

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC CẦN THƠ**  
**KHOA CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC**  
**NGÀNH KỸ THUẬT PHẦN MỀM**

**Đề tài**

**XÂY DỰNG WEBSITE QUẢN LÝ CÔNG ĐOÀN**  
**KHOA CNTT & TT**

**Sinh viên: Lâm Nhật Hảo**

**Mã số: B1605389**

**Khóa: 42**

**Cần Thơ, 01/2021**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC CẦN THƠ**  
**KHOA CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC**  
**NGÀNH KỸ THUẬT PHẦN MỀM**

**Đề tài**

**XÂY DỰNG WEBSITE QUẢN LÝ CÔNG ĐOÀN**  
**KHOA CNTT & TT**

**Người hướng dẫn:**  
**Ths. Trương Thị Thanh Tuyền**

**Sinh viên thực hiện:**  
**Lâm Nhật Hảo**  
**Mã số: B1605389**  
**Khóa: K42**

**Cần Thơ, 01/2021**

## MỤC LỤC

MỤC LỤC .....	3
DANH SÁCH HÌNH VẼ .....	6
DANH SÁCH BẢNG BIỂU .....	8
DANH SÁCH TỪ VIẾT TẮT .....	9
CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI .....	10
1.1    Lý do chọn đề tài .....	10
1.2    Mục tiêu của đề tài .....	10
1.3    Giới hạn và phạm vi của đề tài.....	10
1.4    Nội dung thực hiện .....	11
1.5    Phương pháp tiếp cận .....	11
1.6    Kết luận chương 1 .....	12
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	13
2.1. Tổng quan về ngôn ngữ.....	13
2.1.1. ASP.NET MVC là gì? .....	13
2.1.2. Mô hình lập trình MVC trong ASP.NET.....	13
2.1.3    Ưu điểm của ASP.NET .....	16
2.1.4    Ưu điểm của ASP.NET MVC .....	17
2.1.5    Các tính năng của nền tảng ASP.NET MVC .....	17
2.2    Giới thiệu về Entity Framework.....	19
2.2.1    Entity Framework là gì ? .....	19
2.2.2    Vị trí của Entity Framework trong mô hình Web ? .....	19
2.3    Hệ quản trị Cơ sở dữ liệu MySql .....	22

2.3.1	SQL là gì ? .....	22
2.3.1	Tại sao sử dụng SQL và nó lại quan trọng .....	23
2.3.2	Các câu lệnh trong SQL .....	24
2.4	Kết luận chương 2 .....	25
CHƯƠNG 3: NỘI DUNG THỰC HIỆN .....		26
3.1.	Khảo sát phân tích yêu cầu .....	26
3.1.1.	Thời gian, địa điểm khảo sát .....	26
3.1.2.	Thông tin liên hệ .....	26
3.1.3	Thông tin dự án .....	26
3.1.4	Yêu cầu của hệ thống sau khảo sát .....	26
3.1.5	Mô tả yêu cầu của hệ thống .....	27
3.2.	Phân tích thiết kế với UML. ....	29
3.2.1.	Biểu đồ Use-case .....	31
3.2.2	Biểu đồ lớp .....	41
3.2.3	Biểu đồ tuần tự .....	42
3.3	Thiết kế cơ sở dữ liệu .....	52
3.3.1	Các bảng dữ liệu .....	52
3.4	Thiết kế đặc tả giao diện .....	59
3.4.1	Chức năng hiển thị trang chủ .....	59
3.4.2	Chức năng đăng nhập .....	61
3.4.3	Chức năng ghi chú sự kiện .....	62
3.4.4	Chức năng hiển thị danh sách giáo viên theo khoa .....	63
3.4.5	Chức năng hiển thị danh sách sinh viên theo từng lớp .....	64
3.4.6	Chức năng tạo báo cáo tháng cho giáo viên .....	65

3.4.7	Chức năng thống kê báo cáo tháng của giáo viên chủ nhiệm .....	66
3.4.8	Chức năng quản lý lịch công tác giáo viên .....	67
3.4.9	Chức năng quản lý sinh viên .....	68
3.4.10	Chức năng quản lý giáo viên .....	69
3.4.11	Chức năng quản lý lớp học .....	70
3.4.12	Chức năng quản lý khoa .....	71
3.4.13	Chức năng quản lý menu .....	72
3.4.14	Chức năng quản lý công việc.....	73
3.5	Kết luận chương 3. ....	74
CHƯƠNG 4: KẾT LUẬN.....		76
4.1.	Kết quả đạt được của đề tài.....	76
4.2.	Hạn chế của đề tài .....	76
4.3.	Hướng phát triển của đề tài.....	76
TÀI LIỆU THAM KHẢO.....		77

## **DANH SÁCH HÌNH VẼ**

Hình 3. 1: Biểu đồ usecase tổng quát .....	32
Hình 3. 2: Biểu đồ use-case đăng nhập.....	32
Hình 3. 3: Biểu đồ use-case quản lý lịch công tác.....	33
Hình 3. 4: Biểu đồ Use-case quản lý loại lịch công tác .....	34
Hình 3. 5: Biểu đồ user-case quản lý lớp học .....	36
Hình 3. 6: Biểu đồ user-case quản lý giáo viên.....	37
Hình 3. 7: Biểu đồ user-case quản lý công việc .....	38
Hình 3. 8: Biểu đồ user-case quản lý menu .....	39
Hình 3. 9: Biểu đồ use-case quản lý khoa.....	40
Hình 3. 10: Biểu đồ lớp tổng quát .....	41
Hình 3. 11: Biểu đồ tuần tự cho chức năng quản lý lịch công tác .....	44
Hình 3. 12: Biểu đồ tuần tự cho chức năng quản lý lớp học.....	45
Hình 3. 13: Biểu đồ tuần tự cho chức năng thêm mới menu .....	46
Hình 3. 14: Biểu đồ tuần tự cho chức năng quản lý khoa.....	47
Hình 3. 15: Biểu đồ tuần tự cho chức năng quản lý sinh viên .....	48
Hình 3. 16: Biểu đồ tuần tự cho chức năng quản lý giáo viên.....	49
Hình 3. 17: Biểu đồ tuần tự cho chức năng quản lý loại lịch công tác.....	50
Hình 3. 18: Biểu đồ tuần tự cho chức năng quản lý người dùng .....	51
Hình 3. 19: Biểu đồ tuần tự cho chức năng quản lý công việc .....	52
Hình 3. 20: Giao diện chức năng hiển thị trang chủ.....	59
Hình 3. 21: Giao diện chức năng xem chi tiết lịch công tác giáo viên.....	60
Hình 3. 22: Giao diện chi tiết lịch công tác cấp 2a.....	60

Hình 3. 23: Giao diện chi tiết lịch công tác cấp 2b .....	61
Hình 3. 24: Giao diện chi tiết lịch công tác cấp 2c.....	61
Hình 3. 25: Giao diện chức năng đăng nhập .....	62
Hình 3. 26: Giao diện chức năng ghi chú công việc.....	63
Hình 3. 27: Giao diện chức năng hiển thị danh sách giáo viên theo khoa.....	64
Hình 3. 28: Giao diện trang danh sách sinh viên theo lớp.....	65
Hình 3. 29: Giao diện chức năng tạo báo cáo tháng cho giáo viên .....	66
Hình 3. 30: Giao diện trang thống kê báo cáo tháng của giáo viên.....	67
Hình 3. 31: Giao diện trang quản lý lịch công tác giáo viên.....	68
Hình 3. 32: Giao diện trang quản lý sinh viên .....	69
Hình 3. 33: Giao diện trang quản lý giáo viên .....	70
Hình 3. 34: Giao diện trang quản lý lớp học.....	71
Hình 3. 35: Giao diện trang quản lý khoa.....	72
Hình 3. 36: Giao diện trang quản lý menu.....	73
Hình 3. 37: Giao diện trang quản lý công việc.....	74

## **DANH SÁCH BẢNG BIỂU**

Bảng 3. 1: Các yêu cầu khách hàng.....	26
Bảng 3. 2: Các yêu cầu chức năng.....	27
Bảng 3. 3: Các yêu cầu phi chức năng.....	29
Bảng 3. 4: Các tác nhân.....	31
Bảng 3. 5: Các quyền sử dụng.....	31
Bảng 3. 6: Danh sách các lớp .....	41
Bảng 3. 7: Bảng Science .....	52
Bảng 3. 8: Bảng CalendarWorking.....	53
Bảng 3. 9: Bảng Class .....	54
Bảng 3. 10: Bảng credential .....	54
Bảng 3. 11: Bảng Event .....	55
Bảng 3. 12: Bảng GroupUser .....	55
Bảng 3. 13: Bảng Menu .....	56
Bảng 3. 14: Bảng Role .....	56
Bảng 3. 15: Bảng Student.....	57
Bảng 3. 16: Bảng TypeCalendar.....	57
Bảng 3. 17: Bảng User .....	58
Bảng 3. 18: Bảng Work.....	58



## DANH SÁCH TỪ VIẾT TẮT

Từ viết tắt	Từ đầy đủ	Giải thích
CSDL	Cơ sở dữ liệu	Cơ sở dữ liệu
CMS	Content management system	Hệ thống quản trị nội dung
HTML	Hypertext Markup Language	Ngôn ngữ siêu đánh dấu văn bản
MVC	Model – View – Controller	Mô hình Model-View-Controller
OOP	Object oriented program	Lập trình hướng đối tượng
ORM	Object-Relational Mapping	Bản đồ quan hệ giữa các đối tượng
SQL	Structured Query Language	Ngôn ngữ truy vấn cấu trúc
UML	Unified Modeling Language	Ngôn ngữ mô hình hóa thống nhất

## **CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI**

### **1.1 Lý do chọn đề tài**

Cùng với sự phát triển không ngừng về kỹ thuật máy tính và mạng điện tử, công nghệ thông tin cũng được những công nghệ có đẳng cấp cao và lần lượt chinh phục hết đỉnh cao này đến đỉnh cao khác. Mạng Internet là một trong những sản phẩm có giá trị hết sức lớn lao và ngày càng trở thành một công cụ không thể thiếu, là nền tảng chính cho sự truyền tải, trao đổi thông tin trên toàn cầu.

