datetime	Thời gian phê duyệt
10	teacher_id_leader	varchar	Mã người duyệt
11	active_flag	int	
12	created_by_user_id	uniqueidentifier	
13	created_date_time	datetime	
14	lu_updated	datetime	
15	lu_user_id	uniqueidentifier	

### ❖ Học phần

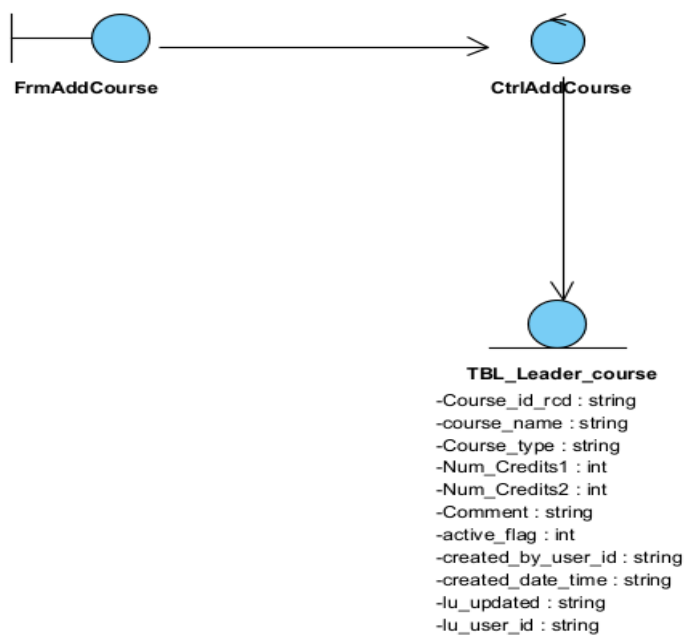
**Bảng 16 Mô tả cấu trúc bảng cơ sở dữ liệu học phần**

STT	Tên trường	Kiểu dữ liệu	Mô tả
1	Course_id_rcd	varchar	Mã học phần
2	course_name	nvarchar	Tên học phần
3	Course_type	varchar	Phân loại học phần theo các khối kiến thức: DC - Đại cương, CSN - Cơ sở ngành, CN- Chuyên ngành , DATT- Đồ án thực tập
4	Num_Credits1	smallint	Số tín chỉ lý thuyết
5	Num_Credits2	smallint	Số tín chỉ thực hành
6	Comment	nvarchar	Ghi chú
7	active_flag	int	
8	created_by_user_id	uniqueidentifier	
9	created_date_time	datetime	
10	lu_updated	datetime	
11	lu_user_id	uniqueidentifier	

### 3.2.2 Thiết kế lớp đối tượng

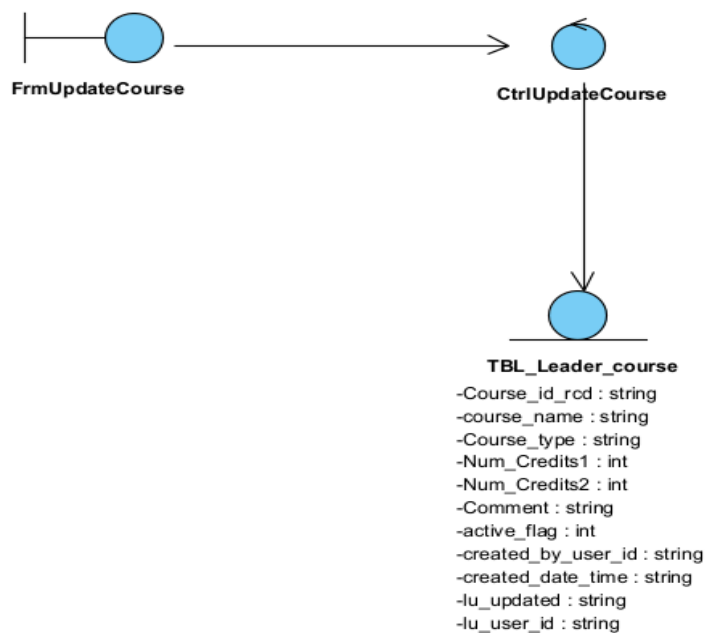
#### ❖ Biểu đồ lớp VOPC của Use Case quản lý học phần

- Thêm học phần



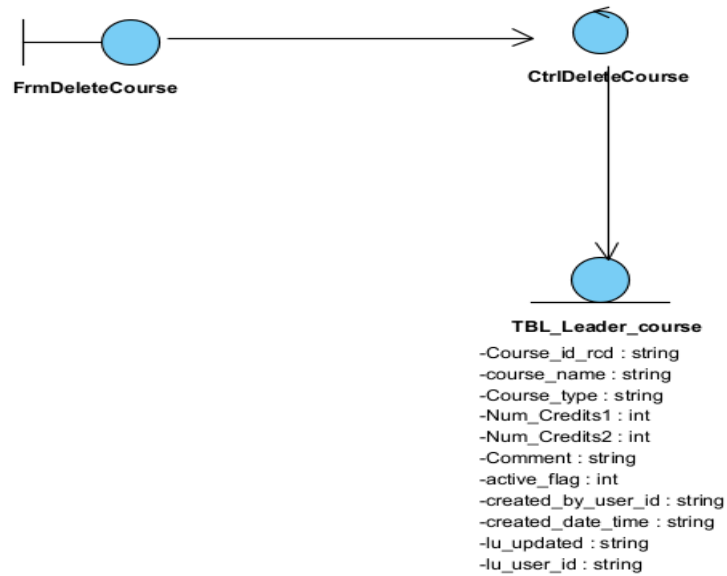
**Hình 7: Biểu đồ lớp VOPC thêm học phần**

