

BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG

---



# ĐỒ ÁN TỐT NGHIỆP

NGÀNH : CÔNG NGHỆ THÔNG TIN

Sinh viên : Nguyễn Thành Dương

Giảng viên hướng dẫn : Ths. Nguyễn Thị Xuân Hương

HẢI PHÒNG – 2021

**BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG**

-----

**XÂY DỰNG WEBSITE KHOA CNTT TRƯỜNG ĐẠI  
HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG**

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY  
NGÀNH: CÔNG NGHỆ THÔNG TIN**

**Sinh viên : Nguyễn Thành Dương  
Giảng viên hướng dẫn : Ths. Nguyễn Thị Xuân Hương**

**HẢI PHÒNG – 2021**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC QUẢN LÝ VÀ CÔNG NGHỆ HẢI PHÒNG**

---

**NHIỆM VỤ ĐỀ TÀI TỐT NGHIỆP**

**Sinh viên: Nguyễn Thành Dương**

**Mã SV: 1612101015**

**Lớp : CT2001C**

**Ngành : Công Nghệ Thông Tin**

**Tên đề tài: Xây dựng website Khoa Công Nghệ Thông Tin - Trường  
Đại học Quản lý và Công nghệ Hải Phòng.**

# NHIỆM VỤ ĐỀ TÀI

## 1. Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp

- Tìm hiểu các thông tin về Khoa Công nghệ Thông Tin Trường Đại học Quản Lý và Công Nghệ Hải Phòng.
- Tìm hiểu ngôn ngữ lập trình PHP, hệ quản trị CSDL MySQL.
- Xây dựng trang web cho Khoa Công nghệ Thông tin Trường Đại học Quản Lý và Công nghệ Hải Phòng.
  - + Phân tích thiết kế hệ thống
  - + Xây dựng Front – End, Back - End
- Demo website thực nghiệm.

## 2. Các tài liệu, số liệu cần thiết

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

## 3. Địa điểm thực tập tốt nghiệp

.....

## **CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP**

**Họ và tên : Nguyễn Thị Xuân Hương**

**Học hàm, học vị : Thạc sỹ**

**Cơ quan công tác : Trường Đại học Quản lý và Công nghệ Hải Phòng**

**Nội dung hướng dẫn: Xây dựng Website Khoa Công nghệ Thông tin**

**Trường Đại học Quản lý và Công nghệ Hải Phòng.**

Đề tài tốt nghiệp được giao ngày .... tháng 12 năm 2021

Yêu cầu phải hoàn thành xong trước ngày 31 tháng 12 năm 2021

Đã nhận nhiệm vụ ĐTTN

*Sinh viên*

Đã giao nhiệm vụ ĐTTN

*Giảng viên hướng dẫn*

*Hải Phòng, ngày tháng năm 2021*

**TRƯỞNG KHOA**

**CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**Độc lập - Tự do - Hạnh phúc**

**PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN TỐT NGHIỆP**

Họ và tên giảng viên: .....

Đơn vị công tác: .....

Họ và tên sinh viên: ..... Ngành:.....

Nội dung hướng dẫn: .....

**1. Tinh thần thái độ của sinh viên trong quá trình làm đề tài tốt nghiệp**

.....  
.....  
.....  
.....  
.....

**2. Đánh giá chất lượng của đề án/khóa luận (so với nội dung yêu cầu đó đề ra trong nhiệm vụ Đ.T. T.N trên các mặt lý luận, thực tiễn, tính toán số liệu...)**

.....  
.....  
.....  
.....  
.....

**3. Ý kiến của giảng viên hướng dẫn tốt nghiệp**

Đạt  Không đạt  Điểm:.....

*Hải Phòng, ngày ... tháng ... năm 2021*

**Giảng viên hướng dẫn**

**CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM**  
**Độc lập - Tự do - Hạnh phúc**

**PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN CHĂM PHẢN BIỆN**

Họ và tên giảng viên: .....

Đơn vị công tác: .....

Họ và tên sinh viên: ..... Ngành: .....

Đề tài tốt nghiệp: .....

.....

**1. Phần nhận xét của giảng viên chăm phản biện**

.....

.....

.....

.....

.....

.....

**2. Những mặt còn hạn chế**

.....

.....

.....

.....

.....

.....

.....

**3. Ý kiến của giảng viên chăm phản biện**

Đạt  Không đạt  Điểm:.....

*Hải Phòng, ngày ... tháng ... năm 2021*

**Giảng viên chăm phản biện**

## LỜI CẢM ƠN

Trong thời gian làm đồ án tốt nghiệp em đã nhận được nhiều sự giúp đỡ, đóng góp ý kiến và chỉ bảo nhiệt tình của thầy cô, gia đình và bạn bè. Em xin gửi lời cảm ơn chân thành đến Ths. Nguyễn Thị Xuân Hương, giảng viên Bộ môn Công Nghệ Thông Tin - Trường Đại học Quản lý và Công nghệ Hải Phòng người đã tận tình hướng dẫn, chỉ bảo em trong suốt quá trình làm khoá luận.

Em cũng xin chân thành cảm ơn các thầy cô giáo trong trường Đại học Quản lý và Công nghệ Hải Phòng nói chung, các thầy cô trong Bộ môn Công Nghệ Thông Tin nói riêng đã dạy dỗ cho em kiến thức về các môn đại cương cũng như các môn chuyên ngành, giúp em có được cơ sở lý thuyết vững vàng và tạo điều kiện giúp đỡ em trong suốt quá trình học tập để em có được ngày hôm nay.

Cuối cùng, em xin chân thành cảm ơn gia đình và bạn bè, đã luôn tạo điều kiện, quan tâm, giúp đỡ, động viên em trong suốt quá trình học tập và hoàn thành khoá luận tốt nghiệp.

**Em xin chân thành cảm ơn!**

*Hải Phòng, ngày...tháng...năm 2021*

**Sinh viên**

**Nguyễn Thành Dương**



## LỜI MỞ ĐẦU

Ngày nay, ứng dụng công nghệ thông tin và việc tin học hóa được xem là một trong những yếu tố mang tính quyết định trong hoạt động của các chính phủ, tổ chức, cũng như của các công ty, doanh nghiệp. Điều này đóng vai trò hết sức quan trọng và là một trong những nền tảng để tạo ra những bước đột phá mạnh mẽ.

Cùng với sự phát triển không ngừng về kỹ thuật máy tính và mạng Internet, công nghệ thông tin đã chứng minh được rằng đây là một trong những ngành mũi nhọn, lần lượt chinh phục hết đỉnh cao này đến đỉnh cao khác. Trên thực tế, mạng Internet là một trong những sản phẩm có giá trị và mức độ ảnh hưởng to lớn và trở thành một công cụ không thể thiếu, là nền tảng chính để giúp truyền tải, trao đổi thông tin trên toàn cầu.

Ngày nay, mọi việc liên quan đến thông tin trở nên thật dễ dàng cho người dùng, chỉ cần có một máy tính kết nối Internet và một dòng dữ liệu truy tìm thì gần như lập tức, cả thế giới về vấn đề mà bạn đang quan tâm sẽ hiện ra, có đầy đủ thông tin, hình ảnh và thậm chí đôi lúc có cả những âm thanh nếu bạn cần, v.v.

Nhận thức được nhu cầu tìm hiểu thông tin, giải trí của xã hội là sự ra đời của hàng loạt website cho các mục đích thương mại, giải trí, tin tức...Để đáp ứng với việc cập nhật thông tin hàng ngày, tình hình xã hội, chính trị, thời sự, và sức khỏe... thì website tin tức ra đời là một nhu cầu tất yếu.

Nhận thấy khoa CNTT trường mình vẫn chưa có website chính thức mà sự cần thiết của một website trong việc cung cấp thông tin giúp các sinh viên có thể nắm bắt tin tức, sự kiện hay thông báo từ khoa là một vấn đề cần thiết. Do đó, em đã vận dụng ngôn ngữ lập trình PHP, hệ quản trị cơ sở dữ liệu MySQL để xây dựng Website về khoa CNTT – Trường Đại học Quản Lý & Công nghệ Hải Phòng.

Với sự hướng dẫn tận tình của cô giáo Nguyễn Thị Xuân Hương em đã hoàn thành đồ án này. Tuy nhiên sẽ không tránh khỏi những thiếu sót. Em rất mong nhận được những ý kiến đóng góp quý báu của Thầy Cô. Em xin chân thành cảm ơn.

## BẢNG CÁC KÝ HIỆU, CHỮ VIẾT TẮT

STT	Từ viết tắt	Tên đầy đủ	Giải thích
1	CSDL	Cơ sở dữ liệu	
2	SEO	Search Engine Optimization	Tối ưu hoá công cụ tìm kiếm
3	HTML	Hypertext Markup Language	Ngôn ngữ đánh dấu siêu văn bản
4	CSS	Cascading Style Sheet	Ngôn ngữ kịch bản quy định cách trình bày của thẻ HTML
5	PHP	Hypertext Preprocessor	Ngôn ngữ lập trình PHP
6	OOP	Object-Oriented Programming	Lập trình hướng đối tượng
7	CNTT	Công nghệ thông tin	
8	WWW	World Wide Web	Mạng lưới toàn cầu

# MỤC LỤC

<b>CHƯƠNG 1: TỔNG QUAN</b> .....	<b>16</b>
1.1 Tổng quan về World Wide Web .....	16
1.1.1 Khái niệm.....	16
1.1.2 Lợi ích của WWW mang lại .....	16
1.1.3 Nguyên tắc hoạt động .....	17
1.1.4 Các chức năng chính.....	17
1.1.5 Sự phát triển của WWW trong thời đại ngày nay.....	17
1.1.6 Tầm quan trọng của WWW .....	18
1.2 Tổng quan về HTML – CSS .....	18
1.2.1 Tổng quan về HTML .....	18
1.2.2. Tổng quan về CSS .....	21
1.3 Ngôn ngữ PHP .....	23
1.3.1 Định nghĩa PHP .....	23
1.3.2 Lý do nên dùng PHP .....	24
1.3.3 Cách thức hoạt động của PHP .....	24
1.3.4 Tổng quan về PHP .....	25
1.3.5 Hàm trong PHP .....	26
1.3.6 Biểu mẫu PHP.....	27
1.3.7 PHP OOP (Object-Oriented Programming.) .....	27
1.3.8 Session và Cookie.....	27
1.4 Hệ quản trị CSDL MySQL .....	28
1.4.1 Giới thiệu về My SQL .....	28
1.4.2 Mục đích sử dụng CSDL .....	28
1.5 Bootstrap .....	29
1.5.1 Định nghĩa Bootstrap .....	29
1.5.2 Lịch sử phát triển của Bootstrap .....	30
1.5.3 Lý do nên sử dụng Bootstrap .....	30
1.5.4 Cấu trúc và Tính năng của Bootstrap.....	31
1.6 Thư viện jQuery .....	32
1.6.1 Định nghĩa.....	32
1.6.2 Tại sao nên chọn jQuery .....	32
1.6.3 Cài đặt tích hợp jQuery vào website.....	33
1.6.4 Cú pháp jQuery .....	33
1.7 Mô hình MVC .....	34
1.7.1 MVC là gì?.....	34
1.7.2 Các thành phần trong MVC .....	34
1.7.3 MVC làm việc như thế nào?.....	35
1.7.4 Ưu điểm và nhược điểm của MVC.....	36
1.8 Ứng dụng Laragon .....	36
1.8.1 Định nghĩa.....	36
1.8.2 Lý do nên sử dụng Laragon thay XAMPP .....	36
<b>CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG</b> .....	<b>37</b>
2.1 Phân tích thiết kế hướng đối tượng UML .....	37

2.1.1	Khái niệm phân tích thiết kế hướng đối tượng .....	37
2.1.2	Khái niệm UML.....	37
2.1.3	Phát biểu bài toán.....	39
2.2	Các yêu cầu phi chức năng.....	42
2.3	Phân tích thiết kế hệ thống .....	42
2.3.1	Thiết kế hệ thống .....	42
2.3.2	Sơ đồ phân cấp chức năng .....	44
2.4	Biểu đồ Use case .....	45
2.4.1	Biểu đồ Usecase tổng quát.....	46
2.4.2	Use case đăng nhập.....	46
2.4.3	Use case quản lý người dùng .....	47
2.4.4	Use case quản lý chuyên mục .....	50
2.4.5	Use case quản lý bài viết .....	54
2.4.6	Use case quản lý giảng viên.....	57
2.4.7	Use case quản lý chuyên ngành .....	60
2.4.9	Use case quản lý liên hệ.....	67
2.4.10	Use case gửi liên hệ .....	68
2.4.11	Use case xem bài viết .....	69
2.5	Biểu đồ tuần tự .....	70
2.5.1	Biểu đồ tuần tự chức năng đăng nhập.....	70
2.5.2	Biểu đồ tuần tự quản lý chuyên mục .....	71
2.5.3	Biểu đồ tuần tự quản lý bài viết.....	72
2.5.4	Biểu đồ tuần tự quản lý giảng viên .....	73
2.5.5	Biểu đồ tuần tự quản lý slide .....	74
2.5.6	Biểu đồ tuần tự quản lý chuyên ngành .....	75
2.5.7	Biểu đồ tuần tự quản lý Users.....	76
2.5.8	Biểu đồ tuần tự quản lý Liên hệ.....	77
2.5.9	Biểu đồ tuần tự gửi Liên hệ .....	77
2.5.10	Biểu đồ tuần tự xem bài viết.....	78
2.6	Biểu đồ cộng tác .....	78
2.6.1	Biểu đồ cộng tác chức năng đăng nhập .....	79
2.6.2	Biểu đồ cộng tác chức năng hiển thị danh sách modul.....	79
2.6.2	Biểu đồ cộng tác chức năng thêm modul.....	80
2.6.3	Biểu đồ cộng tác chức năng sửa modul .....	80
2.6.4	Biểu đồ cộng tác chức năng xóa modul.....	81
2.7	Biểu đồ hoạt động .....	81
2.7.1	Biểu đồ hoạt động chức năng thêm chuyên mục .....	82
2.7.2	Biểu đồ hoạt động chức năng sửa chuyên mục .....	82
2.7.3	Biểu đồ hoạt động chức năng xóa chuyên mục .....	83
2.7.4	Biểu đồ hoạt động chức năng thêm bài viết .....	83
2.7.5	Biểu đồ hoạt động chức năng sửa bài viết.....	84
2.7.6	Biểu đồ hoạt động chức năng xóa bài viết.....	84
2.7.7	Biểu đồ hoạt động chức năng thêm giảng viên.....	85
2.7.8	Biểu đồ hoạt động chức năng sửa giảng viên .....	85
2.7.9	Biểu đồ hoạt động chức năng xóa giảng viên.....	86
2.7.10	Biểu đồ hoạt động chức năng thêm chuyên ngành .....	86

2.7.11 Biểu đồ hoạt động chức năng sửa chuyên ngành .....	87
2.7.12 Biểu đồ hoạt động chức năng xóa chuyên ngành .....	87
2.7.13 Biểu đồ hoạt động chức năng thêm slide.....	88
2.7.14 Biểu đồ hoạt động chức năng sửa slide .....	88
2.7.15 Biểu đồ hoạt động chức năng xóa slide .....	89
2.7.16 Biểu đồ hoạt động chức năng phân quyền người dùng .....	89
2.7.17 Biểu đồ hoạt động chức năng sửa thông tin người dùng .....	90
2.7.18 Biểu đồ hoạt động chức năng khóa tài khoản người dùng .....	90
2.8 Biểu đồ lớp .....	91
2.8.1 Biểu đồ lớp Use case Đăng nhập .....	91
2.8.2 Biểu đồ lớp Use case quản lý người dùng .....	92
2.8.3 Biểu đồ lớp Use case quản lý chuyên mục .....	92
2.8.4 Biểu đồ lớp Use case quản lý bài viết.....	93
2.8.5 Biểu đồ lớp Use case quản lý giảng viên.....	93
2.8.6 Biểu đồ lớp Use case quản lý chuyên ngành .....	94
2.8.7 Biểu đồ lớp Use case quản lý slide .....	94
2.8.9 Biểu đồ lớp Use case quản lý liên hệ.....	95
2.8.10 Biểu đồ lớp Use case gửi liên hệ .....	95
2.8.11 Biểu đồ lớp Use case xem bài viết.....	96
2.9 Thiết kế cơ sở dữ liệu.....	96
2.9.1 Danh sách các bảng.....	96
2.9.2 Bảng Users .....	97
2.9.3 Bảng Major .....	97
2.9.4 Bảng Slide.....	98
2.9.5 Bảng Blog .....	98
2.9.6 Bảng News.....	99
2.9.7 Bảng Lecturer .....	99
2.9.8 Bảng Contact.....	100
2.9.9 Mô hình CSDL.....	101
<b>CHƯƠNG 3. DEMO WEBSITE.....</b>	<b>102</b>
3.1 Cấu trúc thư mục .....	102
3.1.1 App.....	102
3.1.2 Config .....	103
3.1.3 Public .....	103
3.1.4 Core.....	103
3.1.5 Htaccess .....	103
3.1.6 Composer .....	103
3.2 Demo giao diện quản trị viên .....	104
3.2.1 Giao diện đăng nhập .....	104
3.2.2 Giao diện quản lý Slide.....	105
3.2.3 Giao diện quản lý người dùng .....	106
3.2.4 Giao diện quản lý chuyên ngành .....	107
3.2.5 Giao diện quản lý giảng viên .....	108
3.2.6 Giao diện quản lý chuyên mục .....	109
3.2.7 Giao diện quản lý bài viết.....	111
3.2.8 Giao diện quản lý liên hệ .....	112

3.3 Demo giao diện người dùng.....	113
3.3.1 Giao diện trang chủ.....	113
3.3.2 Giao diện danh sách bài viết.....	114
3.3.3 Giao diện chi tiết bài viết.....	115
3.3.4 Giao diện trang liên hệ.....	116
3.3.5 Giao diện website trên thiết bị di động.....	117
KẾT LUẬN .....	118
TÀI LIỆU THAM KHẢO .....	119

## DANH MỤC HÌNH VẼ

Hình 1.1: Cấu trúc của file HTML .....	19
Hình 1.2: Sơ đồ hoạt động của PHP .....	24
Hình 1.3: Cấu trúc mô hình MVC .....	34
Hình 1.4: Cách hoạt động của mô hình MVC .....	35
Hình 2.1: Các phần tử của UML .....	37
Hình 2.2: Sơ đồ phân rã chức năng .....	44
Hình 2.3: Biểu đồ Usecase tổng quát .....	46
Hình 2.4: Biểu đồ Use case đăng nhập.....	46
Hình 2.5: Biểu đồ Use case quản lý người dùng.....	47
Hình 2.6: Biểu đồ Use case quản lý chuyên mục.....	50
Hình 2.7: Biểu đồ Use case quản lý bài viết .....	54
Hình 2.8: Biểu đồ Use case quản lý giảng viên.....	57
Hình 2.9: Biểu đồ Use case quản lý chuyên ngành.....	60
Hình 2.10: Biểu đồ Use case quản lý slide.....	64
Hình 2.11: Biểu đồ Use case quản lý liên hệ.....	67
Hình 2.12: Biểu đồ Use case gửi liên hệ .....	68
Hình 2.13: Biểu đồ Use case xem bài viết .....	69
Hình 2.14: Biểu đồ tuần tự chức năng đăng nhập.....	70
Hình 2.15: Biểu đồ tuần tự quản lý chuyên mục.....	71
Hình 2.16: Biểu đồ tuần tự quản lý bài viết .....	72
Hình 2.17: Biểu đồ tuần tự quản lý giảng viên .....	73
Hình 2.18: Biểu đồ tuần tự quản lý slide.....	74
Hình 2.19: Biểu đồ tuần tự quản lý chuyên ngành.....	75
Hình 2.20: Biểu đồ tuần tự quản lý Users .....	76
Hình 2.21: Biểu đồ tuần tự quản lý liên hệ .....	77
Hình 2.22: Biểu đồ tuần tự gửi liên hệ .....	77
Hình 2.23: Biểu đồ tuần tự xem bài viết .....	78
Hình 2.24: Biểu đồ cộng tác chức năng đăng nhập.....	79
Hình 2.25: Biểu đồ cộng tác chức năng hiển thị danh sách modul .....	79
Hình 2.26: Biểu đồ cộng tác chức năng thêm modul .....	80
Hình 2.27: Biểu đồ cộng tác chức năng sửa modul.....	80
Hình 2.28: Biểu đồ cộng tác chức năng xóa modul .....	81
Hình 2.29: Biểu đồ hoạt động chức năng thêm danh mục .....	82

Hình 2.30: Biểu đồ hoạt động chức năng sửa chuyên mục .....	82
Hình 2.31: Biểu đồ hoạt động chức năng xóa chuyên mục.....	83
Hình 2.32: Biểu đồ hoạt động chức năng thêm bài viết.....	83
Hình 2.33: Biểu đồ hoạt động chức năng sửa bài viết .....	84
Hình 2.34: Biểu đồ hoạt động chức năng xóa bài viết .....	84
Hình 2.35: Biểu đồ hoạt động chức năng thêm giảng viên .....	85
Hình 2.36: Biểu đồ hoạt động chức năng sửa giảng viên.....	85
Hình 2.37: Biểu đồ hoạt động chức năng xóa giảng viên .....	86
Hình 2.38: Biểu đồ hoạt động chức năng thêm chuyên ngành .....	86
Hình 2.39: Biểu đồ hoạt động chức năng sửa chuyên ngành .....	87
Hình 2.40: Biểu đồ hoạt động chức năng xóa chuyên ngành.....	87
Hình 2.41: Biểu đồ hoạt động chức năng thêm slide .....	88
Hình 2.42: Biểu đồ hoạt động chức năng sửa slide.....	88
Hình 2.43: Biểu đồ hoạt động chức năng xóa slide.....	89
Hình 2.44: Biểu đồ hoạt động chức năng phân quyền người dùng .....	89
Hình 2.45: Biểu đồ hoạt động chức năng sửa thông tin người dùng.....	90
Hình 2.46: Biểu đồ hoạt động chức năng khóa tài khoản người dùng .....	90
Hình 2.47: Biểu đồ lớp Use case đăng nhập .....	91
Hình 2.48: Biểu đồ lớp Use case quản lý người dùng.....	92
Hình 2.49: Biểu đồ lớp Use case quản lý chuyên mục.....	92
Hình 2.50: Biểu đồ lớp Use case quản lý bài viết .....	93
Hình 2.51: Biểu đồ lớp Use case quản lý giảng viên .....	93
Hình 2.52: Biểu đồ lớp Use case quản lý chuyên ngành.....	94
Hình 2.53: Biểu đồ lớp Use case quản lý slide.....	94
Hình 2.54: Biểu đồ lớp Use case quản lý liên hệ .....	95
Hình 2.55: Biểu đồ lớp Use case gửi liên hệ.....	95
Hình 2.56: Biểu đồ lớp Use case xem bài viết .....	96
Hình 2.57: Mô hình CSDL .....	101
Hình 3.1: Các thư mục trong source website .....	102
Hình 3.2: Giao diện đăng nhập.....	104
Hình 3.3: Giao diện trang chủ trang quản trị viên.....	104
Hình 3.4: Giao diện quản lý Slide .....	105
Hình 3.5: Giao diện thêm Slide .....	105
Hình 3.6: Giao diện sửa Slide.....	106
Hình 3.7: Giao diện quản lý người dùng .....	106



Hình 3.8: Giao diện quản lý chuyên ngành .....	107
Hình 3.9: Giao diện thêm chuyên ngành .....	107
Hình 3.10: Giao diện quản lý giảng viên.....	108
Hình 3.11: Giao diện thêm giảng viên mới .....	108
Hình 3.12: Giao diện quản lý chuyên mục .....	109
Hình 3.13: Giao diện thêm chuyên mục .....	109
Hình 3.14: Giao diện sửa chuyên mục .....	110
Hình 3.15: Giao diện sửa hoặc xóa bài viết của chuyên mục .....	110
Hình 3.16: Giao diện quản lý bài viết .....	111
Hình 3.17: Giao diện chỉnh sửa bài viết.....	111
Hình 3.18: Giao diện quản lý liên hệ.....	112
Hình 3.19: Giao diện trang chủ Website .....	113
Hình 3.20: Giao diện danh sách bài viết .....	114
Hình 3.21: Giao diện chi tiết bài viết .....	115
Hình 3.22: Giao diện trang liên hệ .....	116
Hình 3.23: Giao diện website trên thiết bị di động .....	117

## DANH MỤC BẢNG

Bảng 2.2: Kịch bản chức năng đăng nhập.....	46
Bảng 2.3: Kịch bản chức năng quản lý người dùng .....	47
Bảng 2.4: Kịch bản chức năng phân quyền người dùng .....	48
Bảng 2.5: Kịch bản chức năng sửa thông tin người dùng .....	49
Bảng 2.6: Kịch bản chức năng khóa tài khoản người dùng .....	50
Bảng 2.7: Kịch bản chức năng quản lý chuyên mục .....	51
Bảng 2.8: Kịch bản chức năng thêm chuyên mục .....	51
Bảng 2.9: Kịch bản chức năng sửa chuyên mục .....	52
Bảng 2.10: Kịch bản chức năng xóa chuyên mục .....	53
Bảng 2.11: Kịch bản chức năng quản lý bài viết.....	54
Bảng 2.12: Kịch bản chức năng thêm bài viết .....	55
Bảng 2.13: Kịch bản chức năng sửa bài viết .....	55
Bảng 2.14: Kịch bản chức năng xóa bài viết.....	56
Bảng 2.15: Kịch bản chức năng quản lý giảng viên.....	57
Bảng 2.16: Kịch bản chức năng thêm giảng viên.....	58
Bảng 2.17: Kịch bản chức năng sửa giảng viên .....	58
Bảng 2.18: Kịch bản chức năng xóa giảng viên.....	59
Bảng 2.19: Kịch bản chức năng quản lý chuyên ngành .....	60
Bảng 2.20: Kịch bản chức năng thêm chuyên ngành .....	61
Bảng 2.21: Kịch bản chức năng sửa chuyên ngành .....	62
Bảng 2.22: Kịch bản chức năng xóa chuyên ngành .....	63
Bảng 2.23: Kịch bản chức năng quản lý slide .....	64
Bảng 2.24: Kịch bản chức năng thêm slide.....	64
Bảng 2.25: Kịch bản chức năng sửa slide .....	65
Bảng 2.26: Kịch bản chức năng xóa slide .....	66
Bảng 2.27: Kịch bản chức năng xóa liên hệ.....	67
Bảng 2.28: Kịch bản chức năng gửi liên hệ .....	68
Bảng 2.29: Kịch bản chức năng xem bài viết.....	69
Bảng 2.30: Danh sách các bảng.....	96
Bảng 2.31: Bảng User.....	97
Bảng 2.32: Bảng Major .....	97
Bảng 2.33: Bảng Slide .....	98
Bảng 2.34: Bảng Blog .....	98

Bảng 2.35: Bảng News .....	99
Bảng 2.36: Bảng Lecturer.....	99
Bảng 2.37: Bảng Contact.....	100

# GIỚI THIỆU

## 1. Mục tiêu đề tài

- Tìm hiểu, nghiên cứu và xây dựng Website Khoa CNTT – Trường Đại học Quản Lý và Công nghệ HP.
- Xây dựng trang website động với nội dung linh hoạt, cấu trúc rõ ràng, bắt mắt.

## 2. Giải pháp và cách thực hiện đề tài

- Tìm hiểu về các website giáo dục phổ biến, tham khảo, lựa chọn bố cục phù hợp.
- Xây dựng một website thì việc lựa chọn ngôn ngữ và ứng dụng khi làm việc là điều rất quan trọng. Trong đề tài lần này em sử dụng ngôn ngữ PHP làm ngôn ngữ lập trình, hệ quản trị cơ sở dữ liệu MySQL và một số thư viện hỗ trợ trong việc xây dựng website.

## 3. Tìm hiểu về Khoa CNTT – Trường Đại học Quản Lý & Công nghệ HP

- *Khoa Công nghệ Thông tin* được thành lập từ ngày 10/06/2011 với khởi đầu là Bộ môn Tin học (năm 1997). Trải qua hơn 22 năm hoạt động, Khoa Công nghệ Thông tin đã có những bước phát triển vững chắc, xứng đáng là nơi đào tạo chủ lực, cung cấp nguồn nhân lực Công nghệ Thông tin uy tín trong nhiều năm qua của *Trường Đại học Dân lập Hải Phòng* (nay là *Trường Đại học Quản lý và Công nghệ Hải Phòng*), địa chỉ: Số 36 Đường Dân Lập, Phường Dư Hàng Kênh, Quận Lê Chân, Thành phố Hải Phòng.

- *Các hệ đào tạo:*
  - Các chuyên ngành bậc đại học:
    - Công nghệ phần mềm.
    - Quản trị và an ninh Mạng.
    - Digital Marketing.
    - Tin học - Kinh tế.
  - Chuyên ngành bậc Thạc sỹ: Hệ thống Thông tin...
- *Hướng nghiên cứu khoa học:*
  - Công nghệ phần mềm

- Hệ thống thời gian thực
- Học máy, trí tuệ nhân tạo
- Xử lý ngôn ngữ Tự nhiên
- Phân tích quan điểm người dùng
- Xử lý dữ liệu lớn
- An ninh mạng
- Các hệ thống phân tán
- An toàn và bảo mật thông tin

#### **4. Ý nghĩa.**

- Đối với người sử dụng:
  - Với người quản trị hệ thống, hệ thống phân quyền quản trị cho người quản trị website, dễ dàng cập nhật, quản trị nội dung một cách tiện lợi nhất.
  - Với người dùng, hệ thống được thiết kế một cách khoa học, dễ dàng truy cập, tìm kiếm thông tin trên website, nắm bắt thông tin nhanh và hiệu quả.
- Đối với bản thân: Tạo ra một website giao diện thân thiện, hiệu quả, tối ưu nhất trong việc quản lý thông tin, tra cứu thông tin giúp sinh viên của khoa nắm bắt thông tin kịp thời nhất.

# CHƯƠNG 1: TỔNG QUAN

## 1.1 Tổng quan về World Wide Web



### 1.1.1 Khái niệm

World Wide Web (WWW) hay còn gọi là web là một dịch vụ phổ biến nhất hiện nay trên Internet, 85% các giao dịch trên Internet ước lượng thuộc về WWW. Ngày nay số website trên thế giới đã đạt tới con số khổng lồ. WWW cho phép truy xuất thông tin văn bản, hình ảnh, âm thanh, video trên toàn thế giới. Thông qua website, các quý công ty có thể giảm thiểu tối đa chi phí in ấn và phân phát tài liệu cho khách hàng ở nhiều nơi.

### 1.1.2 Lợi ích của WWW mang lại

World Wide Web có mối liên hệ mật thiết với mạng internet, vì vậy nó cũng mang lại cho con người rất nhiều lợi ích như:

- World Wide Web giúp con người có thể dễ dàng kết nối với nhau dù ở bất kỳ thời điểm nào, ở bất kỳ nơi đâu trên thế giới. WWW sẽ truyền tải những trang web, hình ảnh, thông tin... đến người dùng trong vòng vài giây.

- WWW sẽ giúp cho các trang Blog, mạng xã hội chia sẻ thông tin đến hàng triệu người sử dụng một lúc chỉ bằng một cái click chuột.

- Hiện nay World Wide Web cũng được các lập trình viên, chú trọng phát triển ứng dụng trên nền tảng điện thoại thông minh, giúp hỗ trợ rất nhiều cho con người trong công việc và cuộc sống.

### 1.1.3 Nguyên tắc hoạt động

World Wide Web là một hệ thống các tài liệu siêu văn bản được liên kết với nhau qua mạng internet. WWW cho phép hiển thị và truy xuất các văn bản, trên máy tính của người dùng. World Wide Web cho phép cá nhân người dùng liên kết các tài liệu và văn bản của họ với nhau để tạo thành mạng lưới các tài liệu.

### 1.1.4 Các chức năng chính

Chức năng chính của WWW đó là cung cấp thông tin cho người dùng, có nhu cầu tìm kiếm những thông tin cụ thể thông qua mạng internet. Như vậy World Wide Web được tạo ra để cung cấp thông tin cần thiết và những thông tin giải trí cho tất cả những người sử dụng mạng internet.

### 1.1.5 Sự phát triển của WWW trong thời đại ngày nay

Hiện nay mọi nội dung website, kế hoạch SEO, Marketing đều hướng tới các thiết bị di động bởi sự phát triển không ngừng của nó.

#### - **Thiết bị di động:**

Trước tiên phải kể đến sự bùng nổ chiều hướng lướt web trên các thiết bị di động. Trong vài năm trở lại đây, thay vì dùng máy tính thông thường, người dùng có chiều hướng dùng điện thoại nhiều hơn. Vì thế, toàn bộ các Website lớn, nhỏ đều thực hiện điều chỉnh các phương thức thiết kế giao diện, nội dung. Tiêu biểu đó là phương pháp đánh giá Mobile-friendly (là yếu tố để Google đánh giá thứ hạng của website) của Google. Mục đích cuối cùng vẫn là đem lại trải nghiệm tốt nhất cho người dùng.

#### - **Tính bảo mật:**

Bên cạnh đó, quyền riêng tư và bảo mật thông tin trên Internet là một vấn đề đang ngày càng được lưu tâm. Khác với trước đây, lượng thông tin cá nhân khổng lồ, bao gồm: Lịch sử tìm kiếm, duyệt web, vị trí địa lý của người dùng tập tính tìm kiếm của người dùng... Những thông tin này nhiều được thu thập để đáp ứng cho mục đích nhắm mục tiêu quảng cáo. Vì thế, các dịch vụ Web proxy sẽ cung cấp cho người dùng trực tuyến mức độ riêng tư cao hơn nhờ định tuyến lại hành vi duyệt web thông qua các máy chủ Web của bên thứ ba.

#### - **Tên miền:**

Các website hiện nay vẫn dùng tên miền và miền mở rộng. Tên miền “.com” vẫn thường nhật và thông dụng nhất. mặc khác, cũng có nhiều tên miền mới hiện có sẵn để đăng ký như “.info”, “.net” và “.asia”.

#### - **Trình duyệt:**

IE và Firefox trước kia là các trình duyệt Web thông dụng nhất, nhưng hiện nay Chrome do Google quản lý vẫn đang thống lĩnh thị trường, mặt khác, mới đây nhất là sự xuất hiện của safari – trình duyệt thường nhật trên danh mục của Apple.

#### - **Công nghệ lập trình:**

HTML5 đã tái thiết lập HTML trở thành công nghệ Web hiện đại sau khi bị đình trệ trong nhiều năm. Tương tự như vậy, các cải tiến hiệu suất trong HTTP 2.0 cũng đảm bảo cho sự tồn tại của giao thức trong tương lai gần.

### **1.1.6 Tầm quan trọng của WWW**

Tầm quan trọng của WWW được thể hiện qua các yếu tố sau:

- Mạng Internet, thông tin đến tất cả mọi người, không những dành cho các nhà khoa học như mục đích thống kê ban đầu.
- Kết nối thế giới theo cách không thể thực hiện được trước đây và giúp mọi người đơn giản nhận thông tin, chia sẻ thông tin và giao tiếp với nhau.
- Cho phép mọi người chia sẻ công việc, suy nghĩ của họ thông qua các trang mạng xã hội, blog và video như hiện nay.
- Đem lại tiện ích cùng trải nghiệm tốt nhất cho người dùng.

## **1.2 Tổng quan về HTML – CSS**

### **1.2.1 Tổng quan về HTML**

**HTML** là chữ viết tắt của **HyperText Markup Language**, dịch ra tiếng Việt có nghĩa là ngôn ngữ đánh dấu siêu văn bản. Nó dùng để định dạng bố cục, các thuộc tính liên quan đến cách hiển thị của một đoạn text và được hiển thị trên một chương trình đặc biệt, ta gọi là trình duyệt browser.

Hiện nay có khá nhiều browser như Firefox, Chrome, Cốc Cốc, ...Tất cả browser đều có điểm chung là giúp người dùng thao tác với website và nó đều có khả năng biên dịch những đoạn mã HTML.



HTML có rất nhiều thẻ khác nhau, và mỗi thẻ sẽ có những nhiệm vụ nhất định.