Trong hệ thống công đoàn, việc quản lý những thông tin của công đoàn viên trong các trường đại học đa phần vẫn còn thực hiện thủ công trên giấy và khó có thể nhất quán về việc quản lý gây nên hiện tượng thất thoát dữ liệu hoặc dữ liệu bị sai lệch.

Chính vì nguyên do đó, việc “xây dựng website quản lý công đoàn khoa CNTT & TT” là thực sự cần thiết trong xã hội 4.0 hiện nay. Nhằm nâng cao khả năng quản lý dữ liệu, sự chuyên nghiệp trong môi trường giáo dục nên hệ thống quản lý công đoàn là cần thiết phải có

### **1.2 Mục tiêu của đề tài**

- Phân tích được bài toán quản lý hệ thống quản lý công đoàn khoa CNTT & TT.
- Tìm hiểu cơ sở lý thuyết của quản lý cơ sở dữ liệu Microsoft SQL Server 2019 và ngôn ngữ C# ASP.NET MVC.
- Phân tích và thiết kế CSDL, các biểu đồ use case và biểu đồ tuần tự.
- Xây dựng Website quản lý công đoàn khoa CNTT & TT.

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

Đề tài bước đầu được xây dựng để áp dụng cho việc quản lý hệ thống công đoàn khoa CNTT & TT và sau đó có thể phát triển thêm và ứng dụng cho trường học trong thực tế.

Đề tài xây dựng sử dụng công nghệ dựa trên ngôn ngữ C# và phát triển trên nền tảng MVC Framework, Entity Framework và công cụ lập trình Visual Studio 2019 và SQL Server 2019.

#### **1.4 Nội dung thực hiện**

Nội dung thực hiện/ nghiên cứu cụ thể của đề tài:

- Hiểu được cách sử dụng cơ bản của ngôn ngữ C# ASP.NET, trong đó có các đối tượng, biến, hàm, phương thức, thuộc tính...
- Tìm hiểu cách sử dụng MVC Framework, Entity Framework trong thiết kế và lập trình website bằng ngôn ngữ C# ASP.NET.
- Tìm hiểu cách sử dụng phần mềm Visual Studio 2019 và SQL Server để thiết kế giao diện và xây dựng các chức năng website quản lý bằng ngôn ngữ C#.
- Biết cách sử dụng hệ quản trị CSDL bằng Microsoft SQL Server 2019.
- Khảo sát và phân tích yêu cầu quy trình quản lý.
- Thiết kế và đặc tả hệ thống.
- Xây dựng cơ sở dữ liệu.
- Lập trình các module của hệ thống.
- Kiểm thử hệ thống.
- Triển khai thực nghiệm hệ thống trên mạng Internet.

#### **1.5 Phương pháp tiếp cận**

- Nghiên cứu quy trình quản lý công đoàn khoa CNTT & TT và mạng internet, nghiên cứu ngôn ngữ C# ASP.NET, MVC Framework, Entity Framework và mô hình Model - View – Controller.
- Sử dụng các phương pháp nghiên cứu:
  - Phương pháp đọc tài liệu.
  - Phương pháp phân tích mẫu, thực nghiệm

## **1.6 Kết luận chương 1**

Trong chương 1 đã nêu nên lý do chọn đề tài xây dựng “*Website quản lý công đoàn khoa CNTT & TT*”. Cũng như mục tiêu, phạm vi, nội dung và phương pháp thực hiện. Chương tiếp theo là tìm hiểu và giới thiệu về công nghệ C# ASP.NET, MVC Framework, Entity Framework và Microsoft SQL Server 2019.

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

### 2.1. Tổng quan về ngôn ngữ

#### 2.1.1. ASP.NET MVC là gì?

Đầu năm 2002, Microsoft giới thiệu một kỹ thuật lập trình Web khá mới mẻ với tên gọi ban đầu là ASP+, tên chính thức sau này là ASP.Net. Với ASP.Net, không những không cần đòi hỏi bạn phải biết các tag HTML, thiết kế web, mà nó còn chăm sóc mạnh lập trình hướng đối tượng trong quá trình xây dựng và phát triển ứng dụng Web. ASP.Net là kỹ thuật lập trình và phát triển ứng dụng web ở phía Server (Server-side) dựa trên nền tảng của Microsoft .Net Framework.

Hầu hết, những người mới đến với lập trình web đều bắt đầu tìm hiểu những kỹ thuật ở phía Client (Client-side) như: HTML, Java Script, CSS (Cascading Style Sheets). Khi Web browser yêu cầu một trang web (trang web sử dụng kỹ thuật client-side), Web server tìm trang web mà Client yêu cầu, sau đó gửi về cho Client. Client nhận kết quả trả về từ Server và hiển thị lên màn hình.

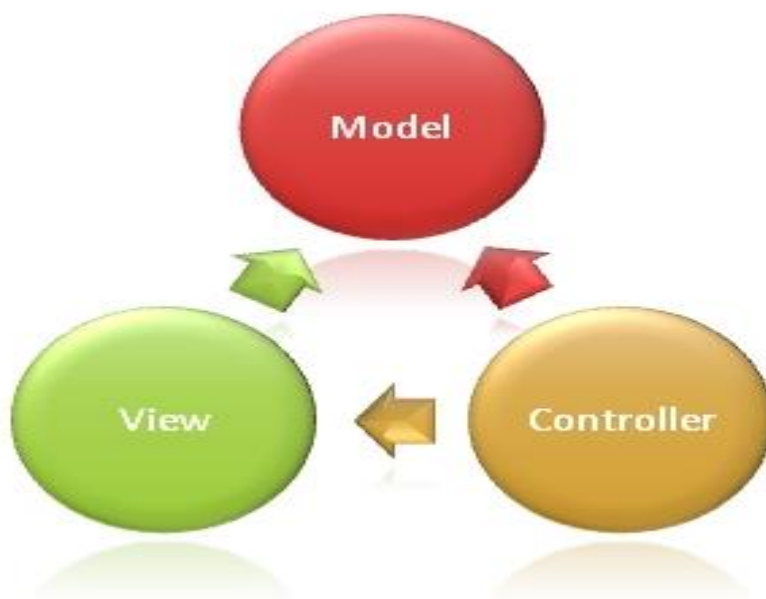
ASP.Net sử dụng kỹ thuật lập trình ở phía server thì hoàn toàn khác, mã lệnh ở phía server (ví dụ: mã lệnh trong trang ASP) sẽ được biên dịch và thi hành tại Web Server. Sau khi được Server đọc, biên dịch và thi hành, kết quả tự động được chuyển sang HTML/JavaScript/CSS và trả về cho Client. Tất cả các xử lý lệnh ASP.Net đều được thực hiện tại Server và do đó, gọi là kỹ thuật lập trình ở phía server.

#### 2.1.2. Mô hình lập trình MVC trong ASP.NET

Mẫu kiến trúc Model – View – Controller được sử dụng nhằm chi ứng dụng thành ba thành phần chính: model, view và controller. Nền tảng ASP.NET MVC giúp cho chúng ta có thể tạo được các ứng dụng web áp dụng mô hình MVC thay vì tạo ứng dụng theo mẫu ASP.NET Web Form. Nền tảng ASP.NET MVC có đặc điểm nổi bật là nhẹ (lighweight), dễ kiểm thử phân giao diện (so với ứng dụng Web Forms), tích hợp các tính năng có sẵn của ASP.NET. Nền tảng ASP.NET MVC được định nghĩa trong namespace System.Web.Mvc và là một phần của name space System.Web.

MVC là một mẫu thiết kế (design pattern) chuẩn mà nhiều lập trình viên đã quen thuộc. Một số loại ứng dụng web sẽ thích hợp với kiến trúc MVC. Một số khác vẫn thích hợp với ASP.NET Web Forms và cơ chế postbacks. Đôi khi có những ứng dụng kết hợp cả hai kiến trúc trên.

Nền tảng MVC bao gồm các thành phần dưới đây:



*Hình 2. 1: Mô hình quan hệ MVC*

**Models:** Các đối tượng Models là một phần của ứng dụng, các đối tượng này thiết lập logic của phần dữ liệu của ứng dụng. Thông thường, các đối tượng model lấy và lưu trạng thái của model trong CSDL. Ví dụ như, một đối tượng Product (sản phẩm) sẽ lấy dữ liệu từ CSDL, thao tác trên dữ liệu và sẽ cập nhật dữ liệu trở lại vào bảng Products ở SQL Server.