- Sửa học phần



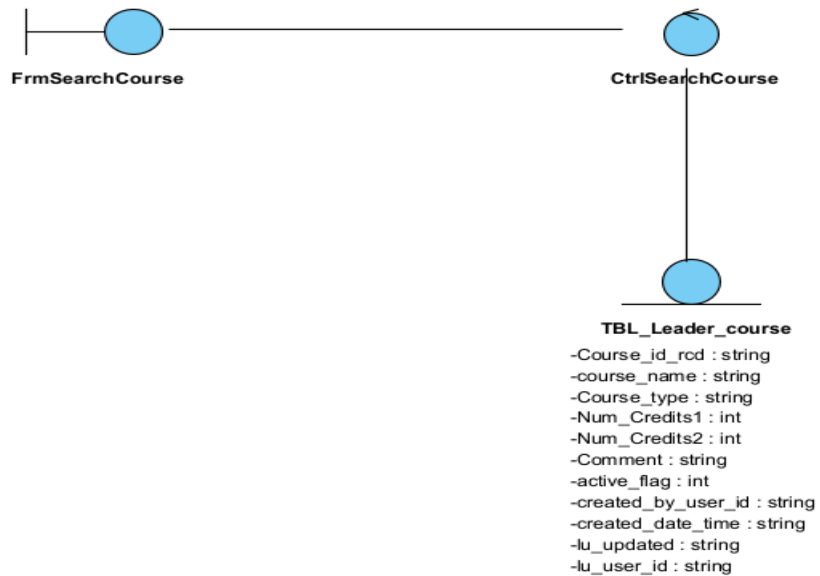
**Hình 8: Biểu đồ lớp VOPC sửa học phần**

- Xóa học phần



*Hình 9: Biểu đồ lớp VOPC xóa học phần*

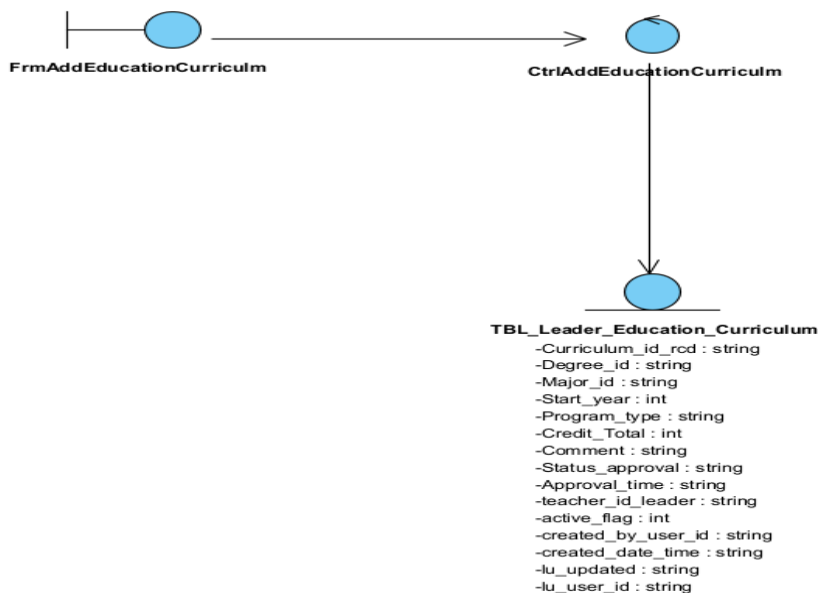
- Tìm kiếm học phần



**Hình 10: Biểu đồ lớp VOPC tìm kiếm học phần**

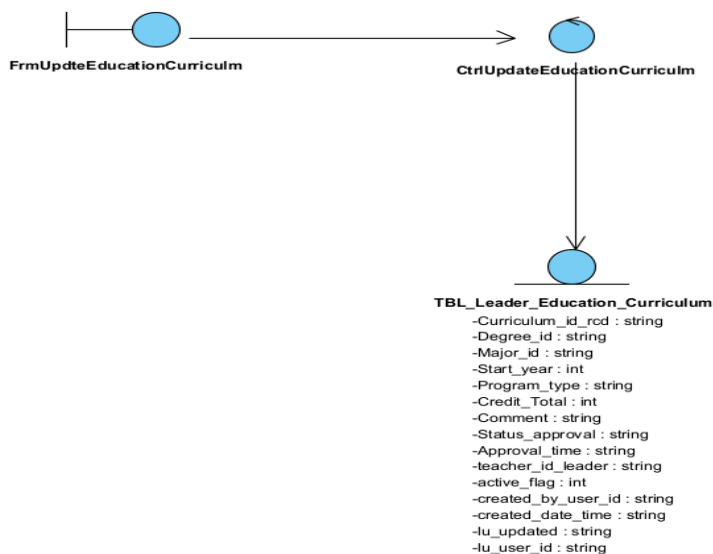
❖ Biểu đồ lớp VOPC của Use Case quản lý danh mục chương trình đào tạo

- Thêm danh mục chương trình đào tạo



**Hình 11: Biểu đồ lớp VOPC thêm danh mục chương trình đào tạo**

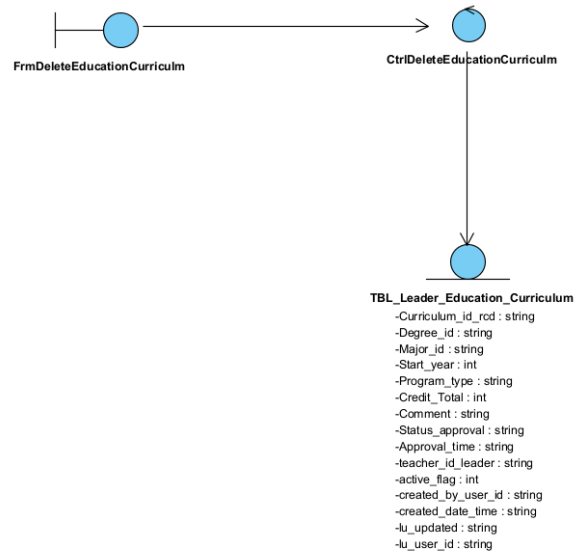
- Sửa danh mục chương trình đào tạo





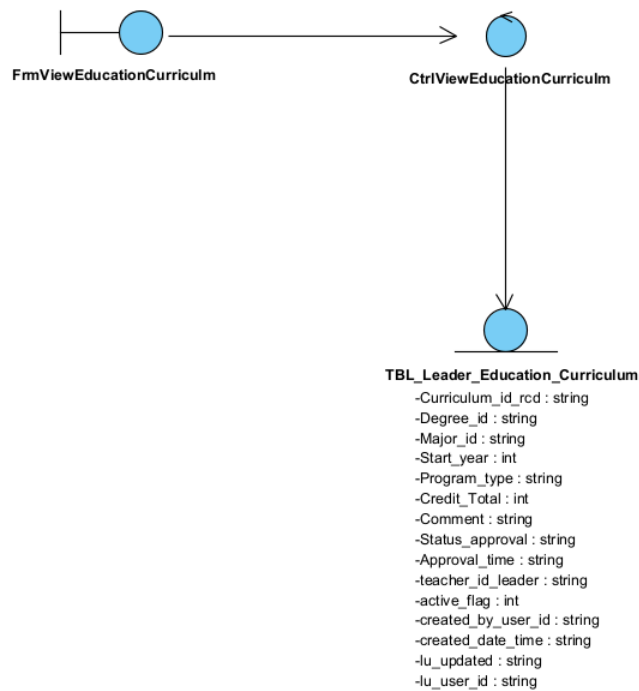
**Hình 12: Biểu đồ lớp VOPC sửa danh mục chương trình đào tạo**