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <title>Document</title>
6 </head>
7 <body>
8
9 </body>
10 </html>
```

Hình 1.1: Cấu trúc của file HTML

Trong đó:

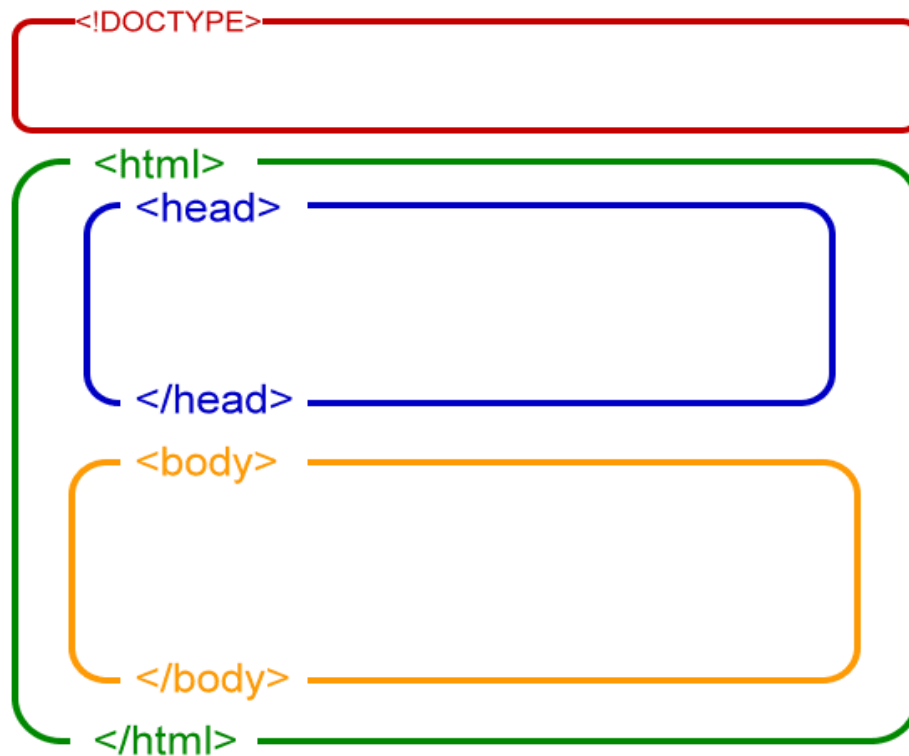
- **<!DOCTYPE html>** là phần khai báo kiểu dữ liệu hiển thị là html để trình duyệt (Browser) biết.
- **<html>** và **</html>** là cặp thẻ nằm ngoài cùng và nó có nhiệm vụ là bao hết nội dung của trang web lại. Thẻ này là bắt buộc.
- **<head>** và **</head>** là phần khai báo thông tin của trang web
- **<title>** và **</title>** nằm bên trong thẻ **<head>** và đây chính là khai báo tiêu đề cho trang web.
- **<body>** và **</body>** là thành phần quan trọng nhất, nó chứa những đoạn mã HTML dùng để hiển thị trên website
- Các thẻ còn lại nằm trong thẻ **<body>** chính là các thẻ định dạng dữ liệu.

Như vậy trong một website chúng ta chia làm 2 phần chính:

- Phần 1: Là những khai báo thông tin cho trang web và ta đặt nó trong thẻ head.

- Phần 2: Là phần hiển thị định dạng nội dung của trang web và ta đặt trong thẻ body.

Và được diễn tả như hình sau:



Ngoài ra trong thẻ **<body>** của HTML còn có rất nhiều thẻ phục vụ các mục đích khác nhau như:

- Thẻ **<div>...</div>** định nghĩa một nội dung.
- Thẻ **<p> ... </p>** tạo một đoạn mới.
- Thẻ **<font> ... </font>** thay đổi phông chữ, kích cỡ và màu kí tự.
- Thẻ **<table> ... </table>** đây là thẻ định dạng bảng trên website. Sau khi khai báo thẻ này, phải khai báo các thẻ hàng **<tr>** và thẻ cột **<td>** cùng với các thuộc tính của nó.
- Thẻ **<img />** cho phép chèn hình ảnh vào website. Thẻ này thuộc loại thẻ không có thẻ đóng.
- Thẻ **<a> ... </a>** là loại thẻ dùng để liên kết giữa các website hoặc liên kết đến địa chỉ Internet, Mail hay Intranet (URL) và địa chỉ trang tin trong mạng cục bộ (UNC).

- Thẻ `<input />` cho phép người dùng nhập dữ liệu hay chỉ thị thực thi một hành động nào đó, thẻ Input bao gồm các loại thẻ như: text, password, submit, button, reset, checkbox, radio, hidden, image.
- Thẻ `< textarea>.... < /textarea>` cho phép người dùng nhập liệu với rất nhiều dòng. Với thẻ này không thể giới hạn chiều dài lớn nhất trên website.
- Thẻ `<select> ... </select>` cho phép người dùng chọn phần tử trong tập phương thức đã được định nghĩa trước. Nếu thẻ `<select>` cho phép người dùng chọn một phần tử trong danh sách phần tử thì thẻ `<select>` sẽ giống như combobox. Nếu thẻ `<select>` cho phép người dùng chọn nhiều phần tử cùng một lần trong danh sách phần tử, thẻ `<select>` đó là dạng listbox.
- Thẻ `<form> ... </form>` khi muốn submit dữ liệu người dùng nhập từ website phía Client lên phía Server, có hai cách để làm điều này ứng với hai phương thức **POST** và **GET** trong thẻ form. Trong một website có thể có nhiều thẻ `<form>` khác nhau, nhưng các thẻ `<form>` này không được lồng nhau, mỗi thẻ form sẽ được khai báo hành động (action) chỉ đến một trang khác....

### 1.2.2. Tổng quan về CSS

CSS là chữ viết tắt của Cascading Style Sheets, nó là một ngôn ngữ được sử dụng để tìm và định dạng lại các phần tử được tạo ra bởi các ngôn ngữ đánh dấu (HTML). Nói ngắn gọn hơn là ngôn ngữ tạo phong cách cho trang web. Bạn có thể hiểu đơn giản rằng, nếu HTML đóng vai trò định dạng các phần tử trên website như việc tạo ra các đoạn văn bản, các tiêu đề, bảng...thì CSS sẽ giúp chúng ta có thể thêm style vào các phần tử HTML đó như đổi bố cục, màu sắc trang, đổi màu chữ, font chữ, thay đổi cấu trúc...

CSS được phát triển bởi W3C (World Wide Web Consortium) vào năm 1996. Nó dùng để tạo phong cách và định kiểu cho những yếu tố được viết dưới dạng ngôn ngữ đánh dấu, như là HTML. Nó có thể điều khiển định dạng của nhiều trang web cùng lúc để tiết kiệm công sức cho người viết web. Nó phân biệt cách hiển thị của trang web với nội dung chính của trang bằng cách điều khiển bố cục, màu sắc, và font chữ...

Mối tương quan giữa HTML và CSS rất mật thiết. HTML là ngôn ngữ markup (nền tảng của site) và CSS định hình phong cách (tất cả những gì tạo nên giao diện website), chúng là không thể tách rời.

Các stylesheet ngoài được lưu trữ dưới dạng các tập tin “.css”.

Bộ cục CSS thường chủ yếu dựa vào hình hộp và mỗi hộp đều chiếm những khoảng trống trên trang của bạn với các thuộc tính như:

- **Padding:** Gồm không gian xung quanh nội dung (ví dụ: xung quanh đoạn văn bản).
- **Border:** Là đường viền nằm ngay bên ngoài phần đệm.
- **Margin:** Là khoảng cách xung quanh bên ngoài của phần tử.

Mỗi đoạn CSS gồm các phần sau:

```
vùng chọn {  
    thuộc tính: giá trị;  
    thuộc tính: giá trị;  
    .....  
}
```

Điều này có nghĩa, CSS được khai báo trong vùng chọn, các thuộc tính cùng giá trị nằm ở cặp dấu “{}” và không giới hạn số lượng các thuộc tính. Từng thuộc tính có một giá trị riêng, tồn tại ở dạng số, hoặc tên giá trị được liệt kê trong danh sách của CSS.

Các phần giá trị và thuộc tính sẽ được phân cách bằng dấu “:” Mỗi dòng khai báo thuộc tính luôn có dấu “;” ở cuối dòng.

Một số thuộc tính cơ bản của CSS có thể kể đến như:

- **Background:** Định dạng nền (background) cho thành phần.
- **Border:** Định dạng đường viền cho thành phần.
- **Color:** Xác định màu sắc cho text.
- **Display:** Xác định loại hiển thị của thành phần.

- **Float:** Xác định có hay không một thành phần được float.
- **Font:** Thiết lập font cho thành phần, bao gồm font chữ, độ rộng, ...
- **Width:** Thiết lập chiều rộng cho thành phần.
- **Height:** Thiết lập chiều cao cho thành phần.

### 1.3 Ngôn ngữ PHP



#### 1.3.1 Định nghĩa PHP

**PHP: Hypertext Preprocessor**, thường được viết tắt thành PHP là một ngôn ngữ lập trình kịch bản hay một loại mã lệnh chủ yếu được dùng để phát triển các ứng dụng viết cho máy chủ, mã nguồn mở, dùng cho mục đích tổng quát. Nó rất thích hợp với web và có thể dễ dàng nhúng vào trang HTML. Do được tối ưu hóa cho các ứng dụng web, tốc độ nhanh, nhỏ gọn, cú pháp giống C và Java, dễ học và thời gian xây dựng sản phẩm tương đối ngắn hơn so với các ngôn ngữ khác nên PHP đã nhanh chóng trở thành một ngôn ngữ lập trình web phổ biến nhất thế giới.

PHP chạy trên môi trường Webserver và lưu trữ dữ liệu thông qua hệ quản trị cơ sở dữ liệu nên PHP thường đi kèm với Apache, MySQL, v.v.

**Apache** là một phần mềm web server có nhiệm vụ tiếp nhận request từ trình duyệt người dùng sau đó chuyển giao cho PHP xử lý và gửi trả lại cho trình duyệt.

**MySQL** cũng tương tự như các hệ quản trị cơ sở dữ liệu khác (Postgress, Oracle, SQL server, v.v.) đóng vai trò là nơi lưu trữ và truy vấn dữ liệu.

### 1.3.2 Lý do nên dùng PHP

PHP được sử dụng làm web động vì nó nhanh, dễ dàng, tốt hơn so với các giải pháp khác.

PHP có khả năng thực hiện và tích hợp chặt chẽ với hầu hết các cơ sở dữ liệu có sẵn, tính linh động, bền vững và khả năng phát triển không giới hạn.

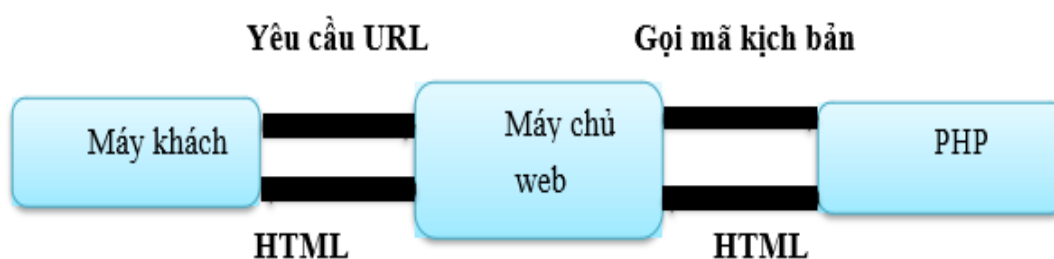
Đặc biệt PHP là mã nguồn mở do đó tất cả các đặc tính trên đều miễn phí, và chính vì mã nguồn mở sẵn có nên cộng đồng các nhà phát triển web luôn có ý thức cải tiến nó, nâng cao để khắc phục các lỗi trong các chương trình này.

PHP vừa dễ với người mới sử dụng vừa có thể đáp ứng mọi yêu cầu của các lập trình viên chuyên nghiệp, mọi ý tưởng của các PHP có thể đáp ứng một cách xuất sắc. PHP đã có mặt trên 13 triệu website.

### 1.3.3 Cách thức hoạt động của PHP

Vì PHP là ngôn ngữ của máy chủ nên mã lệnh của PHP sẽ tập trung trên máy chủ để phục vụ các trang web theo yêu cầu của người dùng thông qua trình duyệt.

Sơ đồ hoạt động:



Hình 1.2: Sơ đồ hoạt động của PHP

Khi người dùng truy cập website viết bằng PHP, máy chủ đọc mã lệnh PHP và xử lý chúng theo các hướng dẫn được mã hóa. Mã lệnh PHP yêu cầu máy chủ gửi một dữ liệu thích hợp (mã lệnh HTML) đến trình duyệt web. Trình duyệt xem nó như là một trang HTML tiêu chuẩn. Như ta đã nói, PHP cũng chính là một trang HTML nhưng có nhúng mã PHP và có phần mở rộng là HTML. Phần mở của PHP được đặt trong thẻ mở, khi trình duyệt truy cập vào một trang PHP, Server sẽ đọc nội dung file PHP lên và lọc ra các đoạn mã PHP và thực thi các đoạn mã đó, lấy kết quả nhận được của đoạn mã PHP thay thế vào chỗ ban đầu của chúng trong file PHP, cuối cùng Server trả về kết quả cuối cùng là một trang nội dung HTML về cho trình duyệt.

### 1.3.4 Tổng quan về PHP

Cấu trúc cơ bản: PHP cũng có thể bắt đầu và kết thúc giống với HTML

Cú pháp chính

```
<?php Mã lệnh php?>
```

Trong PHP để kết thúc 1 dòng lệnh chúng ta sử dụng dấu ";" . Để chú thích một đoạn dữ liệu nào đó trong PHP ta sử dụng dấu "/\*" cho từng dòng hoặc dùng cặp thẻ "/\*\*/" cho từng cụm mã lệnh.

Ví dụ: <?php echo "Hello world!" ;?>

Xuất giá trị ra trình duyệt chúng ta có những dòng cú pháp sau:

```
Echo "Thông tin";  
Printf "Thông tin"; Thông tin bao gồm biến,  
chuỗi, hoặc lệnh html....
```

PHP có các kiểu dữ liệu cơ bản sau:

Số nguyên, số thực, chuỗi, Boolean, mảng, đối tượng, resource, NULL/

Một số thành phần chính trong PHP.

- Biến: Một biến bắt đầu bằng dấu \$, theo sau là tên của biến.
- Tên biến phải bắt đầu bằng một chữ cái hoặc ký tự gạch dưới.
- Một tên biến không thể bắt đầu bằng một số.
- Tên biến chỉ có thể chứa các ký tự chữ và số dưới (Az, 0-9 và \_).
- Tên biến là phân biệt chữ hoa chữ thường.

Chuỗi: là một nhóm các ký tự, số, khoảng trắng, dấu ngắt được đặt trong các dấu nháy, ví dụ: 'Hello'.

Hằng: Một hằng số là một định danh (tên) cho một giá trị đơn giản. Giá trị không thể thay đổi trong tập lệnh. Để tạo một hằng số, sử dụng define() hàm. Các hằng số được tự động toàn cầu và có thể được sử dụng trên toàn bộ tập lệnh.

### 1.3.5 Hàm trong PHP

Sức mạnh thực sự của PHP đến từ các Hàm.

PHP có hơn 1000 hàm dựng sẵn và ngoài ra, bạn có thể tạo các hàm tùy chỉnh.

Để giảm thời gian lặp lại 1 thao tác code nhiều lần, PHP hỗ trợ người lập trình việc tự định nghĩa cho mình những hàm có khả năng lặp lại nhiều lần trong website. Việc này cũng giúp cho người lập trình kiểm soát mã nguồn một cách mạch lạc, đồng thời có thể tùy biến ở mọi trang mà không cần phải khởi tạo hay viết lại mã lệnh như HTML thuần.

Một khai báo hàm do người dùng định nghĩa bắt đầu bằng từ function, tên hàm phải bắt đầu bằng chữ cái hoặc dấu gạch dưới. Tên hàm không phân biệt chữ hoa, chữ thường.

- Tự định nghĩa hàm:

```
Function functionName(){  
    // Lệnh thực thi;  
}
```

- Tự định nghĩa hàm có tham số

```
Function functionName($giatri1, $giatri2){  
    // Lệnh thực thi;  
}
```

- Tự định nghĩa hàm có giá trị trả về

```
Function functionName(){  
    // Lệnh thực thi;  
    Return giatrij;
```

- Hàm có đối số

```
Function functionName($giatri = 1){  
    // Lệnh thực thi;  
}
```



### 1.3.6 Biểu mẫu PHP

Có 2 phương thức được sử dụng trong lập trình PHP là GET và POST.

- Phương thức **GET**: cũng được dùng để lấy dữ liệu từ form nhập liệu. Tuy nhiên nhiệm vụ chính của nó vẫn là lấy nội dung trang dữ liệu từ web server.
  - Phương thức **POST**: phương thức này được sử dụng để lấy dữ liệu từ form nhập liệu và chuyển chúng lên trình chủ webserver.

### 1.3.7 PHP OOP (Object-Oriented Programming.)

OOP là viết tắt của lập trình hướng đối tượng, là việc tạo các đối tượng chứa cả hàm và dữ liệu.

Lập trình hướng đối tượng có một số lợi thế so với lập trình thủ tục:

- OOP nhanh hơn và dễ thực hiện hơn.
- OOP cung cấp một cấu trúc rõ ràng cho các chương trình.
- OOP giúp giữ mã PHP DRY "Đừng lặp lại chính mình" và làm cho mã dễ dàng hơn để duy trì, sửa đổi và gỡ lỗi.
- OOP cho phép tạo các ứng dụng có thể tái sử dụng đầy đủ với ít mã hơn và thời gian phát triển ngắn hơn.

### 1.3.8 Session và Cookie

Cookie và Session là hai phương pháp sử dụng để quản lý các phiên làm việc giữa người sử dụng và hệ thống.

Cookie và Session là hai phương pháp sử dụng để quản lý các phiên làm việc giữa người sử dụng và hệ thống.

Session dùng để lưu dữ liệu trên server, session dùng để lưu trữ thông tin người dùng, hoặc lưu trữ tùy chọn cấu hình hệ thống cho người dùng. Tất cả session đc lưu trữ trong biến toàn cục `$_SESSION`.

Cookie dùng để lưu trữ các tùy chọn riêng của trang web từng user, nó là file nhỏ được chỉ định lưu trên máy tính client và php có thể truy xuất được, cần trình duyệt hỗ trợ chức năng này. Cookie không bị mất khi bị đóng ứng dụng lại, chỉ mất khi hết hạn thời gian thiết lập. Tất cả cookie được lưu trữ trong biến toàn cục `$_COOKIE`.

## 1.4 Hệ quản trị CSDL MySQL



### 1.4.1 Giới thiệu về My SQL

MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (Relational Database Management System, viết tắt là RDBMS) hoạt động theo mô hình client-server. RDBMS là một phần mềm hay dịch vụ dùng để tạo và quản lý các cơ sở dữ liệu (Database) theo hình thức quản lý các mối liên hệ giữa chúng.

MySQL là cơ sở dữ liệu có trình giao diện trên Windows hay Linux, cho phép người sử dụng có thể thao tác các hành động liên quan đến nó. Việc tìm hiểu từng công nghệ trước khi bắt tay vào việc viết mã kịch bản PHP, việc tích hợp hai công nghệ PHP và MySQL là một công việc cần thiết và rất quan trọng.

### 1.4.2 Mục đích sử dụng CSDL

Lưu trữ: Lưu trữ trên đĩa và có thể chuyển đổi dữ liệu từ cơ sở dữ liệu này sang cơ sở dữ liệu khác, nếu sử dụng cho quy mô nhỏ, có thể chọn cơ sở dữ liệu nhỏ như: Microsoft Exel, Microsoft Access, MySQL, Microsoft Visual FoxPro... Nếu ứng dụng có quy mô lớn, có thể chọn cơ sở dữ liệu có quy mô lớn như: Oracle, SQL Server...

Truy cập: Truy cập dữ liệu phụ thuộc vào mục đích và yêu cầu của người sử dụng, ở mức độ mang tính cục bộ, truy cập cơ sở dữ liệu ngay trong cơ sở dữ liệu với nhau, nhằm trao đổi hay xử lý dữ liệu ngay bên trong chính nó, nhưng do mục đích và yêu cầu người dùng vượt ra ngoài cơ sở dữ liệu, nên cần có các phương thức truy cập dữ liệu giữa các cơ sở dữ liệu với nhau như: Microsoft Access với SQL Server, hay SQL Server và cơ sở dữ liệu Oracle....

**Tổ chức:** Tổ chức cơ sở dữ liệu phụ thuộc vào mô hình cơ sở dữ liệu, phân tích và thiết kế cơ sở dữ liệu tức là tổ chức cơ sở dữ liệu phụ thuộc vào đặc điểm riêng của từng ứng dụng. Tuy nhiên khi tổ chức cơ sở dữ liệu cần phải tuân theo một số tiêu chuẩn của hệ thống cơ sở dữ liệu nhằm tăng tính tối ưu khi truy cập và xử lý.

**Xử lý:** Tùy vào nhu cầu tính toán và truy vấn cơ sở dữ liệu với các mục đích khác nhau, cần phải sử dụng các phát biểu truy vấn cùng các phép toán, phát biểu của cơ sở dữ liệu để xuất ra kết quả như yêu cầu. Để thao tác hay xử lý dữ liệu bên trong chính cơ sở dữ liệu ta sử dụng các ngôn ngữ lập trình như: PHP, C++, Java, Visual Basic...

## 1.5 Bootstrap



### 1.5.1 Định nghĩa Bootstrap

Bootstrap là một framework bao gồm các HTML, CSS và JavaScript template dùng để phát triển website chuẩn responsive.

Bootstrap cho phép quá trình thiết kế website diễn ra nhanh chóng và dễ dàng hơn dựa trên những thành tố cơ bản sẵn có như typography, forms, buttons, tables, grids, navigation, image carousels...

Bootstrap là một bộ sưu tập miễn phí của các mã nguồn mở và công cụ dùng để tạo ra một mẫu website hoàn chỉnh. Với các thuộc tính về giao diện được quy định sẵn như kích thước, màu sắc, độ cao, độ rộng..., các designer có thể sáng tạo nhiều sản phẩm mới mẻ nhưng vẫn tiết kiệm thời gian khi làm việc với framework này trong quá trình thiết kế giao diện website.

## 1.5.2 Lịch sử phát triển của Bootstrap

Bootstrap là sản phẩm của Mark Otto và Jacob Thornton tại Twitter. Nó được xuất bản như là một mã nguồn mở vào ngày 19/8/2011 trên GitHub. Tên gọi ban đầu là Twitter Blueprint.

Đến ngày 31/1/2012, Bootstrap phiên bản 2 đã được phát hành. Bootstrap 2 được bổ sung bố cục lưới 12 cột với thiết kế tùy chỉnh đáp ứng cho nhiều màn hình kích thước. Tiếp nối sự thành công của phiên bản 2, Bootstrap 3 ra đời vào ngày 19/8/2013 với giao diện tương thích với smartphone.

Chỉ 3 năm sau ngày ra mắt, Bootstrap đã trở thành No.1 project trên GitHub. Vào tháng 10/2014, Mark Otto công bố phiên bản Bootstrap 4 đang được phát triển. Phiên bản alpha đầu tiên của Bootstrap 4 đã được triển khai vào tháng 8/2015. Phiên bản mới nhất của Bootstrap được giới thiệu đến người dùng là Bootstrap 4.3.1. Cho đến nay, Bootstrap vẫn là một trong những framework thiết kế website có lượng người dùng “khủng” nhất.

## 1.5.3 Lý do nên sử dụng Bootstrap

Giữa muôn vàn ứng dụng thiết kế website hiện nay, Bootstrap vẫn có khả năng cạnh tranh cao là nhờ những đặc điểm nổi bật sau:

- Dễ dàng thao tác
- Tùy chỉnh dễ dàng
- Chất lượng sản phẩm đầu ra hoàn hảo
- Độ tương thích cao
- Dễ dàng thao tác: Cơ chế hoạt động của Bootstrap là dựa trên xu hướng mã nguồn mở HTML, CSS và Javascript. Người dùng cần trang bị kiến thức cơ bản 3 mã này mới có thể sử dụng Bootstrap hiệu quả. Bên cạnh đó, các mã nguồn này cũng có thể dễ dàng thay đổi và chỉnh sửa tùy ý.
- Tùy chỉnh dễ dàng: Bootstrap được tạo ra từ các mã nguồn mở cho phép designer linh hoạt hơn. Giờ đây có thể lựa chọn những thuộc tính, phần tử phù hợp với dự án họ đang theo đuổi. CDN Bootstrap còn giúp bạn tiết kiệm dung lượng vì không cần tải mã nguồn về máy.

- Chất lượng sản phẩm đầu ra hoàn hảo: Bootstrap là sáng tạo của các lập trình viên giỏi trên khắp thế giới. Bootstrap đã được nghiên cứu và thử nghiệm trên các thiết bị. Được kiểm tra nhiều lần trước khi đưa vào sử dụng. Do đó, khi chọn Bootstrap, bạn có thể tin rằng mình sẽ tạo nên những sản phẩm với chất lượng tốt nhất.
- Độ tương thích cao: Điểm cộng lớn nhất của Bootstrap là khả năng tương thích với mọi trình duyệt và nền tảng. Đây là một điều cực kì quan trọng và cần thiết trong trải nghiệm người dùng. Sử dụng Grid System cùng với hai bộ tiền xử lý Less và Sass, Bootstrap mặc định hỗ trợ Responsive và ưu tiên cho các giao diện trên thiết bị di động hơn. Bootstrap có khả năng tự động điều chỉnh kích thước website theo khung browser. Mục đích để phù hợp với màn hình của máy tính để bàn, tablet hay laptop.

#### 1.5.4 Cấu trúc và Tính năng của Bootstrap

- Bootstrap chứa các tập tin JavaScript, CSS và fonts đã được biên dịch và nén lại. Ngoài ra, Bootstrap được thiết kế dưới dạng các mô-đun. Do đó, dễ dàng tích hợp với hầu hết các mã nguồn mở như WordPress, Joomla, Magento, ... Trong đó, Bootstrap mang đến nhiều chức năng nổi bật.
- Bootstrap cho phép người dùng truy cập vào thư viện “khổng lồ” các thành tố dùng để tạo nên giao diện của một website hoàn chỉnh như font, typography, form, table, grid...
- Bootstrap cho phép bạn tùy chỉnh framework của website trước khi tải xuống và sử dụng nó tại website của khung.
- Tái sử dụng các thành phần lặp đi lặp lại trên website.
- Bootstrap được tích hợp jQuery. Bạn chỉ cần khai báo chính xác các tính năng trong quá trình lập trình web của bạn.
- Định nghĩa glyphicons nhằm giảm thiểu việc sử dụng hình ảnh làm biểu tượng và tăng tốc độ tải trang.

## 1.6 Thư viện jQuery



### 1.6.1 Định nghĩa

jQuery là một thư viện JavaScript nhỏ gọn, chạy nhanh. jQuery giải quyết các vấn đề tương tác với HTML, bắt sự kiện, hiệu ứng động ... trở lên rất đơn giản.

Mọi tính năng của jQuery thực ra nó bao bọc các lệnh, hàm của JavaScript nên để hiểu jQuery bạn cần hiểu về HTML cơ bản cũng như CSS cơ bản trước.

Các tính năng chính jQuery cung cấp gồm:

- Tương tác với HTML/DOM.
- Tương tác với CSS.
- Bắt và xử lý sự kiện HTML.
- Các hiệu ứng và chuyển động trong HTML.
- AJAX (Asynchronous JavaScript and XML).
- JSON parsing.
- Các tiện ích xây dựng sẵn.

### 1.6.2 Tại sao nên chọn jQuery

Hiện nay có nhiều JavaScript Framework xuất hiện nhưng jQuery vẫn là Framework phổ biến nhất và nhiều thành phần mở rộng cài thêm (extend). Nhiều công ty công nghệ lớn sử dụng jQuery như Google, Microsoft, IBM, Netflix ...

Ngoài ra jQuery còn tương thích với nhiều trình duyệt khác nhau thậm chí là IE6, bạn thật sự không cần bận tâm nhiều về việc code của mình không chạy được trên các trình duyệt khác nhau.

### 1.6.3 Cài đặt tích hợp jQuery vào website

Để sử dụng jQuery trong trang HTML, bạn cần đảm bảo trang HTML của mình load chính xác thư viện jQuery bằng cách sử dụng thẻ `<script>` ở phần `<head>` của HTML

```
<html>

  <head>

    <script src="file-jquery.js"></script>

  </head>

  <body>

    ....

  </body>

</html>
```

**Cách 1:** tải về file js thư viện tại jQuery download (nên chọn bản đã nén có chữ .min.js), sau đó tích hợp vào trang bằng thẻ `<script>`. Ví dụ: `<script src="/public/jquery-3.1.1.min.js"></script>`

**Cách 2:** thay vì phải tải về file thư viện, ta dùng luôn từ các CDN (Content Delivery Network) giúp cho trang tải nhanh hơn.

Dùng CDN Google: Lấy link phiên bản jQuery tại jQuery Google, sau đó tích hợp vào Website, ví dụ:

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js">

</script>
```

### 1.6.4 Cú pháp jQuery

Khi sử dụng jQuery, bạn cần chọn các phần tử(query) sau đó thực hiện các hành động trên chúng, điều này được thể hiện bằng cú pháp sử dụng jQuery: `$("selector").action()`

- \$ là ký hiệu cho biết bạn truy cập jQuery
- (selector) phần tìm phần tử HTML, theo cách chọn phần tử như đã biết trong CSS (xem thêm chọn phần tử HTML). Ví dụ chọn tất cả các phần tử p là \$('p'), ví dụ chọn phần tử có id là exam là \$('#exam'), ví dụ chọn các phần tử có class là examclass là \$('.examclass')
- action() là các hành động trên phần tử đã chọn, là các hàm mà jQuery cung cấp, bạn sẽ học dần các hàm này

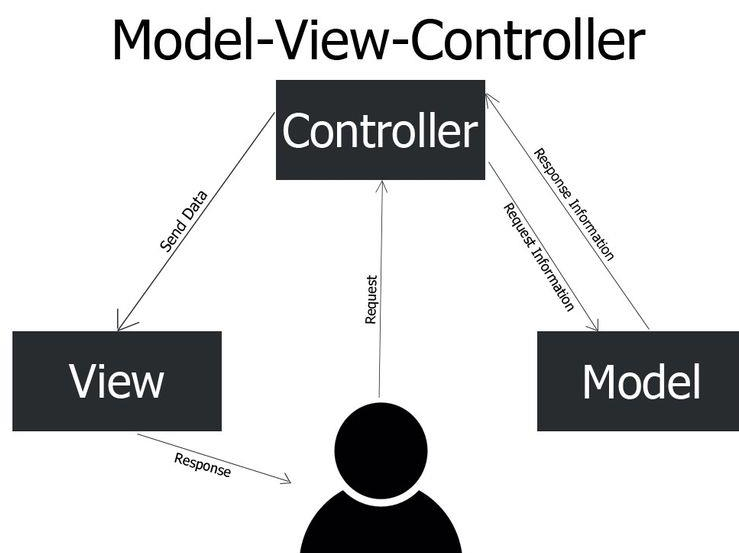
Xem lại ví dụ trên, câu lệnh theo cú pháp đó là:

```
$("#start").html("Go!");
```

## 1.7 Mô hình MVC

### 1.7.1 MVC là gì?

Mô hình MVC (Model – View – Controller) là một kiến trúc phần mềm hay mô hình thiết kế được sử dụng trong kỹ thuật phần mềm. Nó giúp cho các developer tách ứng dụng của họ ra 3 thành phần khác nhau Model, View và Controller. Mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác.



Hình 1.3: Cấu trúc mô hình MVC

### 1.7.2 Các thành phần trong MVC

- **Model:**



Đây là thành phần chứa tất cả các nghiệp vụ logic, phương thức xử lý, truy xuất database, đối tượng mô tả dữ liệu như các Class, hàm xử lý...

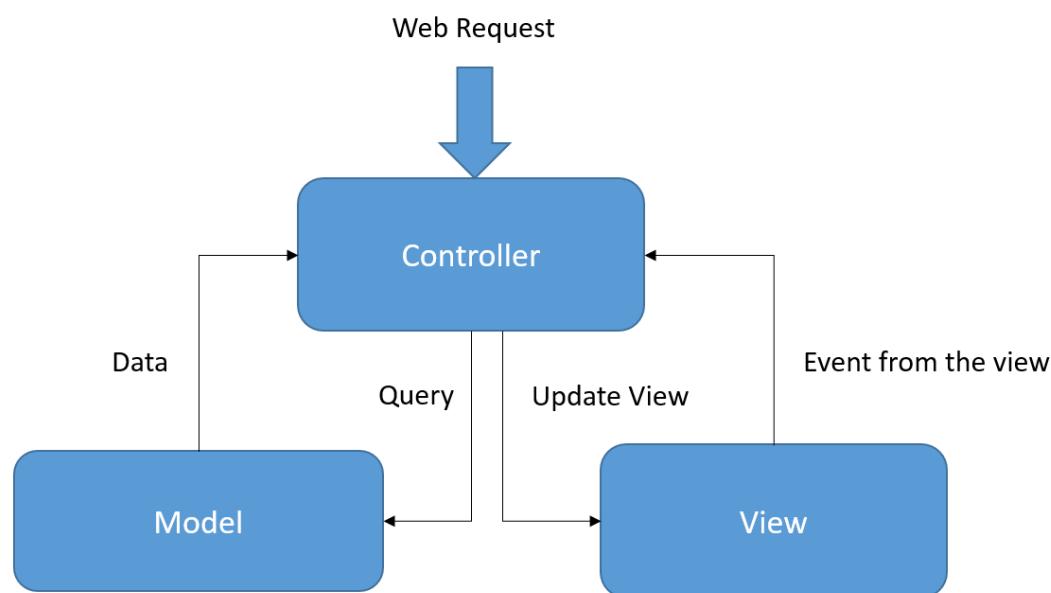
**- View:**

Đảm nhận việc hiển thị thông tin, tương tác với người dùng, nơi chứa tất cả các đối tượng GUI như textbox, images...Hiểu một cách đơn giản, nó là tập hợp các form hoặc các file HTML.

**- Controller:**

Giữ nhiệm vụ nhận điều hướng các yêu cầu từ người dùng và gọi đúng những phương thức xử lý chúng... Chẳng hạn thành phần này sẽ nhận request từ url và form để thao tác trực tiếp với Model.

### 1.7.3 MVC làm việc như thế nào?



Hình 1.4: Cách hoạt động của mô hình MVC

Đây là một cách đơn giản để mô tả lại luồng sự kiện được xử lý trong MVC:

- Giữa Client và Database có mô hình MVC.
- Khi người dùng gõ từ khóa trên trình duyệt chính là lúc tác động lên Controller, khi đó giữa Controller và database sẽ làm một hàm được gọi trong Model, Model sẽ lấy giữ liệu từ cơ sở dữ liệu, trả dữ liệu về Model, và trả ngược lại Controller, và trả ngược lại Client thông qua View. View chính là kết quả nhìn thấy.

- Model là kho tạo ra những hàm gọi ra từ cơ sở dữ liệu, sau đó trả về Controller và trả về View.
- View và Model không liên quan đến nhau, View chỉ được gọi hàm liên quan đến truy vấn dữ liệu thông qua Controller.

#### 1.7.4 Ưu điểm và nhược điểm của MVC

- Ưu điểm:

Thể hiện tính chuyên nghiệp trong lập trình, phân tích thiết kế. Do được chia thành các thành phần độc lập nên giúp phát triển ứng dụng nhanh, đơn giản, dễ nâng cấp, bảo trì...

- Nhược điểm:

Đối với dự án nhỏ việc áp dụng mô hình MC gây công kênh, tốn thời gian trong quá trình phát triển. Tốn thời gian trung chuyển dữ liệu của các thành phần.

## 1.8 Ứng dụng Laragon

### 1.8.1 Định nghĩa

Laragon là ứng dụng dùng để tạo web-server, môi trường phát triển web trên Windows. Đặc biệt Laragon được tác giả xây dựng theo dạng module, điều đó giúp bạn có thể tùy chỉnh môi trường phát triển thích hợp với nhu cầu cá nhân hoặc từng dự án riêng biệt. Bản Laragon Full đã có tích hợp sẵn Apache, Nginx, MySQL, PHP, Redis, Memcached, Node.js, npm, yarn, git ...

### 1.8.2 Lý do nên sử dụng Laragon thay XAMPP

- Nhanh và mượt hơn XAMPP rất nhiều
- Rất ít khi gặp lỗi (từ lúc mình sử dụng chưa gặp lỗi lần nào)
- Tên miền ảo và SSL được cài đặt tự động
- Dễ dàng tùy chỉnh, nâng cấp các Module
- Cài đặt nhanh các ứng dụng, CMS, Framework như WordPress, Laravel, Joomla, Drupla ...

## CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### 2.1 Phân tích thiết kế hướng đối tượng UML

#### 2.1.1 Khái niệm phân tích thiết kế hướng đối tượng

Khái niệm phân tích thiết kế hướng đối tượng (OOP): phân tích xem hệ thống gồm những đối tượng nào trong hệ thống và chúng tương tác, liên kết với nhau như thế nào, từ việc mô tả được tất cả các đối tượng và sự tương tác của chúng sẽ giúp chúng ta hiểu rõ về hệ thống và cài đặt được nó.

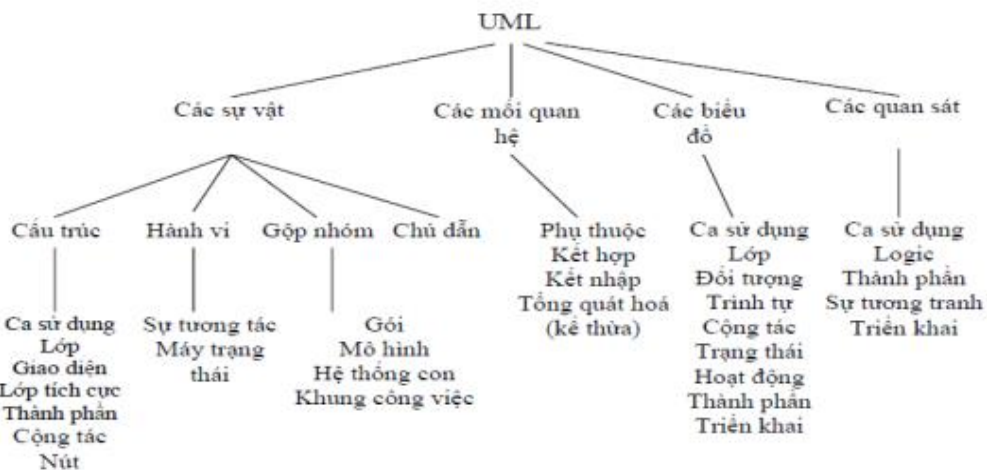
Ưu điểm của phân tích thiết kế hướng đối tượng: gần gũi với thế giới thực, dễ tái sử dụng, có thể thừa kế từ đó làm giảm chi phí và tính mở cao, hệ thống đáng tin cậy hơn.

Nhược điểm: phức tạp, khó theo dõi luồng dữ liệu hơn phân tích thiết kế hệ thống hướng chức năng.

#### 2.1.2 Khái niệm UML

Khái niệm UML (Unified Modeling Language): Ngôn ngữ mô hình hóa thống nhất, là ngôn ngữ dùng để đặc tả, trực quan hóa và tư liệu hóa phần mềm hướng đối tượng.

Các phần tử của UML:



Hình 2.1: Các phần tử của UML

- Các quan sát (góc nhìn, view) theo các phương diện khác nhau của hệ thống cần phân tích, thiết kế. Dựa vào các quan sát để thiết lập kiến trúc cho hệ thống cần

phát triển. Có năm loại quan sát: quan sát theo ca sử dụng, quan sát logic, quan sát thành phần, quan sát tương tranh và quan sát triển khai.

- Quan sát các ca sử dụng (Usecase View): Mô tả các chức năng, nhiệm vụ của hệ thống. Quan sát này thể hiện mọi yêu cầu của hệ thống.
  - Quan sát Logic (Logical View) biểu diễn cách tổ chức logic của các lớp và các quan hệ của chúng với nhau. Nó mô tả cấu trúc tĩnh của các lớp đối tượng và sự liên hệ của chúng thể hiện mối liên kết động thông qua sự trao đổi các thông điệp.
  - Quan sát thành phần (Component View) xác định các mô đun vật lý hay tệp mã chương trình và sự liên hệ giữa chúng để tổ chức thành hệ thống phần mềm.
  - Quan sát tiến trình (Process view) biểu diễn sự phân chia các luồng thực hiện công việc, các lớp đối tượng cho các tiến trình và sự đồng bộ giữa các luồng trong hệ thống.
  - Quan sát triển khai (Deployment view) mô tả sự phân bố tài nguyên và nhiệm vụ trong hệ thống. Nó liên quan đến các tầng kiến trúc của phần mềm.
- Biểu đồ (Diagram):

Là đồ thị biểu diễn đồ họa về tập các phần tử trong mô hình và mối quan hệ của chúng. Biểu đồ chứa đựng các nội dung của các quan sát dưới các góc độ khác nhau, một thành phần của hệ thống có thể xuất hiện trong một hay nhiều biểu đồ, có 9 loại biểu đồ khác nhau và được sử dụng kết hợp với nhau trong các trường hợp để cung cấp tất cả các hướng nhìn của một hệ thống.

- Biểu đồ ca sử dụng (Usecase diagram) mô tả sự tương tác giữa các tác nhân ngoài và hệ thống thông qua các ca sử dụng.
- Biểu đồ lớp (Class diagram): mô tả cấu trúc tĩnh, mô tả mô hình khái niệm bao gồm các lớp đối tượng và các mối quan hệ của chúng trong hệ thống hướng đối tượng.
- Biểu đồ đối tượng (Object diagram): là một phiên bản của biểu đồ lớp và thường cũng sử dụng các ký hiệu như biểu đồ lớp.

- Biểu đồ trình tự (Sequence diagram) thể hiện sự tương tác của các đối tượng với nhau.
  - Biểu đồ cộng tác (Collaboration diagram) tương tự như biểu đồ trình tự nhưng nhấn mạnh vào sự tương tác của các đối tượng trên cơ sở cộng tác với nhau bằng cách trao đổi các thông điệp để thực hiện các yêu cầu theo ngữ cảnh công việc.
  - Biểu đồ trạng thái (State diagram): thể hiện chu kỳ hoạt động của các đối tượng, của các hệ thống con và của cả hệ thống.
  - Biểu đồ hành động (Activity diagram) chỉ ra dòng hoạt động của hệ thống.
  - Biểu đồ thành phần (Component diagram) chỉ ra cấu trúc vật lý của các thành phần trong hệ thống.
  - Biểu đồ triển khai (Deployment Diagram) chỉ ra cách bố trí vật lý các thành phần theo kiến trúc được thiết kế của hệ thống.
- Các mối quan hệ (Relationship)
    - Hiện thực hoá (Realization): là quan hệ ngữ nghĩa giữa giao diện và lớp.
    - Phụ thuộc (Dependency): là quan hệ ngữ nghĩa giữa hai phần tử, trong đó sự thay đổi của một tử sẽ tác động đến ngữ nghĩa của phần tử phụ thuộc.
    - Kế thừa (Generalization): là quan hệ mô tả sự khái quát hoá mà trong đó một số đối tượng cụ thể (của lớp con) sẽ được kế thừa các thuộc tính, các phương thức của các đối tượng tổng quát (lớp cơ sở). Kết hợp (Association) là quan hệ cấu trúc xác định mối liên kết giữa các lớp đối tượng.
    - Các sự vật (Các phần tử mô hình\ model element): Các khái niệm được sử dụng trong các biểu đồ được gọi là các phần tử mô hình, thể hiện các khái niệm hướng đối tượng quen thuộc. Một phần tử mô hình thường được sử dụng trong nhiều biểu đồ khác nhau, nhưng nó luôn luôn có chỉ một ý nghĩa và một kí hiệu.

### 2.1.3 Phát biểu bài toán

Khoa Công nghệ Thông tin được thành lập từ ngày 10/06/2011 với khởi đầu là Bộ môn Tin học (năm 1997). Trải qua hơn 22 năm hoạt động, Khoa Công nghệ Thông

tin đã có những bước phát triển vững chắc, xứng đáng là nơi đào tạo chủ lực, cung cấp nguồn nhân lực Công nghệ Thông tin uy tín trong nhiều năm qua của Trường Đại học Dân lập Hải Phòng (nay là Trường Đại học Quản lý và Công nghệ Hải Phòng).

Khoa Công nghệ Thông tin xác định sứ mệnh của Khoa trong Lĩnh vực Công nghệ Thông tin là:

1. Đào tạo nguồn nhân lực Công nghệ Thông tin chất lượng cao, có kiến thức - kỹ năng - đạo đức nghề nghiệp đáp ứng nhu cầu nhân lực công nghệ thông tin trong các lĩnh vực chuyên môn khác nhau.
2. Nghiên cứu khoa học, ứng dụng và chuyển giao công nghệ.

Các chuyên ngành đào tạo bậc đại học:

- Công nghệ phần mềm,
- Quản trị và an ninh Mạng,
- Digital Marketing,
- Tin học - Kinh tế.

Chuyên ngành đào tạo bậc Thạc sỹ: Hệ thống Thông tin.

Các hướng nghiên cứu khoa học:

- Công nghệ phần mềm
- Hệ thống thời gian thực
- Học máy, trí tuệ nhân tạo
- Xử lý ngôn ngữ Tự nhiên
- Phân tích quan điểm người dùng
- Xử lý dữ liệu lớn
- An ninh mạng
- Các hệ thống phân tán
- An toàn và bảo mật thông tin.

Website của Khoa cần có 2 giao diện gồm:

- **Giao diện người dùng:** Cần có các danh mục như:
  - Danh mục “*Giới thiệu*” nhằm giới thiệu thông tin về khoa.
  - Danh mục “*Tin tức - Sự kiện*” giúp người dùng có thể nắm bắt các tin tức, sự kiện, thông báo, cũng như thông tin tuyển sinh nhanh và hiệu quả nhất.
  - Danh mục “*Đào tạo*” gồm các chương trình đào tạo của Khoa.
  - Danh mục “*Sinh viên*” chứa hoạt động của sinh viên khoa, cơ hội việc làm, tài liệu cũng như câu hỏi thường gặp giúp hỗ trợ sinh viên nắm bắt thông tin 1 cách nhanh và hiệu quả nhất.
  - Danh mục “*Quan hệ hợp tác*” là nơi chứa thông tin tuyển dụng, nơi mà các doanh nghiệp, tập đoàn liên kết, hợp tác với khoa.
  - Danh mục “*Liên hệ*” giúp người dùng nếu có bất kỳ câu hỏi hay thắc mắc nào có thể nhắn tin trực tiếp và trực tiếp được quản trị viên của Khoa trả lời.
- **Giao diện Quản trị viên:** bao gồm đầy đủ chức năng thêm, sửa, xóa, hiển thị các danh mục như:
  - *Slide:* Quản lý các ảnh slide chạy trên trang index của hệ thống.
  - *Tài khoản:* Quản lý các tài khoản người dùng bao gồm cả tài khoản admin và tài khoản user. Quản trị viên được quyền xóa các tài khoản user.
  - *Chuyên ngành:* Quản lý các chuyên ngành có trong chương trình của Khoa.
  - *Giảng viên:* Quản lý thông tin, trạng thái các giảng viên đã và đang công tác tại Khoa.
  - *Chuyên mục:* Quản lý các chuyên mục mà quản trị viên muốn hiển thị trên thanh nav của website.
  - *Bài viết:* Quản lý các bài viết, tin tức, sự kiện... của Khoa.
  - *Liên hệ:* Quản lý các liên hệ giúp giải đáp các thắc mắc của người dùng tại giao diện “*Liên Hệ*” của website.