Trong các ứng dụng nhỏ, model thường là chỉ là một khái niệm nhằm phân biệt hơn là được cài đặt thực thụ, ví dụ, nếu ứng dụng chỉ đọc dữ liệu từ CSDL và gửi chúng đến view, ứng dụng không cần phải có tầng model và các lớp liên quan. Trong trường hợp này, dữ liệu được lấy như là một đối tượng model (hơn là tầng model

**Views:** Views là các thành phần dùng để hiển thị giao diện người dùng (UI). Thông thường, view được tạo dựa vào thông tin dữ liệu model. Ví dụ như, view dùng

để cập nhật bảng Products sẽ hiển thị các hộp văn bản, drop-down list, và các check box dựa trên trạng thái hiện tại của một đối tượng Product.

**Controllers:** Controller là các thành phần dùng để quản lý tương tác người dùng, làm việc với model và chọn view để hiển thị giao diện người dùng. Trong một ứng dụng MVC, view chỉ được dùng để hiển thị thông tin, controller chịu trách nhiệm quản lý và đáp trả nội dung người dùng nhập và tương tác với người dùng. Ví dụ, controller sẽ quản lý các dữ liệu người dùng gửi lên (query-string values) và gửi các giá trị đó đến model, model sẽ lấy dữ liệu từ CSDL nhờ vào các giá trị này.

Mẫu MVC giúp bạn tạo được các ứng dụng mà chúng phân tách rạch ròi các khía cạnh của ứng dụng (logic về nhập liệu, logic xử lý tác vụ và logic về giao diện). Mẫu MVC chỉ ra mỗi loại logic kể trên nên được thiết lập ở đâu trên ứng dụng. Logic giao diện (UI logic) thuộc về views. Logic nhập liệu (input logic) thuộc về controller. Và logic tác vụ (Business logic – là logic xử lý thông tin, mục đích chính của ứng dụng) thuộc về model. Sự phân chia này giúp bạn giảm bớt được sự phức tạp của ứng dụng và chỉ tập trung vào mỗi khía cạnh cần được cài đặt ở mỗi thời điểm. Ví dụ như bạn chỉ cần tập trung vào giao diện (views) mà không phải quan tâm đến logic xử lý thông tin của ứng dụng.

Để quản lý sự phức tạp của ứng dụng, mẫu MVC giúp cho chúng ta có thể kiểm thử ứng dụng dễ dàng hơn hẳn so với khi áp dụng mẫu Web Forms. Ví dụ, trong một ứng dụng ASP.NET Web Forms, một lớp thường được sử dụng để hiển thị thông tin xuất ra cho người dùng và đồng thời xử lý thông tin người dùng nhập. Việc xây dựng các bộ test tự động cho ứng dụng Web Forms là rất phức tạp, bởi để kiểm thử mỗi trang web, bạn phải khởi tạo đối tượng trang, khởi tạo tất cả các control được sử dụng trong trang và các lớp phụ thuộc trong ứng dụng. Và bởi vì có quá nhiều lớp cần được khởi tạo để chạy được trang, thật khó để có thể viết các test chỉ tập trung vào một khía cạnh nào đó của ứng dụng. Và vì thế, kiểm thử đối với các ứng dụng dựa trên nền tảng Web Forms sẽ khó khăn hơn nhiều so với khi áp dụng trên ứng dụng MVC. Hơn thế nữa, việc kiểm thử trên nền tảng Web Forms yêu cầu phải sử dụng đến web server.

Nền tảng MVC phân tách các thành phần và sử dụng các interface (khái niệm giao diện trong lập trình hướng đối tượng), và nhờ đó có thể kiểm thử các thành phần riêng biệt trong tình trạng phân lập với các yếu tố còn lại của ứng dụng. Sự phân tách rạch ròi ba thành phần của ứng dụng MVC còn giúp cho việc lập trình diễn ra song song. Ví dụ như một lập trình viên làm việc với view, lập trình viên thứ hai lo cài đặt logic của controller và lập trình viên thứ ba có thể tập trung vào logic tác vụ của model tại cùng một thời điểm.

Bạn cần phải xem xét kỹ càng việc áp dụng mô hình ASP.NET MVC hay mô hình ASP.NET Web Forms khi xây dựng một ứng dụng. Mô hình MVC không phải là mô hình thay thế cho Web Forms, bạn có thể dùng một trong hai mô hình.

Trước khi quyết định sử dụng MVC hay Web Forms cho một web site cụ thể, bạn cần phải phân tích lợi ích khi chọn một trong hai hướng.

Một ứng dụng web được phát triển theo mô hình MVC, có nguyên lý hoạt động xử lý yêu cầu của người dùng như sau:

- Khi có yêu cầu phát sinh từ người dùng (phía Client), yêu cầu này sẽ được gửi đến phía Server, tại đây Controller sẽ tiếp nhận yêu cầu để xử lý.
- Bên trong Controller chứa nhiều phương thức (action), mỗi action tương ứng với một yêu cầu cụ thể từ phía người dùng. Controller sẽ lựa chọn action phù hợp với yêu cầu để xử lý. Trong quá trình xử lý, Controller sẽ tương tác với Model để có được dữ liệu mà người dùng mong muốn.
- Sau khi Controller làm việc với Model để có được dữ liệu theo yêu cầu, Controller sẽ gửi về cho View thành phần Data Model và nhiệm vụ của View là chuyển Data Model nhận được thành dữ liệu ứng dụng và gửi trả về phía Client để hiển thị kết quả yêu cầu.

### **2.1.3 Ưu điểm của ASP.NET**

ASP.NET được xây dựng với bộ thư viện đa dạng và phong phú của .Net Framework, chăm sóc vượt trội về XML và khả năng truy cập cơ sở dữ liệu qua ADO.net. Bên cạnh đó, ASP và ASPX có thể hoạt động trong cùng một ứng dụng



mang lại hiệu năng cao cho website. Có thể nhận thấy ASP.NET mang một phong cách hoàn toàn mới, bạn có thể tách giao diện và code thành 2 phần khác biệt. Điều này giúp cho việc quản lý và bảo trì trở nên rõ ràng và đơn giản hơn.

Thiết kế web bằng ASP.NET giúp cho hệ điều hành hoạt động trơn tru cũng như tăng tốc độ tải cho trang web của bạn. Với kiến trúc tương tự như các ứng dụng trên Windows, việc sử dụng cũng trở nên dễ dàng hơn rất nhiều. Bên cạnh đó ASP.NET có khả năng tự động tạo ra mã HTML cho các Server để website có thể hoạt động trên hầu hết các trình duyệt được khách hàng sử dụng. Đặc biệt, ASP.NET là một ngôn ngữ mất phí nên bạn có thể hoàn toàn yên tâm về vấn đề bảo mật của nó. Với những ưu điểm vượt trội như vậy, ASP.NET luôn được rất nhiều nhà thiết kế web chuyên nghiệp ưa chuộng và sử dụng. Tuy nhiên, nó vẫn còn một số nhược điểm cần khắc phục mà bạn cần chú ý khi xây dựng web doanh nghiệp.

#### **2.1.4 Ưu điểm của ASP.NET MVC**

- Dễ dàng quản lý sự phức tạp của ứng dụng bằng cách chia ứng dụng thành ba thành phần model, view, controller.
- Nó không sử dụng view state hoặc server-based form. Điều này tốt cho những lập trình viên muốn quản lý hết các khía cạnh của một ứng dụng.
- Nó sử dụng mẫu Front Controller, mẫu này giúp quản lý các requests (yêu cầu) chỉ thông qua một Controller. Nhờ đó bạn có thể thiết kế một hạ tầng quản lý định tuyến. Để có nhiều thông tin hơn, bạn nên xem phần Front Controller trên web site MSDN.
- Hỗ trợ tốt hơn cho mô hình phát triển ứng dụng hướng kiểm thử (TDD).
- Nó hỗ trợ tốt cho các ứng dụng được xây dựng bởi những đội có nhiều lập trình viên và thiết kế mà vẫn quản lý được tính năng của ứng dụng.

#### **2.1.5 Các tính năng của nền tảng ASP.NET MVC**

– Tách bạch các tác vụ của ứng dụng (logic nhập liệu, business logic, và logic giao diện), dễ dàng kiểm thử và mặc định áp dụng hướng phát triển TDD. Tất cả các tính năng chính của mô hình MVC được cài đặt dựa trên interface và được kiểm thử

bằng cách sử dụng các đối tượng mocks, mock object là các đối tượng mô phỏng các tính năng của những đối tượng thực sự trong ứng dụng. Bạn có thể kiểm thử unit-test cho ứng dụng mà không cần chạy controller trong tiến trình ASP.NET, và điều đó giúp unit test được áp dụng nhanh chóng và tiện dụng. Bạn có thể sử dụng bất kỳ nền tảng unit-testing nào tương thích với nền tảng .NET.