- Xóa danh mục chương trình đào tạo



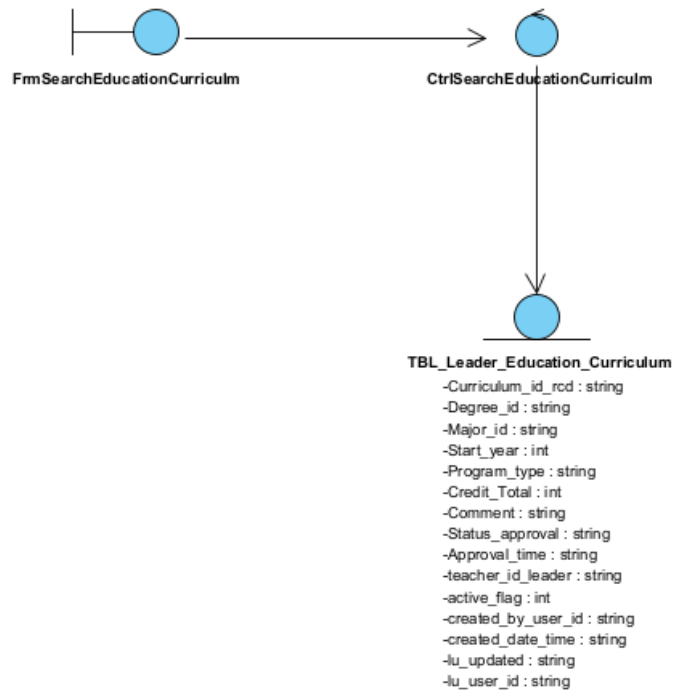
**Hình 13: Biểu đồ lớp VOPC xóa danh mục chương trình đào tạo**

- Xem danh mục chương trình đào tạo



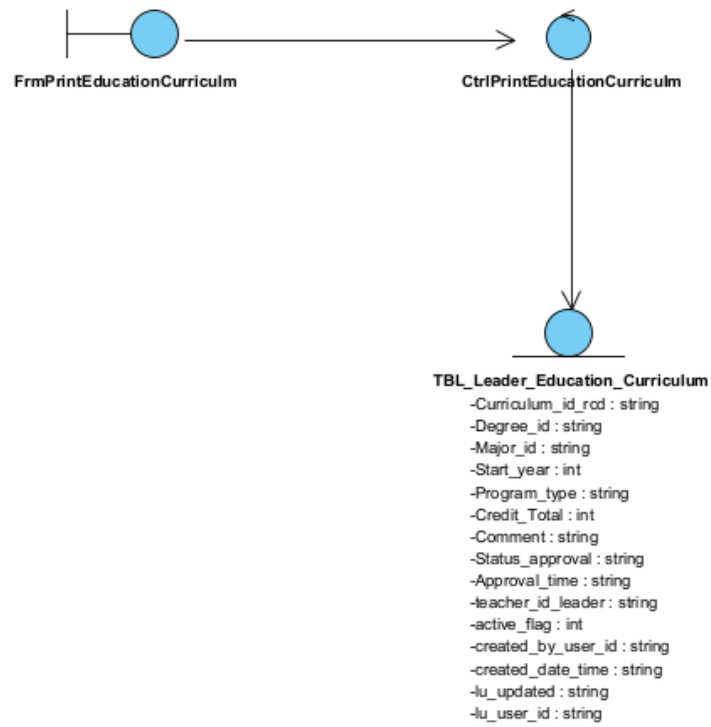
**Hình 14: Biểu đồ lớp VOPC xem danh mục chương trình đào tạo**

- Tìm kiếm danh mục chương trình đào tạo



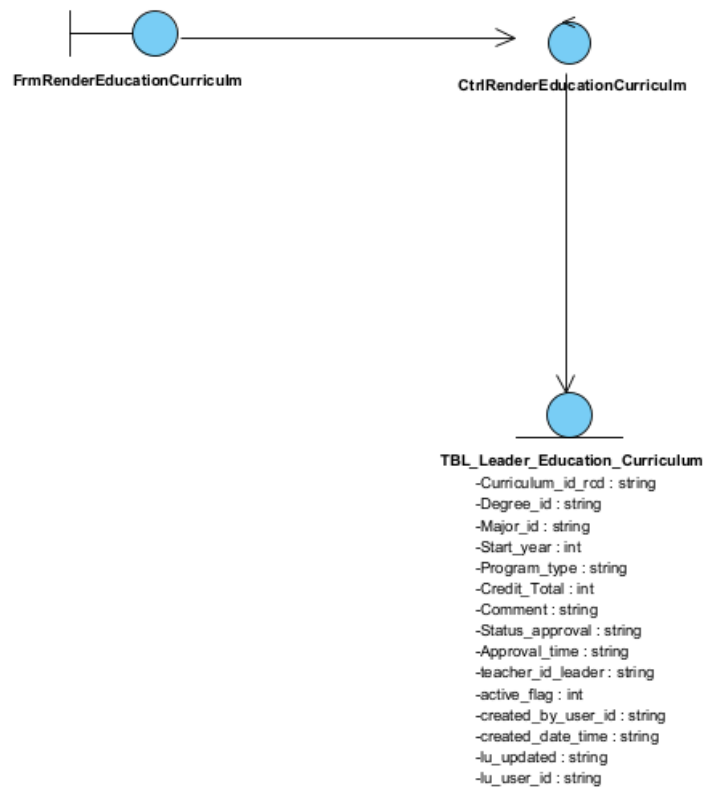
**Hình 15: Biểu đồ lớp VOPC tìm kiếm danh mục chương trình đào tạo**