Việc tạo ra 1 website riêng cho khoa là 1 công việc cần thiết và cấp bách. Thông qua website, khoa CNTT sẽ tiếp cận đến sinh viên của khoa nói chung cũng như mọi người nói riêng 1 cách nhanh nhất, hiệu quả nhất, nắm thông tin kịp thời cũng như giải

đáp thắc mắc người dùng mà nếu như không có website thì đôi lúc sẽ tốn rất nhiều thời gian.

Tất nhiên, website được tạo ra sẽ phải tương thích với các thiết bị di động phổ biến trên thị trường, hướng đến trải nghiệm người dùng cũng như tối ưu chuẩn SEO giúp cho mọi người tìm kiếm dễ dàng hơn trên các trình duyệt.

## 2.2 Các yêu cầu phi chức năng

Bảng 2.1: Các yêu cầu phi chức năng

Mục	Tên yêu cầu	Mô tả yêu cầu
1	Giao diện	Giao diện hệ thống phải dễ sử dụng, trực quan, thân thiện với mọi người dùng.
2	Tốc độ xử lý	Hệ thống phải xử lý nhanh chóng, tối ưu và chính xác.
3	Tương thích	Tương thích với đa phần các trình duyệt web trên điện thoại và máy tính.

## 2.3 Phân tích thiết kế hệ thống

### 2.3.1 Thiết kế hệ thống

Website được chia làm 2 phần:

#### - **Front-End:**

Front - end của Website được xây dựng trên các tiêu chí:

- Giao diện thân thiện đơn giản dễ sử dụng.
- Nội dung hấp dẫn, ngắn gọn và dễ hiểu.
- Thao tác người dùng trên hệ thống thuận tiện, dễ dàng.
- Các thông tin bài viết, giới thiệu quản trị viên luôn được cập nhật và có thể chia sẻ bài viết lên các trang mạng xã hội như Facebook, Zalo...
- Cho phép gửi thông tin liên hệ.

Để xây dựng được Front - end:

- Website sử dụng HTML5, được phát triển trên nền tảng ngôn ngữ HTML và quan trọng nhất của world wide web.



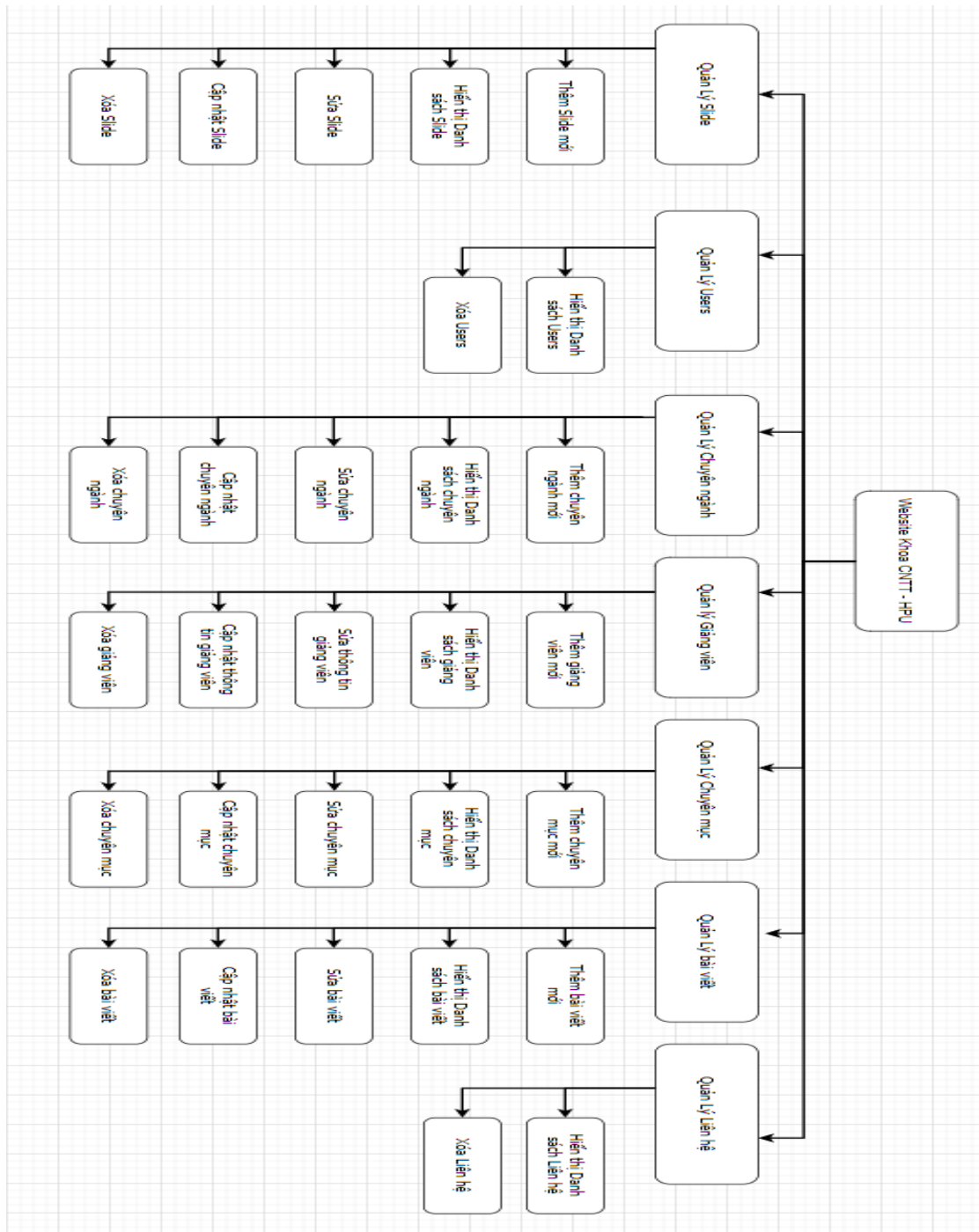
- Website sử dụng Css làm ngôn ngữ tạo phong cách cho trang web.
- Website sử dụng framework Bootstrap 4, bao gồm HTML, CSS và JavaScript template để phát triển website chuẩn Responsive tương thích trên các loại màn hình trên điện thoại và máy tính bảng...
- Website sử dụng ngôn ngữ lập trình Javascript và thư viện JQuery.
- Website sử dụng font chữ là: Inter

- **Back – End:**

Quản trị hệ thống có quyền quản lý toàn bộ hoạt động của Website:

- *Quản lý Slide:* Cho phép admin được quyền xem, thêm, sửa, xóa Slide.
- *Quản lý User:* Cho phép quyền xem danh sách người dùng, danh sách quản trị viên và xóa thông tin người dùng.
- *Quản lý Chuyên ngành:* Cho phép admin quyền thêm, sửa hoặc xóa chuyên ngành khi trường có các cải cách hoặc đổi mới chuyên ngành đào tạo.
- *Quản lý Giảng viên:* Cho phép thêm, sửa, xóa, thay đổi trạng thái của Giảng viên.
- *Quản lý Chuyên mục:* Cho phép thêm, sửa, xóa chuyên mục để phù hợp với nhu cầu mong muốn.
- *Quản lý Bài viết:* Các bài viết sẽ thuộc các chuyên mục khác nhau. Admin được quyền thêm, sửa, xóa bài viết.
- *Quản lý Liên hệ:* Cho phép xem, trả lời và xóa liên hệ.

### 2.3.2 Sơ đồ phân cấp chức năng



Hình 2.2: Sơ đồ phân rã chức năng

Mô tả chức năng:

- *Quản lý Slide:*

Hệ thống cho phép quản lý slide chạy tại trang index của website gồm các chức năng thêm, sửa, xóa tại giao diện của quản trị viên. Mỗi Slide sẽ có ảnh slide, tiêu đề slide và mô tả slide giúp hỗ trợ hơn trong SEO, sắp xếp Slide giúp việc quản lý thứ tự ưu tiên slide trên website.

- *Quản lý chuyên mục:*

Hệ thống cho phép quản lý danh sách các chuyên mục cha – con và có thể thêm, sửa, xóa các chuyên mục. Khi xóa chuyên mục cha, hệ thống sẽ tự động xóa các chuyên mục là chuyên mục con của chuyên mục đó. Mỗi chuyên mục sẽ có thêm cột slug tức là lấy ra phần tên viết không dấu và nối nhau bằng dấu “-” để tạo querystring, cột mô tả chuyên mục, các cột này khi điều hướng trang web sẽ hỗ trợ SEO.

- *Quản lý bài viết:*

Hệ thống cho phép quản lý các danh sách bài viết, mỗi bài viết khác nhau có chuyên mục khác nhau, có các chức năng thêm, sửa, xóa và xem chi tiết bài viết. Một bài viết sẽ gồm tên bài viết, bài viết thuộc chuyên mục nào, ảnh bài viết, tác giả bài viết, ảnh bài viết, file đính kèm (nếu có), nội dung bài viết và slug nhằm hỗ trợ SEO website.

- *Quản lý chuyên ngành:*

Hệ thống cho phép quản trị viên thực hiện đầy đủ các chức năng như thêm, sửa, xóa. Mỗi chuyên ngành gồm thông tin cơ bản như số sinh viên đang theo học, hình thức đào tạo, thời gian đào tạo, ảnh đại diện cho chuyên ngành, nội dung chuyên ngành và slug.

- *Quản lý giảng viên:*

Quản lý trạng thái của các giảng viên trong Khoa, truy cập hệ thống sẽ có quyền thêm, sửa, xóa giảng viên.

- *Quản lý Users:*

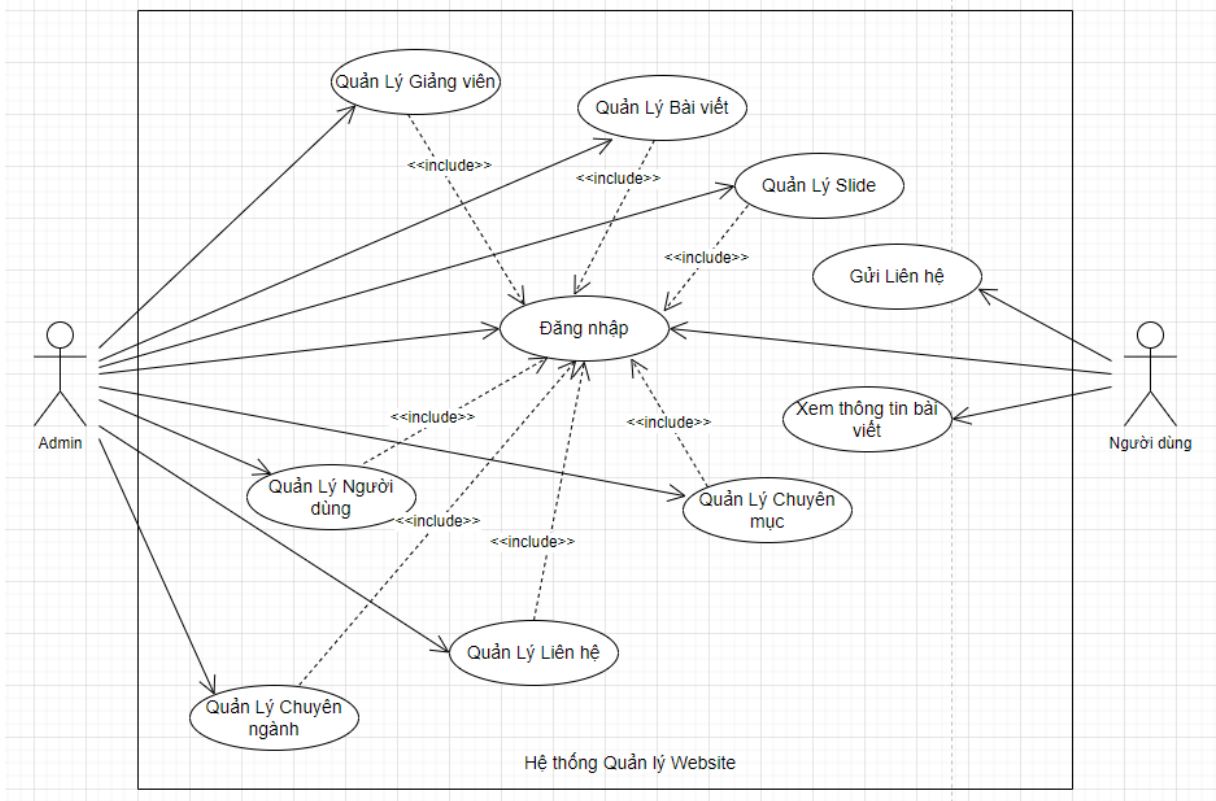
Hệ thống sẽ có các chức năng thêm, sửa, xóa người dùng, hiển thị danh sách người dùng và danh sách quản trị viên.

- *Quản lý Liên hệ:* Cho phép quản trị viên xem danh sách liên hệ, xóa liên hệ.

## **2.4 Biểu đồ Use case**

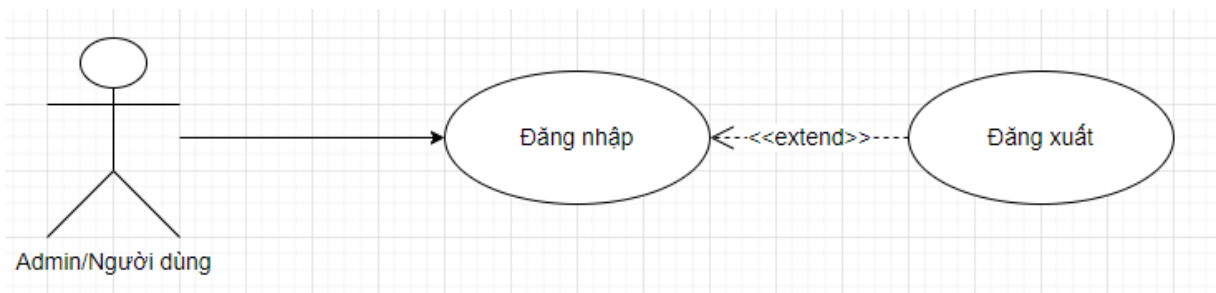
Biểu đồ ca sử dụng (Usecase diagram) mô tả sự tương tác giữa các tác nhân ngoài và hệ thống thông qua các ca sử dụng.

### 2.4.1 Biểu đồ Usecase tổng quát



Hình 2.3: Biểu đồ Usecase tổng quát

### 2.4.2 Use case đăng nhập



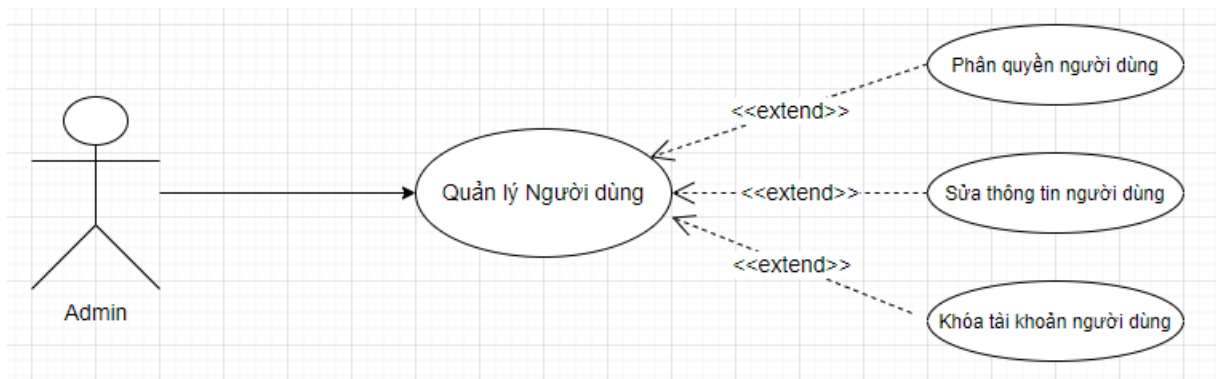
Hình 2.4: Biểu đồ Use case đăng nhập

Bảng 2.2: Kịch bản chức năng đăng nhập

Tên Usecase	Đăng nhập
Tác nhân chính	Admin/User
Người chịu trách nhiệm:	Admin/User
Tiền điều kiện:	Admin/User chưa đăng nhập vào hệ thống.
Đảm bảo tối thiểu:	Nhập đầy đủ thông tin đăng nhập.

Đảm bảo thành công:	Đăng nhập hệ thống thành công.
Kích hoạt:	Người dùng chọn chức năng đăng nhập.
Chuỗi sự kiện chính: <ol style="list-style-type: none"> <li>1. Khi chọn chức năng đăng nhập.</li> <li>2. Hệ thống hiển thị form đăng nhập và yêu cầu người dùng điền vào thông tin đăng nhập.</li> <li>3. Hệ thống kiểm tra thông tin đăng nhập và xác nhận thông tin đăng nhập hợp lệ.</li> <li>4. Hệ thống thông báo đã đăng nhập thành công.</li> </ol>	
Ngoại lệ: <ol style="list-style-type: none"> <li>1. Hệ thống báo thông tin đăng nhập sai hoặc không tồn tại.</li> <li>2. Hệ thống yêu cầu nhập lại thông tin đăng nhập.</li> <li>3. Admin/User nhập lại thông tin đăng nhập.</li> </ol>	

### 2.4.3 Use case quản lý người dùng



Hình 2.5: Biểu đồ Use case quản lý người dùng

Với use case quản lý người dùng, ta xác định được kịch bản: Xóa người dùng.

Bảng 2.3: Kịch bản chức năng quản lý người dùng

Tên Usecase	Quản lý Người dùng
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Đăng nhập

Đảm bảo tối thiểu:	Admin đã đăng nhập thành công
Đảm bảo thành công:	Admin thấy được danh sách người dùng và tiến hành hành động
Kích hoạt:	Admin chọn mục “Tài Khoản”
Chuỗi sự kiện chính: <ol style="list-style-type: none"> <li>Admin chọn mục “Tài Khoản”.</li> <li>Hệ thống truy xuất thông tin rồi trả về danh sách người dùng.</li> </ol>	

*Bảng 2.4: Kịch bản chức năng phân quyền người dùng*

Tên Usecase	Phân quyền Người dùng
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống và vào mục Tài khoản.
Đảm bảo tối thiểu:	Hiển thị danh sách người dùng.
Đảm bảo thành công:	Admin thấy được danh sách người dùng và tiến hành hành động.
Kích hoạt:	Quản lý chọn chức năng phân quyền trong mục phân quyền Người dùng.
Chuỗi sự kiện chính: <ol style="list-style-type: none"> <li>Admin chọn vào cột “Phân quyền”.</li> <li>Admin nhấn nút “ADMIN” hoặc “USERS”.</li> </ol>	
Ngoại lệ: <ol style="list-style-type: none"> <li>Admin không nhấn “ADMIN” hoặc “USERS”.</li> </ol>	

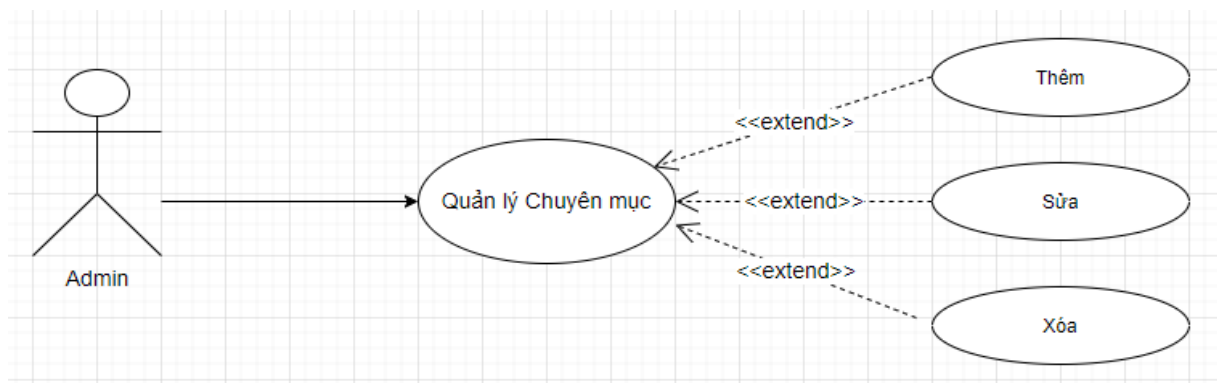
Bảng 2.5: Kịch bản chức năng sửa thông tin người dùng

Tên Usecase	Sửa thông tin người dùng
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống và vào mục Tài khoản.
Đảm bảo tối thiểu:	Hiện thị form sửa thông tin người dùng.
Đảm bảo thành công:	Chức năng sửa thông tin người dùng sẽ được thực hiện trong CSDL.
Kích hoạt:	Admin chọn chức năng sửa trong danh sách người dùng.
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>1. Admin lựa chọn tên người dùng cần chỉnh sửa.</li> <li>2. Hệ thống hiển thị form sửa thông tin người dùng.</li> <li>3. Admin nhập các thông tin cần sửa và bấm hoàn thành</li> <li>4. Hệ thống thông báo “Cập nhật thông tin người dùng thành công”</li> <li>5. Hệ thống thoát khỏi chức năng sửa thông tin người dùng.</li> </ol>	
<p>Ngoại lệ:</p> <ol style="list-style-type: none"> <li>1. Admin sửa để trống form tên, email người dùng.</li> <li>2. Hệ thống hiển thị thông báo “Vui lòng nhập tên(hoặc email) Người dùng!”.</li> </ol>	

Bảng 2.6: Kịch bản chức năng khóa tài khoản người dùng

Tên Usecase	Khóa tài khoản Người dùng
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống và vào mục Tài khoản.
Đảm bảo tối thiểu:	Hệ thống tự loại bỏ người dùng đã chọn.
Đảm bảo thành công:	Admin thấy được danh sách người dùng và tiến hành hành động.
Kích hoạt:	Quản lý chọn chức năng khóa tài khoản trong mục quản lý Người dùng.
Chuỗi sự kiện chính:	<ol style="list-style-type: none"> <li>Admin chọn vào nút khóa tài khoản người dùng.</li> <li>Hệ thống thông báo “khóa tài khoản người dùng thành công”.</li> <li>Admin trở về danh sách người dùng.</li> </ol>
Ngoại lệ:	<ol style="list-style-type: none"> <li>Admin không nhấn “OK” mà nhấn “Hủy”.</li> </ol>

#### 2.4.4 Use case quản lý chuyên mục



Hình 2.6: Biểu đồ Use case quản lý chuyên mục



Với use case quản lý chuyên mục, ta xác định được các kịch bản như sau:

- Thêm chuyên mục
- Sửa chuyên mục
- Xóa chuyên mục

*Bảng 2.7: Kịch bản chức năng quản lý chuyên mục*

Tên Usecase	Quản lý Chuyên mục
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Đăng nhập
Đảm bảo tối thiểu:	Admin đã đăng nhập thành công
Đảm bảo thành công:	Admin thấy được danh sách chuyên mục và tiến hành hành động
Kích hoạt:	Admin chọn mục “Danh sách Chuyên mục”
Chuỗi sự kiện chính: <ol style="list-style-type: none"> <li>1. Admin chọn mục “Danh sách Chuyên mục”.</li> <li>2. Hệ thống truy xuất thông tin rồi trả về danh sách chuyên mục.</li> </ol>	

*Bảng 2.8: Kịch bản chức năng thêm chuyên mục*

Tên Usecase	Quản lý Chuyên mục
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập thành công
Đảm bảo tối thiểu:	Hiện thị form thêm chuyên mục

Đảm bảo thành công:	Chuyên mục được lưu vào CSDL, hệ thống thông báo thêm chuyên mục thành công.
Kích hoạt:	Admin chọn chức năng “Thêm Chuyên mục”.
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>1. Admin ấn nút “Thêm Chuyên mục”.</li> <li>2. Hệ thống hiển thị form thêm chuyên mục</li> <li>3. Admin nhập các thông tin trong form thêm chuyên mục</li> <li>4. Admin bấm nút “Thêm”.</li> <li>5. Hệ thống thông báo thêm thành công</li> <li>6. Hệ thống thoát khỏi chức năng thêm chuyên mục.</li> </ol>	

*Bảng 2.9: Kịch bản chức năng sửa chuyên mục*

Tên Usecase	Sửa chuyên mục
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống và vào mục quản lý chuyên mục.
Đảm bảo tối thiểu:	Hiển thị form sửa chuyên mục
Đảm bảo thành công:	Chức năng sửa chuyên mục sẽ được thực hiện trong CSDL.
Kích hoạt:	Admin chọn chức năng sửa chuyên mục trong danh sách chuyên mục.

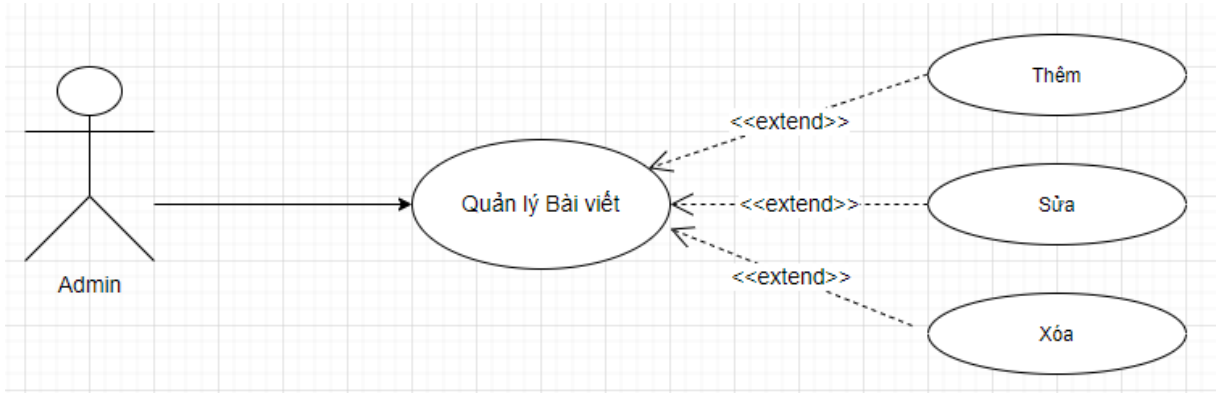
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>1. Admin lựa chọn tên chuyên mục cần chỉnh sửa.</li> <li>2. Hệ thống hiển thị form sửa chuyên mục</li> <li>3. Admin nhập các thông tin cần sửa và bấm hoàn thành</li> <li>4. Hệ thống thông báo “Cập nhật chuyên mục thành công”</li> <li>5. Hệ thống thoát khỏi chức năng sửa chuyên mục.</li> </ol>
<p>Ngoại lệ:</p> <ol style="list-style-type: none"> <li>1. Admin sửa để trống form tên chuyên mục.</li> <li>2. Hệ thống hiển thị thông báo “Vui lòng nhập tên Chuyên mục!”.</li> </ol>

*Bảng 2.10: Kịch bản chức năng xóa chuyên mục*

Tên Usecase	Xóa chuyên mục
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống.
Đảm bảo tối thiểu:	Hệ thống tự loại bỏ người dùng đã chọn.
Đảm bảo thành công:	Chức năng xóa sẽ được thực hiện trong CSDL
Kích hoạt:	Quản lý chọn chức năng xóa trong mục quản lý Chuyên mục.
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>1. Admin lựa chọn chuyên mục cần xóa.</li> <li>2. Hệ thống hiển thị thông báo “Xóa mà không thể khôi phục. Ban có chắc?”.</li> <li>3. Admin nhấn “OK”.</li> <li>4. Hệ thống thông báo “Xóa thành công”.</li> </ol>	

5. Admin trở về danh sách chuyên mục.
Ngoại lệ: 1. Admin không nhấn “OK” mà nhấn “Hủy”.

#### 2.4.5 Use case quản lý bài viết



Hình 2.7: Biểu đồ Use case quản lý bài viết

Với use case quản lý bài viết, ta xác định được các kịch bản như sau:

- Thêm bài viết
- Sửa thông tin bài viết
- Xóa bài viết

Bảng 2.11: Kịch bản chức năng quản lý bài viết

Tên Usecase	Quản lý Bài viết
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Đăng nhập
Đảm bảo tối thiểu:	Admin đã đăng nhập thành công
Đảm bảo thành công:	Admin thấy được danh sách bài viết và tiến hành hành động
Kích hoạt:	Admin chọn mục “Danh sách Bài viết”
Chuỗi sự kiện chính:	<ol style="list-style-type: none"> <li>1. Admin chọn mục “Danh sách Bài viết”.</li> <li>2. Hệ thống truy xuất thông tin rồi trả về danh sách bài viết.</li> </ol>

Bảng 2.12: Kịch bản chức năng thêm bài viết

Tên Usecase	Quản lý Bài viết
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập thành công
Đảm bảo tối thiểu:	Hiện thị form thêm bài viết
Đảm bảo thành công:	Bài viết được lưu vào CSDL, hệ thống thông báo thêm bài viết thành công.
Kích hoạt:	Admin chọn chức năng thêm bài viết trong “Danh sách Bài viết”.
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>1. Admin nút thêm chuyên mục trong mục “Bài viết”.</li> <li>2. Hệ thống hiển thị form thêm bài viết</li> <li>3. Admin nhập các thông tin trong form thêm bài viết</li> <li>4. Admin bấm nút “Thêm”.</li> <li>5. Hệ thống thông báo thêm thành công</li> <li>6. Hệ thống thoát khỏi chức năng thêm bài viết.</li> </ol>	

Bảng 2.13: Kịch bản chức năng sửa bài viết

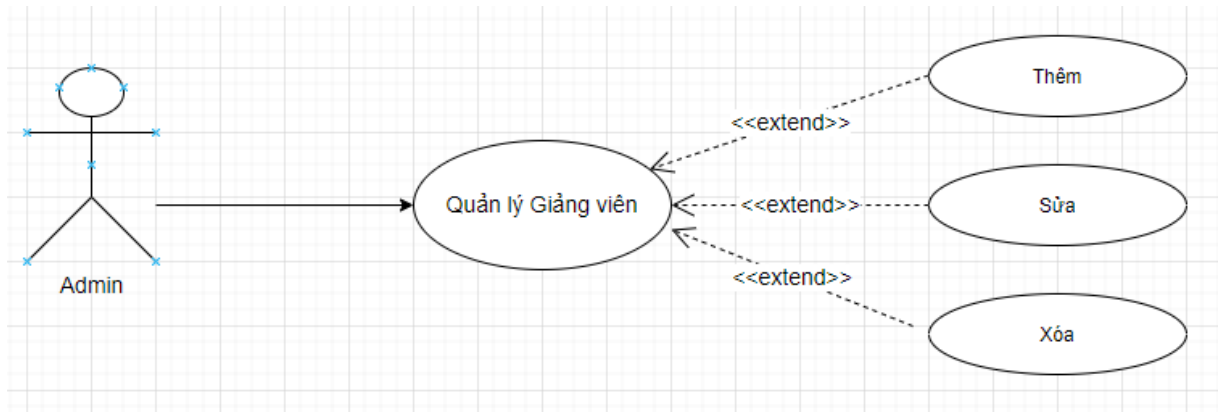
Tên Usecase	Sửa bài viết
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống và vào mục quản lý bài viết.
Đảm bảo tối thiểu:	Hiện thị form sửa bài viết
Đảm bảo thành công:	Chức năng sửa bài viết sẽ được thực hiện trong CSDL.
Kích hoạt:	Admin chọn chức năng sửa bài viết trong danh sách bài viết.
<p>Chuỗi sự kiện chính:</p>	

<ol style="list-style-type: none"> <li>Admin lựa chọn tên bài viết cần chỉnh sửa.</li> <li>Hệ thống hiển thị form sửa bài viết</li> <li>Admin nhập các thông tin cần sửa và bấm hoàn thành</li> <li>Hệ thống thông báo “Cập nhật bài viết thành công”</li> <li>Hệ thống thoát khỏi chức năng sửa bài viết.</li> </ol>
<p>Ngoại lệ:</p> <ol style="list-style-type: none"> <li>Admin sửa để trống form tên bài viết.</li> <li>Hệ thống hiển thị thông báo “Vui lòng nhập tiêu đề Bài viết!”.</li> </ol>

Bảng 2.14: Kịch bản chức năng xóa bài viết

Tên Usecase	Xóa bài viết
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống.
Đảm bảo tối thiểu:	Hệ thống tự loại bỏ người dùng đã chọn.
Đảm bảo thành công:	Chức năng xóa sẽ được thực hiện trong CSDL
Kích hoạt:	Quản lý chọn chức năng xóa trong mục quản lý Bài viết.
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>Admin lựa chọn bài viết cần xóa.</li> <li>Hệ thống hiển thị thông báo “Xóa mà không thể khôi phục. Bạn có chắc?”.</li> <li>Admin nhấn “OK”.</li> <li>Hệ thống thông báo “Xóa thành công”.</li> <li>Admin trở về danh sách bài viết.</li> </ol>	
<p>Ngoại lệ:</p> <ol style="list-style-type: none"> <li>Admin không nhấn “OK” mà nhấn “Hủy”.</li> </ol>	

## 2.4.6 Use case quản lý giảng viên



Hình 2.8: Biểu đồ Use case quản lý giảng viên

Với use case quản lý giảng viên, ta xác định được các kịch bản như sau:

- Thêm giảng viên mới
- Sửa thông tin giảng viên
- Xóa giảng viên

Bảng 2.15: Kịch bản chức năng quản lý giảng viên

Tên Usecase	Quản lý Giảng viên
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Đăng nhập
Đảm bảo tối thiểu:	Admin đã đăng nhập thành công
Đảm bảo thành công:	Admin thấy được danh sách giảng viên và tiến hành hành động
Kích hoạt:	Admin chọn mục “Danh sách Giảng viên”
Chuỗi sự kiện chính:	<ol style="list-style-type: none"> <li>1. Admin chọn mục “Danh sách Giảng viên”.</li> <li>2. Hệ thống truy xuất thông tin rồi trả về danh sách giảng viên.</li> </ol>

*Bảng 2.16: Kịch bản chức năng thêm giảng viên*

Tên Usecase	Quản lý Giảng viên
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập thành công
Đảm bảo tối thiểu:	Hiện thị form thêm giảng viên
Đảm bảo thành công:	Bài viết được lưu vào CSDL, hệ thống thông báo thêm chuyên mục thành công.
Kích hoạt:	Admin chọn chức năng thêm giảng viên trong “Thêm giảng viên”.
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>1. Admin chọn nút “Thêm giảng viên”.</li> <li>2. Hệ thống hiển thị form thêm giảng viên</li> <li>3. Admin nhập các thông tin trong form thêm giảng viên</li> <li>4. Admin bấm nút “Thêm”.</li> <li>5. Hệ thống thông báo thêm thành công</li> <li>6. Hệ thống thoát khỏi chức năng thêm giảng viên.</li> </ol>	

*Bảng 2.17: Kịch bản chức năng sửa giảng viên*

Tên Usecase	Sửa giảng viên
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống và vào mục quản lý giảng viên.



Đảm bảo tối thiểu:	Hiện thị form sửa giảng viên
Đảm bảo thành công:	Chức năng sửa giảng viên sẽ được thực hiện trong CSDL.
Kích hoạt:	Admin chọn chức năng sửa giảng viên trong danh sách giảng viên.
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>1. Admin lựa chọn tên giảng viên cần chỉnh sửa.</li> <li>2. Hệ thống hiển thị form sửa giảng viên</li> <li>3. Admin nhập các thông tin cần sửa và bấm hoàn thành</li> <li>4. Hệ thống thông báo “Cập nhật thông tin giảng viên thành công”</li> <li>5. Hệ thống thoát khỏi chức năng sửa giảng viên.</li> </ol>	
<p>Ngoại lệ:</p> <ol style="list-style-type: none"> <li>1. Admin sửa để trống form tên giảng viên.</li> <li>2. Hệ thống hiển thị thông báo “Vui lòng nhập tên giảng viên!”.</li> </ol>	

Bảng 2.18: Kịch bản chức năng xóa giảng viên

Tên Usecase	Xóa giảng viên
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống.
Đảm bảo tối thiểu:	Hệ thống tự loại bỏ người dùng đã chọn.
Đảm bảo thành công:	Chức năng xóa sẽ được thực hiện trong CSDL
Kích hoạt:	Quản lý chọn chức năng xóa trong mục quản lý Giảng viên.

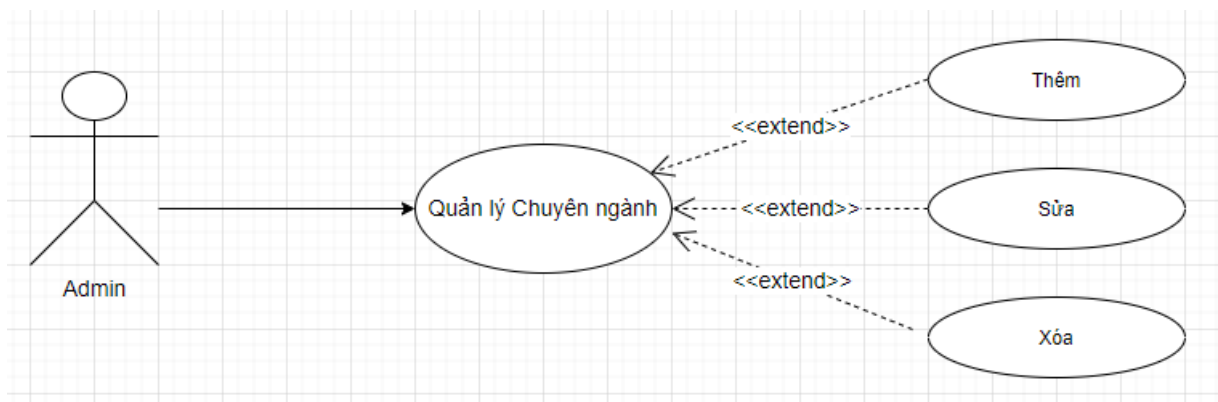
Chuỗi sự kiện chính:

1. Admin lựa chọn giảng viên cần xóa.
2. Hệ thống hiển thị thông báo “Xóa mà không thể khôi phục. Bạn có chắc?”.
3. Admin nhấn “OK”.
4. Hệ thống thông báo “Xóa thành công”.
5. Admin trở về danh sách giảng viên.

Ngoại lệ:

1. Admin không nhấn “OK” mà nhấn “Hủy”.

#### 2.4.7 Use case quản lý chuyên ngành



Hình 2.9: Biểu đồ Use case quản lý chuyên ngành

Với use case quản lý chuyên ngành, ta xác định được các kịch bản như sau:

- Thêm chuyên ngành
- Sửa chuyên ngành
- Xóa chuyên ngành

Bảng 2.19: Kịch bản chức năng quản lý chuyên ngành

Tên Usecase	Quản lý Chuyên ngành
Tác nhân chính	Admin

Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Đăng nhập
Đảm bảo tối thiểu:	Admin đã đăng nhập thành công
Đảm bảo thành công:	Admin thấy được danh sách chuyên ngành và tiến hành hành động
Kích hoạt:	Admin chọn mục “Danh sách Chuyên ngành”
Chuỗi sự kiện chính: <ol style="list-style-type: none"> <li>1. Admin chọn mục “Danh sách Chuyên ngành”.</li> <li>2. Hệ thống truy xuất thông tin rồi trả về danh sách chuyên ngành.</li> </ol>	

*Bảng 2.20: Kịch bản chức năng thêm chuyên ngành*

Tên Usecase	Quản lý Chuyên ngành
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập thành công
Đảm bảo tối thiểu:	Hiện thị form thêm chuyên ngành
Đảm bảo thành công:	Bài viết được lưu vào CSDL, hệ thống thông báo thêm chuyên ngành thành công.
Kích hoạt:	Admin chọn chức năng thêm chuyên ngành trong “Thêm chuyên ngành”.

*Chuỗi sự kiện chính:*

1. Admin chọn nút “Thêm chuyên ngành”.
2. Hệ thống hiển thị form thêm chuyên ngành
3. Admin nhập các thông tin trong form thêm chuyên ngành
4. Admin bấm nút “Thêm”.
5. Hệ thống thông báo thêm thành công
6. Hệ thống thoát khỏi chức năng thêm chuyên ngành.

*Bảng 2.21: Kịch bản chức năng sửa chuyên ngành*

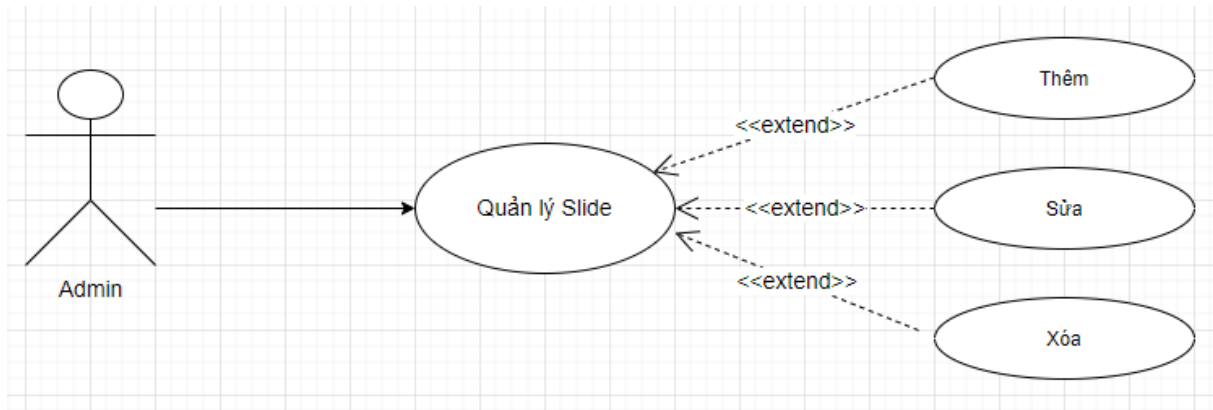
Tên Usecase	Sửa chuyên ngành
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống và vào mục quản lý chuyên ngành.
Đảm bảo tối thiểu:	Hiện thị form sửa chuyên ngành
Đảm bảo thành công:	Chức năng sửa chuyên ngành sẽ được thực hiện trong CSDL.
Kích hoạt:	Admin chọn chức năng sửa chuyên ngành trong danh sách chuyên ngành.
<i>Chuỗi sự kiện chính:</i> <ol style="list-style-type: none"><li>1. Admin lựa chọn tên chuyên ngành cần chỉnh sửa.</li><li>2. Hệ thống hiển thị form sửa chuyên ngành</li><li>3. Admin nhập các thông tin cần sửa và bấm hoàn thành</li><li>4. Hệ thống thông báo “Cập nhật chuyên ngành thành công”</li></ol>	

5. Hệ thống thoát khỏi chức năng sửa chuyên ngành
Ngoại lệ:
3. Admin sửa để trống form tên chuyên ngành.
4. Hệ thống hiển thị thông báo “Vui lòng nhập tên Chuyên ngành!”.