- MVC là một nền tảng khả mở rộng (extensible) & khả nhúng (pluggable). Các thành phần của ASP.NET MVC được thiết kế để chúng có thể được thay thế một cách dễ dàng hoặc dễ dàng tùy chỉnh. Bạn có thể nhúng thêm view engine, cơ chế định tuyến cho URL, cách kết xuất tham số của action-method và các thành phần khác. ASP.NET MVC cũng hỗ trợ việc sử dụng Dependency Injection (DI) và Inversion of Control (IoC). DI cho phép bạn gắn các đối tượng vào một lớp cho lớp đó sử dụng thay vì buộc lớp đó phải tự mình khởi tạo các đối tượng. IoC quy định rằng, nếu một đối tượng yêu cầu một đối tượng khác, đối tượng đầu sẽ lấy đối tượng thứ hai từ một nguồn bên ngoài, ví dụ như từ tập tin cấu hình. Và nhờ vậy, việc sử dụng DI và IoC sẽ giúp kiểm thử dễ dàng hơn.

- ASP.NET MVC có thành phần ánh xạ URL mạnh mẽ cho phép bạn xây dựng những ứng dụng có các địa chỉ URL xúc tích và dễ tìm kiếm. Các địa chỉ URL không cần phải có phần mở rộng của tên tập tin và được thiết kế để hỗ trợ các mẫu định dạng tên phù hợp với việc tối ưu hóa tìm kiếm (URL) và phù hợp với lập địa chỉ theo kiểu REST.

- Hỗ trợ sử dụng đặc tả (các thẻ) của các trang ASP.NET(.aspx), điều khiển người dùng (.ascx) và trang master page (.master). Bạn có thể sử dụng các tính năng có sẵn của ASP.NET như là sử dụng lồng các trang master page, sử dụng in-line expression (<%= %>), sử dụng server controls, mẫu, data-binding, địa phương hóa (localization) và hơn thế nữa.

- Hỗ trợ các tính năng có sẵn của ASP.NET như cơ chế xác thực người dùng, quản lý thành viên, quyền, output caching và data caching, session và profile, quản lý tình trạng ứng dụng, hệ thống cấu hình...

ASP.NET MVC 5 còn bổ sung một view engine mới là Razor View Engine cho phép thiết lập các view nhanh chóng, dễ dàng và tốn ít công sức hơn so với việc sử dụng Web Forms view engine

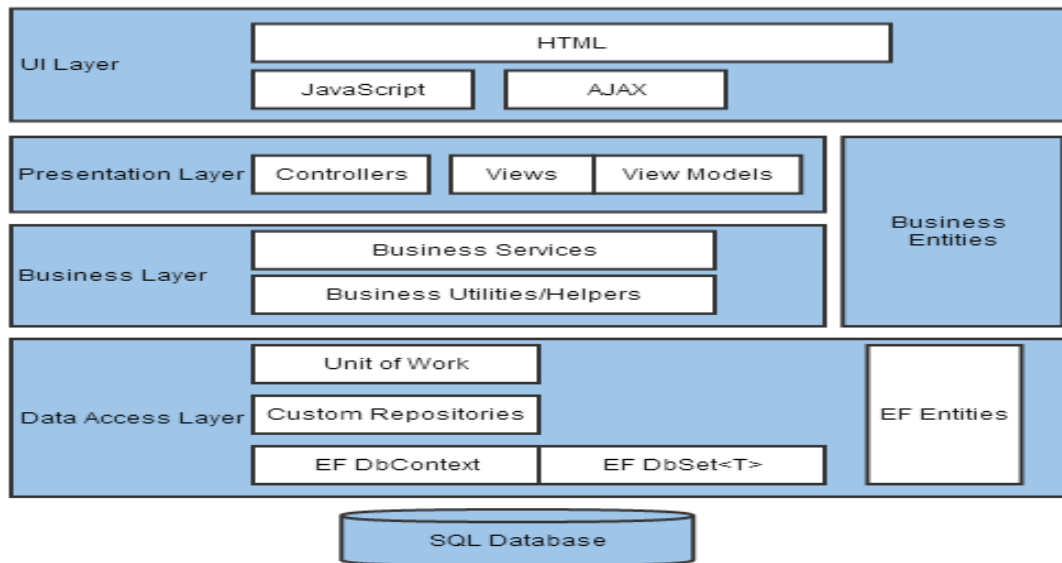
## **2.2 Giới thiệu về Entity Framework**

### **2.2.1 Entity Framework là gì ?**

**Entity Framework (EF)** là một framework ánh xạ quan hệ đối tượng (ORM) dành cho ADO.NET, là 1 phần của .NET Framework. EF cho phép các nhà phát triển Web tương tác với dữ liệu quan hệ theo phương pháp hướng đối tượng đặc trưng. Lợi ích lớn nhất của EF là giúp lập trình viên giảm thiểu việc lập trình mã nguồn cần thiết để truy cập và tương tác với cơ sở dữ liệu. EF được Microsoft hỗ trợ phát triển lâu dài và bền vững, vì vậy EF là 1 framework mạnh nhất hiện nay để phát triển ứng dụng Web với sự hỗ trợ đông đảo của các nhà phát triển Web.

### **2.2.2 Vị trí của Entity Framework trong mô hình Web ?**

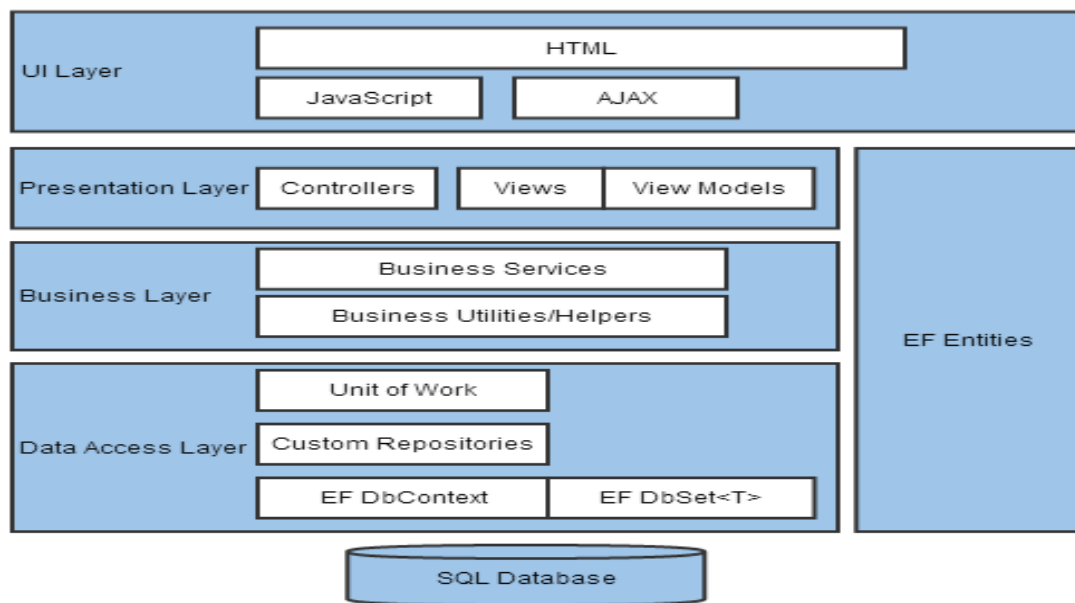
Chắc hẳn, bạn đã biết về mô hình Web 3 tầng (n tầng) khi đọc sách, giáo trình, bài giảng về nội dung lập trình Web. Tuy nhiên, vấn đề bạn cần phải hiểu rõ là: việc định rõ số lượng các tầng và các mối liên hệ giữa các tầng trong mô hình phát triển Web cũng rất đa dạng, tùy theo cách hiểu các lập trình viên và dự án Web. Do đó, việc hiểu Entity Framework nằm ở đâu trong mô hình Web 3 tầng cũng không hoàn toàn dễ dàng. Trong phần này, tôi có tham khảo từ Stephen M. Redd nhằm giúp các bạn hiểu rõ bản chất vấn đề hơn. Đầu tiên, chúng ta có vị trí của EF trong mô hình dự án Web ASP.NET MVC.



*Hình 2. 2: Mô hình dự án ASP.NET MVC*

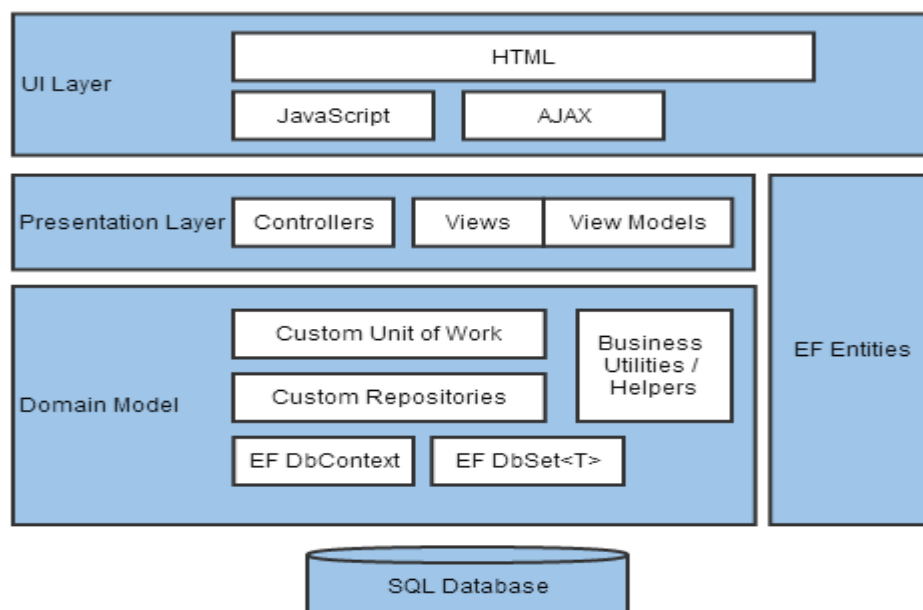
Trong mô hình trên, chúng ta thấy vị trí của EF nằm trọn trong tầng Data Access Layer, tức là EF đóng vai trò là nơi trung gian để tương tác dữ liệu từ database với các tầng cao hơn, chẳng hạn như Business Layer. Hay nói cách khác, EF đóng vai trò như là phiên bản mới của ADO.NET. Tuy nhiên vị trí của EF trong mô hình này chỉ có trên lý thuyết.