- In danh mục chương trình đào tạo



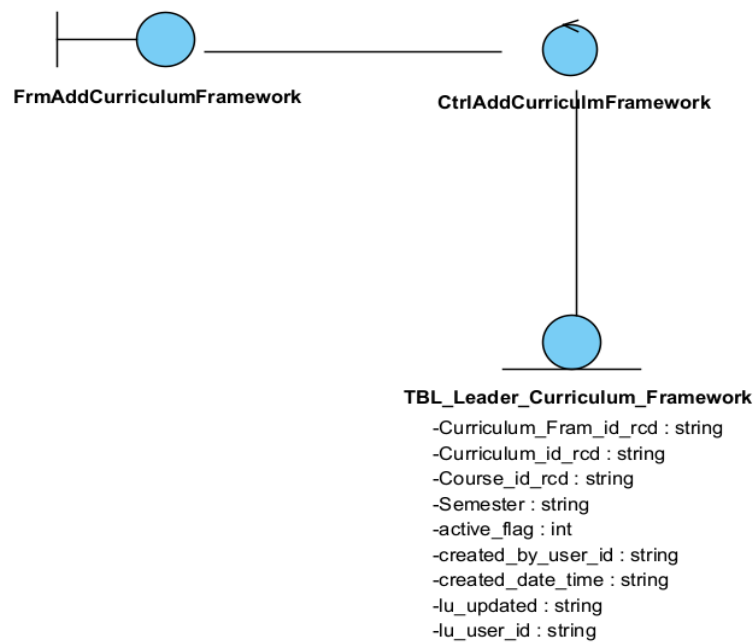
**Hình 16: Biểu đồ lớp VOPC in danh mục chương trình đào tạo**

- Kết xuất danh sách chương trình đào tạo



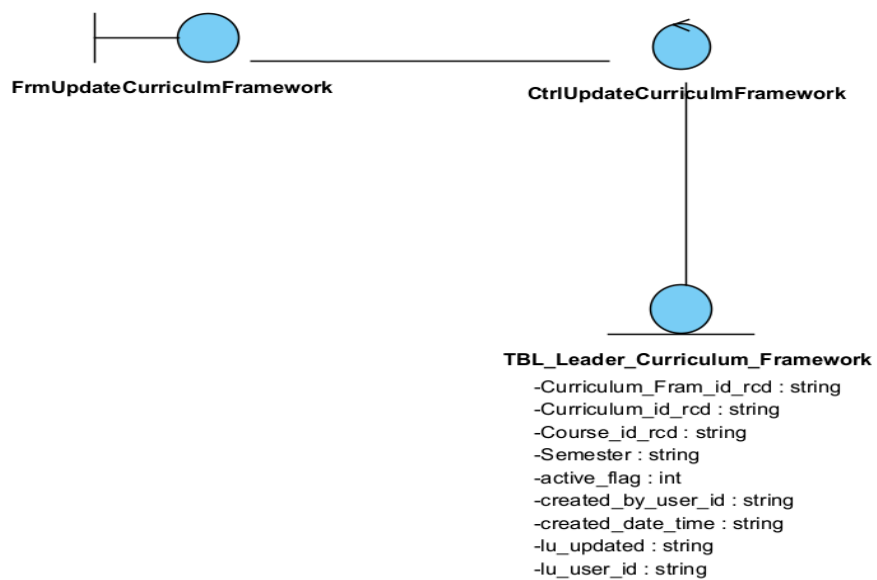
**Hình 17: Biểu đồ lớp VOPC kết xuất danh sách danh mục chương trình đào tạo**

- ❖ Biểu đồ lớp VOPC của Use Case quản lý khung chương trình đào tạo
  - Thêm khung chương trình đào tạo