Bảng 2.22: Kịch bản chức năng xóa chuyên ngành

Tên Usecase	Xóa chuyên ngành
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống.
Đảm bảo tối thiểu:	Hệ thống tự loại bỏ người dùng đã chọn.
Đảm bảo thành công:	Chức năng xóa sẽ được thực hiện trong CSDL
Kích hoạt:	Quản lý chọn chức năng xóa trong mục quản lý Chuyên ngành.
Chuỗi sự kiện chính:	
<ol style="list-style-type: none"> <li>1. Admin lựa chọn chuyên ngành cần xóa.</li> <li>2. Hệ thống hiển thị thông báo “Xóa mà không thể khôi phục. Bạn có chắc?”.</li> <li>3. Admin nhấn “OK”.</li> <li>4. Hệ thống thông báo “Xóa thành công”.</li> <li>5. Admin trở về danh sách chuyên ngành.</li> </ol>	
Ngoại lệ:	
<ol style="list-style-type: none"> <li>1. Admin không nhấn “OK” mà nhấn “Hủy”.</li> </ol>	

## 2.4.8 Use case quản lý slide



Hình 2.10: Biểu đồ Use case quản lý slide

Với use case quản lý slide, ta xác định được các kịch bản như sau:

- Thêm slide
- Sửa slide
- Xóa slide

Bảng 2.23: Kịch bản chức năng quản lý slide

Tên Usecase	Quản lý Slide
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Đăng nhập
Đảm bảo tối thiểu:	Admin đã đăng nhập thành công
Đảm bảo thành công:	Admin thấy được danh sách slide và tiến hành hành động
Kích hoạt:	Admin chọn mục “Danh sách Slide”
Chuỗi sự kiện chính:	<ol style="list-style-type: none"> <li>1. Admin chọn mục “Danh sách Slide”.</li> <li>2. Hệ thống truy xuất thông tin rồi trả về danh sách slide.</li> </ol>

Bảng 2.24: Kịch bản chức năng thêm slide

Tên Usecase	Quản lý Slide
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập thành công
Đảm bảo tối thiểu:	Hiện thị form thêm slide
Đảm bảo thành công:	Chuyên mục được lưu vào CSDL, hệ thống thông báo thêm chuyên mục thành công.
Kích hoạt:	Admin chọn chức năng “Thêm Slide”.
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>1. Admin ấn nút “Thêm Slide”.</li> <li>2. Hệ thống hiển thị form thêm slide</li> <li>3. Admin nhập các thông tin trong form thêm slide</li> <li>4. Admin bấm nút “Thêm”.</li> <li>5. Hệ thống thông báo thêm thành công</li> <li>6. Hệ thống thoát khỏi chức năng thêm slide.</li> </ol>	

*Bảng 2.25: Kịch bản chức năng sửa slide*

Tên Usecase	Sửa slide
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống và vào mục quản lý slide.
Đảm bảo tối thiểu:	Hiện thị form sửa slide

Đảm bảo thành công:	Chức năng sửa slide sẽ được thực hiện trong CSDL.
Kích hoạt:	Admin chọn chức năng sửa slide trong danh sách slide.
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>1. Admin lựa chọn tên slide cần chỉnh sửa.</li> <li>2. Hệ thống hiển thị form sửa slide</li> <li>3. Admin nhập các thông tin cần sửa và bấm hoàn thành</li> <li>4. Hệ thống thông báo “Cập nhật slide thành công”</li> <li>5. Hệ thống thoát khỏi chức năng sửa slide.</li> </ol>	
<p>Ngoại lệ:</p> <ol style="list-style-type: none"> <li>1. Admin sửa để trống form tên slide.</li> <li>2. Hệ thống hiển thị thông báo “Vui lòng nhập tiêu đề slide!”.</li> <li>3. Admin sửa để trống ảnh slide.</li> <li>4. Hệ thống hiển thị thông báo “Vui lòng thêm ảnh slide!”.</li> </ol>	

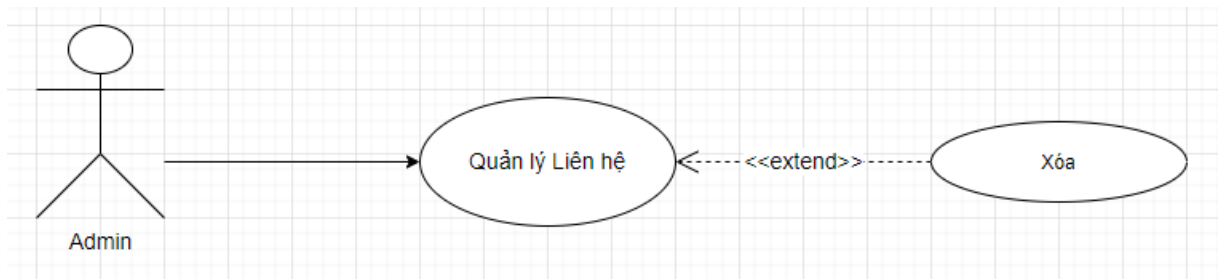
Bảng 2.26: Kịch bản chức năng xóa slide

Tên Usecase	Xóa slide
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống.
Đảm bảo tối thiểu:	Hệ thống tự loại bỏ người dùng đã chọn.
Đảm bảo thành công:	Chức năng xóa sẽ được thực hiện trong CSDL
Kích hoạt:	Quản lý chọn chức năng xóa trong mục quản lý



	Slide.
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>1. Admin lựa chọn chuyên mục cần xóa.</li> <li>2. Hệ thống hiển thị thông báo “Xóa mà không thể khôi phục. Bạn có chắc?”.</li> <li>3. Admin nhấn “OK”.</li> <li>4. Hệ thống thông báo “Xóa thành công”.</li> <li>5. Admin trở về danh sách slide.</li> </ol>	
<p>Ngoại lệ:</p> <ol style="list-style-type: none"> <li>1. Admin không nhấn “OK” mà nhấn “Hủy”.</li> </ol>	

#### 2.4.9 Use case quản lý liên hệ



Hình 2.11: Biểu đồ Use case quản lý liên hệ

Với use case quản lý liên hệ, ta xác định được các kịch bản như sau:

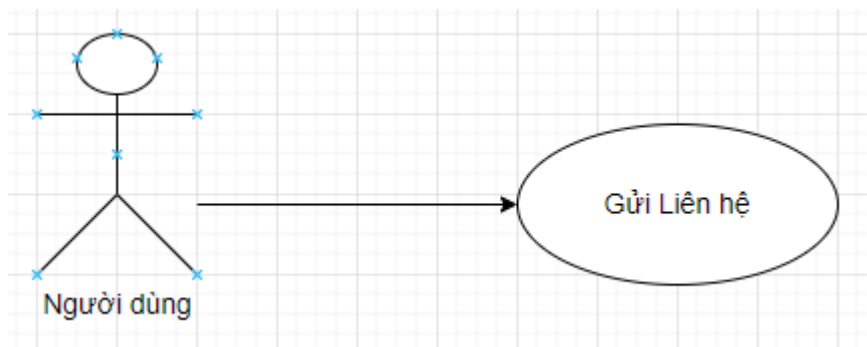
- Xóa liên hệ

Bảng 2.27: Kịch bản chức năng xóa liên hệ

Tên Usecase	Xóa liên hệ
Tác nhân chính	Admin
Người chịu trách nhiệm:	Admin
Tiền điều kiện:	Admin đã đăng nhập vào hệ thống.

Đảm bảo tối thiểu:	Hệ thống tự loại bỏ người dùng đã chọn.
Đảm bảo thành công:	Chức năng xóa sẽ được thực hiện trong CSDL
Kích hoạt:	Quản lý chọn chức năng xóa trong mục Liên hệ.
<p>Chuỗi sự kiện chính:</p> <ol style="list-style-type: none"> <li>Admin lựa chọn liên hệ cần xóa.</li> <li>Hệ thống hiển thị thông báo “Xóa mà không thể khôi phục. Bạn có chắc?”.</li> <li>Admin nhấn “OK”.</li> <li>Hệ thống thông báo “Xóa thành công”.</li> <li>Admin trở về danh sách slide.</li> </ol>	
<p>Ngoại lệ:</p> <ol style="list-style-type: none"> <li>Admin không nhấn “OK” mà nhấn “Hủy”.</li> </ol>	

#### 2.4.10 Use case gửi liên hệ



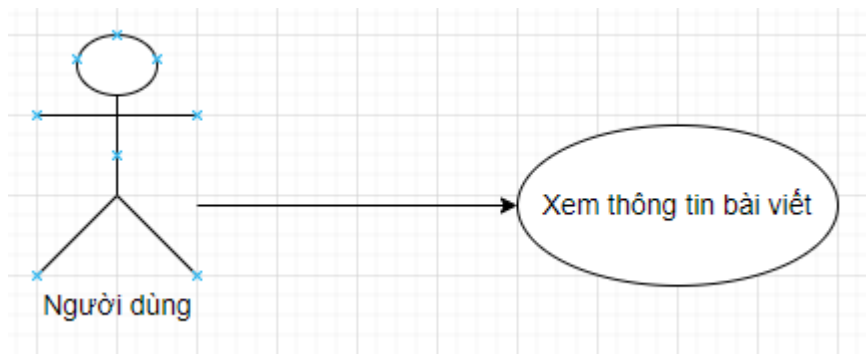
Hình 2.12: Biểu đồ Use case gửi liên hệ

Bảng 2.28: Kịch bản chức năng gửi liên hệ

Tên Usecase	Gửi liên hệ
Tác nhân chính	Người dùng
Người chịu trách nhiệm:	Người dùng

Tiền điều kiện:	
Đảm bảo tối thiểu:	Người dùng nhập các thông tin tại form liên hệ
Đảm bảo thành công:	Người dùng thấy được form liên hệ tại mục “Liên hệ” giao diện người dùng
Kích hoạt:	Người dùng chọn mục “Liên Hệ”
Chuỗi sự kiện chính: <ol style="list-style-type: none"> <li>1. Người dùng chọn mục “Liên Hệ”.</li> <li>2. Hệ thống hiển thị form liên hệ</li> <li>3. Người dùng nhập các thông tin trong form liên hệ</li> <li>4. Người dùng bấm nút “Gửi”.</li> <li>5. Hệ thống thông báo Gửi Liên hệ Thành công.</li> </ol>	

#### 2.4.11 Use case xem bài viết



Hình 2.13: Biểu đồ Use case xem bài viết

Bảng 2.29: Kịch bản chức năng xem bài viết

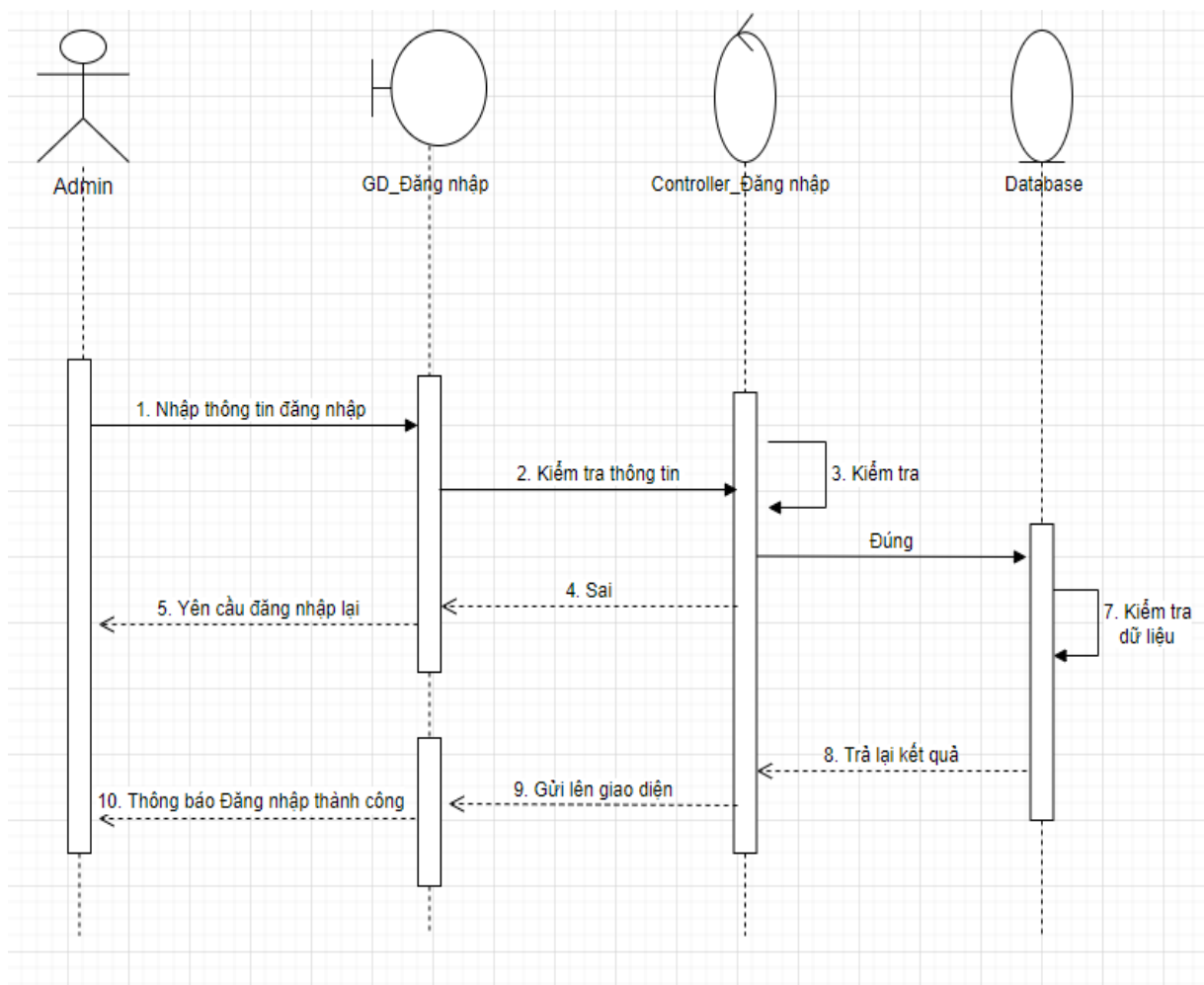
Tên Usecase	Xem thông tin bài viết
Tác nhân chính	Người dùng
Người chịu trách nhiệm:	Người dùng
Tiền điều kiện:	

Đảm bảo tối thiểu:	
Đảm bảo thành công:	Người dùng thấy được danh sách các bài viết
Kích hoạt:	Người dùng click vào link hoặc ảnh bài viết
Chuỗi sự kiện chính:	<ol style="list-style-type: none"> <li>1. Người dùng click vào link hoặc ảnh bài viết</li> <li>2. Hệ thống hiển thị bài viết.</li> </ol>

## 2.5 Biểu đồ tuần tự

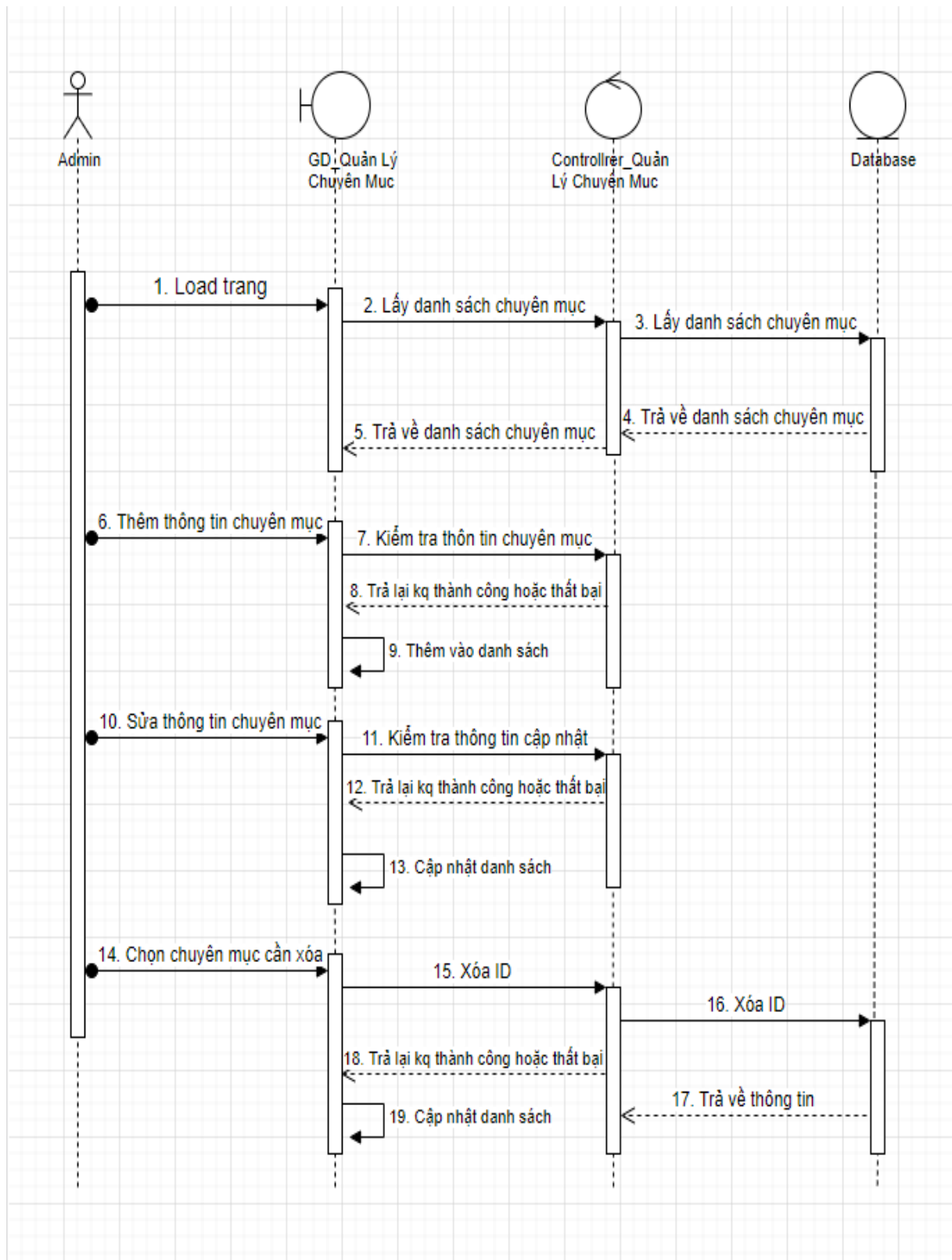
Biểu đồ tuần tự dùng để xác định các trình tự diễn ra sự kiện của một nhóm đối tượng nào đó. Nó miêu tả chi tiết các thông điệp được gửi và nhận giữa các đối tượng đồng thời cũng chú trọng đến việc trình tự về mặt thời gian gửi và nhận các thông điệp đó.

### 2.5.1 Biểu đồ tuần tự chức năng đăng nhập



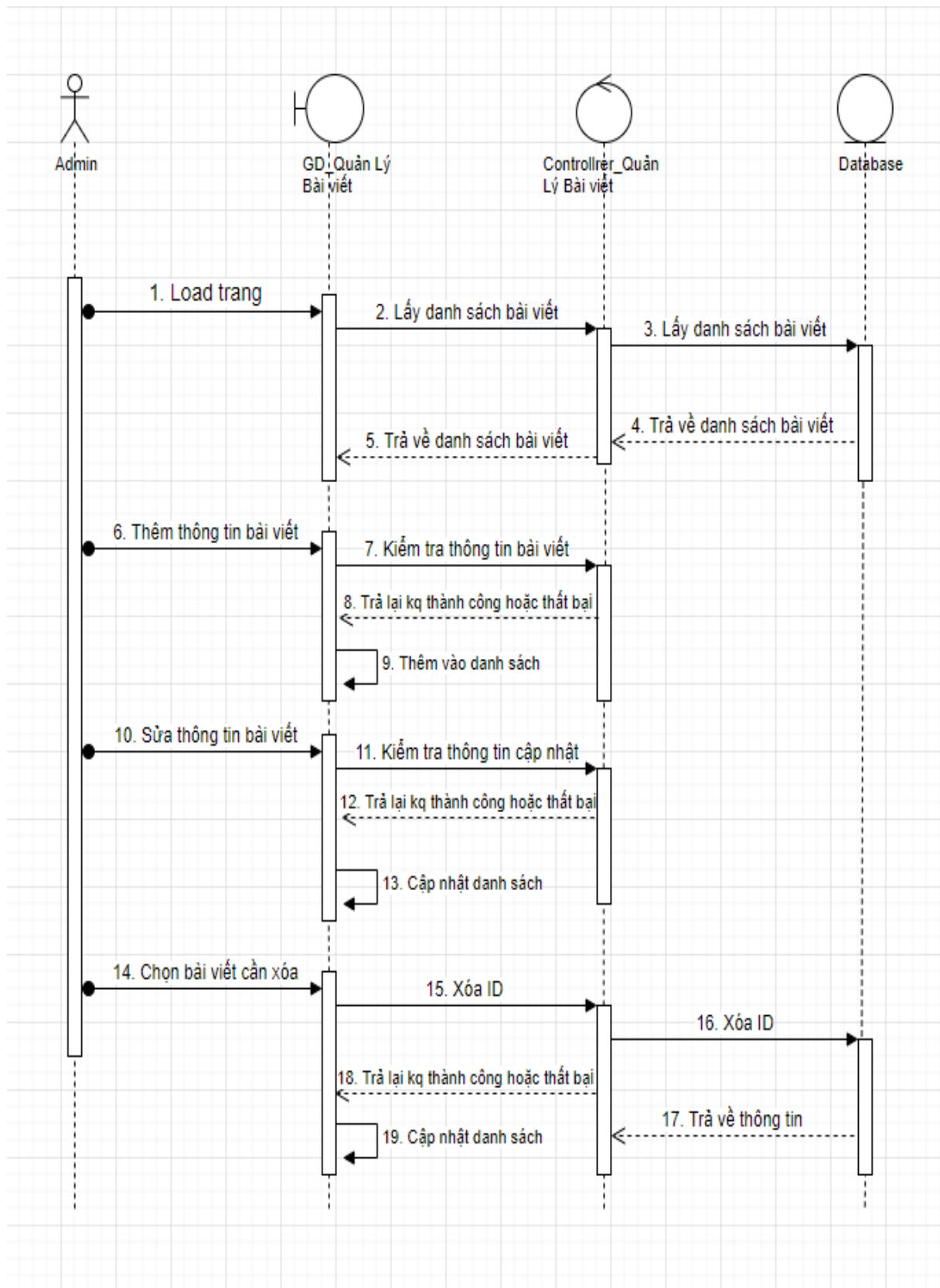
Hình 2.14: Biểu đồ tuần tự chức năng đăng nhập

## 2.5.2 Biểu đồ tuần tự quản lý chuyên mục



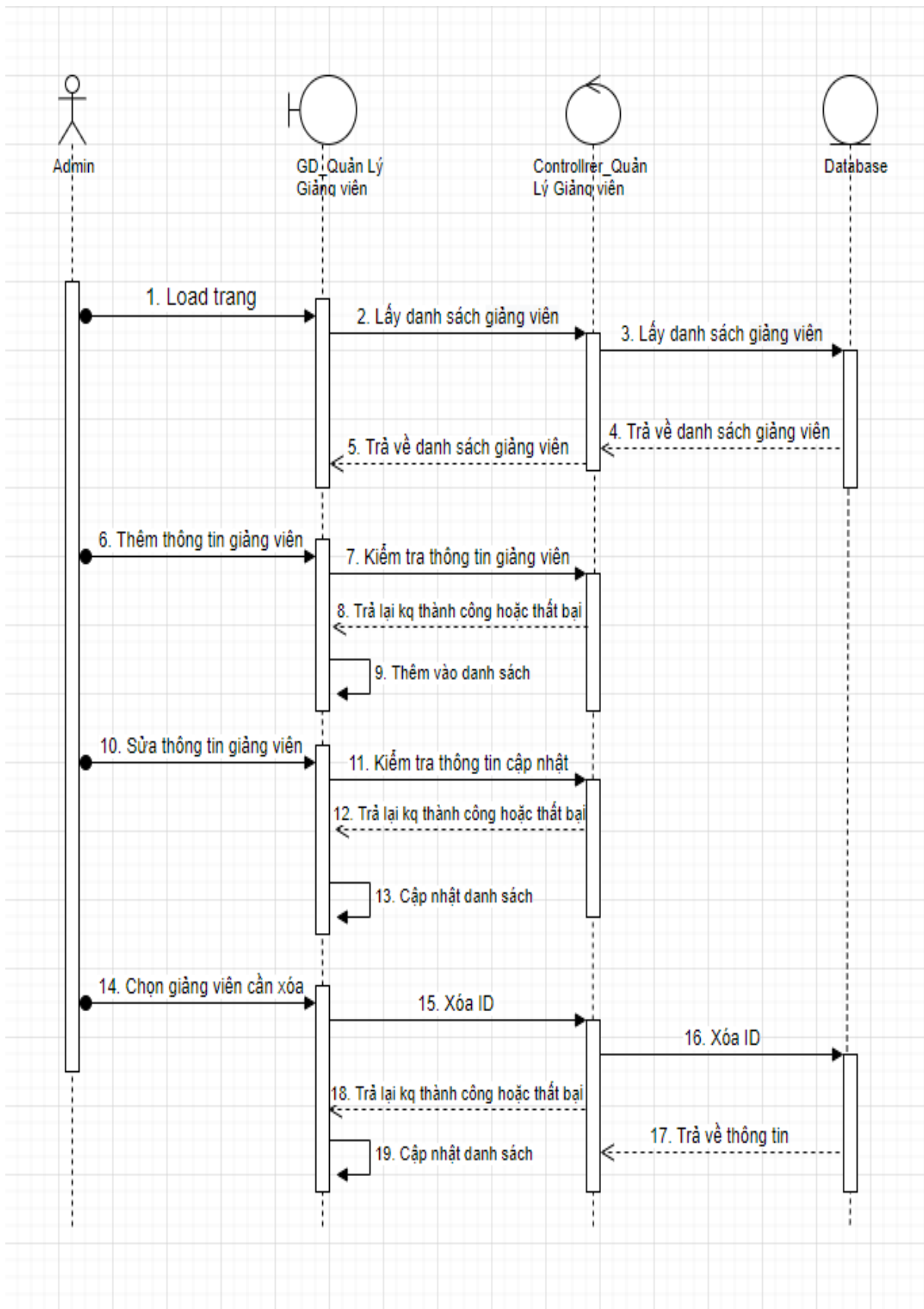
Hình 2.15: Biểu đồ tuần tự quản lý chuyên mục

### 2.5.3 Biểu đồ tuần tự quản lý bài viết



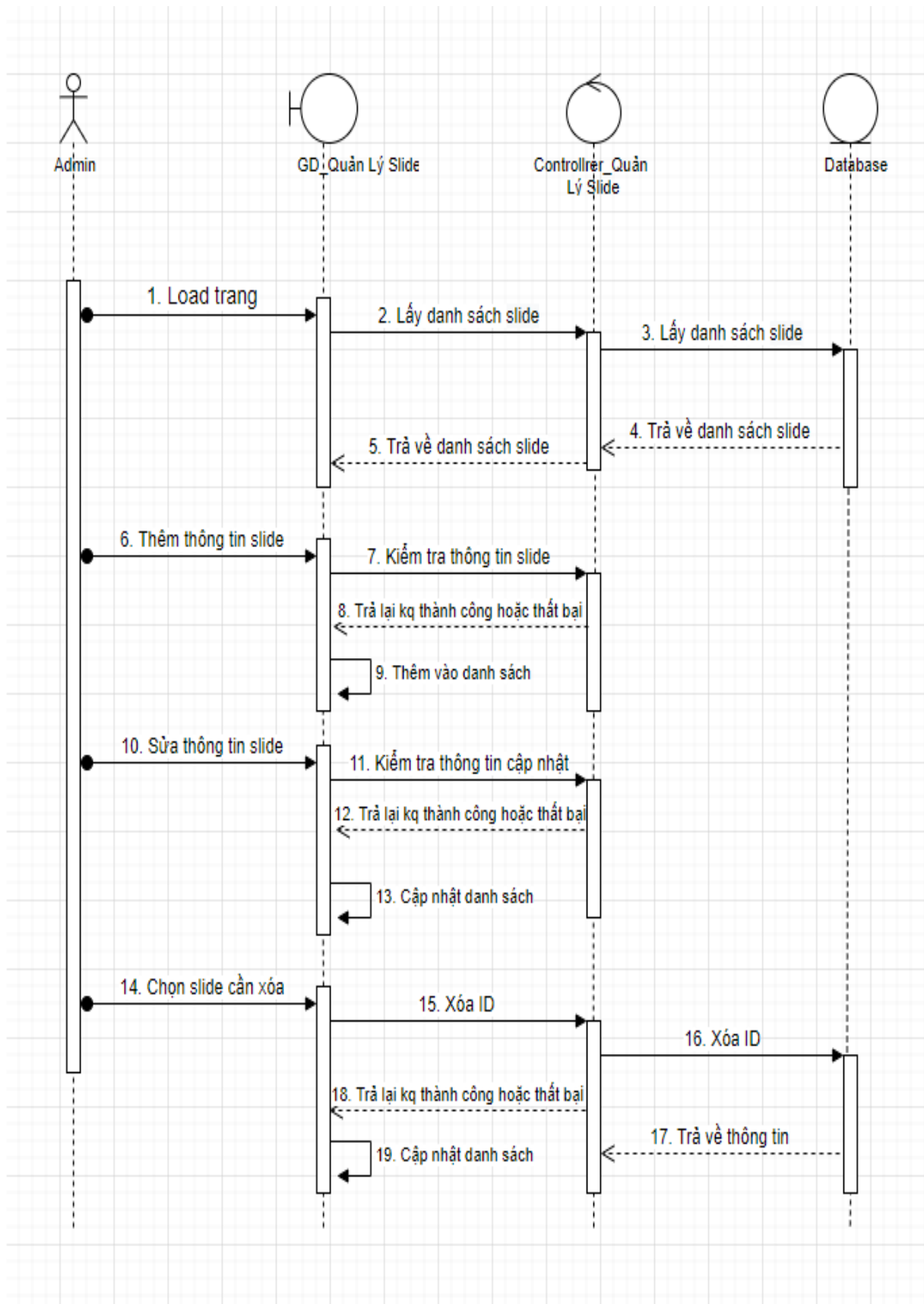
Hình 2.16: Biểu đồ tuần tự quản lý bài viết

## 2.5.4 Biểu đồ tuần tự quản lý giảng viên



Hình 2.17: Biểu đồ tuần tự quản lý giảng viên

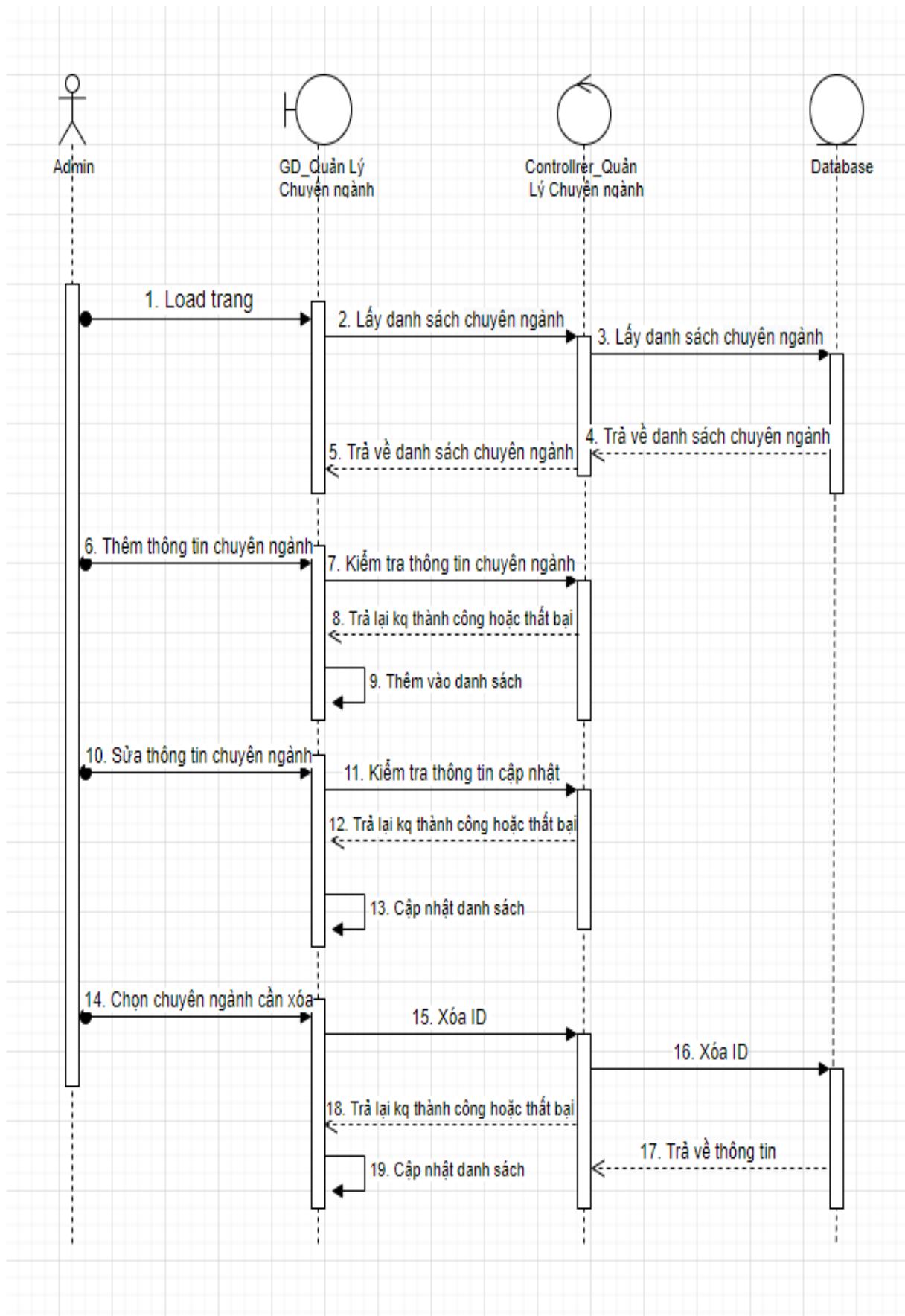
## 2.5.5 Biểu đồ tuần tự quản lý slide



Hình 2.18: Biểu đồ tuần tự quản lý slide

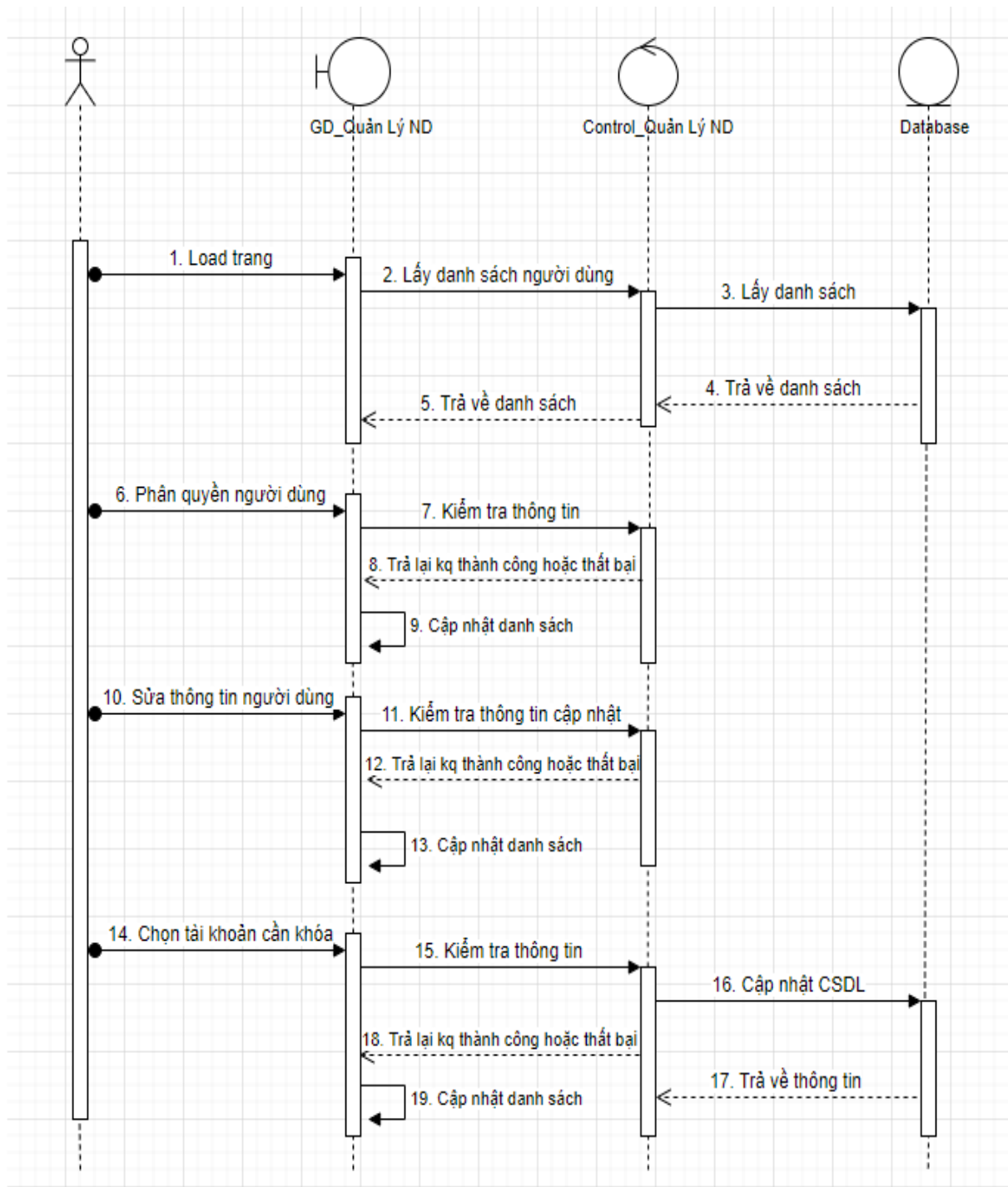


## 2.5.6 Biểu đồ tuần tự quản lý chuyên ngành



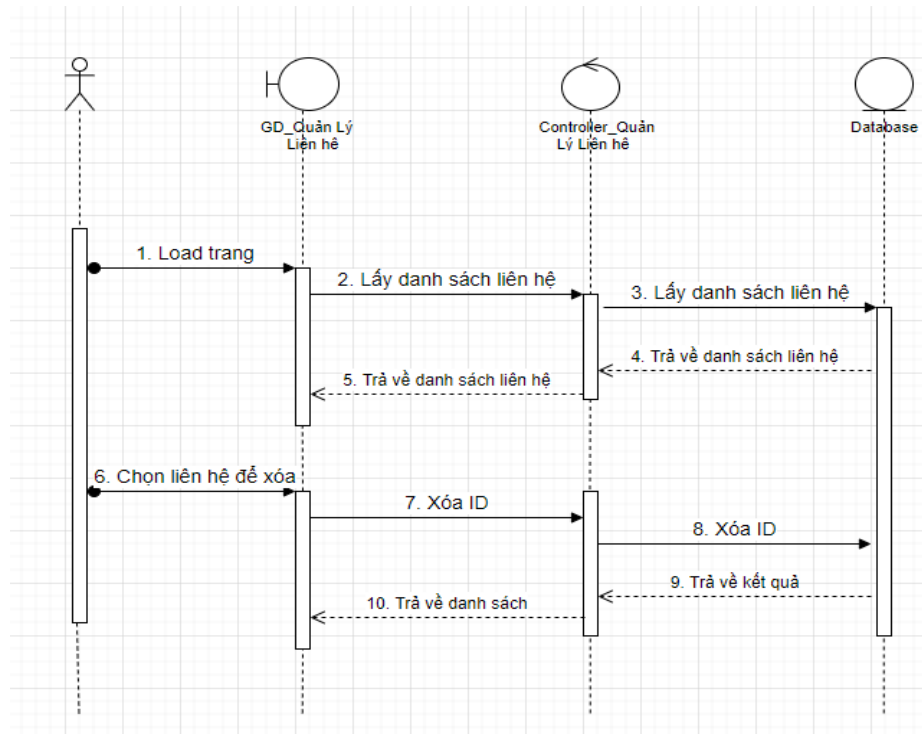
Hình 2.19: Biểu đồ tuần tự quản lý chuyên ngành