Thật sự, khi bắt tay xây dựng nhiều dự án Web, vị trí thường thấy EF ở các dự án website trên thực tế như mô hình sau.



*Hình 2. 3: Mô hình dự án ASP.NET MVC*

Trong hình trên, EF dường như có vị trí rất tự do, xuất hiện ở cả 3 tầng Data Access Layer, Business Layer, và cả Presentation Layer. Các bạn có thể hình dung EF như là 1 thư viện, hễ tầng nào cần thì chỉ cần gọi đến. Mô hình này xuất phát có thể là sự cầu thả của lập trình viên khi không thích tuân theo tiêu chuẩn quan hệ giữa các tầng hoặc cách giải thích khác là sự lỏng lẻo giúp cho việc lập trình thuận tiện hơn. Trong vài mô hình người ta gom tầng Data Access Layer và Business Layer chỉ làm tầng Data Model, và vị trí EF cũng bao hàm tầng Data Model và Presentation Layer.



Hình 2. 4: Mô hình dự án ASP.NET

Đến đây, bạn có thể hiểu 1 cách chung nhất EF có vị trí trung gian, đóng vai trò kết nối giữa cơ sở dữ liệu và các thành phần khác của 1 dự án Web khi cần đến. Ngoài ra, có nhiều cách hiểu về vị trí của EF ở đâu trong mô hình Web, bạn sẽ dần khám phá để đưa ra cách hiểu và định nghĩa riêng trong quá trình thiết kế và xây dựng dự án Web ASP.NET.

## 2.3 Hệ quản trị Cơ sở dữ liệu MySql

### 2.3.1 SQL là gì ?

**SQL** là viết tắt của **Structured Query Language** là ngôn ngữ truy vấn có cấu trúc. Nó là một ngôn ngữ, là tập hợp các lệnh để tương tác với cơ sở dữ liệu. Dùng để lưu trữ, thao tác và truy xuất dữ liệu được lưu trữ trong một cơ sở dữ liệu quan hệ. Trong thực tế, SQL là ngôn ngữ chuẩn được sử dụng hầu hết cho hệ cơ sở dữ liệu quan hệ. Tất cả các hệ thống quản lý cơ sở dữ liệu quan hệ (RDMS) như MySQL, MS Access, Oracle, Postgres và SQL Server... đều sử dụng SQL làm ngôn ngữ cơ sở dữ liệu chuẩn.

Hầu như công ty nào lớn cũng cần xây dựng một hệ thống để lưu trữ cơ sở dữ liệu. Mọi thứ trong cơ sở dữ liệu này sẽ được diễn tả ra thành nhiều bảng, có mối

quan hệ với nhau. Để truy vấn và lấy dữ liệu từ các bảng này nhằm tổng hợp thành thông tin nào đó, người ta dùng đến SQL thông qua các câu query.

SQL bắt đầu từ những năm 1970, khi các kỹ sư của **IBM** là **Donald Chamberlin** và **Raymond Boyce** thiết kế phiên bản đầu tiên để tương tác và lấy dữ liệu được lưu trữ trong hệ thống cơ sở dữ liệu của công ty. Họ gọi nó là **SEQUEL**, mặc dù sau đó họ buộc phải thay đổi nó do các vấn đề bản quyền.

### 2.3.1 Tại sao sử dụng SQL và nó lại quan trọng

SQL được sử dụng phổ biến vì nó có các ưu điểm sau:

- Cho phép truy cập dữ liệu trong các hệ thống quản lý cơ sở dữ liệu quan hệ.
- Cho phép mô tả dữ liệu.
- Cho phép xác định dữ liệu trong cơ sở dữ liệu và thao tác dữ liệu đó.
- Cho phép nhúng trong các ngôn ngữ khác sử dụng mô-đun SQL, thư viện và trình biên dịch trước.
- Cho phép tạo và thả các cơ sở dữ liệu và bảng.
- Cho phép tạo chế độ view, thủ tục lưu trữ, chức năng trong cơ sở dữ liệu.
- Cho phép thiết lập quyền trên các bảng, thủ tục và view.

Thường thì doanh nghiệp luôn cần một hệ thống để quản trị thông tin, họ phải thiết kế ra một cơ sở dữ liệu để lưu trữ thông tin. Nếu họ lưu trữ ở dạng giấy hoặc excel thì sẽ chứa nhiều rủi ro bảo mật thông tin như bị mất, sửa, xóa...

**SQL** sẽ giúp quản lý hiệu quả và truy vấn thông tin nhanh hơn, giúp bảo trì, bảo mật thông tin dễ dàng hơn.

Ví dụ: trước đây, trường đại học thường lưu trữ thông tin sinh viên bằng hồ sơ giấy. Sau đó, cất giữ hồ sơ trong kho. Khi cần tìm kiếm hoặc thêm/xóa/sửa thông tin nào đó, họ phải mất rất nhiều thời gian để lục tìm lại hồ sơ.

Trong khi, nếu lưu trữ thông tin vào một hệ thống cơ sở dữ liệu, họ chỉ cần gõ một câu lệnh SQL ngắn là đã có thể trích xuất được thông tin cần. Việc thêm/xóa/sửa cũng được thực hiện một cách dễ dàng, nhanh chóng.

### 2.3.2 Các câu lệnh trong SQL

Các lệnh SQL tiêu chuẩn để tương tác với cơ sở dữ liệu quan hệ là CREATE, SELECT, INSERT, UPDATE, DELETE và DROP. Các lệnh này có thể được phân thành các nhóm sau dựa trên bản chất của chúng.

DDL – Ngôn ngữ định nghĩa dữ liệu(Data Definition Language)

Lệnh	Mô Tả
<b>CREATE</b>	Tạo ra một bảng mới hoặc các đối tượng khác trong cơ sở dữ liệu.
<b>ALTER</b>	Sửa đổi một đối tượng cơ sở dữ liệu hiện có, chẳng hạn như một bảng.
<b>DROP</b>	Xoá toàn bộ một bảng hoặc các đối tượng khác trong cơ sở dữ liệu.

*Bảng 2. 1: Ngôn ngữ định nghĩa dữ liệu*

DML – Ngôn ngữ thao tác dữ liệu (Data Manipulation Language)

Lệnh	Mô Tả
<b>SELECT</b>	Lấy ra các bảng ghi nhất định từ một hoặc nhiều bảng.
<b>INSERT</b>	Tạo một bảng ghi.
<b>UPDATE</b>	Chỉnh sửa bảng ghi.
<b>DELETE</b>	Xoá bảng ghi.

*Bảng 2. 2: Ngôn ngữ thao tác dữ liệu*



## **2.4 Kết luận chương 2**

Tóm lại, trong chương 2 em đã tìm hiểu được cơ sở lý thuyết về ngôn ngữ C# ASP.NET và tầm quan trọng của MVC Framework, Entity Framework. Chương tiếp theo sẽ tiến hành xây dựng “*Website quản lý công đoàn khoa CNTT & TT*”.

## CHƯƠNG 3: NỘI DUNG THỰC HIỆN

### 3.1. Khảo sát phân tích yêu cầu

#### 3.1.1. Thời gian, địa điểm khảo sát

- Tham khảo thực tế hệ thống quản lý công đoàn hiện tại của một số trường thông qua Internet.
- Khảo sát một số website quản lý về nhân sự, công văn.

#### 3.1.2. Thông tin liên hệ

- Bộ môn CNPM – Khoa Công Nghệ Thông Tin – Trường Đại Học Cần Thơ.

#### 3.1.3 Thông tin dự án

- Tên dự án: “XÂY DỰNG WEBSITE QUẢN LÝ CÔNG ĐOÀN KHOA CNTT & TT”
- Người thực hiện: Lâm Nhật Hảo.
- Người hướng dẫn: Thạc sĩ Trương Thị Thanh Tuyền.
- Ngày thực hiện: 20/09/2020.
- Dự kiến ngày hoàn thành: 12/01/2020.