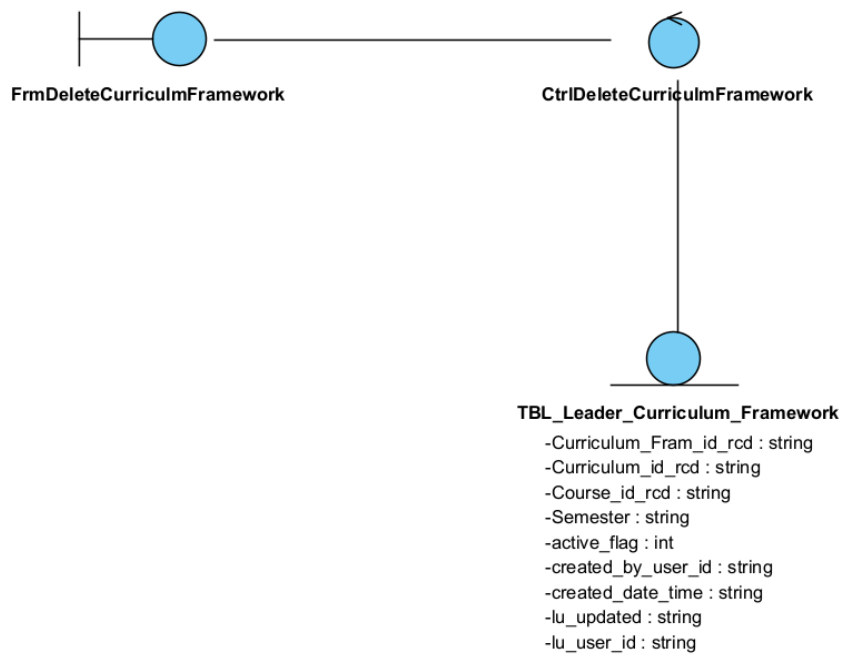


**Hình 18: Biểu đồ lớp VOPC thêm khung chương trình đào tạo**

- Sửa khung chương trình đào tạo

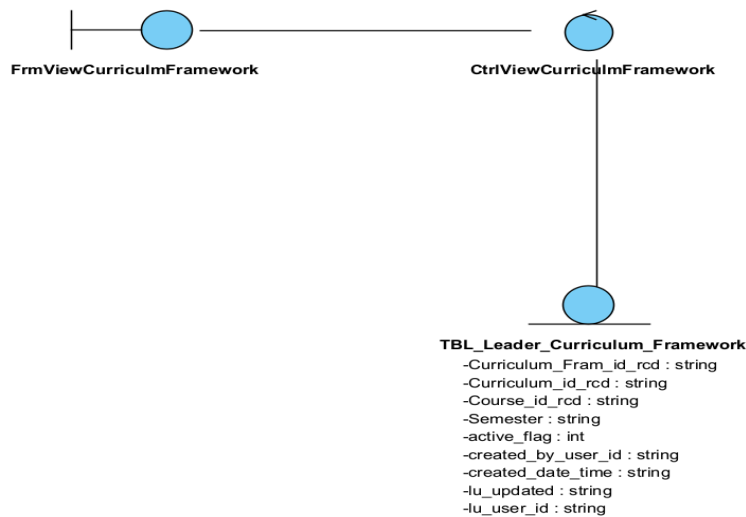


**Hình 19: Biểu đồ lớp VOPC sửa khung chương trình đào tạo**



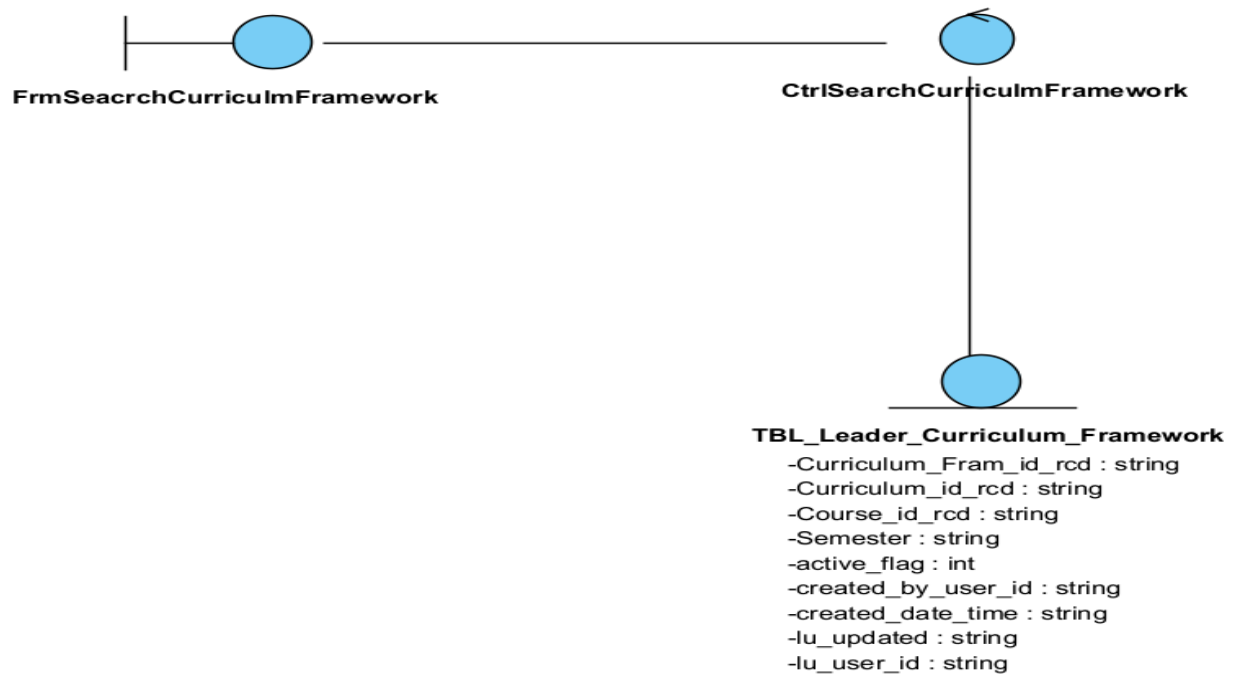
**Hình 20: Biểu đồ lớp VOPC xóa khung chương trình đào tạo**

- Xem khung chương trình đào tạo



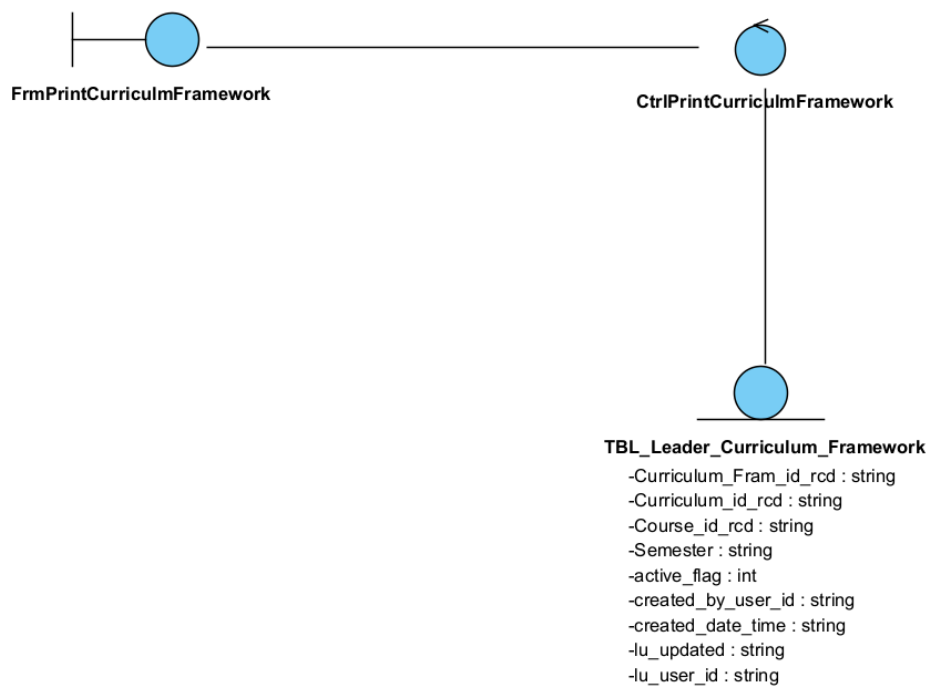
**Hình 21: Biểu đồ lớp VOPC xem khung chương trình đào tạo**

- Tìm kiếm khung chương trình đào tạo



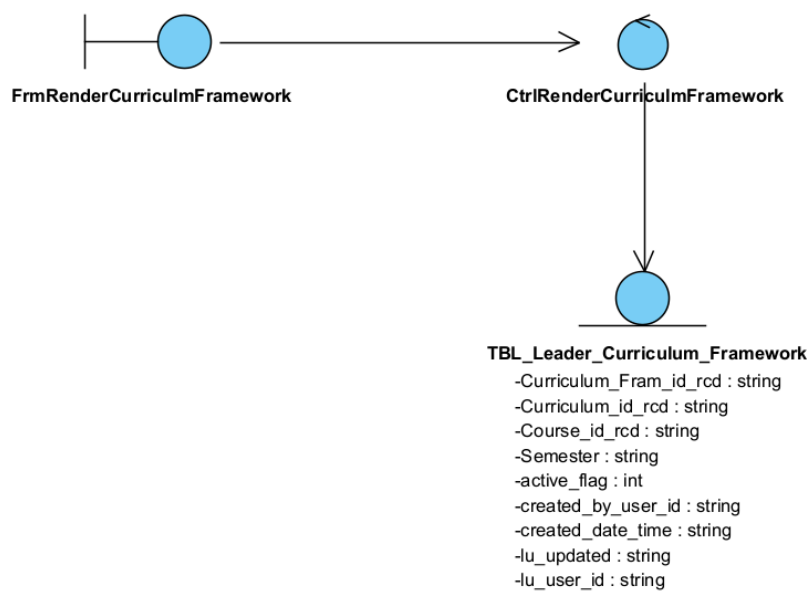
**Hình 22: Biểu đồ lớp VOPC tìm kiếm khung chương trình đào tạo**