## 2.5.7 Biểu đồ tuần tự quản lý Users



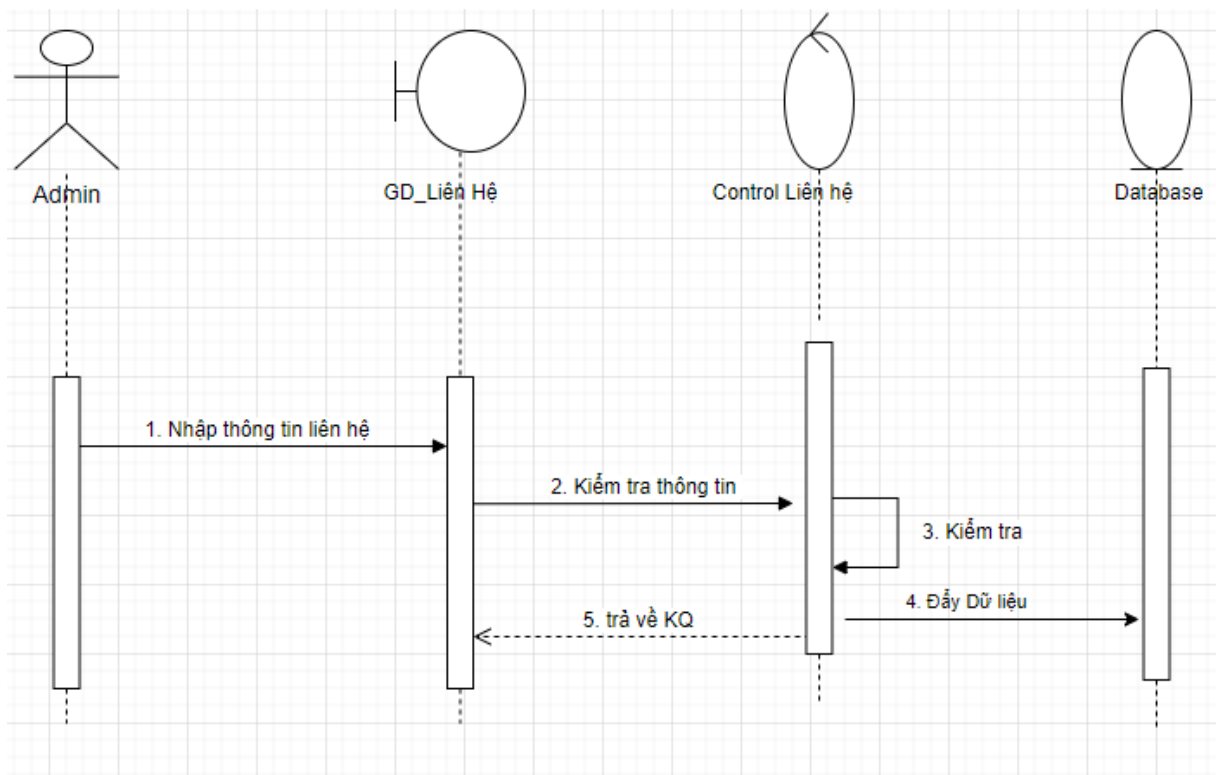
Hình 2.20: Biểu đồ tuần tự quản lý Users

## 2.5.8 Biểu đồ tuần tự quản lý Liên hệ



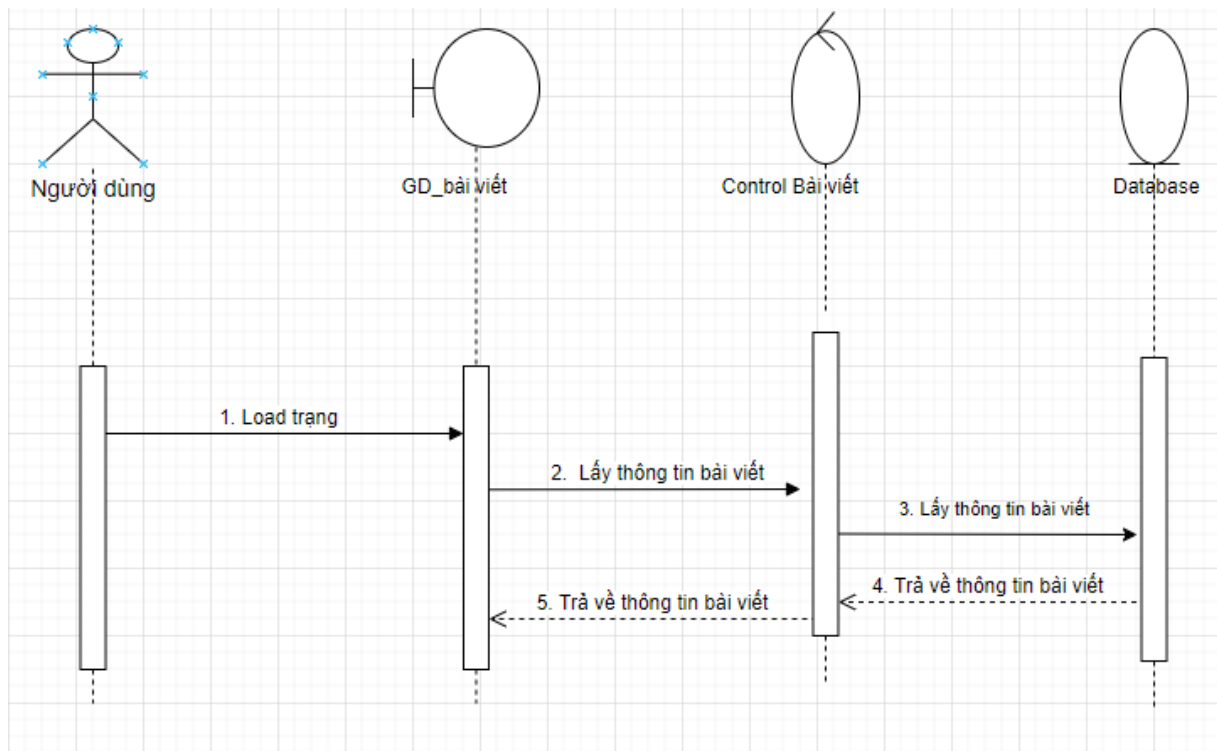
Hình 2.21: Biểu đồ tuần tự quản lý liên hệ

## 2.5.9 Biểu đồ tuần tự gửi Liên hệ



Hình 2.22: Biểu đồ tuần tự gửi liên hệ

## 2.5.10 Biểu đồ tuần tự xem bài viết



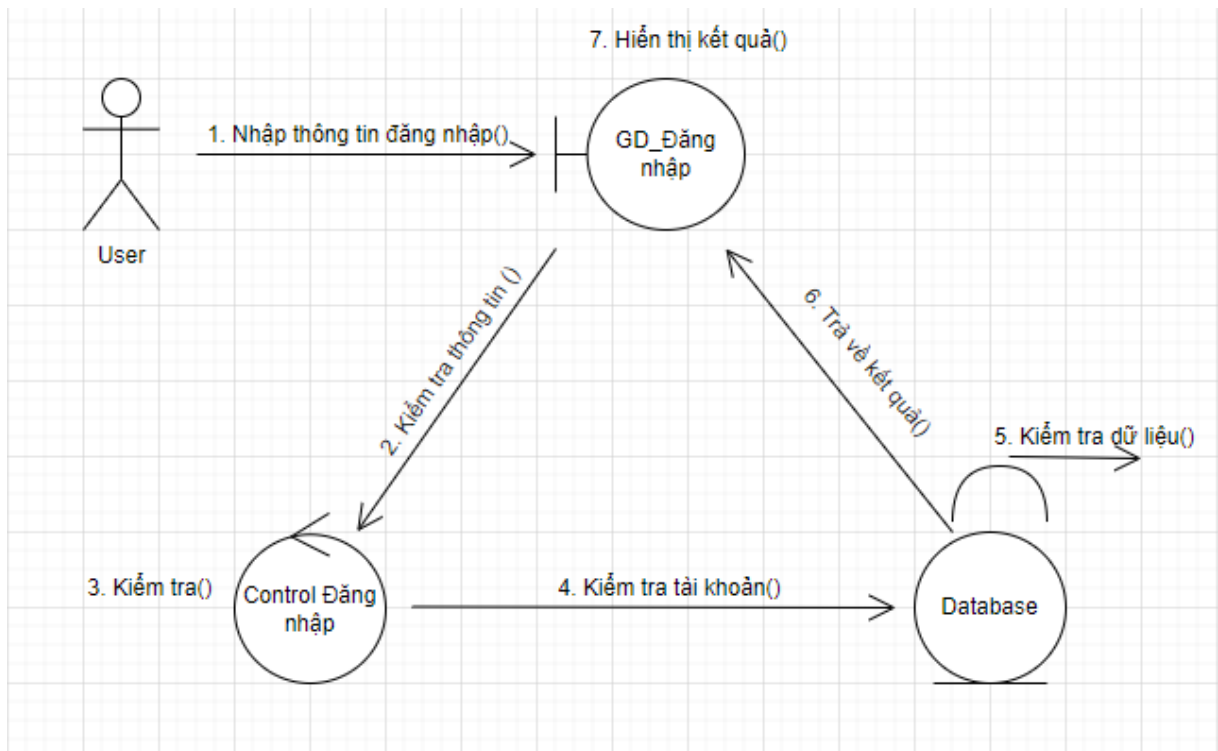
Hình 2.23: Biểu đồ tuần tự xem bài viết

## 2.6 Biểu đồ cộng tác

Biểu đồ cộng tác nhấn mạnh vào sự tương tác của các đối tượng trên cơ sở cộng tác với nhau bằng cách trao đổi các thông điệp để thực hiện các yêu cầu theo ngữ cảnh công việc. Dùng để:

- Mô tả tương tác giữa actor và đối tượng hệ thống
- Dùng để biểu diễn một kịch bản khai thác của 1 use case
- Có thể tạo nhiều sơ đồ cộng tác cho 1 use case
- Có thể xác định được các lớp đối tượng và mối quan hệ giữa các lớp từ Sơ đồ cộng tác

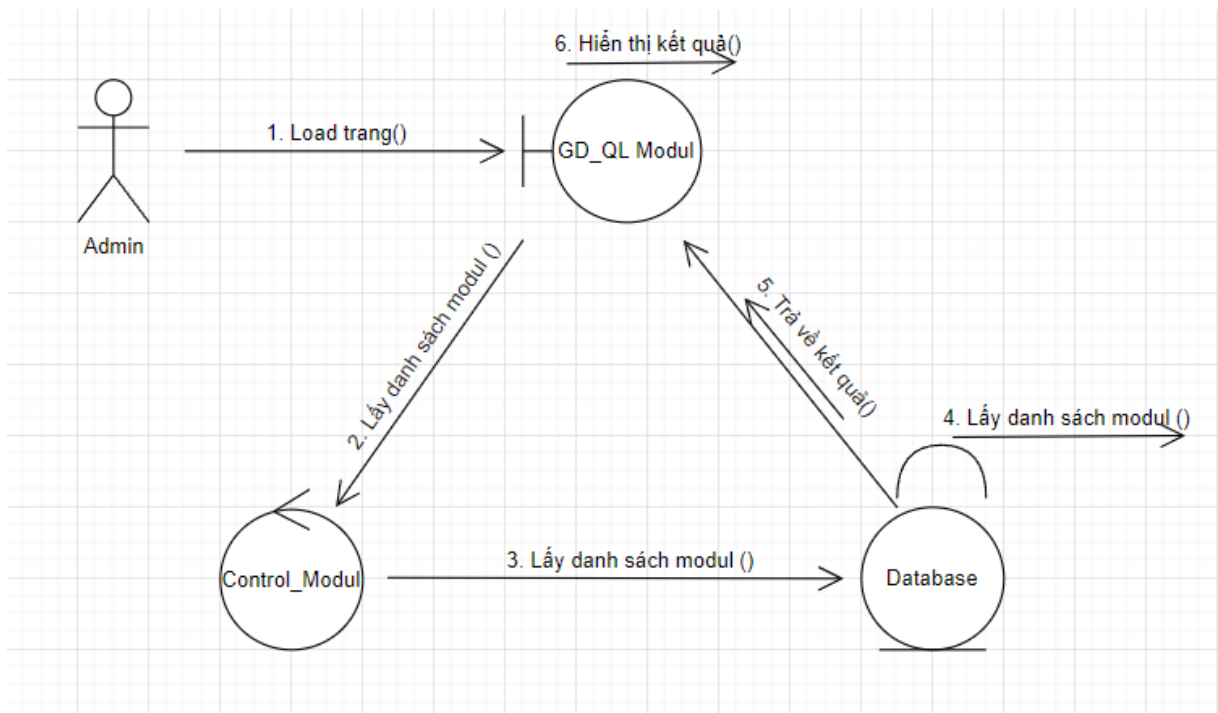
### 2.6.1 Biểu đồ cộng tác chức năng đăng nhập



Hình 2.24: Biểu đồ cộng tác chức năng đăng nhập

### 2.6.2 Biểu đồ cộng tác chức năng hiển thị danh sách modul

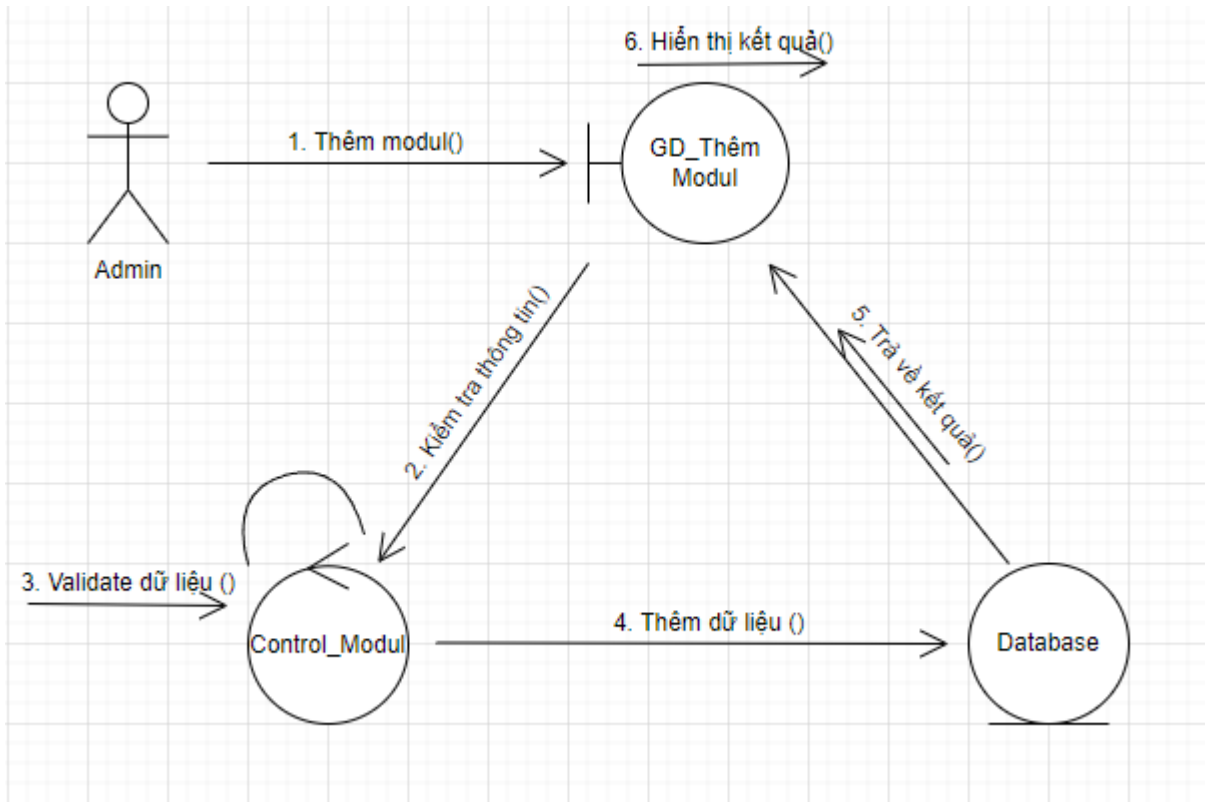
Ở đây, thao tác hiển thị danh sách các modul như slide, chuyên ngành, giảng viên, chuyên mục đều có quy trình giống nhau nên ta có được biểu đồ cộng tác chức năng hiển thị danh sách slide, chuyên ngành, giảng viên, chuyên mục tổng quát:



Hình 2.25: Biểu đồ cộng tác chức năng hiển thị danh sách modul

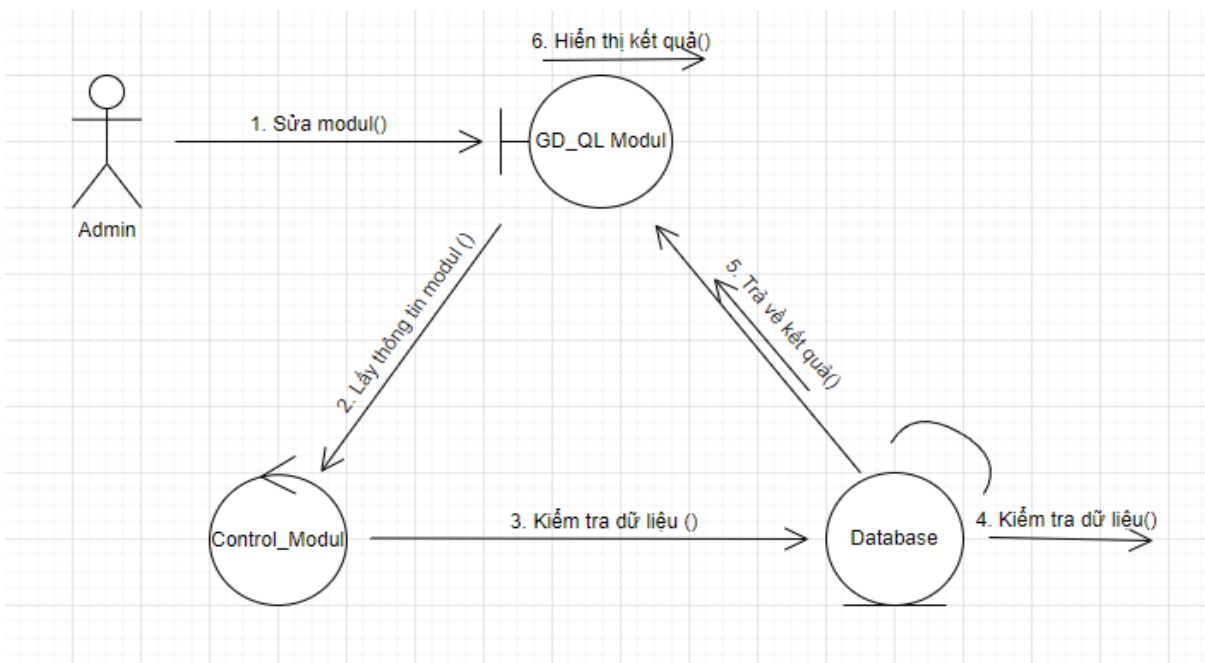
## 2.6.2 Biểu đồ cộng tác chức năng thêm modul

Tương tự, thao tác thêm các modul như slide, chuyên ngành, giảng viên, chuyên mục đều có quy trình giống nhau nên ta có được biểu đồ cộng tác chức năng thêm slide, chuyên ngành, giảng viên, chuyên mục tổng quát:



Hình 2.26: Biểu đồ cộng tác chức năng thêm modul

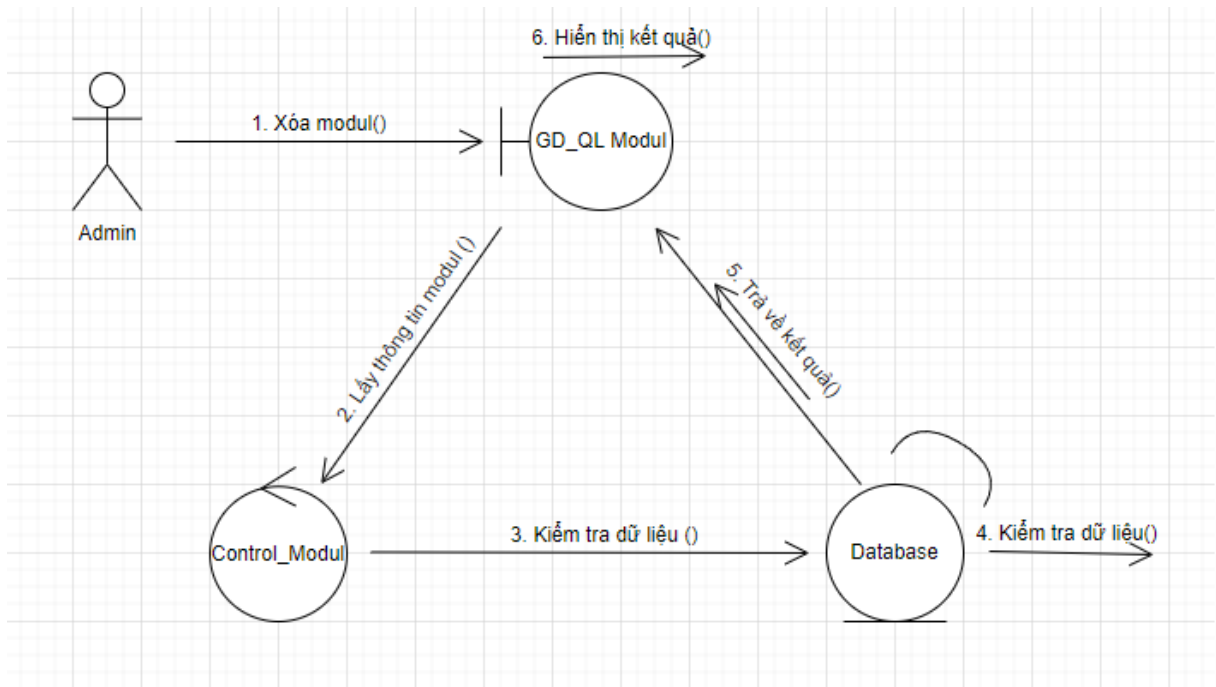
## 2.6.3 Biểu đồ cộng tác chức năng sửa modul



Hình 2.27: Biểu đồ cộng tác chức năng sửa modul

### 2.6.4 Biểu đồ cộng tác chức năng xóa modul

Các modul ở đây bao gồm: chuyên mục, giảng viên, bài viết, liên hệ, chuyên ngành, người dùng.

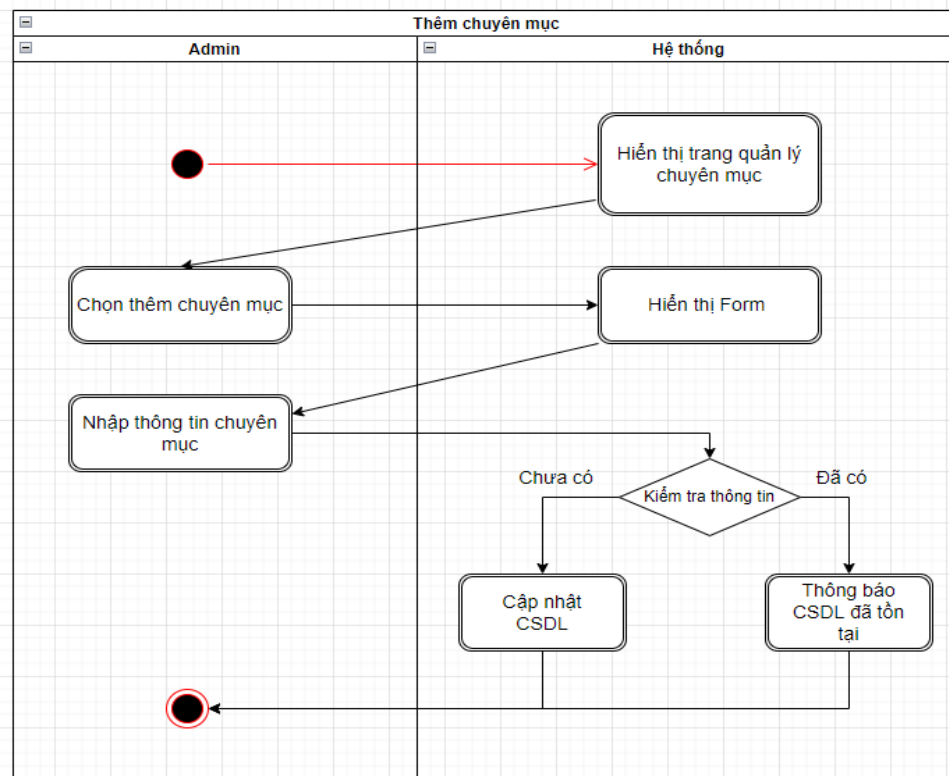


Hình 2.28: Biểu đồ cộng tác chức năng xóa modul

### 2.7 Biểu đồ hoạt động

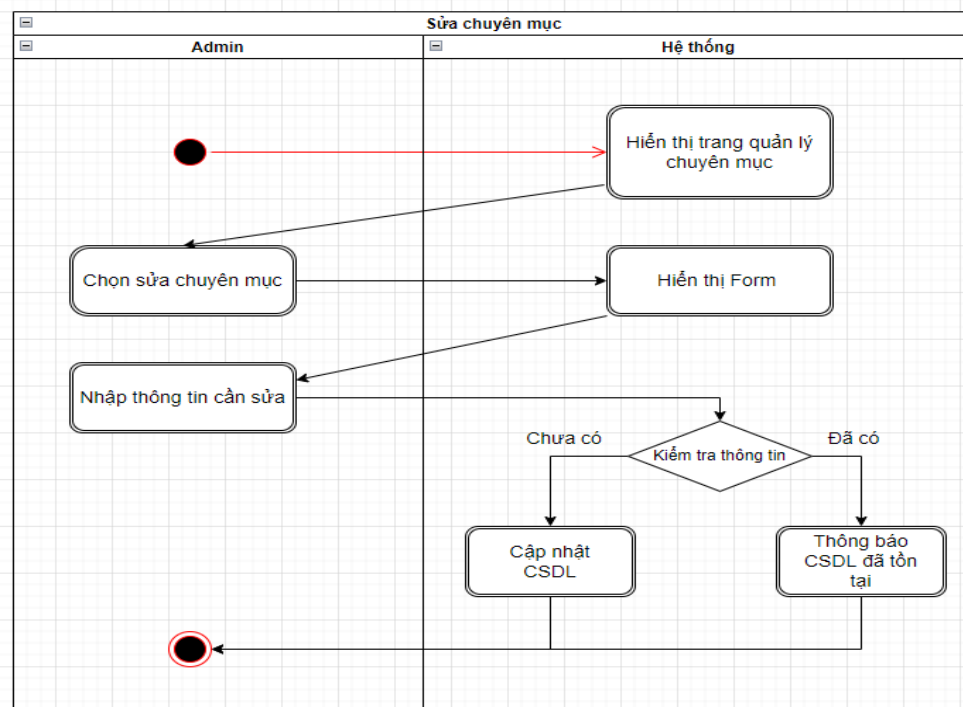
Biểu đồ hoạt động (Activity diagram) chỉ ra dòng hoạt động của hệ thống.

## 2.7.1 Biểu đồ hoạt động chức năng thêm chuyên mục



Hình 2.29: Biểu đồ hoạt động chức năng thêm danh mục

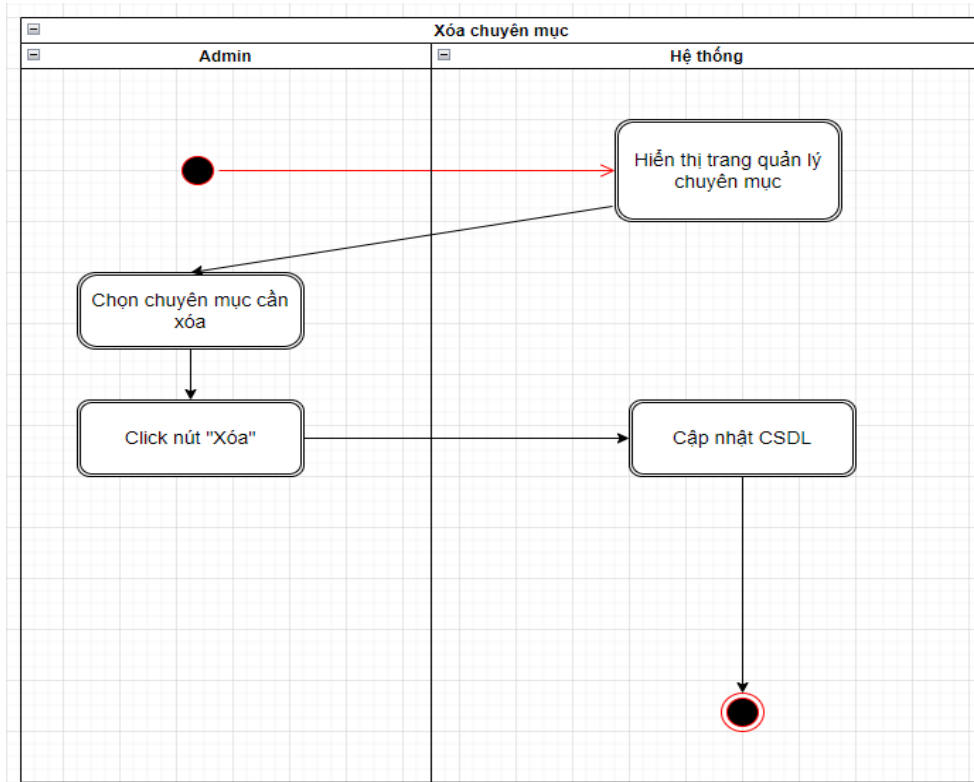
## 2.7.2 Biểu đồ hoạt động chức năng sửa chuyên mục



Hình 2.30: Biểu đồ hoạt động chức năng sửa chuyên mục

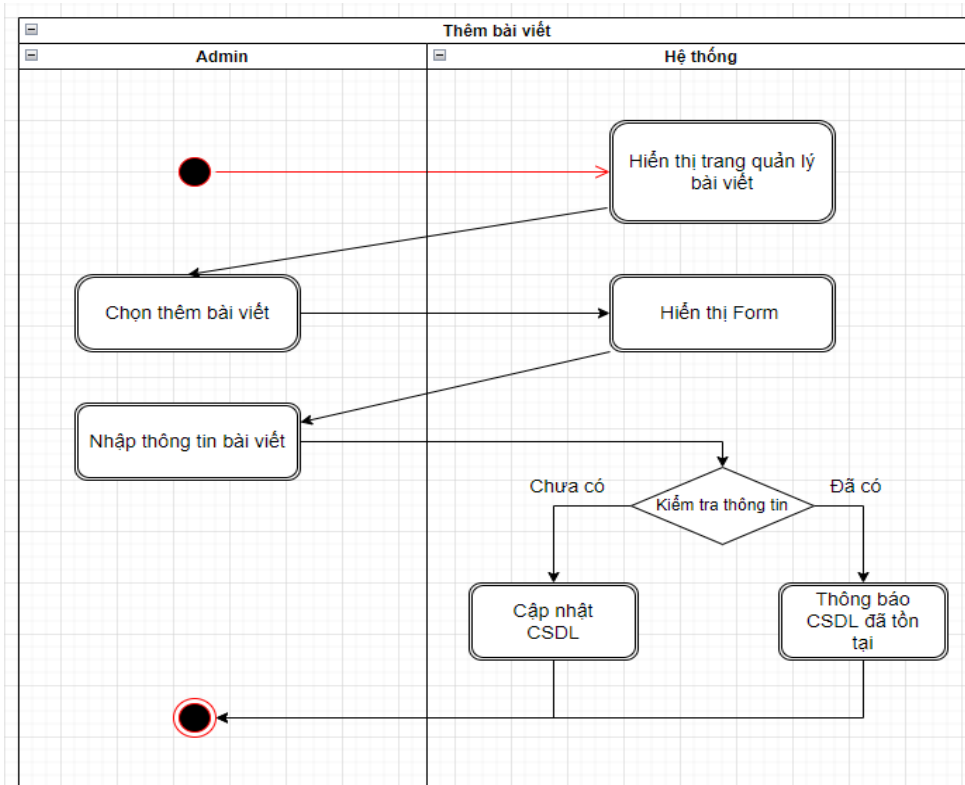


### 2.7.3 Biểu đồ hoạt động chức năng xóa chuyên mục



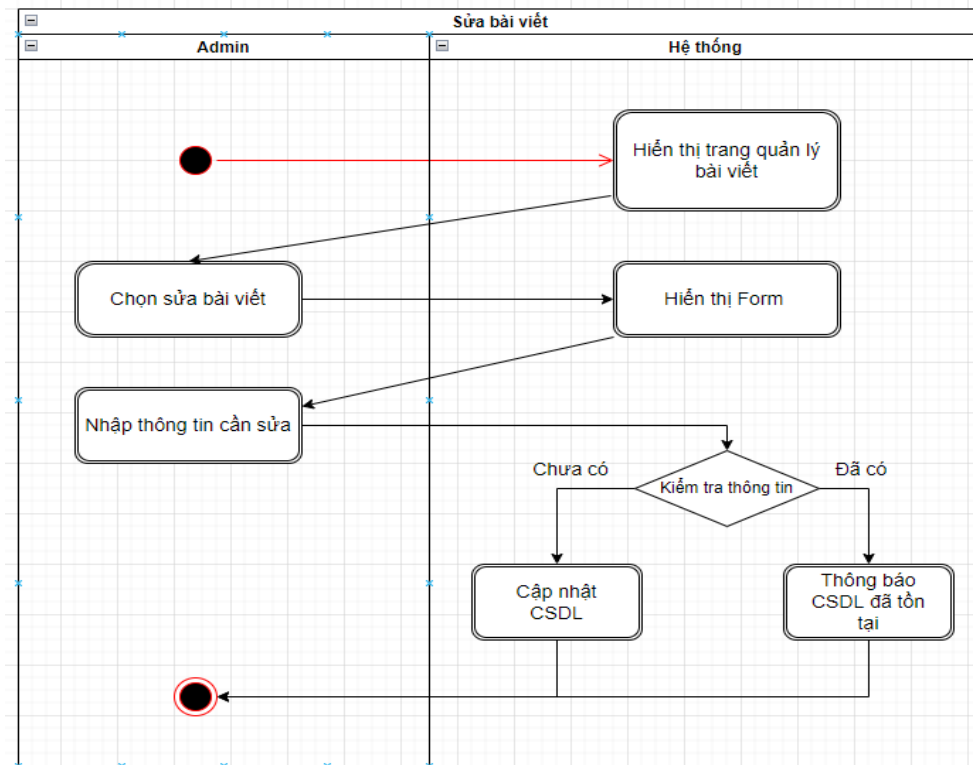
Hình 2.31: Biểu đồ hoạt động chức năng xóa chuyên mục

### 2.7.4 Biểu đồ hoạt động chức năng thêm bài viết



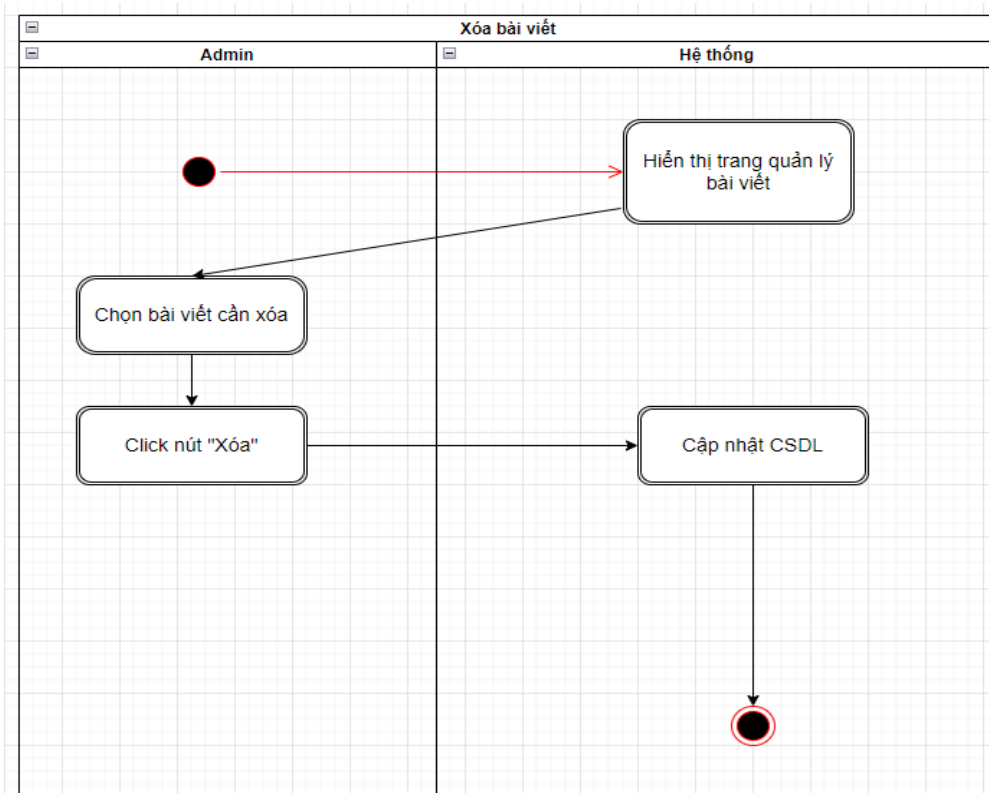
Hình 2.32: Biểu đồ hoạt động chức năng thêm bài viết

### 2.7.5 Biểu đồ hoạt động chức năng sửa bài viết



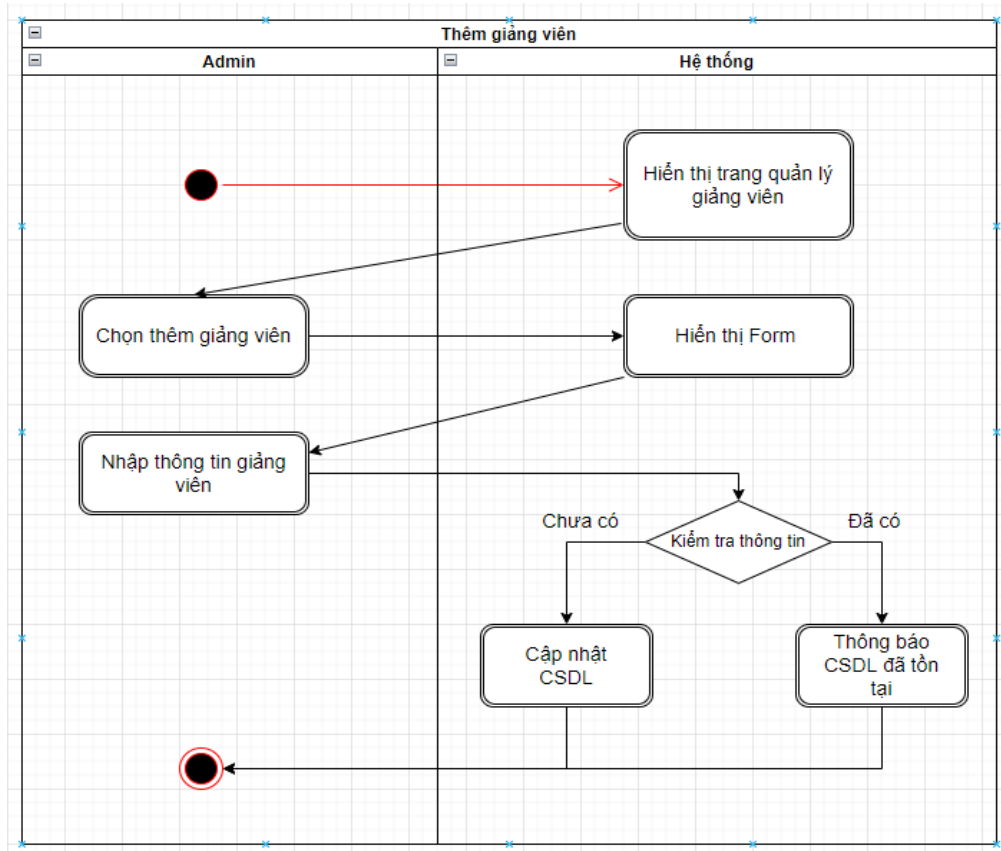
Hình 2.33: Biểu đồ hoạt động chức năng sửa bài viết

### 2.7.6 Biểu đồ hoạt động chức năng xóa bài viết



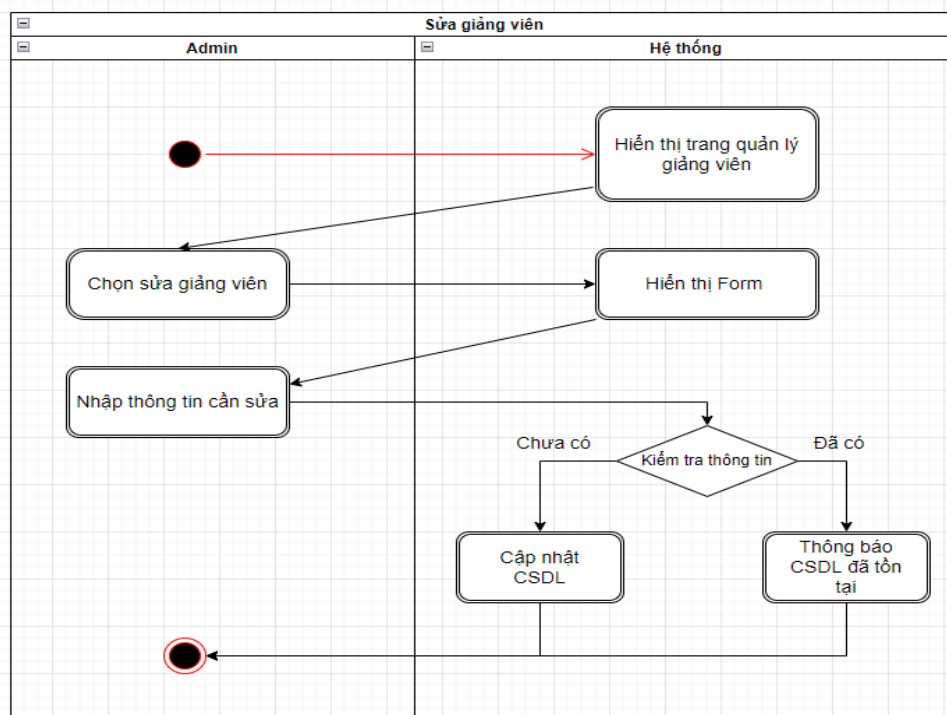
Hình 2.34: Biểu đồ hoạt động chức năng xóa bài viết