#### 3.1.4 Yêu cầu của hệ thống sau khảo sát

*Bảng 3. 1: Các yêu cầu khách hàng*

STT	Tên yêu cầu
1	Đăng nhập vào hệ thống quản trị
2	Hiển thị thông tin tin tức – sự kiện
3	Hiển thị các hoạt động công đoàn
4	Hiển thị thông tin khoa, công đoàn viên
5	Hiển thị thông tin kết quả tìm kiếm
6	Chức năng thông kê chi phí tổng
7	Chức năng thông kê chi phí hàng tháng
8	Chức năng quản lý công đoàn viên

STT	Tên yêu cầu
9	Chức năng quản lý danh sách ban chấp hành
10	Chức năng ghi chú sự kiện trên lịch
11	Chức năng tiện ích.
12	Quản lý hoạt động
13	Quản lý công đoàn phí
14	Quản lý tin tức sự kiện
15	Quản lý văn bản mẫu
16	Quản lý tài khoản
17	Quản lý menu
18	Quản lý các đơn vị

### 3.1.5 Mô tả yêu cầu của hệ thống

#### ❖ Các yêu cầu chức năng

Bảng 3. 2: Các yêu cầu chức năng

STT	Tên yêu cầu	Mô tả yêu cầu
<b>I</b>	<b>Các yêu cầu chức năng nghiệp vụ</b>	Là các chức năng của phần mềm tương ứng với các công việc trong thế giới thực.
1	Module tin tức – sự kiện	Hiển thị thông tin từ cơ bản đến chi tiết của tin tức sự kiện và hiển thị đầy đủ các thông tin như: Tên công việc, cán bộ phụ trách, tình trạng thực hiện, loại công việc, thời gian thực hiện, nội dung công việc và văn bản đính kèm theo tin tức sự kiện.

STT	Tên yêu cầu	Mô tả yêu cầu
2	Module đăng ký	Khi người dùng muốn được gia nhập hệ thống quản lý công đoàn nội bộ thì người quản trị sẽ truy cập hệ thống với tài khoản quản trị viên và tạo tài khoản với quyền tương ứng. Sau đó bàn giao lại tài khoản cho công đoàn viên.
3	Module Đăng nhập hệ thống	Mỗi người dùng hệ thống có quyền đăng ký một tài khoản riêng. Tài khoản này được sử dụng khi hệ thống yêu cầu. Một tài khoản người dùng đăng ký sẽ lưu trữ các thông tin của một thành viên.
4	Module tìm kiếm	Giúp người dùng tìm nhanh hơn, chỉ cần nhập tên gần đúng, hệ thống sẽ trả về danh sách gần đúng
5	Module thống kê chi phí theo tháng	Hệ thống tổng hợp loại chi phí được phân theo từng tháng gồm các loại chi phí khác nhau như: công đoàn phí, quỹ từ thiện, quyên góp.
<b>II</b>	<b>Các yêu cầu chức năng hệ thống</b>	Là các chức năng của phần mềm được phát sinh thêm khi thực hiện một công việc trên máy tính.
1	Quản lý hệ thống	<ul style="list-style-type: none"> <li>+ Quản lý thông tin tài khoản</li> <li>+ Thêm tài khoản</li> <li>+ Phân quyền tài khoản</li> </ul>

STT	Tên yêu cầu	Mô tả yêu cầu
2	Quản lý tin tức – sự kiện	+ Quản lý danh sách tin tức sự kiện + Quản lý công việc
3	Thống kê	+ Thống kê chi phí theo tháng
4	Quản lý người dùng	+ Quản lý quyền + Quản lý ban chấp hành
5	Quản lý đơn vị	+ Quản lý danh sách công đoàn viên + Quản lý danh sách đơn vị

#### ❖ Các yêu cầu phi chức năng

*Bảng 3. 3: Các yêu cầu phi chức năng*

STT	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 và chính xác.
3	Bảo mật dữ liệu	Tính bảo mật và độ an toàn cao.
4	Tương thích	Tương thích với đa phần các trình duyệt web hiện tại.

### 3.2. Phân tích thiết kế với UML.

Ngôn ngữ mô hình hóa thống nhất (tiếng Anh: Unified Modeling Language, viết tắt thành UML) là một ngôn ngữ mô hình gồm các ký hiệu đồ họa mà các phương

pháp hướng đối tượng sử dụng để thiết kế các hệ thống thông tin một cách nhanh chóng.

Cách xây dựng các mô hình trong UML phù hợp mô tả các hệ thống thông tin cả về cấu trúc cũng như hoạt động. Cách tiếp cận theo mô hình của UML giúp ích rất nhiều cho những người thiết kế và thực hiện hệ thống thông tin cũng như những người sử dụng nó; tạo nên một cái nhìn bao quát và đầy đủ về hệ thống thông tin dự định xây dựng.

Cái nhìn bao quát này giúp nắm bắt trọn vẹn các yêu cầu của người dùng; phục vụ từ giai đoạn phân tích đến việc thiết kế, thẩm định và kiểm tra sản phẩm ứng dụng công nghệ thông tin. Các mô hình hướng đối tượng được lập cũng là cơ sở cho việc ứng dụng các chương trình tự động sinh mã trong các ngôn ngữ lập trình hướng đối tượng, chẳng hạn như ngôn ngữ C++, Java,... Phương pháp mô hình này rất hữu dụng trong lập trình hướng đối tượng. Các mô hình được sử dụng bao gồm Mô hình đối tượng (mô hình tĩnh) và Mô hình động.

Để phân tích và thiết kế một phần mềm thì có nhiều cách làm, một trong những cách làm đó là xem hệ thống gồm những đối tượng sống trong đó và tương tác với nhau. Việc mô tả được tất cả các đối tượng và sự tương tác của chúng sẽ giúp chúng ta hiểu rõ hệ thống và cài đặt được nó. Phương thức này gọi là Phân tích thiết kế hướng đối tượng (OOAD).



OOAD cần các bản vẽ để mô tả hệ thống được thiết kế, còn UML là ngôn ngữ mô tả các bản vẽ nên cần nội dung thể hiện. Do vậy, chúng ta phân tích và thiết kế theo hướng đối tượng và sử dụng UML để biểu diễn các thiết kế đó nên chúng thường đi đôi với nhau. Nguyên tắc phân tích, thiết kế một hệ thống phần mềm cũng không khác việc xây dựng một cái nhà trong xây dựng. Hãy nhớ cách tiếp cận này để dễ hiểu hơn trong việc phân tích và thiết kế hệ thống.

### 3.2.1. Biểu đồ Use-case

Qua khảo sát hiện trạng thực tế, kết hợp với, việc tham khảo tài liệu và phân tích ý kiến của người dùng, hệ thống quản lý đồ án có các tác nhân sau:

#### ❖ Các tác nhân

*Bảng 3. 4: Các tác nhân*

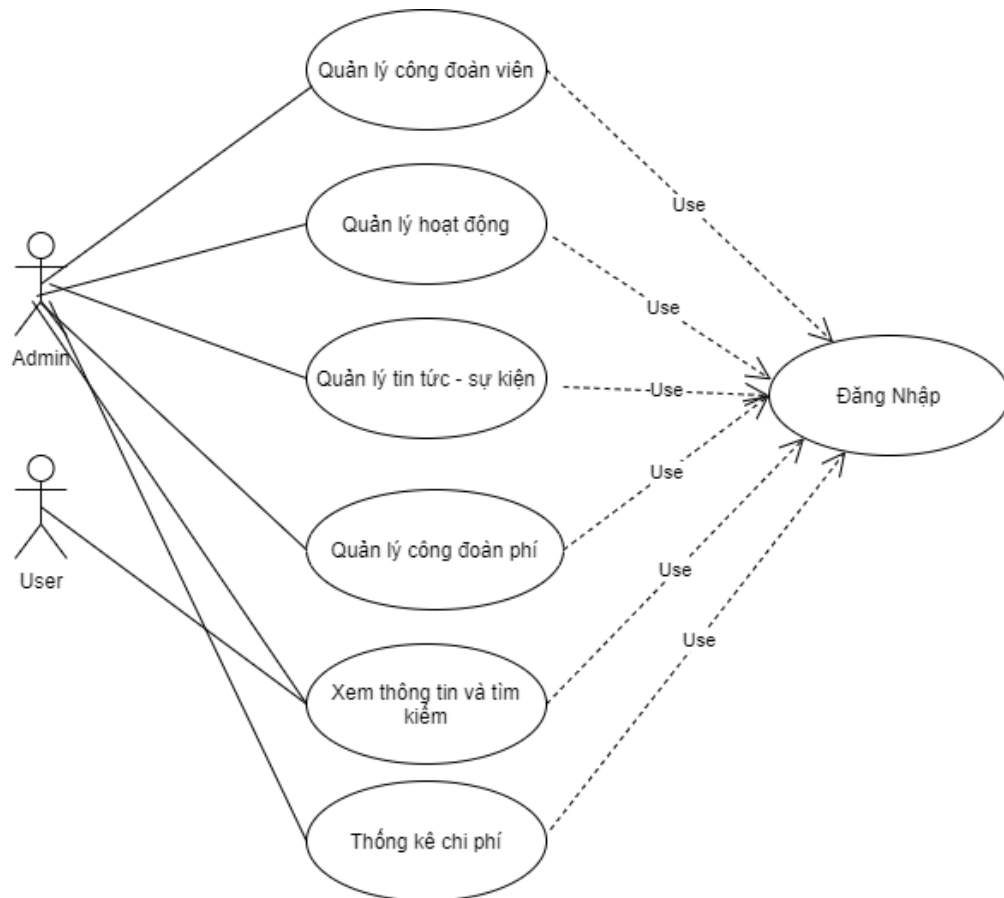
STT	Tác Nhân	Diễn Giải
1	 Admin	Những người thuộc nhóm quyền này có thể tạo tài khoản người dùng và phân quyền cho từng tài khoản đó
2	 Admin	Những người thuộc quyền này sẽ được Admin cấp cho tài khoản và được vào trang .