- In khung chương trình đào tạo



**Hình 23: Biểu đồ lớp VOPC in khung chương trình đào tạo**

- Kết xuất khung chương trình đào tạo



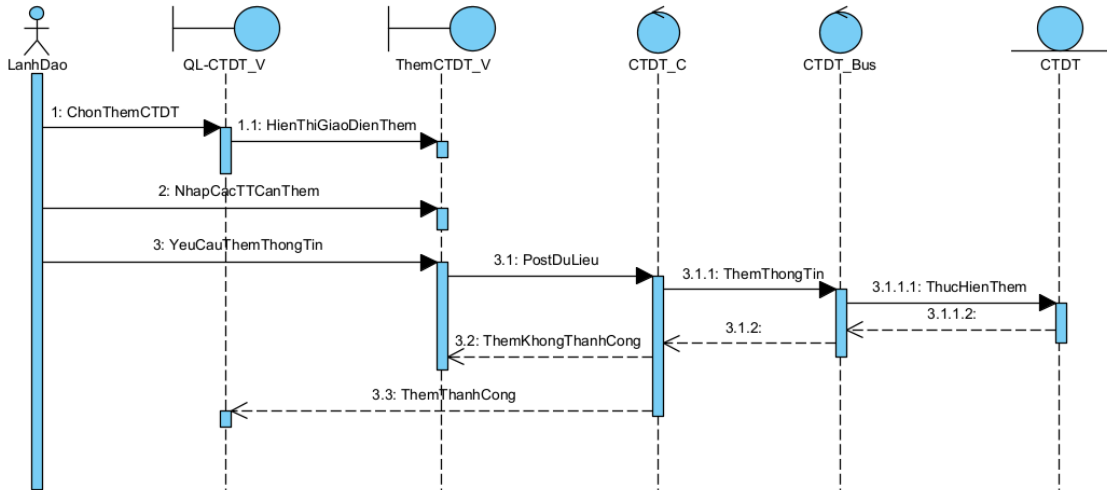
**Hình 24: Biểu đồ lớp VOPC kết xuất khung chương trình đào tạo**



### 2.3.1.1 Biểu đồ tuần tự

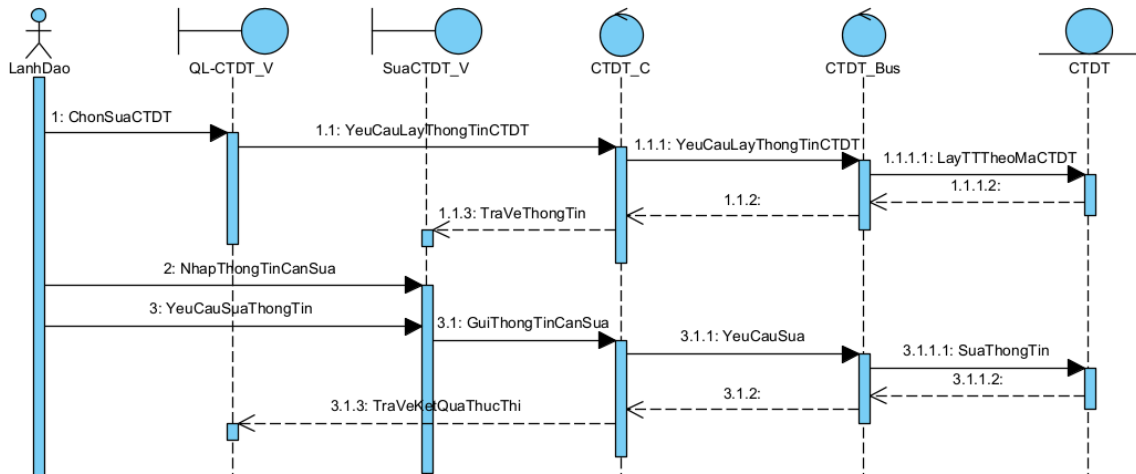
#### ❖ Biểu đồ tuần tự của Use Case quản lý khung chương trình đào tạo