### 2.7.7 Biểu đồ hoạt động chức năng thêm giảng viên



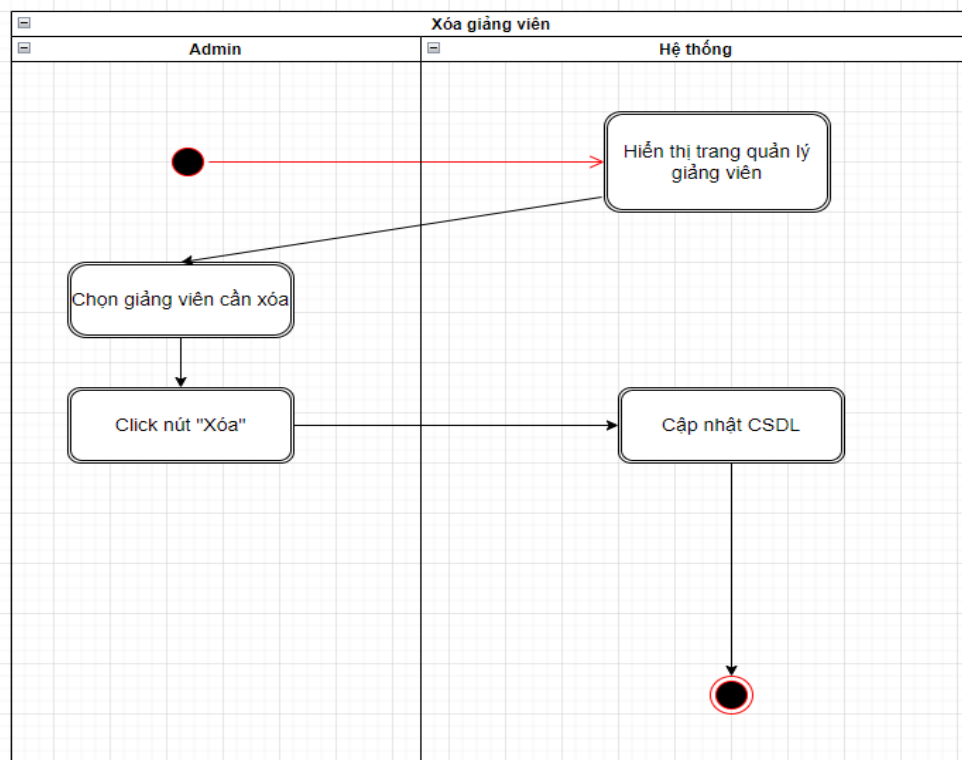
Hình 2.35: Biểu đồ hoạt động chức năng thêm giảng viên

### 2.7.8 Biểu đồ hoạt động chức năng sửa giảng viên



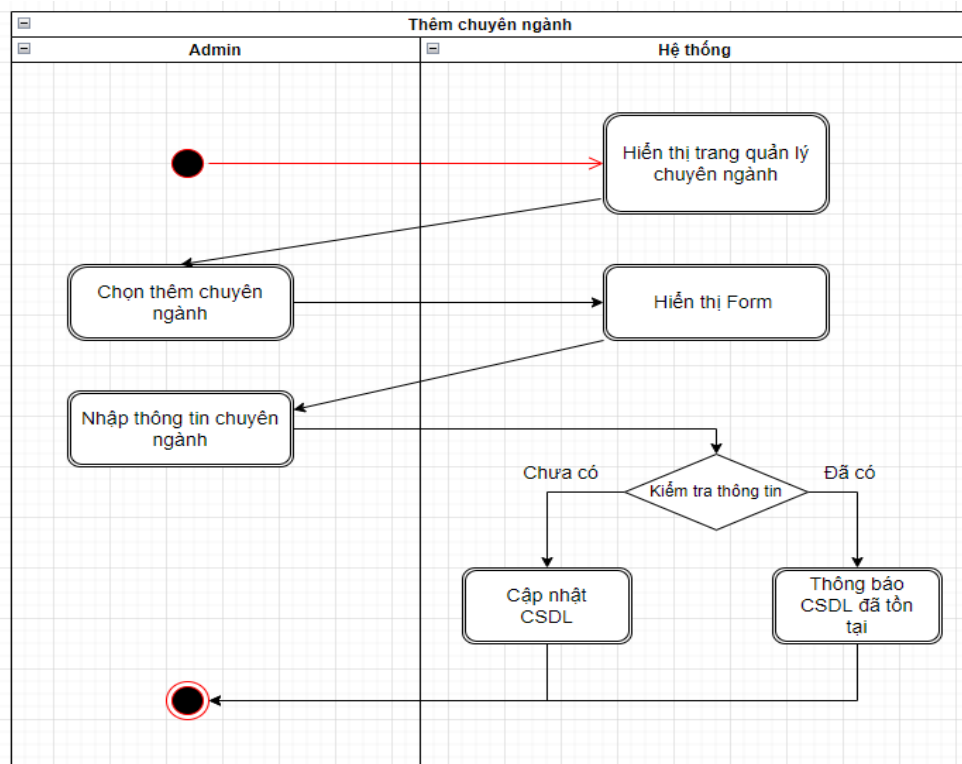
Hình 2.36: Biểu đồ hoạt động chức năng sửa giảng viên

## 2.7.9 Biểu đồ hoạt động chức năng xóa giảng viên



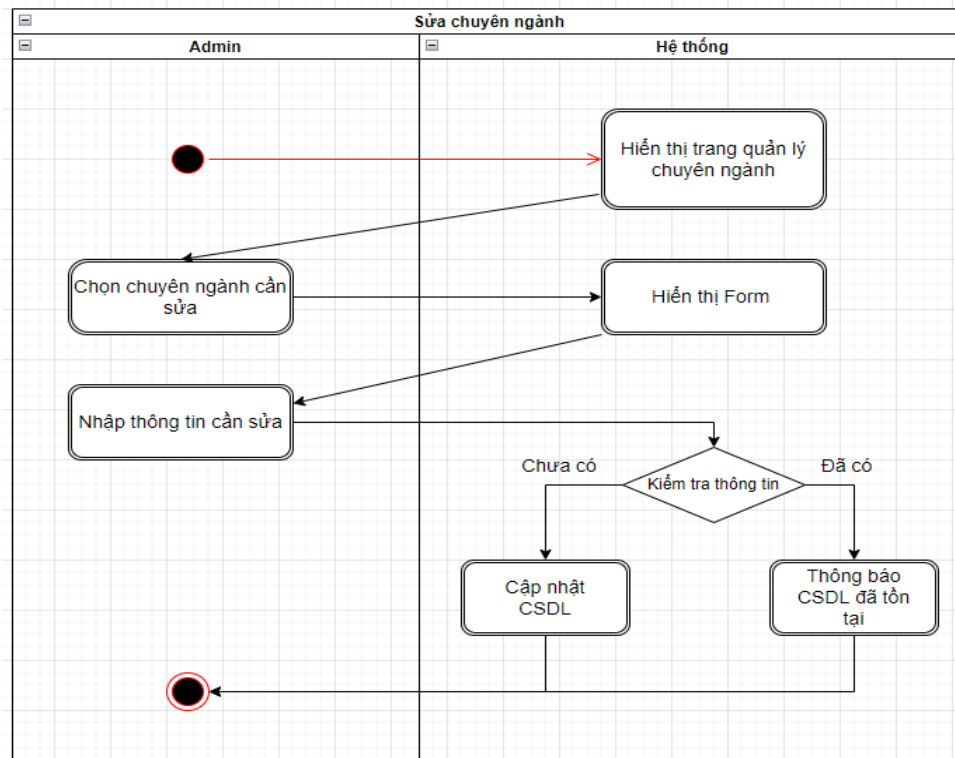
Hình 2.37: Biểu đồ hoạt động chức năng xóa giảng viên

## 2.7.10 Biểu đồ hoạt động chức năng thêm chuyên ngành



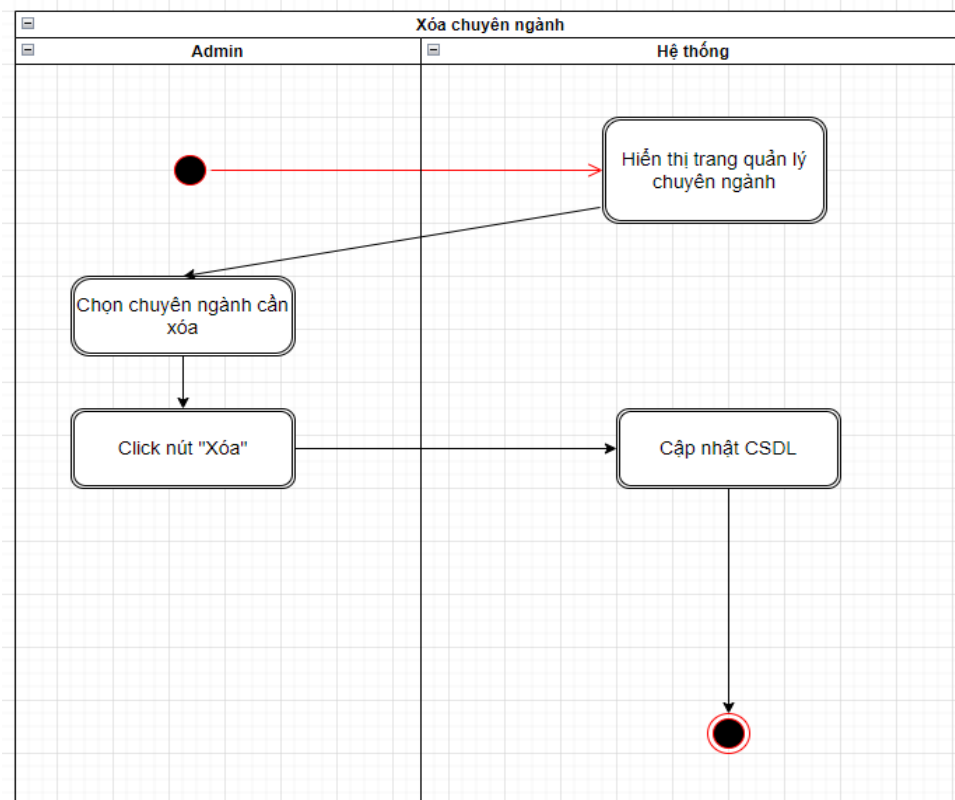
Hình 2.38: Biểu đồ hoạt động chức năng thêm chuyên ngành

### 2.7.11 Biểu đồ hoạt động chức năng sửa chuyên ngành



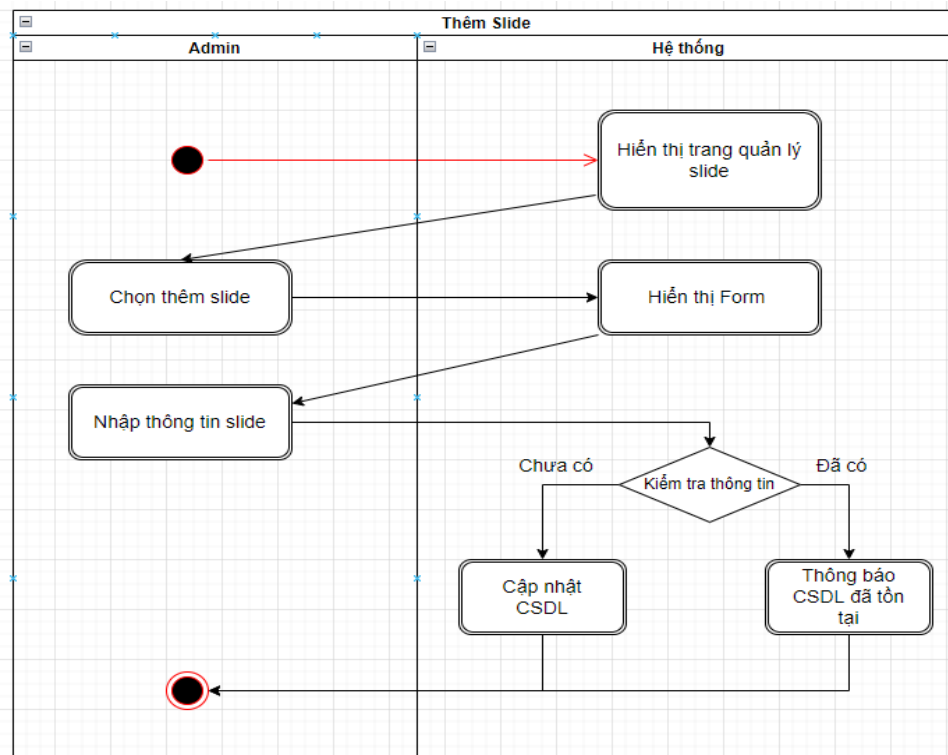
Hình 2.39: Biểu đồ hoạt động chức năng sửa chuyên ngành

### 2.7.12 Biểu đồ hoạt động chức năng xóa chuyên ngành



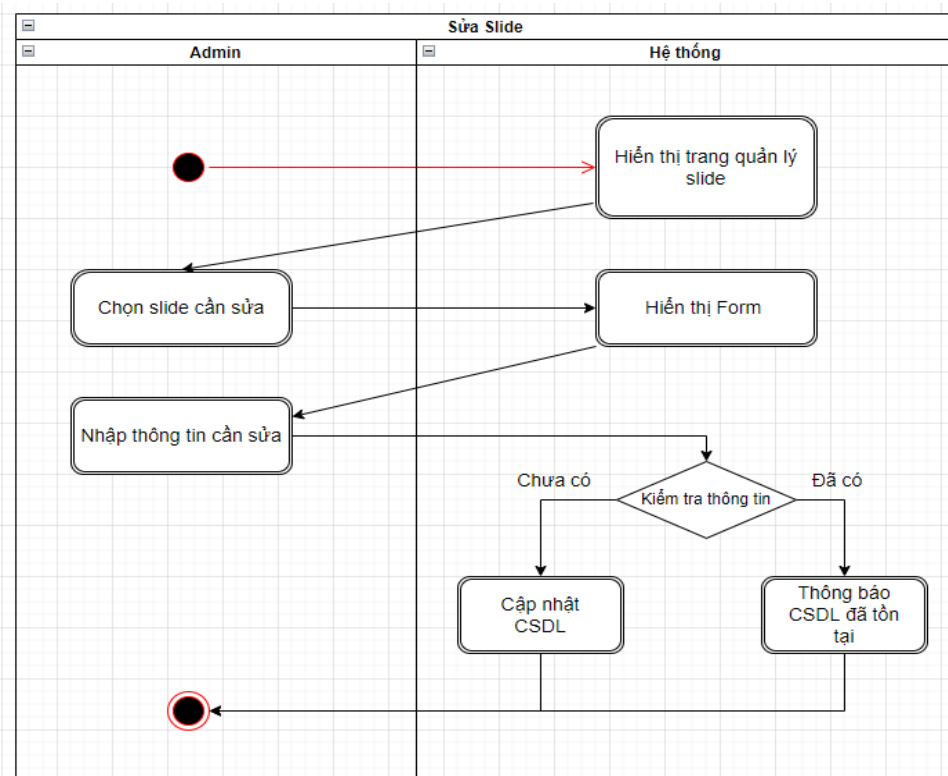
Hình 2.40: Biểu đồ hoạt động chức năng xóa chuyên ngành

### 2.7.13 Biểu đồ hoạt động chức năng thêm slide



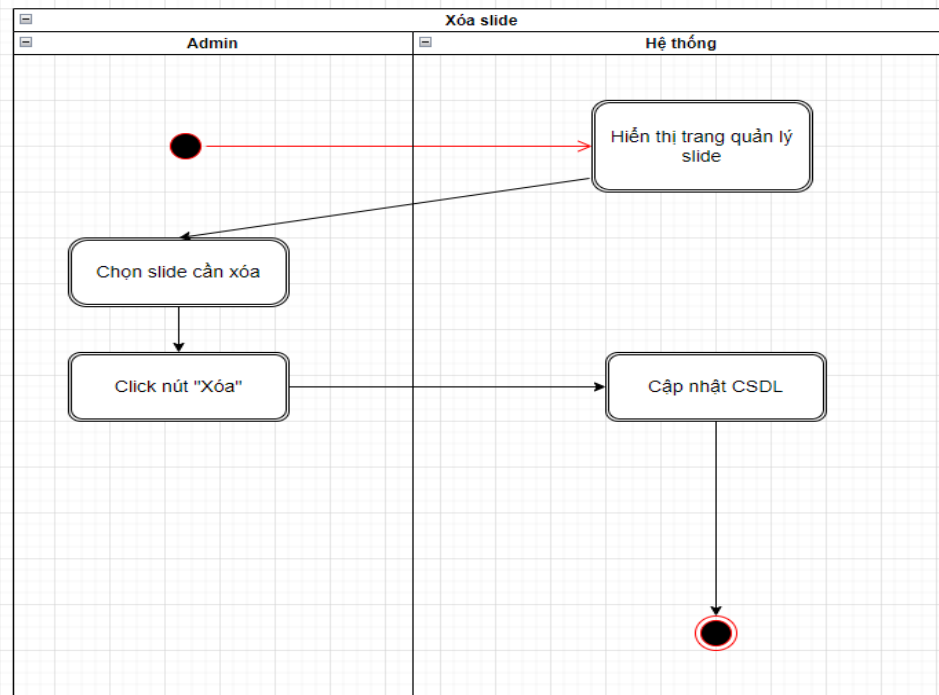
Hình 2.41: Biểu đồ hoạt động chức năng thêm slide

### 2.7.14 Biểu đồ hoạt động chức năng sửa slide



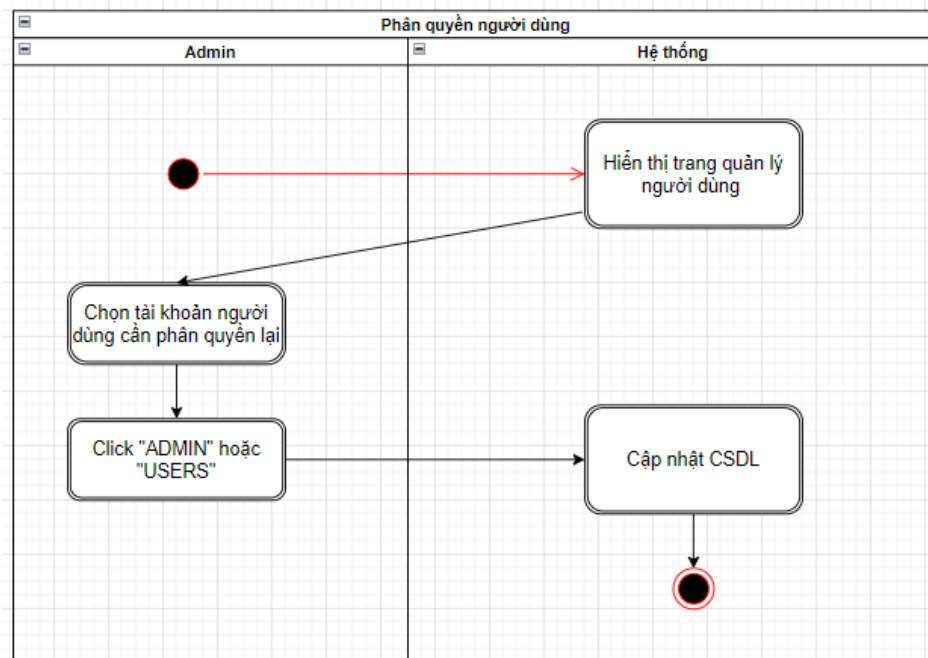
Hình 2.42: Biểu đồ hoạt động chức năng sửa slide

### 2.7.15 Biểu đồ hoạt động chức năng xóa slide



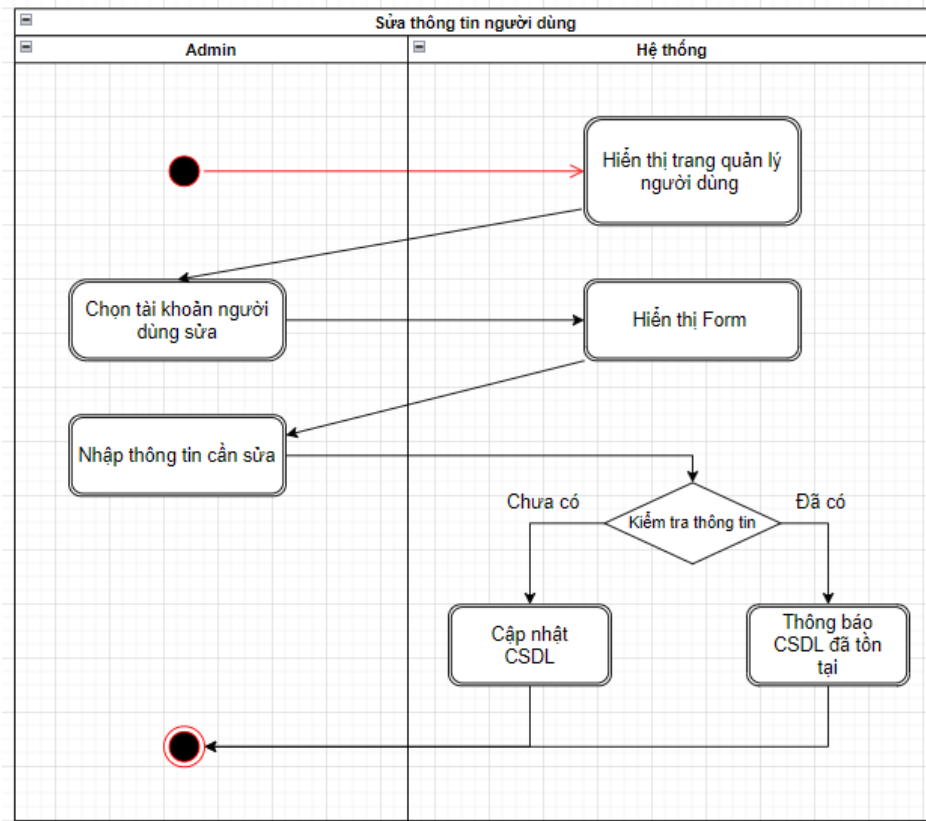
Hình 2.43: Biểu đồ hoạt động chức năng xóa slide

### 2.7.16 Biểu đồ hoạt động chức năng phân quyền người dùng



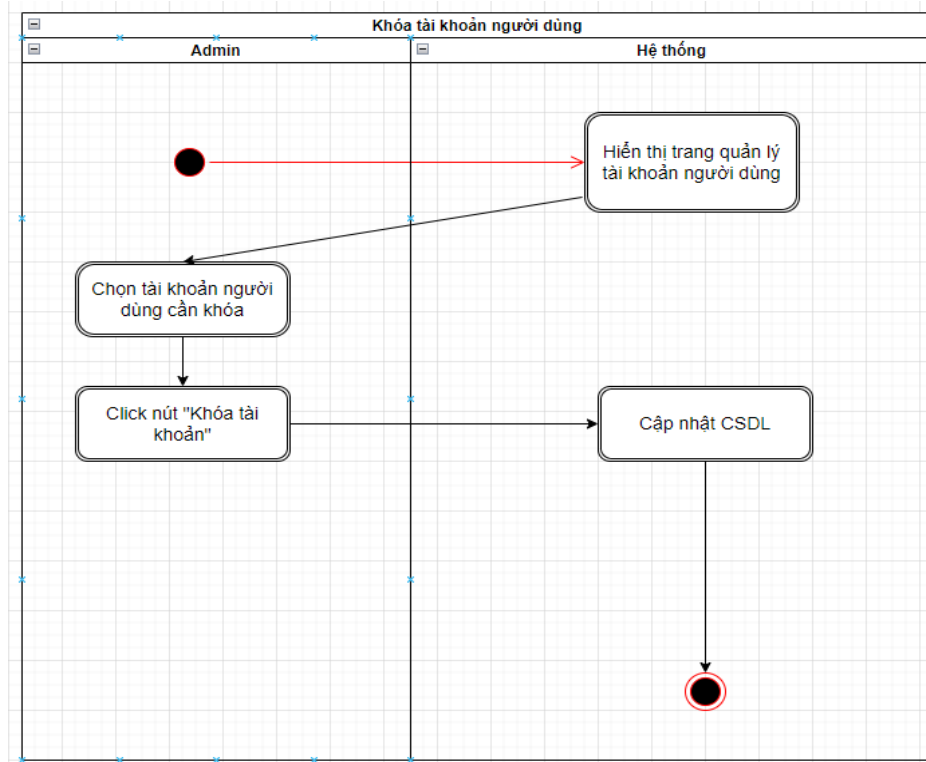
Hình 2.44: Biểu đồ hoạt động chức năng phân quyền người dùng

### 2.7.17 Biểu đồ hoạt động chức năng sửa thông tin người dùng



Hình 2.45: Biểu đồ hoạt động chức năng sửa thông tin người dùng

### 2.7.18 Biểu đồ hoạt động chức năng khóa tài khoản người dùng



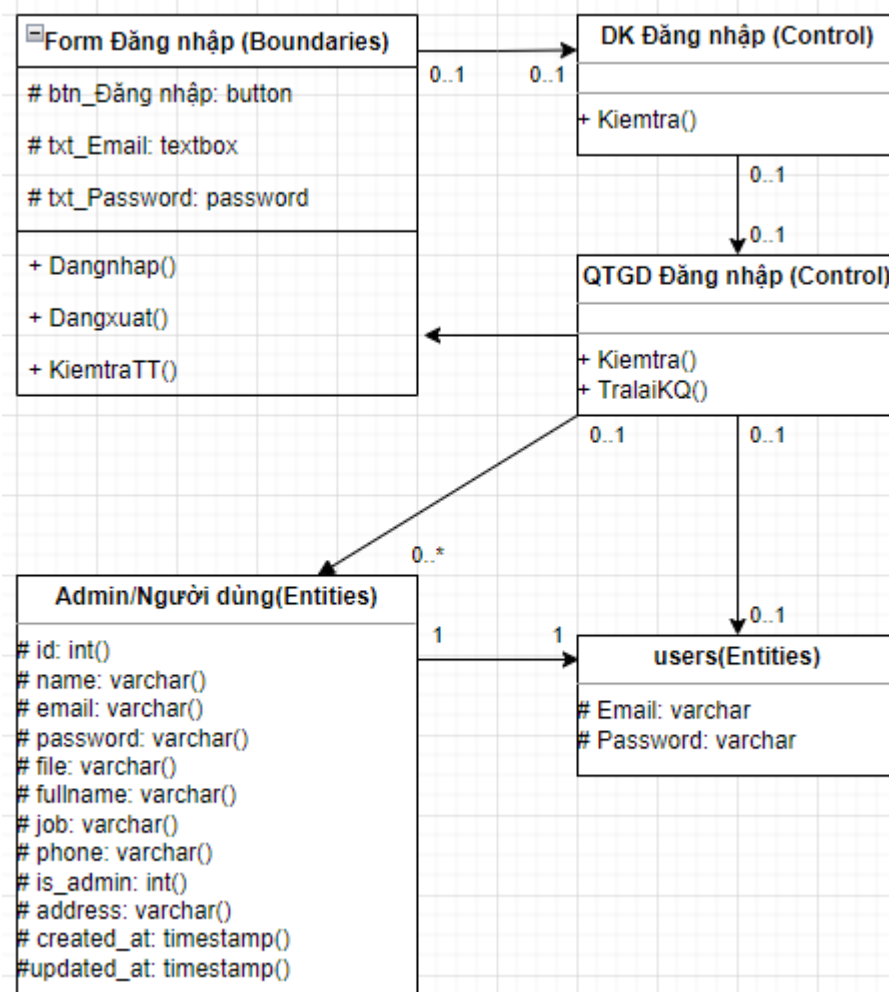
Hình 2.46: Biểu đồ hoạt động chức năng khóa tài khoản người dùng



## 2.8 Biểu đồ lớp

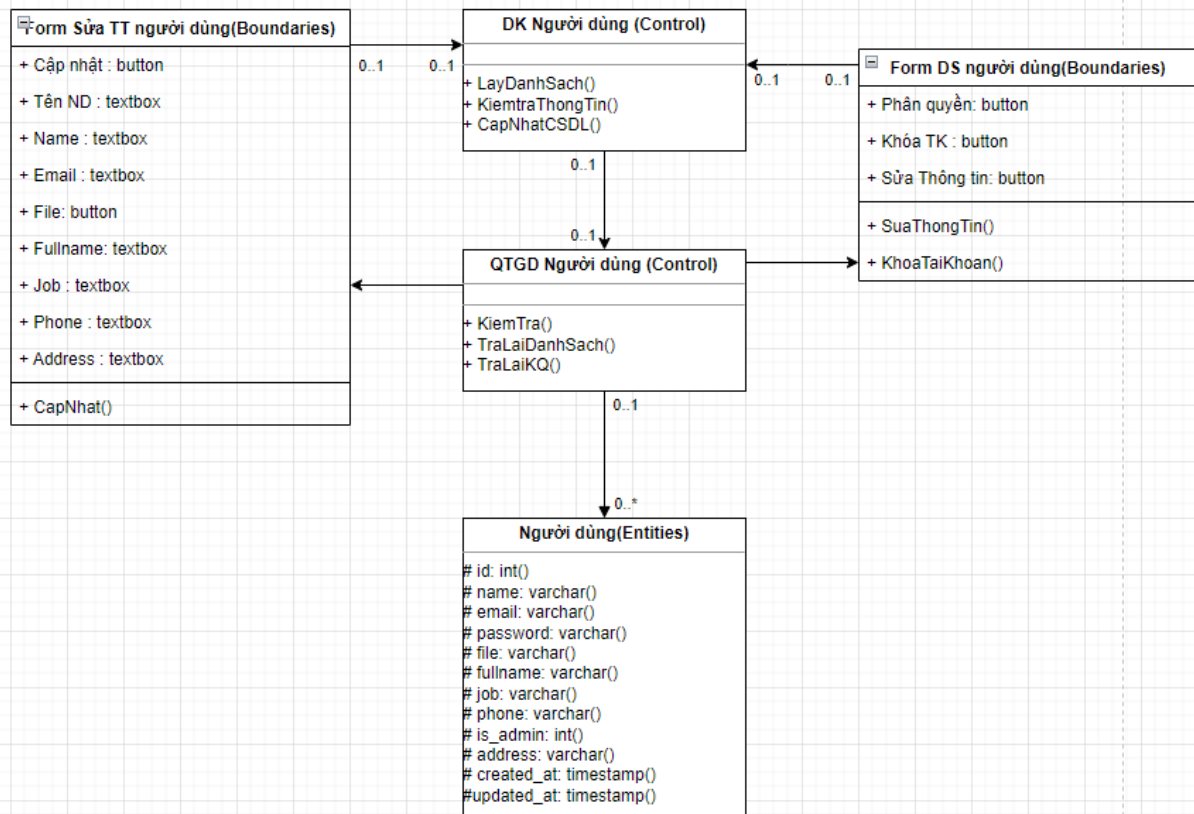
Biểu đồ lớp là biểu đồ dạng mô hình tĩnh nhằm mô tả hướng cách nhìn tĩnh về một hệ thống bằng các khái niệm lớp, các thuộc tính, phương thức của lớp và mối quan hệ giữa chúng với nhau.

### 2.8.1 Biểu đồ lớp Use case Đăng nhập



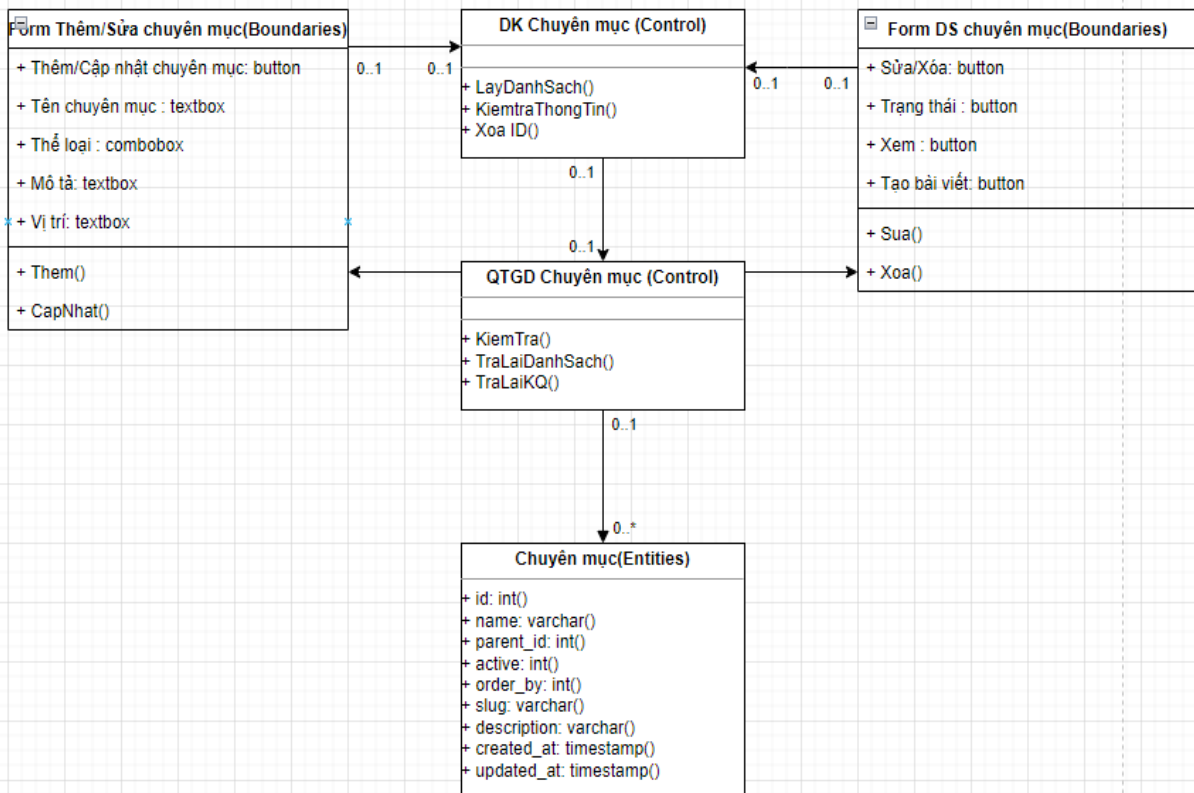
Hình 2.47: Biểu đồ lớp Use case đăng nhập

## 2.8.2 Biểu đồ lớp Use case quản lý người dùng



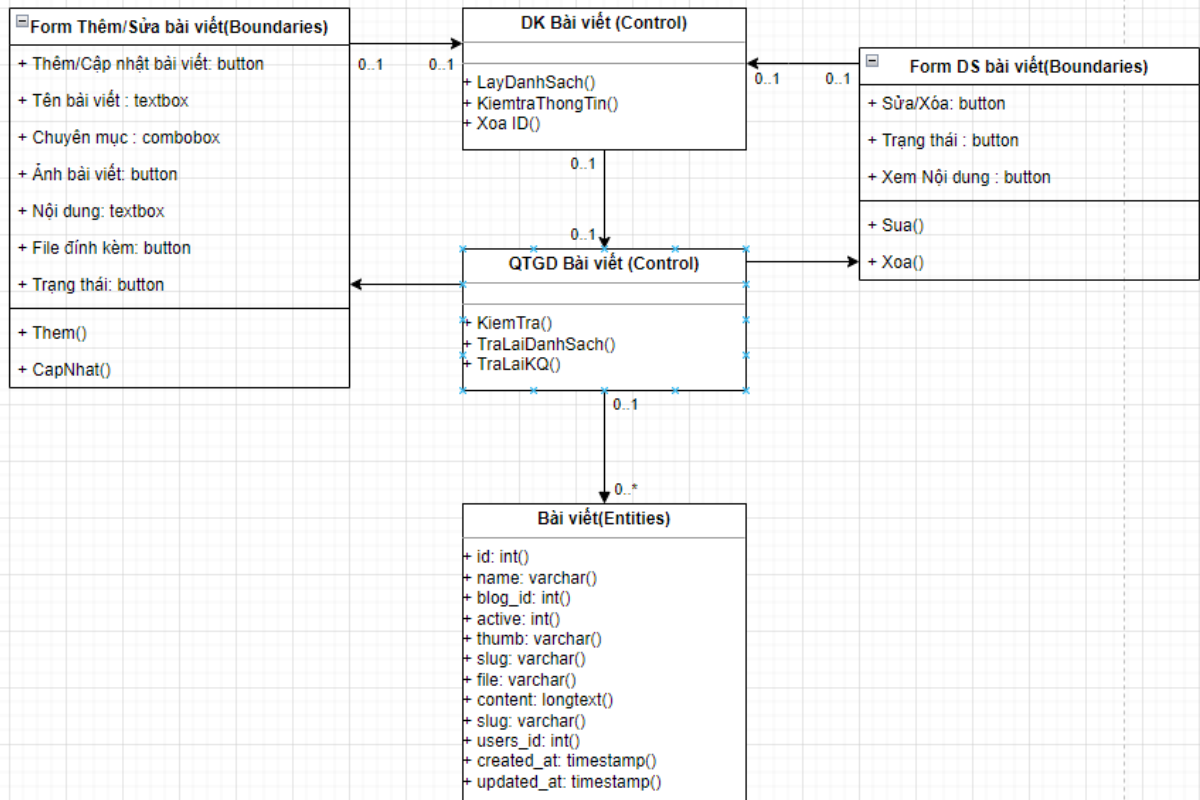
Hình 2.48: Biểu đồ lớp Use case quản lý người dùng

## 2.8.3 Biểu đồ lớp Use case quản lý chuyên mục



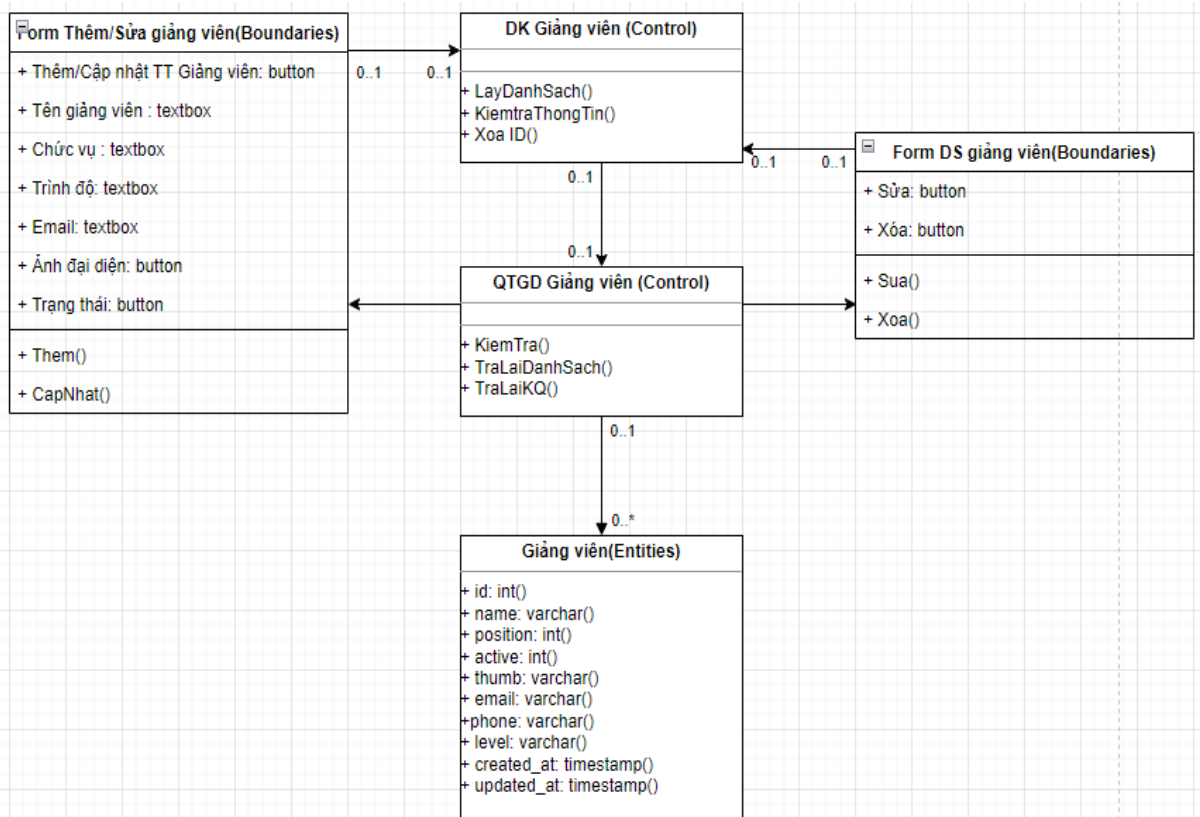
Hình 2.49: Biểu đồ lớp Use case quản lý chuyên mục

## 2.8.4 Biểu đồ lớp Use case quản lý bài viết



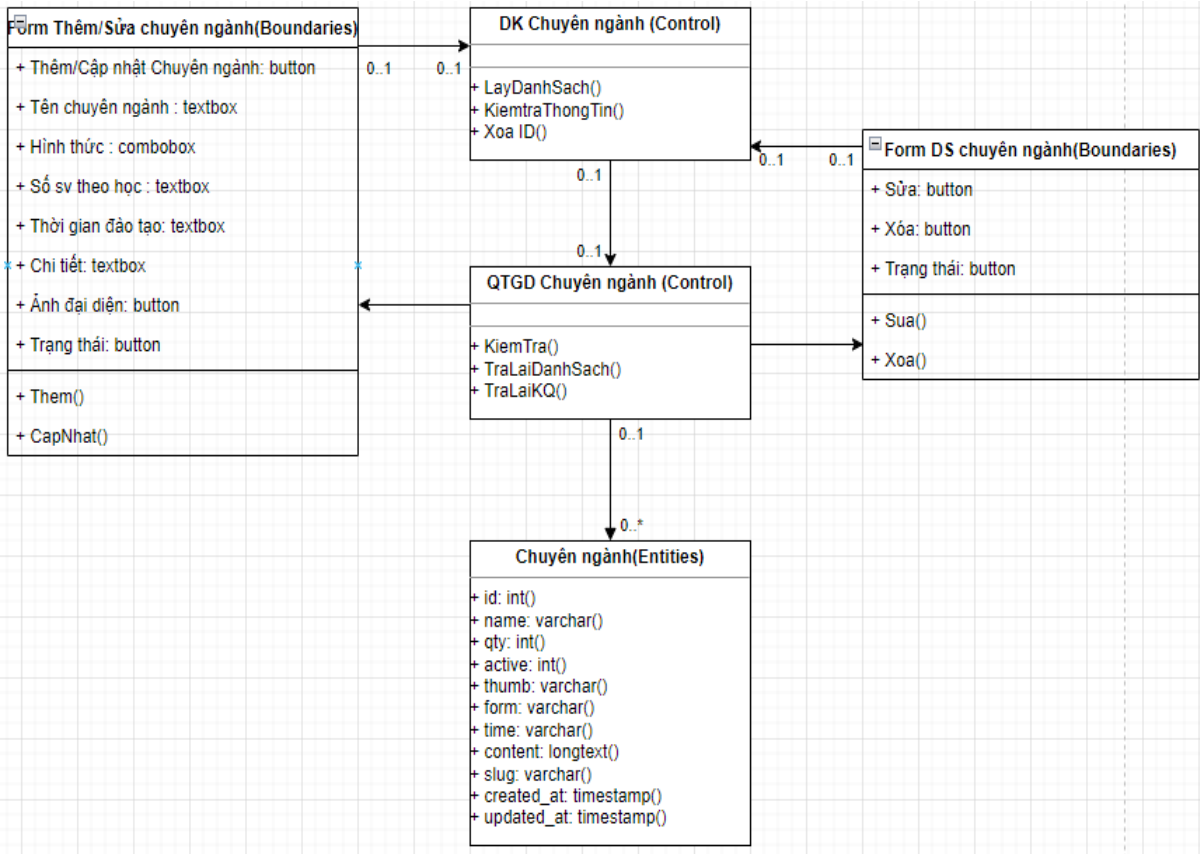
Hình 2.50: Biểu đồ lớp Use case quản lý bài viết