#### ❖ Các quyền sử dụng

*Bảng 3. 5: Các quyền sử dụng*

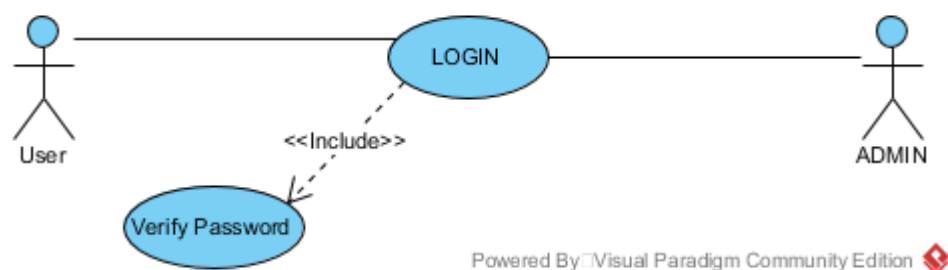
STT	Tác nhân chính	Tên Use case
1	Admin	Thực hiện các chức năng như quản lý tin tức – sự kiện, hoạt động công đoàn, người dùng... Để thực hiện chức năng này admin phải đăng nhập.
2	Người dùng	Thực hiện các chức năng như xem tin tức sự kiện, hoạt động công đoàn, thông tin các nhân.

### ❖ Biểu đồ Use-case tổng quát



Hình 3. 1: Biểu đồ usecase tổng quát

### ❖ Biểu đồ Use-case đăng nhập



Powered By Visual Paradigm Community Edition

Hình 3. 2: Biểu đồ use-case đăng nhập

Tác nhân: Admin, người dùng.

Mô tả: Use case cho admin, người dùng đăng nhập vào hệ thống.

Điều kiện trước: admin, người dùng chưa đăng nhập vào hệ thống.



Dòng sự kiện chính:

- Chọn chức năng đăng nhập.
- Giao diện đăng nhập hiển thị.
- Nhập username, mật khẩu vào giao diện đăng nhập.
- Hệ thống kiểm tra username và mật khẩu nhập của admin, nhân viên. Nếu nhập sai username hoặc mật khẩu thì chuyển sang dòng sự kiện rẽ nhánh A1. Nếu nhập đúng thì hệ thống kiểm tra đến quyền, nếu quyền là admin sẽ chuyển tới trang quản trị.

Use case kết thúc.

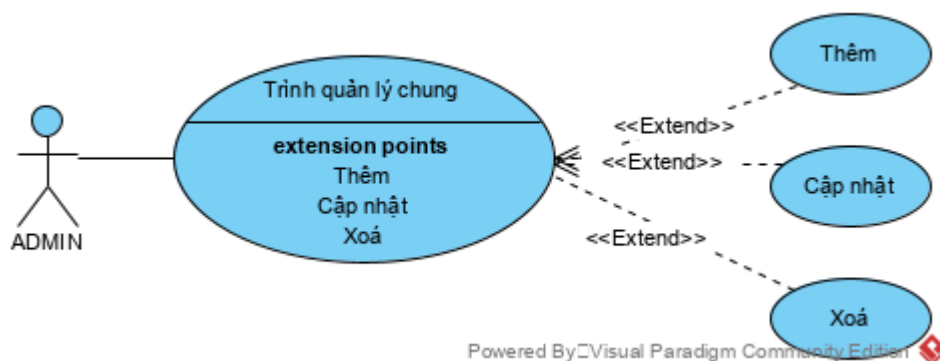
Dòng sự kiện rẽ nhánh:

- Dòng rẽ nhánh A1: admin, nhân viên đăng nhập không thành công.
- Hệ thống thông báo quá trình đăng nhập không thành công do sai username hoặc mật khẩu.
- Chọn nhập lại hệ thống yêu cầu nhập lại username, mật khẩu.

Use case kết thúc.

Hậu điều kiện: admin, nhân viên đăng nhập thành công và có thể sử dụng các chức năng tương ứng trong trang quản trị.

#### ❖ Biểu đồ Use-case quản lý tin tức – sự kiện



Hình 3. 3: Biểu đồ use-case quản lý tin tức sự kiện

Tác nhân: Admin

Mô tả: use case cho phép xem, thêm, sửa, xóa lịch công tác trong hệ thống.

Điều kiện trước: admin đã đăng nhập vào hệ thống.

Dòng sự kiện chính:

Người sử dụng chọn kiểu tác động: thêm, sửa, xóa tin tức – sự kiện.

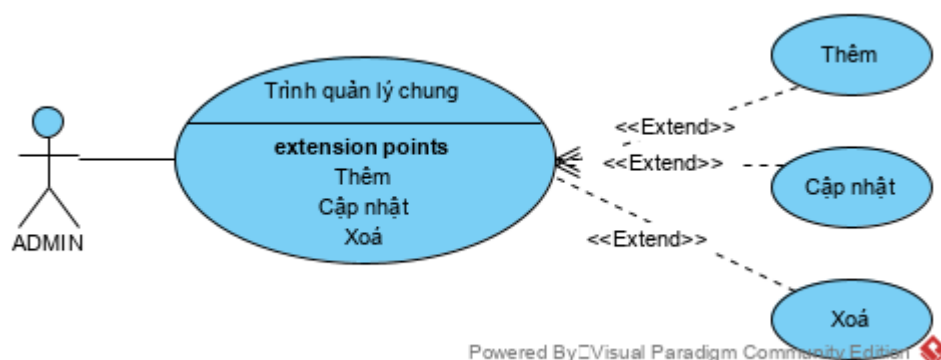
- Thêm tin tức sự kiện: chọn thêm tin tức – sự kiện, hệ thống hiển thị giao diện nhập tên tin tức – sự kiện, người sử dụng nhập thông tin của sự kiện đó, nếu thành công hệ thống đưa ra thông báo, nếu sai thực hiện dòng rẽ nhánh A1, lưu thông tin vào danh sách lịch công tác.
- Sửa thông tin lịch công tác: hệ thống hiển thị danh sách tin tức sự kiện, chọn tin tức – sự kiện cần sửa, nhập các thông tin cần thay đổi, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách lịch công tác.
- Xóa thông tin lịch công tác: hệ thống hiển thị danh sách tin tức sự kiện, chọn tin tức – sự kiện cần xóa, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách lịch công tác.

Use case kết thúc.

Dòng sự kiện rẽ nhánh A1: hệ thống thông báo việc nhập dữ liệu không hợp lệ, nhập lại thông tin, quay lại bước 1 của dòng sự kiện chính.

Hậu điều kiện: các thông tin về lịch công tác được cập nhật trong cơ sở dữ liệu.

#### ❖ Biểu đồ Use-case quản lý hoạt động



Hình 3. 4: Biểu đồ Use-case quản lý hoạt động

Tác nhân: Admin

Mô tả: use case cho phép xem, thêm, sửa, xóa thông tin loại hoạt động trong hệ thống.

Điều kiện trước: admin đã đăng nhập vào hệ thống.

Dòng sự kiện chính:

Người sử dụng chọn kiểu tác động: thêm, sửa, xóa thông tin hoạt động.

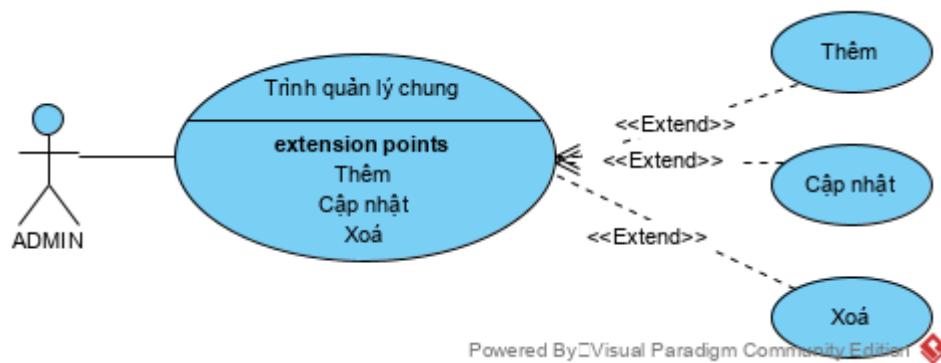
- Thêm thông tin loại hoạt động: chọn thêm loại hoạt động, hệ thống hiển thị giao diện nhập thông tin loại hoạt động, người sử dụng nhập thông tin loại hoạt động nếu thành công hệ thống đưa ra thông báo, nếu sai thực hiện dòng rẽ nhánh A1, lưu thông tin vào danh sách loại lịch công tác.
- Sửa thông tin hoạt động: hệ thống hiển thị danh sách loại hoạt động, chọn loại hoạt động cần sửa, nhập các thông tin cần thay đổi, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách loại hoạt động.
- Xóa thông tin loại hoạt động: hệ thống hiển thị danh sách loại hoạt động, chọn loại hoạt động cần xóa, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách loại hoạt động.

Use case kết thúc

Dòng sự kiện rẽ nhánh A1: hệ thống thông báo việc nhập dữ liệu không hợp lệ, nhập lại thông tin, quay lại bước 1 của dòng sự kiện chính.

Hậu điều kiện: các thông tin về loại lịch công tác được cập nhật trong cơ sở dữ liệu.