- Thêm khung chương trình đào tạo



**Hình 25: Biểu đồ tuần tự thêm khung chương trình đào tạo**

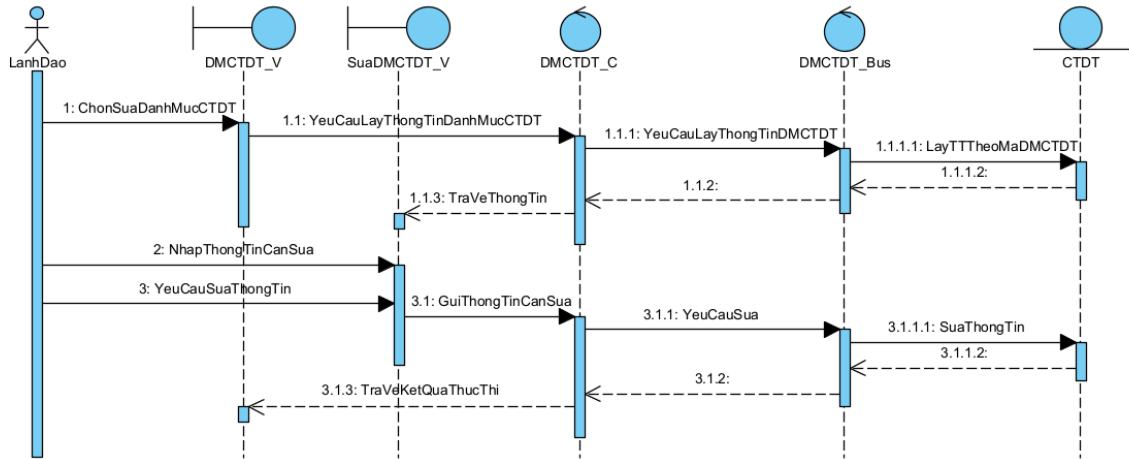
- Sửa khung chương trình đào tạo



**Hình 26: Biểu đồ tuần tự sửa khung chương trình đào tạo**

❖ Biểu đồ tuần tự của Use Case quản lý danh mục chương trình đào tạo

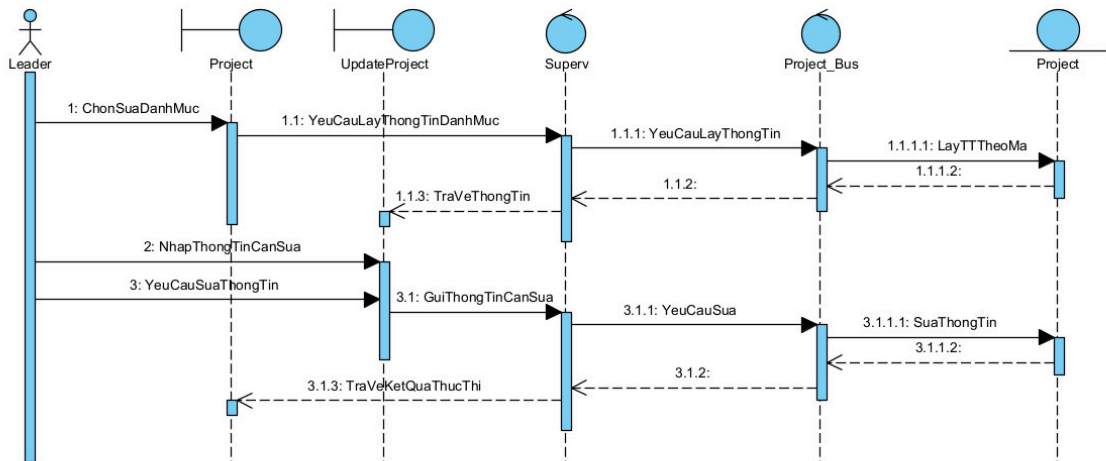
- Sửa danh mục chương trình đào tạo



**Hình 27: Biểu đồ tuần tự sửa danh mục chương trình đào tạo**

❖ Biểu đồ tuần tự của Use Case quản lý hướng dẫn đề án

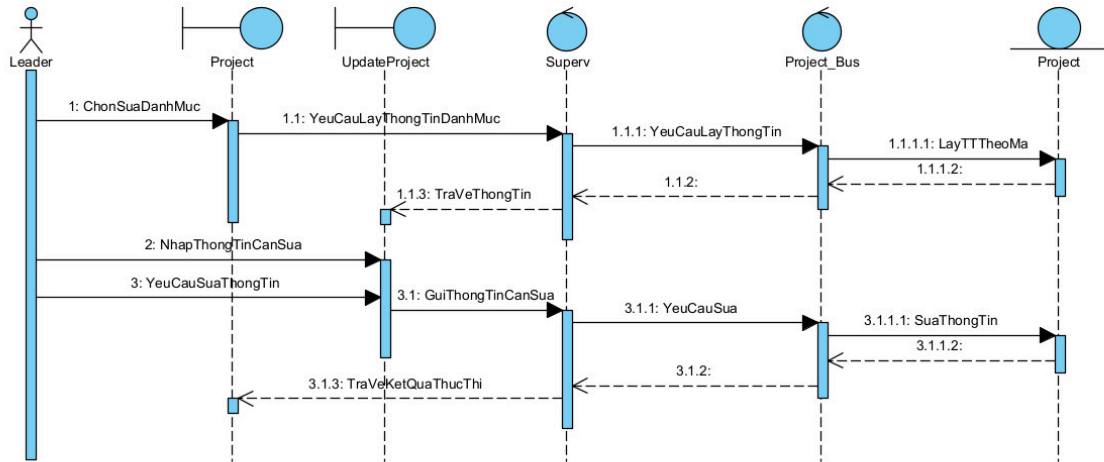
- quản lý hướng dẫn đề án



**Hình 28 Biểu đồ tuần tự sửa danh mục hướng dẫn đề án**

❖ Biểu đồ tuần tự của Use Case quản lý danh mục giáo viên

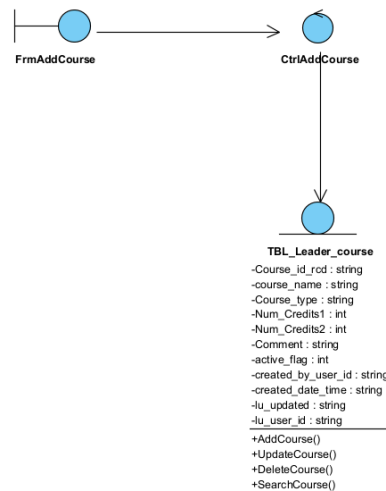
● Giáo viên chủ nhiệm



**Hình 29 Biểu đồ tuần tự sửa danh mục giáo viên chủ nhiệm**

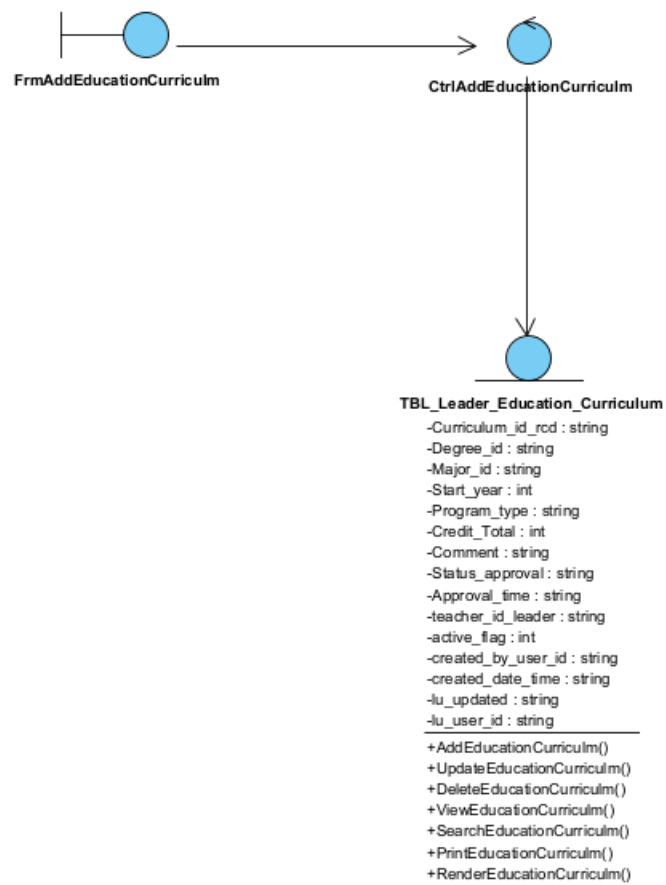
### 2.3.1.2 Biểu đồ lớp chi tiết

❖ Biểu đồ lớp chi tiết của Use Case quản lý học phần



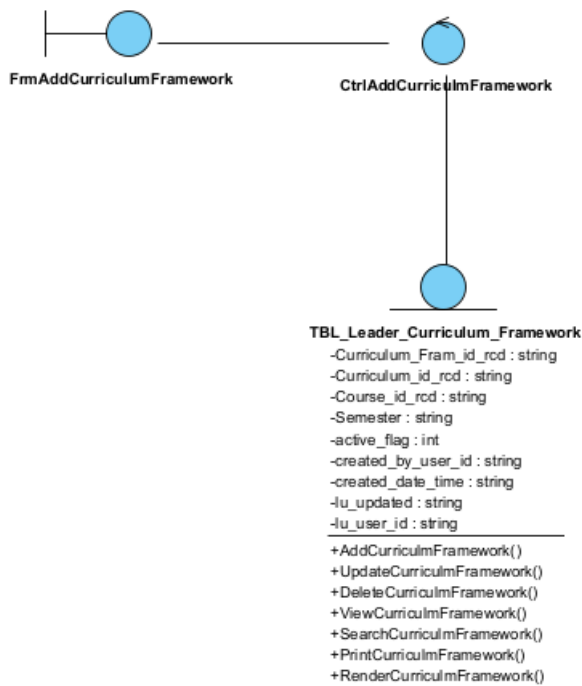
**Hình 30: Biểu đồ lớp chi tiết thêm học phần**

❖ Biểu đồ lớp chi tiết của Use Case quản lý danh mục chương trình đào tạo



**Hình 31: Biểu đồ lớp chi tiết thêm danh mục chương trình đào tạo**

❖ Biểu đồ lớp chi tiết của Use Case quản lý khung chương trình đào tạo



**Hình 32: Biểu đồ lớp chi tiết thêm khung chương trình đào tạo**

### 3.3 Thiết kế giao diện

- Giao diện quản lý chương trình đào tạo:

**Hình 33 Giao diện quản lý chương trình đào tạo**

- Giao diện thêm:

Thêm học phần

Khối kiến thức

Tên học phần\*

TCHT\*

Số tín chỉ TH\*

Số tiết thực hành\*

Số tiết thảo luận\*

Tổng tiết\*

Mã học phần\*

Học kỳ\*

TCHP\*

Số tín chỉ LT\*

Số tiết lý thuyết\*

Số tiết bài tập\*

Số tiết tự học\*

Hình thức thi\*

Làm mới

Đồng ý

Thoát

**Hình 34** Giao diện quản lý thêm học phần

– Giao diện quản lý đồ án:

Trang chủ

Quản lý đồ án

Thêm

Tìm kiếm

Tên đồ án

Loại đồ án

Tìm kiếm

	STT	Tên đồ án	Loại đồ án	Số tín 1	Số tín 2	Nhận xét	Tác vụ