## 2.8.5 Biểu đồ lớp Use case quản lý giảng viên



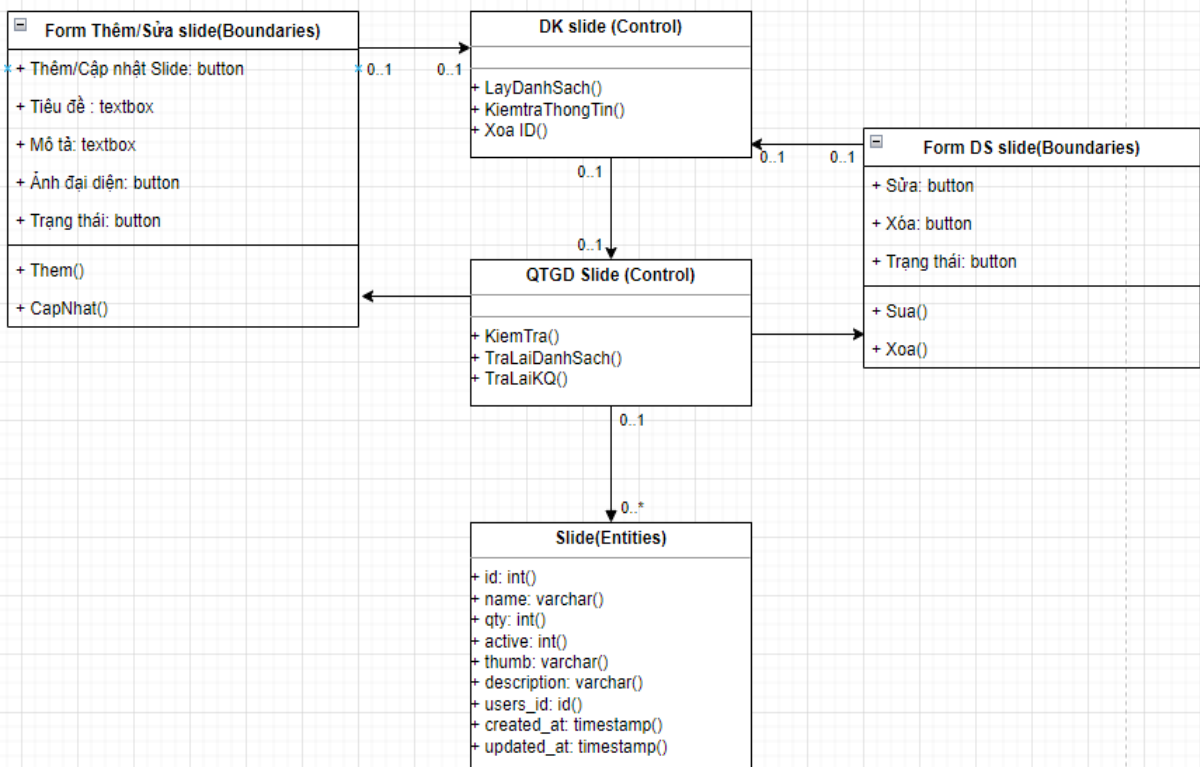
Hình 2.51: Biểu đồ lớp Use case quản lý giảng viên

## 2.8.6 Biểu đồ lớp Use case quản lý chuyên ngành



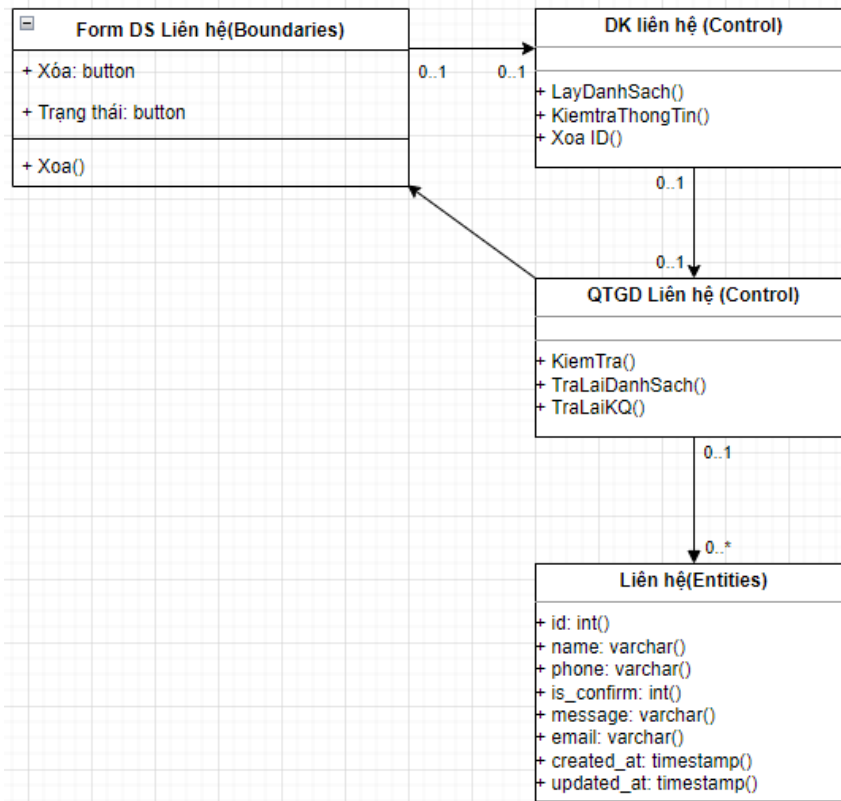
Hình 2.52: Biểu đồ lớp Use case quản lý chuyên ngành

## 2.8.7 Biểu đồ lớp Use case quản lý slide



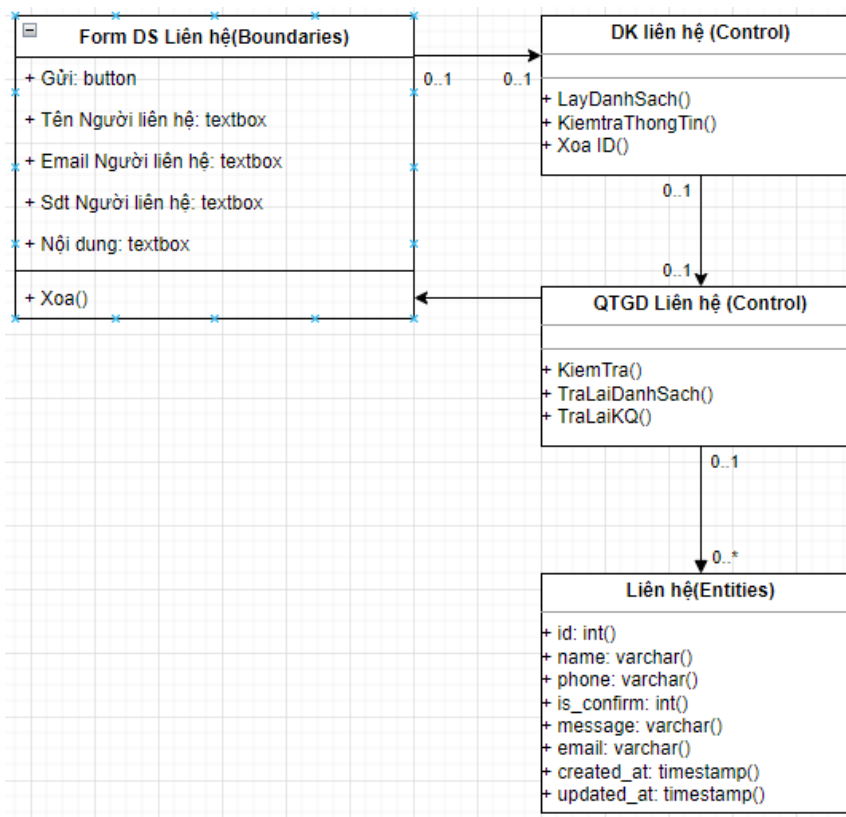
Hình 2.53: Biểu đồ lớp Use case quản lý slide

### 2.8.9 Biểu đồ lớp Use case quản lý liên hệ



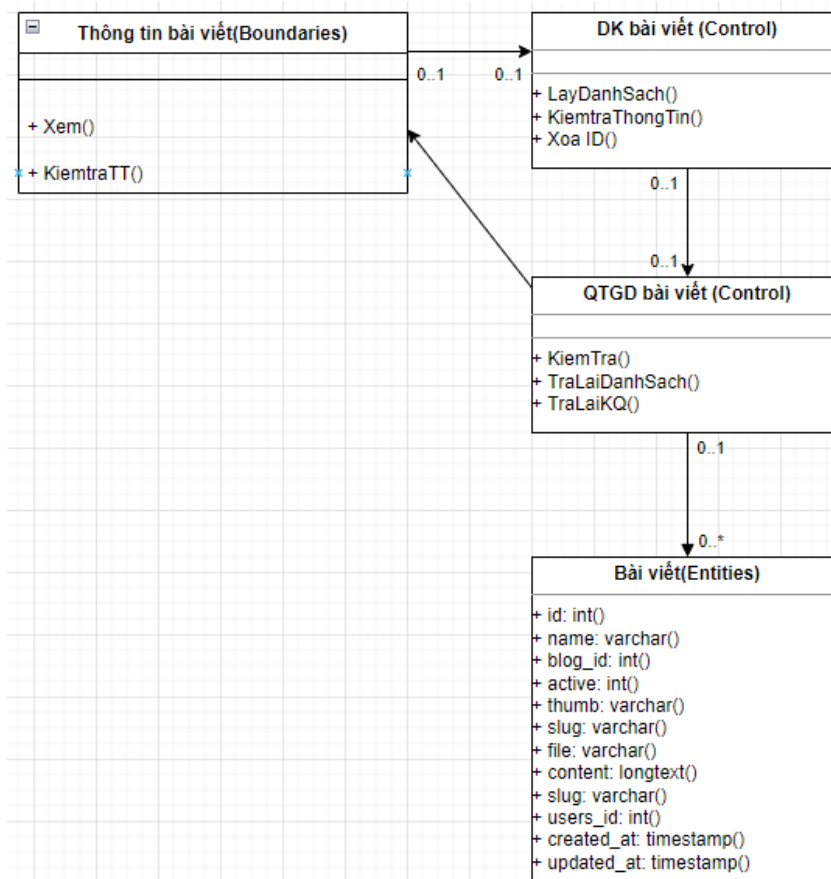
Hình 2.54: Biểu đồ lớp Use case quản lý liên hệ

### 2.8.10 Biểu đồ lớp Use case gửi liên hệ



Hình 2.55: Biểu đồ lớp Use case gửi liên hệ

## 2.8.11 Biểu đồ lớp Use case xem bài viết



Hình 2.56: Biểu đồ lớp Use case xem bài viết

## 2.9 Thiết kế cơ sở dữ liệu

### 2.9.1 Danh sách các bảng

Bảng 2.30: Danh sách các bảng

STT	Tên bảng	Mô tả
1	Users	Lưu thông tin quản lý người dùng
2	Major	Lưu thông tin quản lý chuyên ngành
3	Slide	Lưu thông tin quản lý slide
4	Blog	Lưu thông tin quản lý chuyên mục
5	News	Lưu thông tin quản lý bài viết
6	Lecturer	Lưu thông tin quản lý giảng viên
7	Contact	Lưu thông tin quản lý liên hệ

## 2.9.2 Bảng Users

Bảng 2.31: Bảng User

STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	Id	<i>int (6)</i>	<i>Khóa chính</i>	Khóa chính của bảng
2	Is_admin	<i>int (1)</i>	<i>Not null</i>	Vai trò
3	Name	<i>varchar (255)</i>	<i>Null</i>	Tên
4	Email	<i>varchar (255)</i>	<i>Not null</i>	Email (tên TK đăng nhập)
5	Password	<i>varchar (255)</i>	<i>Not null</i>	Mật khẩu
6	File	<i>varchar (255)</i>	<i>Null</i>	Ảnh đại diện
7	Fullname	<i>varchar (255)</i>	<i>Null</i>	Tên đầy đủ
8	Job	<i>varchar (255)</i>	<i>Null</i>	Nghề nghiệp
9	Phone	<i>varchar (255)</i>	<i>Null</i>	Số điện thoại
10	Address	<i>varchar (255)</i>	<i>Null</i>	Địa chỉ
11	Facebook	<i>varchar (255)</i>	<i>Null</i>	Link facebook
12	Instagram	<i>varchar (255)</i>	<i>Null</i>	Link instagram
13	Created_at	<i>timestamp</i>	<i>Null</i>	Ngày tạo
14	Updated_at	<i>timestamp</i>	<i>Null</i>	Ngày cập nhật

## 2.9.3 Bảng Major

Bảng 2.32: Bảng Major

STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	Id	<i>int (6)</i>	<i>Khóa chính</i>	Khóa chính của bảng
2	Name	<i>varchar (255)</i>	<i>Not null</i>	Tên chuyên ngành
3	Qty	<i>int (11)</i>	<i>Null</i>	Số sinh viên theo học
4	Form	<i>varchar (255)</i>	<i>Null</i>	Hình thức đào tạo
5	Time	<i>varchar (255)</i>	<i>Null</i>	Thời gian đào tạo

6	Active	<i>int (1)</i>	<i>Null</i>	Trạng thái
7	File	<i>varchar (255)</i>	<i>Null</i>	Ảnh đại diện
8	Content	<i>longtext</i>	<i>Null</i>	Nội dung chính
9	Slug	<i>varchar (255)</i>	<i>Null</i>	Slug của tên chuyên ngành
10	Created_at	<i>timestamp</i>	<i>Null</i>	Ngày tạo
11	Updated_at	<i>timestamp</i>	<i>Null</i>	Ngày cập nhật

### 2.9.4 Bảng Slide

Bảng 2.33: Bảng Slide

STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	Id	<i>int (6)</i>	<i>Khóa chính</i>	Khóa chính của bảng
2	Name	<i>varchar (255)</i>	<i>Not null</i>	Tiêu đề Slide
3	Description	<i>varchar (255)</i>	<i>Null</i>	Mô tả
4	Users_id	<i>int (6)</i>	<i>Null</i>	ID của Users
4	Active	<i>int (1)</i>	<i>Null</i>	Trạng thái
5	Thumb	<i>varchar (255)</i>	<i>Null</i>	Ảnh Slide
6	Created_at	<i>timestamp</i>	<i>Null</i>	Ngày tạo
7	Updated_at	<i>timestamp</i>	<i>Null</i>	Ngày cập nhật

### 2.9.5 Bảng Blog

Bảng 2.34: Bảng Blog

STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	Id	<i>int (6)</i>	<i>Khóa chính</i>	Khóa chính của bảng
2	Name	<i>varchar (255)</i>	<i>Not null</i>	Tên chuyên mục
3	Parent_id	<i>int (11)</i>	<i>Null</i>	ID của danh mục cha
4	Description	<i>varchar (255)</i>	<i>Null</i>	Mô tả chuyên mục
5	Active	<i>int (1)</i>	<i>Null</i>	Trạng thái
6	Order_by	<i>int (6)</i>	<i>Null</i>	Thứ tự sắp xếp



7	Slug	<i>varchar (255)</i>	<i>Null</i>	Slug của tên chuyên mục
8	Created_at	<i>timestamp</i>	<i>Null</i>	Ngày tạo
9	Updated_at	<i>timestamp</i>	<i>Null</i>	Ngày cập nhật

### 2.9.6 Bảng News

Bảng 2.35: Bảng News

STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	Id	<i>int (6)</i>	<i>Khóa chính</i>	Khóa chính của bảng
2	Name	<i>varchar (255)</i>	<i>Not null</i>	Tên bài viết
3	Blog_id	<i>int (11)</i>	<i>Null</i>	ID của chuyên mục
4	Thumb	<i>varchar (255)</i>	<i>Null</i>	Ảnh đại diện
5	Users_id	<i>int (11)</i>	<i>Null</i>	ID của Users
6	File	<i>varchar (255)</i>	<i>Null</i>	File đính kèm
7	Content	<i>longtext</i>	<i>Not null</i>	Nội dung bài viết
8	Active	<i>int (1)</i>	<i>Null</i>	Trạng thái
9	Slug	<i>varchar (255)</i>	<i>Null</i>	Slug của tên bài viết
10	Created_at	<i>timestamp</i>	<i>Null</i>	Ngày tạo
11	Updated_at	<i>timestamp</i>	<i>Null</i>	Ngày cập nhật

### 2.9.7 Bảng Lecturer

Bảng 2.36: Bảng Lecturer

STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	Id	<i>int (6)</i>	<i>Khóa chính</i>	Khóa chính của bảng
2	Name	<i>varchar (255)</i>	<i>Not null</i>	Tên giảng viên
3	Level	<i>varchar (255)</i>	<i>Null</i>	Trình độ
4	Thumb	<i>varchar (255)</i>	<i>Null</i>	Ảnh đại diện
5	Position	<i>varchar (255)</i>	<i>Null</i>	Chức vụ
6	Phone	<i>varchar (255)</i>	<i>Null</i>	Số điện thoại

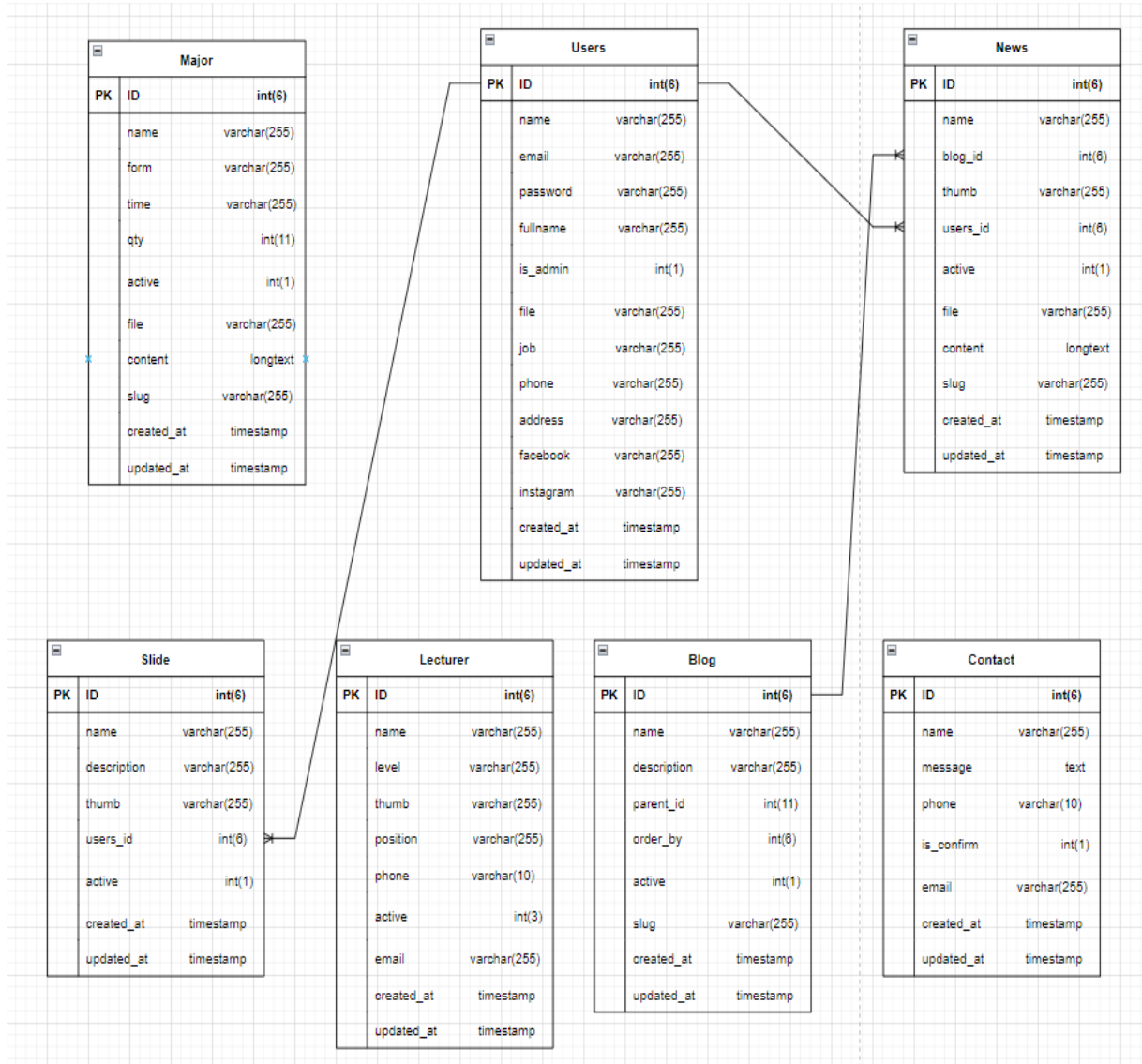
7	Active	<i>int (3)</i>	<i>Null</i>	Trạng thái
8	Email	<i>varchar (255)</i>	<i>Null</i>	Email giảng viên
9	Created_at	<i>timestamp</i>	<i>Null</i>	Ngày tạo
10	Updated_at	<i>timestamp</i>	<i>Null</i>	Ngày cập nhật

### 2.9.8 Bảng Contact

Bảng 2.37: Bảng Contact

STT	Tên thực thể	Kiểu dữ liệu	Ràng buộc	Mô tả
1	Id	<i>int (6)</i>	<i>Khóa chính</i>	Khóa chính của bảng
2	Name	<i>varchar (255)</i>	<i>Not null</i>	Tên người liên hệ
3	Phone	<i>varchar (10)</i>	<i>Null</i>	Số điện thoại
4	Email	<i>varchar (255)</i>	<i>Null</i>	Email người liên hệ
5	Message	<i>Text</i>	<i>Not null</i>	Nội dung
6	Is_confirm	<i>int (1)</i>	<i>Null</i>	Trạng thái
7	Created_at	<i>timestamp</i>	<i>Null</i>	Ngày tạo

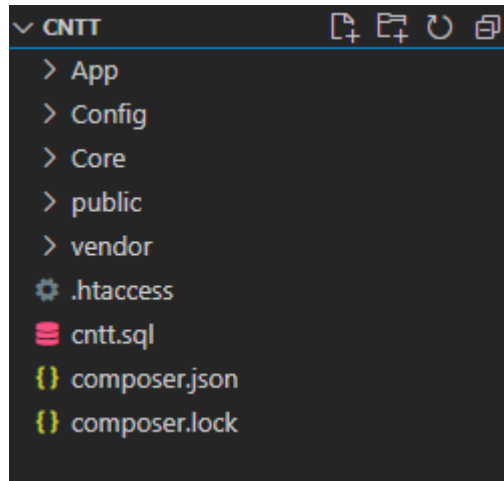
## 2.9.9 Mô hình CSDL



Hình 2.57: Mô hình CSDL

## CHƯƠNG 3. DEMO WEBSITE

### 3.1 Cấu trúc thư mục



Hình 3.1: Các thư mục trong source website

Trong dự án gồm nhiều thư mục, mỗi thư mục khác nhau sẽ giữ vai trò và chức năng khác nhau:

#### 3.1.1 App

Thư mục App chứa toàn bộ thông tin người dùng code. Trong App sẽ bao gồm các thư mục như:

- *Controller*: Giữ nhiệm vụ tiếp nhận, điều hướng yêu cầu từ end-user để gọi đúng phương thức xử lý, thao tác trực tiếp với Model và trả về dữ liệu cho View.
- *Model*: Là nơi chứa các logic, nghiệp vụ tương tác với dữ liệu hoặc hệ quản trị cơ sở dữ liệu (MySQL, SQL Server...), nó sẽ bao gồm các phương thức xử lý kết nối database, truy vấn dữ liệu. Là nơi lưu giữ các đối tượng mô tả dữ liệu, như là Class và các hàm xử lý get, set của Class...
- *View*: Đảm nhận việc hiển thị, trả về thông tin, dữ liệu cho end-user. Ví dụ như hiển thị UI/UX, hiển thị dữ liệu ra cho người dùng xem website, hoặc có thể là một đoạn XML hoặc JSON...
- *Core*: Là thư mục chứa class có nhiệm vụ phân quyền truy cập giữa quản trị viên và người dùng.
- *Helper*: Là thư mục chứa các function dạng tĩnh, mục đích để xử lý các nhiệm vụ phát sinh ...

### **3.1.2 Config**

Chứa các file config như DB dùng để kết nối đến CSDL, Route dùng để định hướng các request, các hằng số ...

### **3.1.3 Public**

Thư mục public chứa các file dạng công khai như ảnh, các file javascript, css ...

### **3.1.4 Core**

Là phần cốt lõi của dự án, trong Core chủ yếu là các hàm php có nhiệm vụ xử lý thông tin người dùng gọi lên sever để các hàm trong thư mục App xử lý như App, Controller, Model, Session...

### **3.1.5 Htaccess**

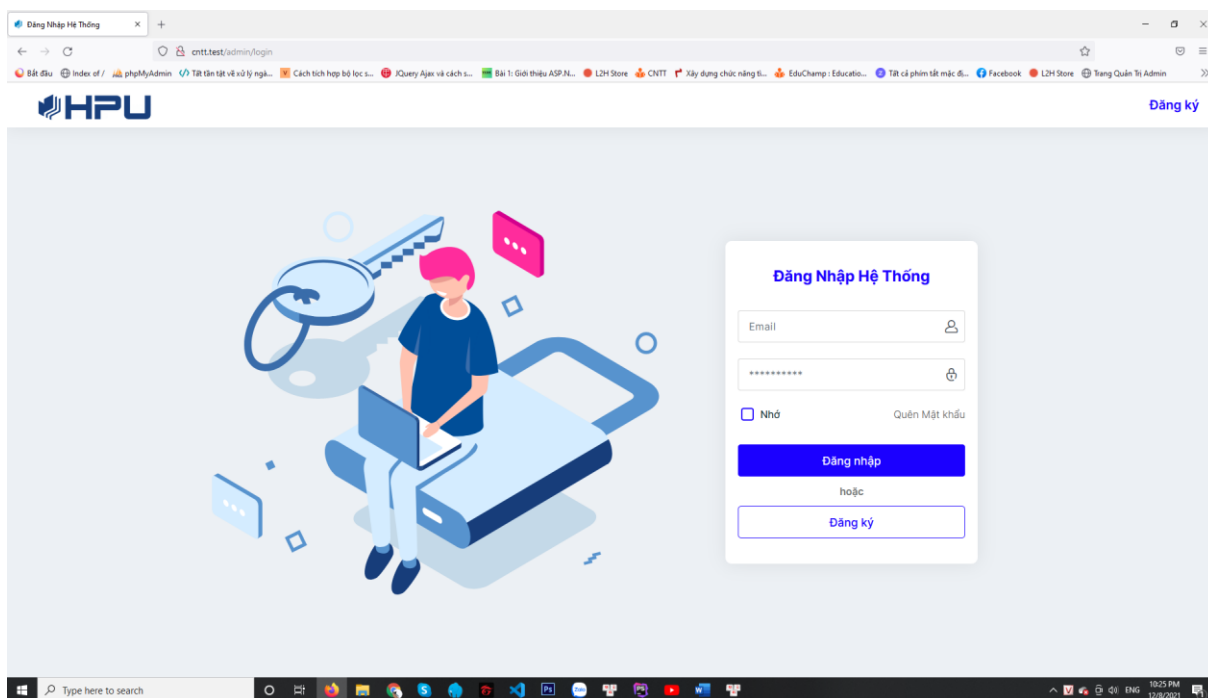
Là một tập tin dùng để cấu hình máy chủ web Apache. File được máy chủ chấp nhận như là một thành phần và cho phép chúng ta thực hiện điều hướng và bật các tính năng một cách linh hoạt hoặc bảo vệ một phần folder nào đó của trang web.

### **3.1.6 Composer**

Là công cụ quản lý các thư viện mà project PHP. Nó quản lý sự phụ thuộc các tài nguyên trong dự án, cho phép khai báo các thư viện mà dự án sử dụng, tự động tải code của các thư viện...

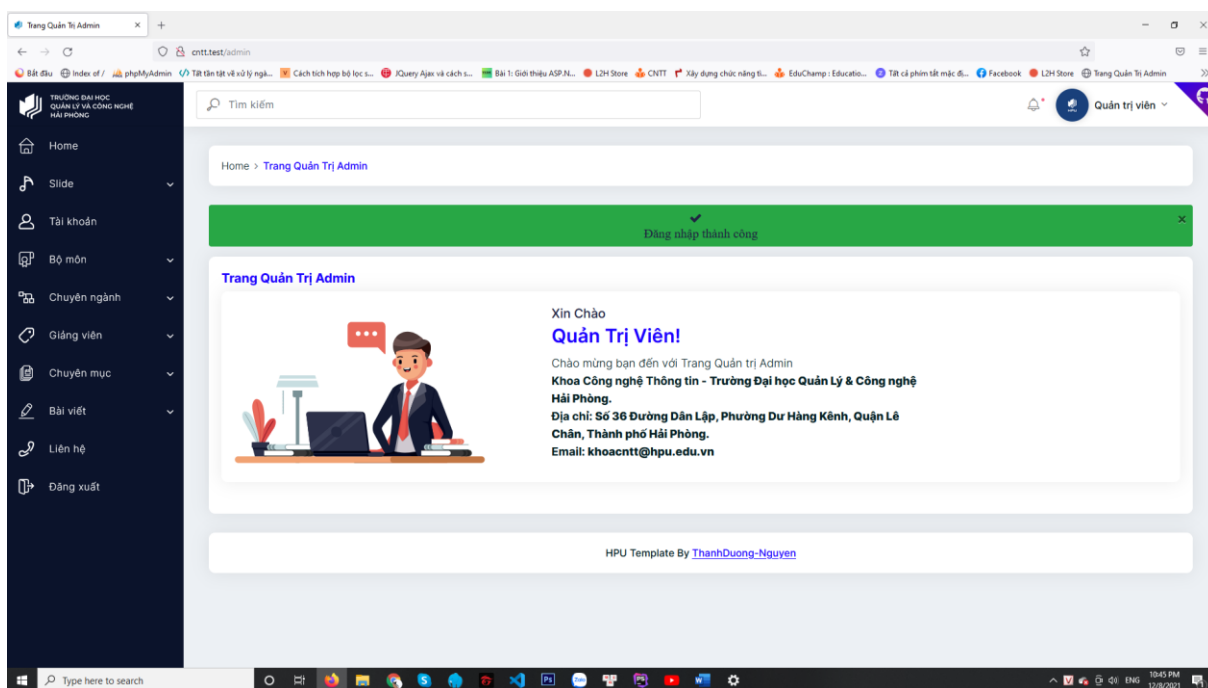
## 3.2 Demo giao diện quản trị viên

### 3.2.1 Giao diện đăng nhập



Hình 3.2: Giao diện đăng nhập

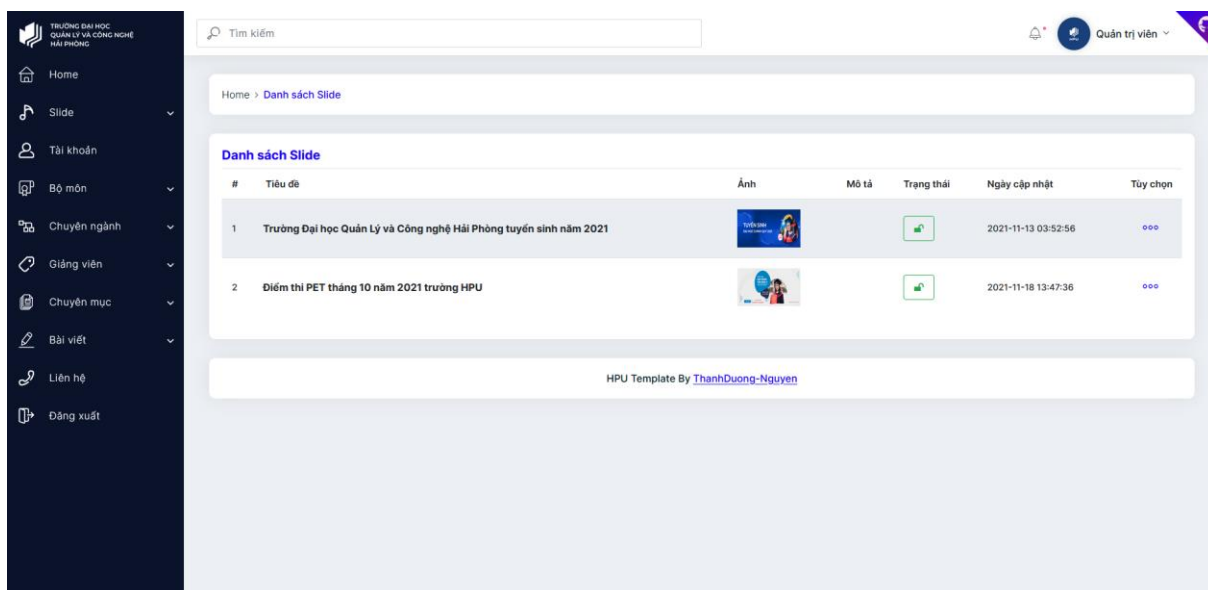
Giao diện đăng nhập để có thể vào thì người dùng phải có tài khoản quản trị viên. Khi nhập thông tin email hoặc mật khẩu sai, hệ thống sẽ thông báo và không cho người dùng truy cập vào được trang quản trị viên.



Hình 3.3: Giao diện trang chủ trang quản trị viên

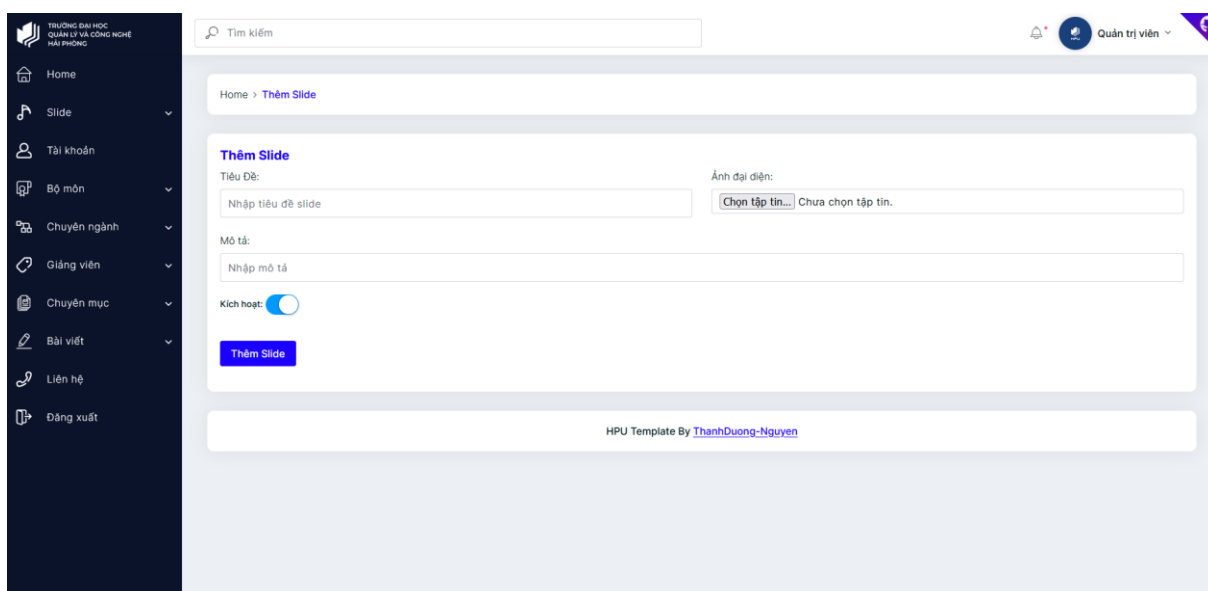
Khi đăng nhập thành công hệ thống sẽ đưa người dùng đến giao diện trang chủ quản trị viên, tới đây người dùng đã nắm mọi quyền quản trị của website.

### 3.2.2 Giao diện quản lý Slide



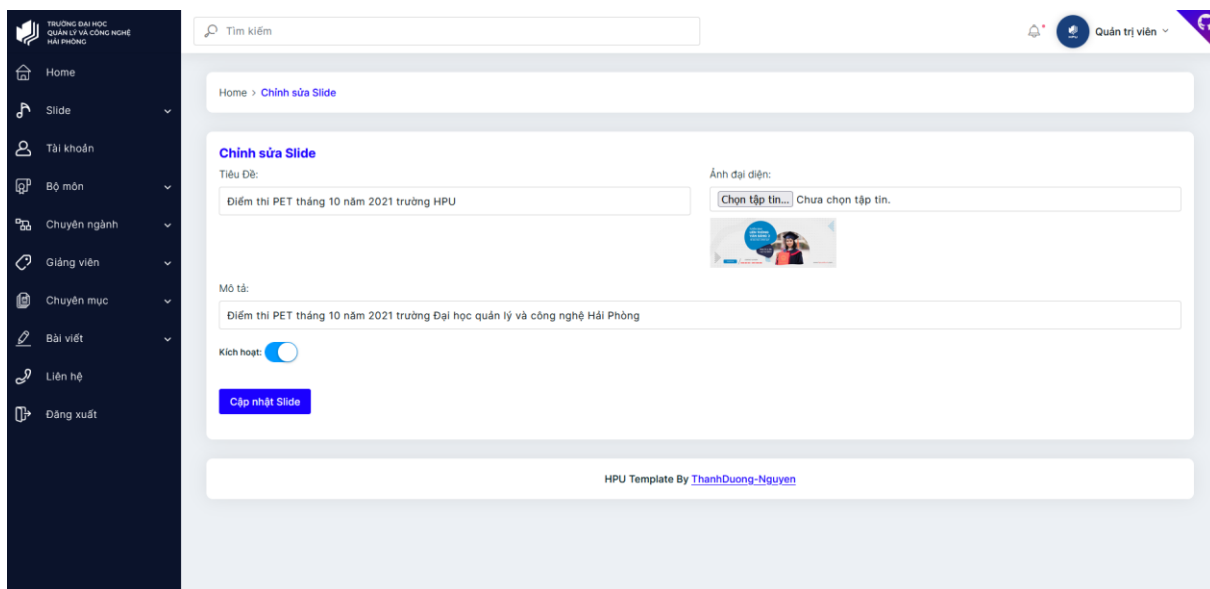
Hình 3.4: Giao diện quản lý Slide

Giao diện quản lý Slide hiển thị thông tin của các slide như tiêu đề, ảnh slide, mô tả, trạng thái, ngày cập nhật của slide.



Hình 3.5: Giao diện thêm Slide

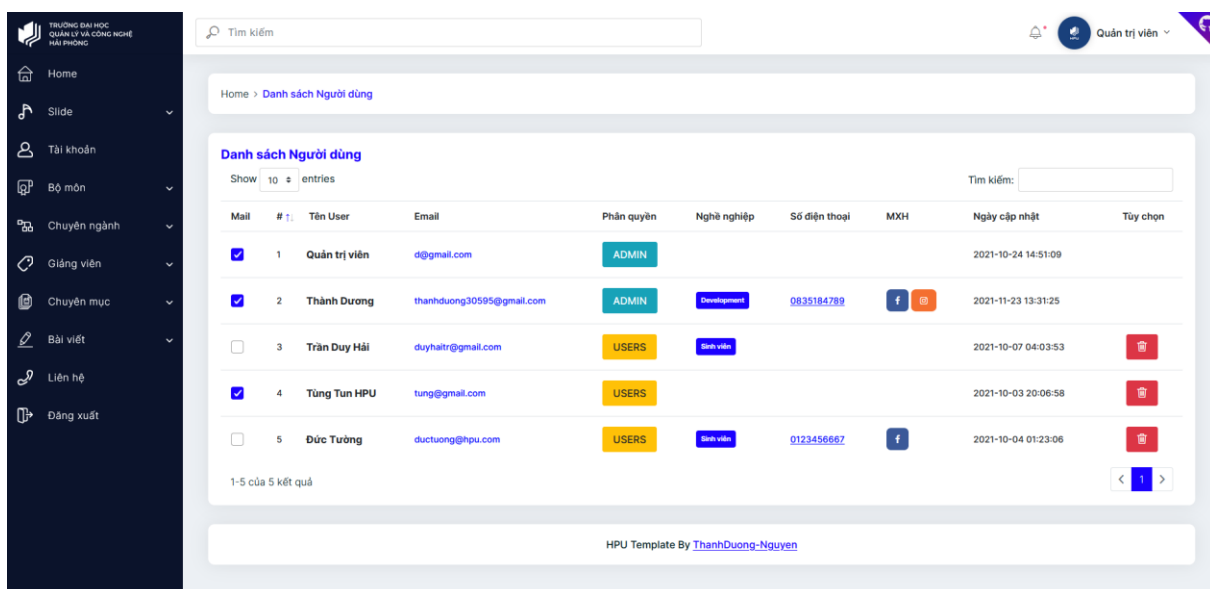
Giao diện thêm Slide là form để người dùng nhập các thông tin cần thiết để thêm được slide mới.



Hình 3.6: Giao diện sửa Slide

Giao diện sửa Slide cho phép quản trị viên sửa các thông tin có sẵn của slide và cập nhật lại nó lên hệ thống.

### 3.2.3 Giao diện quản lý người dùng

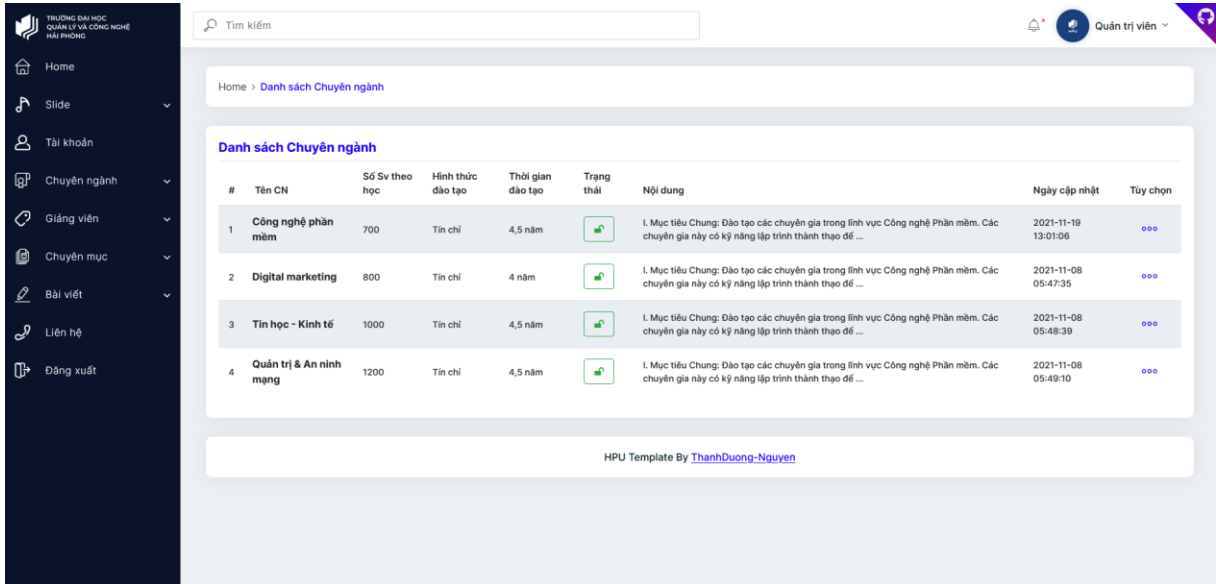


Hình 3.7: Giao diện quản lý người dùng

Giao diện quản lý Người dùng hiển thị thông tin của tất cả người dùng và quản trị viên. Giao diện hiển thị đầy đủ thông tin như tên, tên đầy đủ, phân quyền truy cập người dùng, nghề nghiệp, số điện thoại, facebook, instagram, ngày cập nhật tài khoản của người dùng.



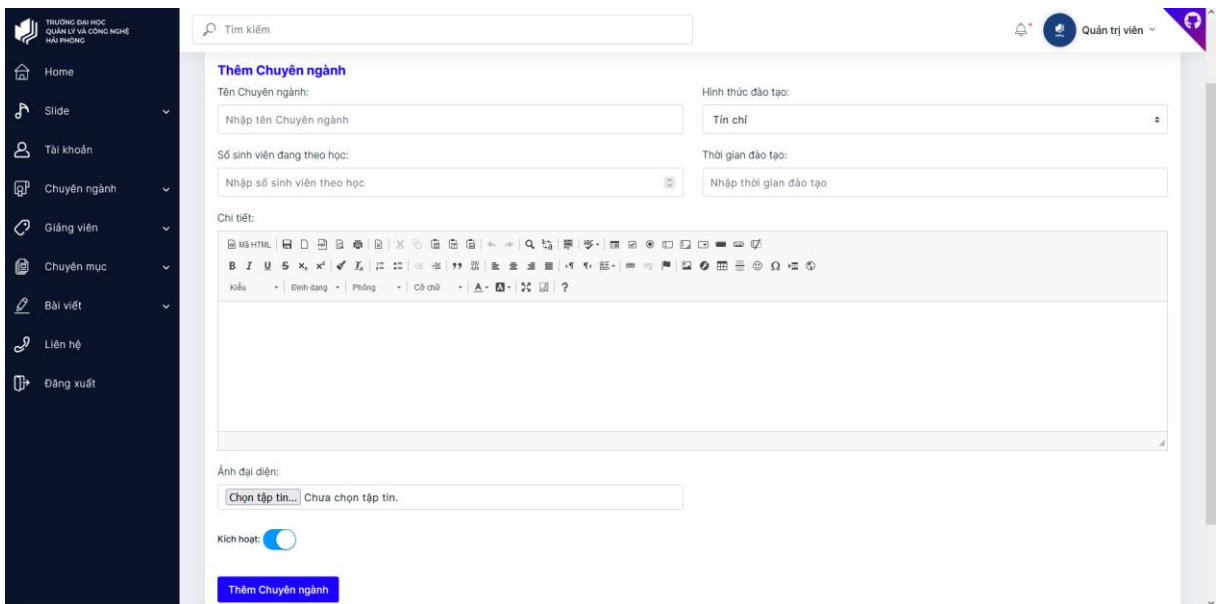
### 3.2.4 Giao diện quản lý chuyên ngành



Hình 3.8: Giao diện quản lý chuyên ngành

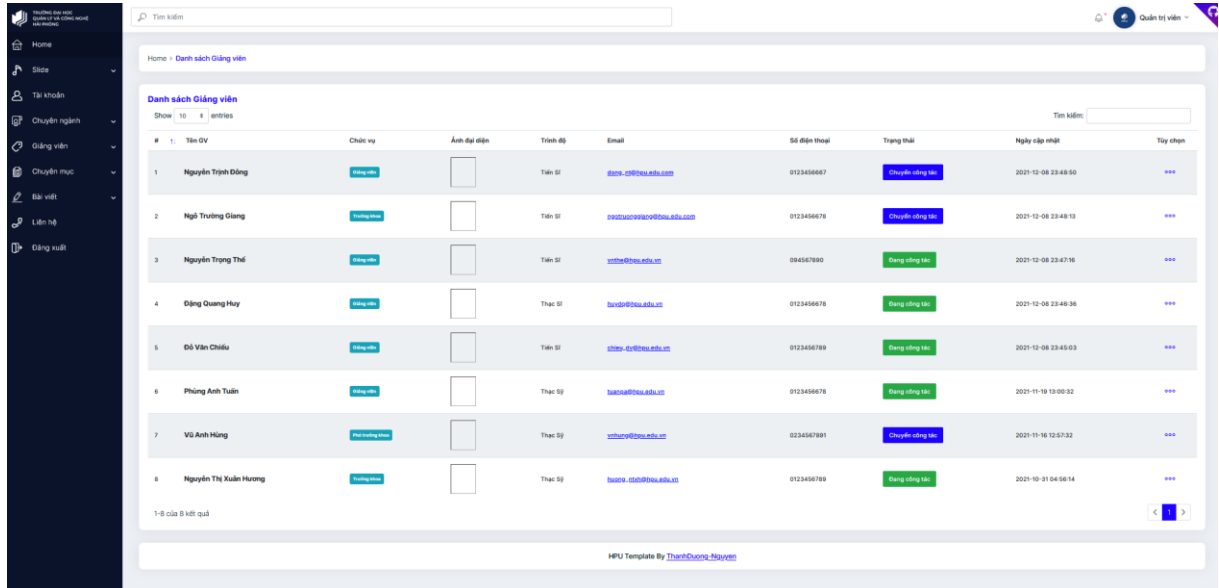
Tương tự, giao diện quản lý chuyên ngành sẽ hiển thị các thông tin của chuyên ngành như tên chuyên ngành, số sinh viên đang theo học chuyên ngành đó, hình thức đào tạo, thời gian đào tạo, nội dung, trạng thái và ngày cập nhật của chuyên ngành.

Còn đây là giao diện khi quản trị viên muốn thêm chuyên ngành mới.



Hình 3.9: Giao diện thêm chuyên ngành

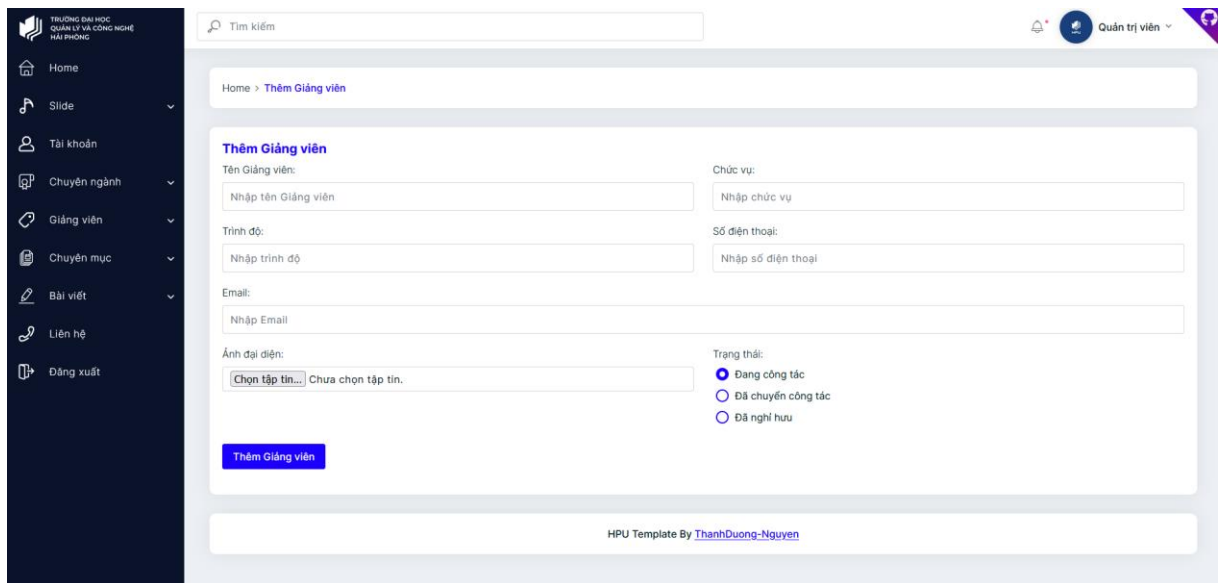
### 3.2.5 Giao diện quản lý giảng viên



Hình 3.10: Giao diện quản lý giảng viên

Giao diện quản lý giảng viên sẽ hiển thị danh sách các giảng viên đã và đang công tác tại Khoa, hiển thị đầy đủ các thông tin như họ tên, ảnh đại diện, trình độ, chức vụ, email, số điện thoại, tình trạng cũng như ngày cập nhật của các giảng viên.

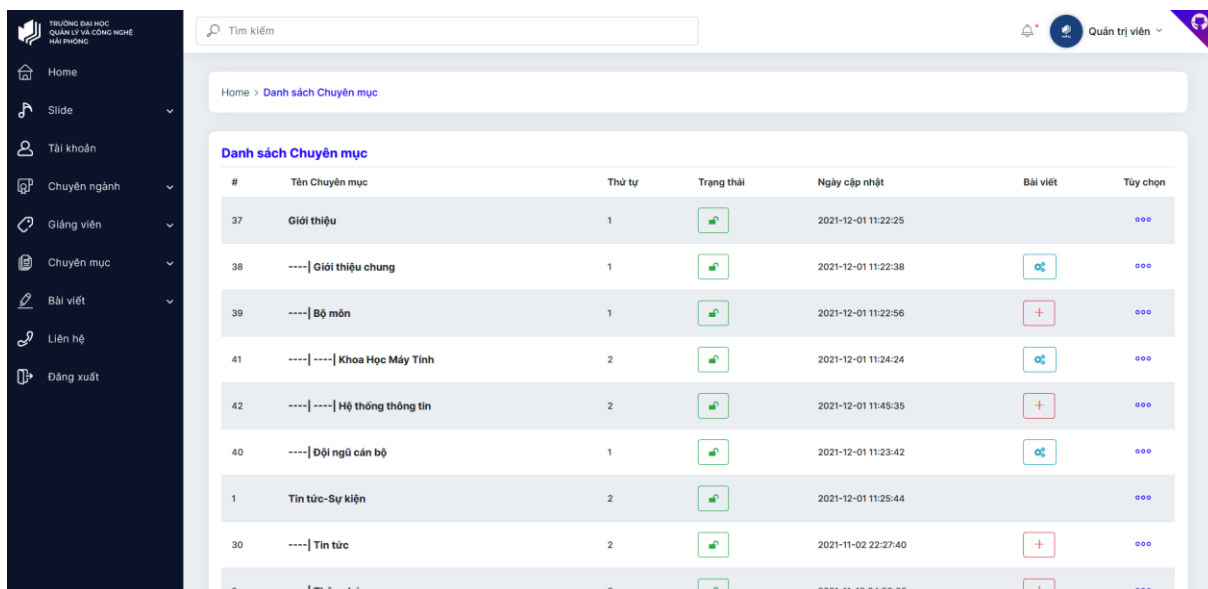
Còn khi muốn thêm mới 1 giảng viên nào đó, hệ thống sẽ hiển thị ra form như sau:



Hình 3.11: Giao diện thêm giảng viên mới

### 3.2.6 Giao diện quản lý chuyên mục

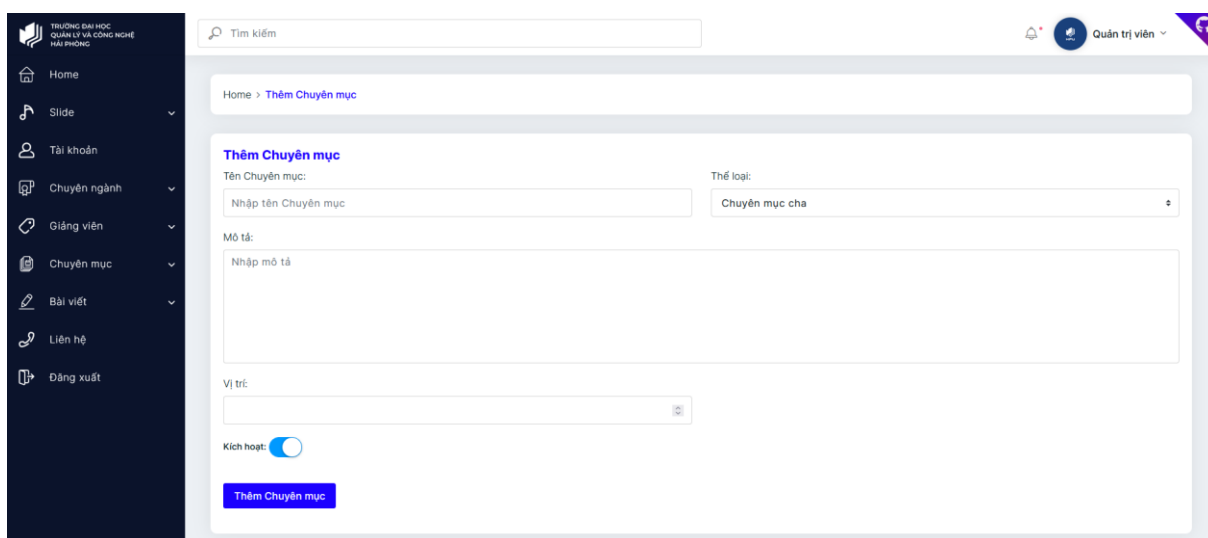
Giao diện chuyên mục là các chuyên mục xuất hiện trên trang chủ của giao diện người dùng, các chuyên mục sẽ có thể có các chuyên mục con, các chuyên mục có thể bao gồm 1 list các bài viết (VD: chuyên mục cơ hội việc làm gồm tất cả bài viết về cơ hội việc làm cho sinh viên của Khoa) hay cũng có thể là 1 bài viết riêng (VD: chuyên mục giới thiệu chung là bài viết giới thiệu chung về thông tin của Khoa như các chuyên ngành đào tạo, tôn chỉ, lịch sử hình thành...)



#	Tên Chuyên mục	Thứ tự	Trạng thái	Ngày cập nhật	Bài viết	Tùy chọn
37	Giới thiệu	1		2021-12-01 11:22:25		...
38	----  Giới thiệu chung	1		2021-12-01 11:22:38		...
39	----  Bộ môn	1		2021-12-01 11:22:56		...
41	---- ----  Khoa Học Máy Tính	2		2021-12-01 11:24:24		...
42	---- ----  Hệ thống thông tin	2		2021-12-01 11:45:35		...
40	----  Đội ngũ cán bộ	1		2021-12-01 11:23:42		...
1	Tin tức-Sự kiện	2		2021-12-01 11:25:44		...
30	----  Tin tức	2		2021-11-02 22:27:40		...
2	----  Thông báo	3		2021-11-13 04:53:33		...

Hình 3.12: Giao diện quản lý chuyên mục

Nhìn vào hình ta có thể thấy được thông tin của các chuyên mục, các chuyên mục con của nó, thứ tự xuất hiện trên website, trạng thái, chuyên mục dạng list hay dạng bài viết...



Thêm Chuyên mục

Tên Chuyên mục:

Thể loại:

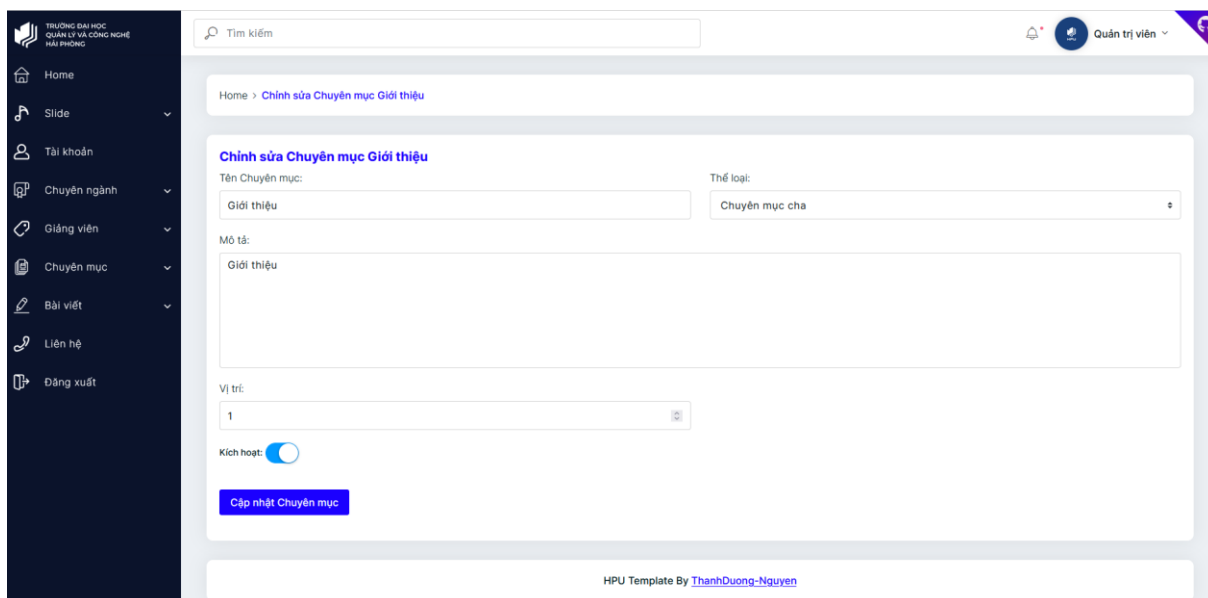
Mô tả:

Vị trí:




Kích hoạt:

Hình 3.13: Giao diện thêm chuyên mục

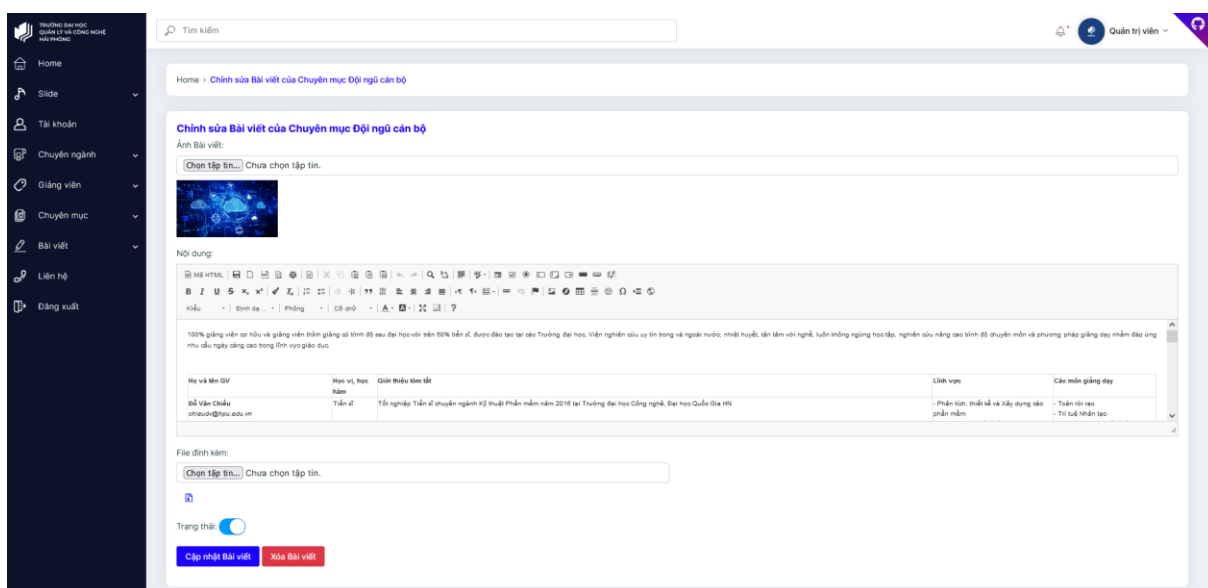
Giao diện sửa chuyên mục thực chất giống giao diện thêm chuyên mục.



Hình 3.14: Giao diện sửa chuyên mục

Khi muốn tạo bài viết cho chuyên mục, ta ấn vào biểu tượng , còn khi chuyên mục đã tồn tại bài viết sẵn, hệ thống sẽ có biểu tượng . Ta có thể sửa hoặc xóa bài viết đó khi ấn vào biểu tượng  đó.

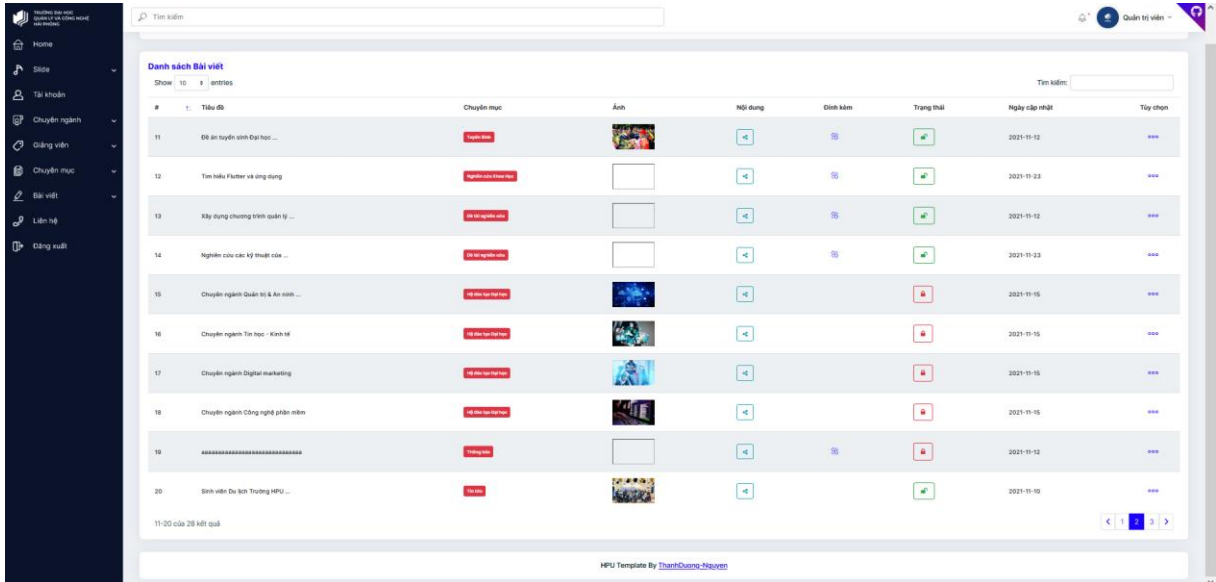
Khi đó, giao diện sẽ hiển thị:



Hình 3.15: Giao diện sửa hoặc xóa bài viết của chuyên mục

Bằng cách ấn vào 2 nút ở cuối form, ta đã có thể cập nhật lại thông tin hoặc xóa bài viết cho chuyên mục này nếu muốn.

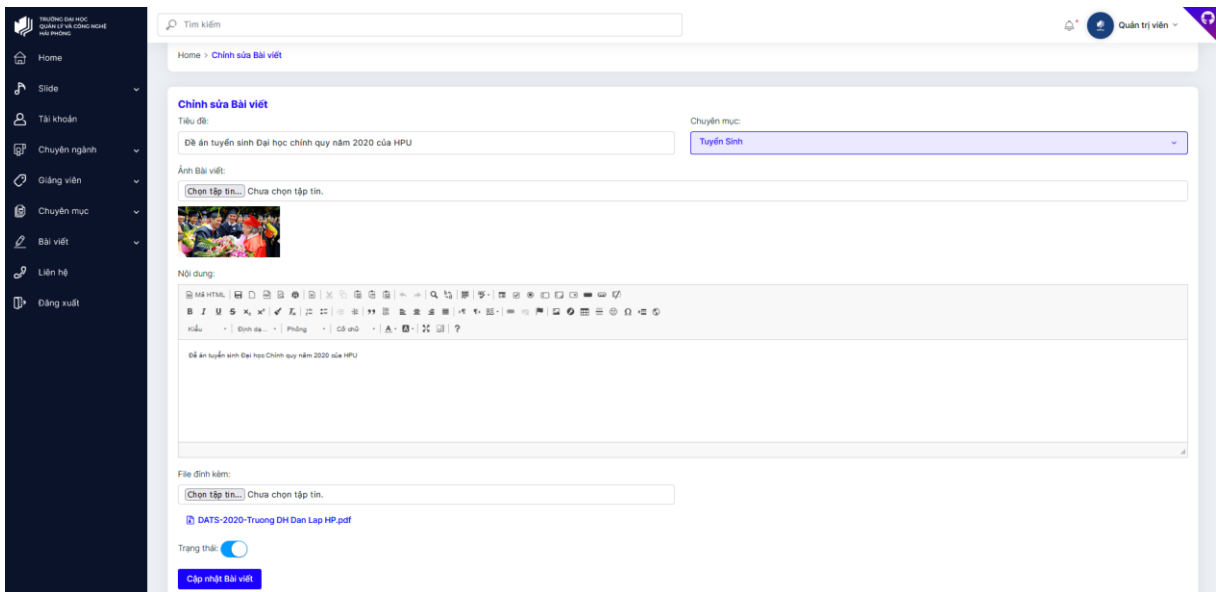
### 3.2.7 Giao diện quản lý bài viết



Hình 3.16: Giao diện quản lý bài viết

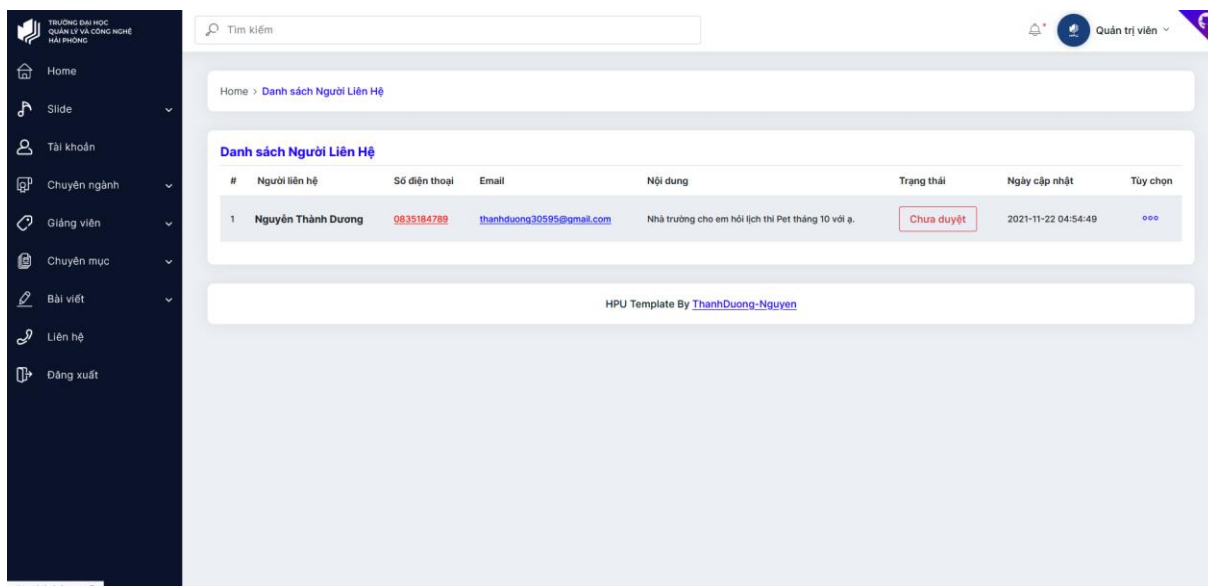
Giao diện quản lý bài viết hiển thị danh sách các bài viết, thông tin các bài viết gồm tiêu đề bài viết, chuyên mục, ảnh đại diện, nội dung bài viết, các tệp đính kèm, trạng thái và ngày cập nhật.

Nếu muốn sửa bài viết nào, ta chỉ cần ấn vào mục “Tùy chọn” chọn sửa. Hệ thống sẽ hiển thị ngay phần form chỉnh sửa bài viết đó.



Hình 3.17: Giao diện chỉnh sửa bài viết

### 3.2.8 Giao diện quản lý liên hệ

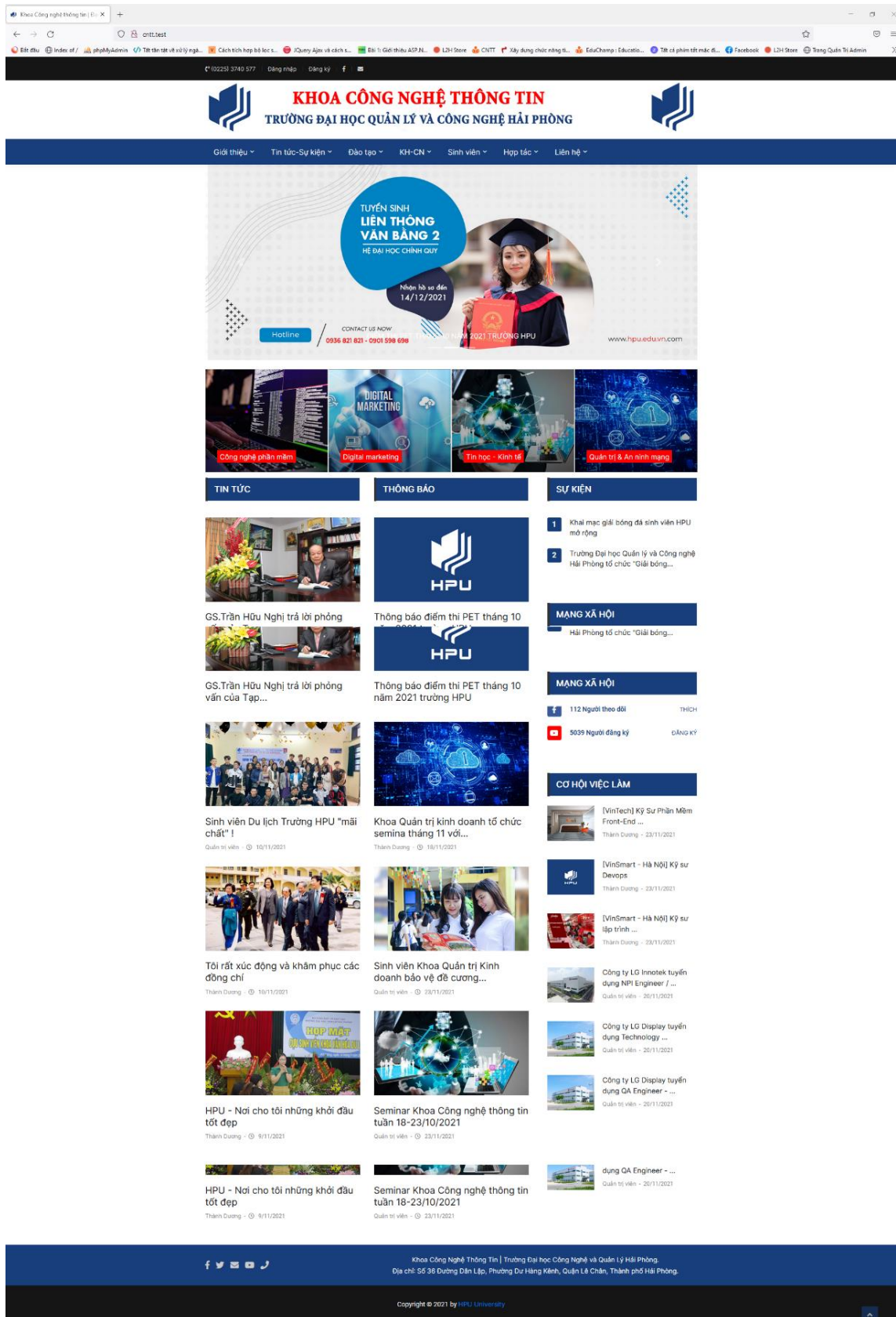


Hình 3.18: Giao diện quản lý liên hệ

Quản trị viên có thể nhìn thấy các liên hệ phản hồi như họ tên, email, số điện thoại, nội dung và quản trị viên có thể xóa các liên hệ nếu muốn.

## 3.3 Demo giao diện người dùng

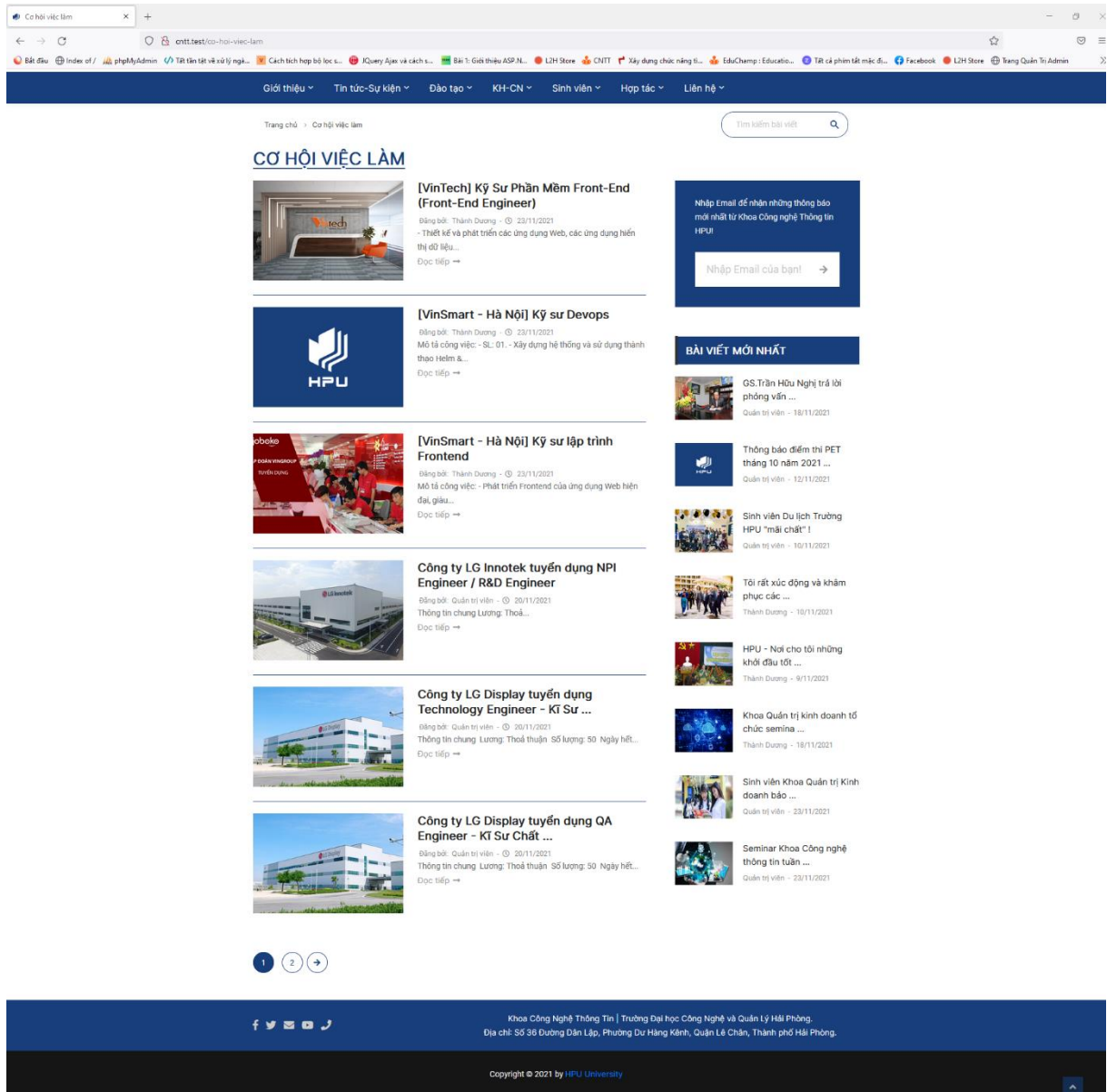
### 3.3.1 Giao diện trang chủ



Hình 3.19: Giao diện trang chủ Website

Tại giao diện trang chủ sẽ hiện các thông tin như banner, logo của Khoa, danh sách các chuyên mục, chuyên ngành, các bài viết thuộc chuyên mục tin tức, thông báo và sự kiện. Ngoài ra phần sidebar có sự kiện, cơ hội việc làm cho sinh viên ...

### 3.3.2 Giao diện danh sách bài viết



Hình 3.20: Giao diện danh sách bài viết

Giao diện danh sách bài viết sẽ hiển thị ra danh sách các bài viết thuộc chuyên mục, phần sidebar gồm những bài viết mới nhất và gửi Mail.



### 3.3.3 Giao diện chi tiết bài viết



Hình 3.21: Giao diện chi tiết bài viết

Trương tự, giao diện chi tiết bài viết sẽ có ảnh của bài viết, nội dung bài viết, tác giả và thời gian bài viết.

### 3.3.4 Giao diện trang liên hệ

The screenshot shows the contact page of the Faculty of Information Technology, Hanoi University of Management and Technology. The page has a dark blue header with the university's name and logo. Below the header is a navigation menu with links for 'Giới thiệu', 'Tin tức-Sự kiện', 'Đào tạo', 'KH-CN', 'Sinh viên', 'Hợp tác', and 'Liên hệ'. The main content area is white and contains a contact form titled 'Liên Hệ'. The form has four input fields: 'Nhập tên của bạn (\*)', 'Email của bạn (\*)', 'Số điện thoại (\*)', and 'Nhập nội dung (\*)'. A 'Gửi' button is located below the form. To the right of the form is a blue sidebar with a search bar and a button labeled 'Nhập Email của bạn!' with a right arrow. The footer is dark blue and contains social media icons and contact information.

Hình 3.22: Giao diện trang liên hệ

Giao diện trang liên hệ là 1 form để người dùng nhập các thông tin như tên người liên hệ, email, số điện thoại và nội dung liên hệ. Khi người dùng nhấn “Gửi” ngay lập tức thông tin sẽ được gửi lên sever và quản trị viên sẽ đọc và trả lời cho họ.

### 3.3.5 Giao diện website trên thiết bị di động



Hình 3.23: Giao diện website trên thiết bị di động

## KẾT LUẬN

Về cơ bản hiểu được quá trình xây dựng một website bằng một ngôn ngữ lập trình PHP. Xây dựng được trang với giao diện thân thiện, từ ngữ dễ hiểu dễ sử dụng với người dùng. Đồng thời tạo được các chức năng quản lý nội dung khiến người dùng dễ dàng thao tác, và quản lý. Xây dựng chức năng phân quyền người dùng khi đăng nhập để quản lý nội dung. Xây dựng trang web hiển thị nội dung linh hoạt, giúp cập nhật tin tức, hình ảnh, bài viết dễ dàng. Thiết kế giao diện với màu sắc và bố cục hài hòa tạo cho người sử dụng cảm thấy thoải mái khi sử dụng.

Tuy nhiên vẫn còn một số hạn chế, website còn nhỏ. Thiếu chức năng tìm kiếm nội dung, chức năng đăng ký, chức năng comment, xem số view và chức năng chat trực tuyến với người dùng. Nội dung cũng chưa đáp ứng đầy đủ tới người dùng.

Trong thời gian tới em sẽ hoàn thành các chức năng tìm kiếm nội dung, chức năng đăng ký và chức năng chat trực tuyến với người dùng.

Tạo cơ sở dữ liệu phong phú tối ưu hơn, xây dựng các trang chi tiết đa dạng hơn, nội dung thuận tiện tới người dùng.

## TÀI LIỆU THAM KHẢO

- [1.] Trần Đình Quế, Nguyễn Mạnh Sơn, Phân tích và thiết kế hệ thống thông tin, Học viện Công nghệ Bưu chính Viễn Thông, 2007.
- [2.] Nguyễn Văn Vị (2002), Phân tích và thiết kế hệ thống thông tin quản lý, NXB Thống kê.
- [3.] <https://laravel.com/docs/7.x>
- [4.] <https://khoapham.vn/lap-trinh-php.html>
- [5.] <https://github.com/dropways/deskapp>
- [6.] <https://fullstack.edu.vn/courses/html-css>
- [7.] <https://fullstack.edu.vn/courses/javascript-co-ban>
- [8.] <https://getbootstrap.com/>
- [9.] <https://www.w3schools.com/>