<input type="checkbox"/>	1	ĐA5: Phát triển UD thương mại điện tử	DACN	30	15	lì	<a href="#">Chỉnh sửa</a> <a href="#">Xóa</a>
<input type="checkbox"/>	2	ĐA4: XDHT Web với PHP & MySQL	DACN	30	15		<a href="#">Chỉnh sửa</a> <a href="#">Xóa</a>
<input type="checkbox"/>	3	ĐA3: XD hệ thống website với mô hình MVC	DACN	30	15		<a href="#">Chỉnh sửa</a> <a href="#">Xóa</a>
<input type="checkbox"/>	4	ĐA2: TK Web với HTML5,CSS3,....	DACN	30	15		<a href="#">Chỉnh sửa</a> <a href="#">Xóa</a>
<input type="checkbox"/>	5	ĐA1: Xây dựng ứng dụng trên C Sharp	DACN	30	15		<a href="#">Chỉnh sửa</a> <a href="#">Xóa</a>

Tất cả

1

10

1 / 1

Hiện thị bản ghi từ 1 đến 5 trên tổng số 5

**Hình 35** Giao diện quản lý đồ án

– Giao diện phân công:

**Phân công**

Lớp: 101192

Lớp: 101192 Chuyên ngành: 2 Sĩ số: 29 + Thêm

<input type="checkbox"/>	STT	Mã GV	Tên giảng viên	Số lượng sinh viên GS	Tác vụ
<input type="checkbox"/>	1	4	Nguyễn Duy Tân		10 <span>↑</span> <span>↓</span> <span>🗑</span>
<input type="checkbox"/>	2	5	Chu Thị Minh Huệ		10 <span>↑</span> <span>↓</span> <span>🗑</span>
<input type="checkbox"/>	3	1	Nguyễn Văn Quyết		9 <span>↑</span> <span>↓</span> <span>🗑</span>

1 / 1 X Xác nhận

**Hình 36 Giao diện phân công**

– Giao diện giảng viên:

**Tìm kiếm**

Tên giảng viên  🔍 Tìm kiếm

Công nghệ phần mềm Khoa học máy tính

STT	Họ và tên	Chức vụ
1	Nguyễn Văn Quyết	
2	Chu Thị Minh Huệ	
3	Nguyễn Minh Tiến	
4	Nguyễn Văn Hậu	

1 / 1 X Hiện thị bản ghi từ 1 đến 4 trên tổng số 4

**Hình 37 Giao diện quản lý giảng viên**

## KẾT LUẬN

- ❖ Những điều đạt được:
  - o Hiểu hơn về lập trình web Api, angular
  - o Hoàn thiện các bảng cơ sở dữ liệu
- ❖ Hạn chế:
  - o Do điều kiện của các nhân tố khách quan nên chưa thể hoàn thiện đầy đủ các chức năng.



## **TÀI LIỆU THAM KHẢO**

- [1] Đề cương môn lập trình cơ bản với C# - Trường ĐHSPKTHY
- [2] Đề cương môn phân tích thiết kế hướng đối tượng - Trường ĐHSPKTHY
- [3] Đề cương môn Web API - Trường ĐHSPKTHY
- [4] Đề cương môn phân tích thiết kế hướng đối tượng - Trường ĐHSPKTHY