#### ❖ Biểu đồ Use-case quản lý công đoàn viên



Hình 3. 5: Biểu đồ user-case quản lý công đoàn viên

Tác nhân: Admin

Mô tả: use case cho phép xem, thêm, sửa, xóa thông tin công đoàn viên trong hệ thống.

Điều kiện trước: admin đã đăng nhập vào hệ thống.

Dòng sự kiện chính:

Người sử dụng chọn kiểu tác động: thêm, sửa, xóa thông tin công đoàn viên.

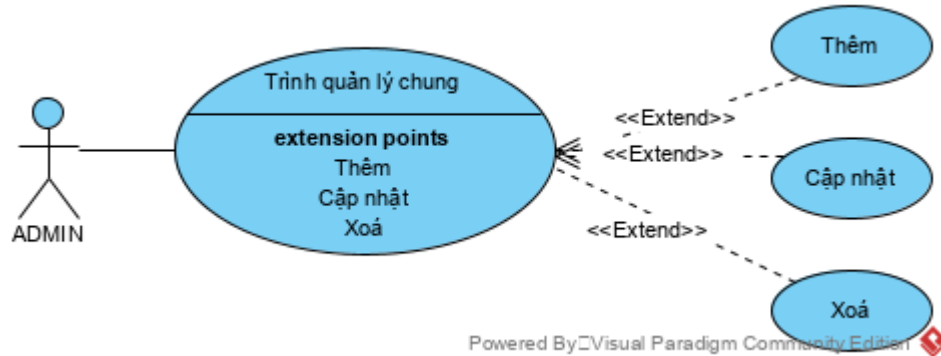
- Thêm thông tin công đoàn viên: chọn thêm công đoàn viên, hệ thống hiển thị giao diện nhập thông tin công đoàn viên, người sử dụng nhập thông tin công đoàn viên nếu thành công hệ thống đưa ra thông báo, nếu sai thực hiện dòng rẽ nhánh A1, lưu thông tin vào danh sách lớp học.
- Sửa thông tin công đoàn viên: hệ thống hiển thị danh sách công đoàn viên, chọn công đoàn viên cần sửa, nhập các thông tin cần thay đổi, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách công đoàn viên.
- Xóa thông tin lớp học: hệ thống hiển thị danh sách công đoàn viên, chọn công đoàn viên cần xóa, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách công đoàn viên.

Use case kết thúc

Dòng sự kiện rẽ nhánh A1: hệ thống thông báo việc nhập dữ liệu không hợp lệ, nhập lại thông tin, quay lại bước 1 của dòng sự kiện chính.

Hậu điều kiện: các thông tin về hàng hóa được cập nhật trong cơ sở dữ liệu.

❖ **Biểu đồ Use-case quản lý ban chấp hành**



Hình 3. 6: Biểu đồ user-case quản lý ban chấp hành

Tác nhân: Admin

Mô tả: use case cho phép xem, thêm, sửa, xóa thông tin ban chấp hành trong hệ thống.

Điều kiện trước: admin đã đăng nhập vào hệ thống.

Dòng sự kiện chính:

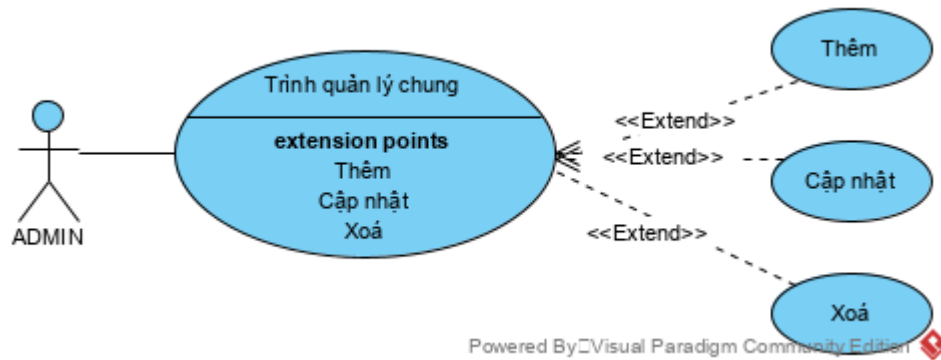
Người sử dụng chọn kiểu tác động: thêm, sửa, xóa thông tin ban chấp hành.

- Thêm thông tin ban chấp hành: chọn thêm giáo viên, hệ thống hiển thị giao diện nhập thông tin ban chấp hành, người sử dụng nhập thông tin ban chấp hành nếu thành công hệ thống đưa ra thông báo, nếu sai thực hiện dòng rẽ nhánh A1, lưu thông tin vào danh sách ban chấp hành.
- Sửa thông tin giáo viên: hệ thống hiển thị danh sách ban chấp hành, chọn ban chấp hành cần sửa, nhập các thông tin cần thay đổi, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách ban chấp hành.
- Xóa thông tin giáo viên: hệ thống hiển thị danh sách ban chấp hành, chọn ban chấp hành cần xóa, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách ban chấp hành.

Use case kết thúc

Dòng sự kiện rẽ nhánh A1: hệ thống thông báo việc nhập dữ liệu không hợp lệ, nhập lại thông tin, quay lại bước 1 của dòng sự kiện chính.

❖ **Biểu đồ Use-case quản lý văn bản mẫu**



Hình 3. 7: Biểu đồ user-case quản lý văn bản mẫu

Tác nhân: Admin

Mô tả: use case cho phép xem, thêm, sửa, xóa thông tin văn bản mẫu trong hệ thống.

Điều kiện trước: admin đã đăng nhập vào hệ thống.

Dòng sự kiện chính:

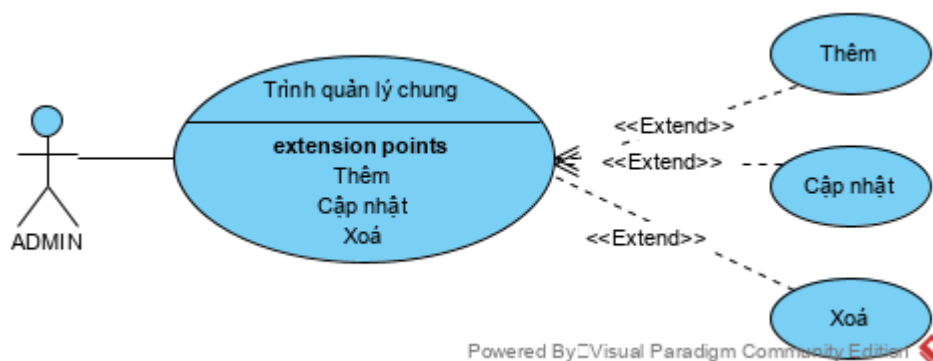
Người sử dụng chọn kiểu tác động: thêm, sửa, xóa thông tin văn bản mẫu.

- Thêm thông tin văn bản mẫu: chọn thêm văn bản mẫu, hệ thống hiển thị giao diện nhập thông tin văn bản mẫu, người sử dụng nhập thông tin quyền như tên công việc nếu thành công hệ thống đưa ra thông báo, nếu sai thực hiện dòng rẽ nhánh A1, lưu thông tin vào danh sách văn bản mẫu.
- Sửa thông tin công việc: hệ thống hiển thị danh sách văn bản mẫu, chọn văn bản mẫu cần sửa, nhập các thông tin cần thay đổi, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách văn bản mẫu.
- Xóa thông tin công việc: hệ thống hiển thị danh sách văn bản mẫu, chọn văn bản mẫu cần xóa, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách văn bản mẫu.

Use case kết thúc

Dòng sự kiện rẽ nhánh A1: hệ thống thông báo việc nhập dữ liệu không hợp lệ, nhập lại thông tin, quay lại bước 1 của dòng sự kiện chính.

#### ❖ Biểu đồ Use-case quản lý menu



Hình 3. 8: Biểu đồ user-case quản lý menu

Tác nhân: Admin

Mô tả: use case cho phép xem, thêm, sửa, xóa thông tin menu trong hệ thống.

Điều kiện trước: admin đã đăng nhập vào hệ thống.

Dòng sự kiện chính:

Người sử dụng chọn kiểu tác động: thêm, sửa, xóa thông tin menu.

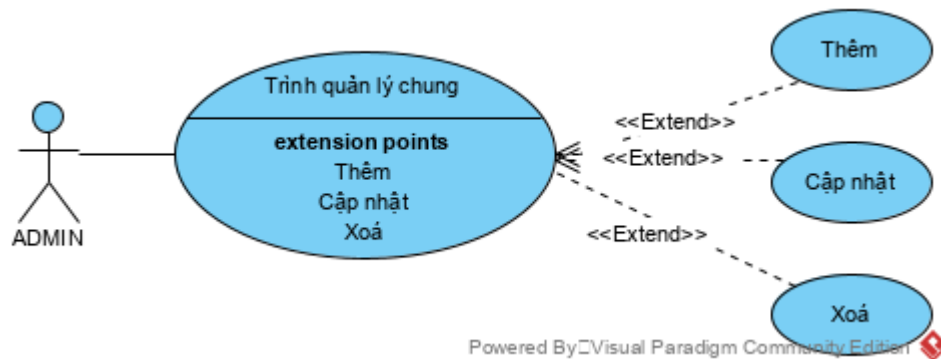
- Thêm menu: chọn thêm menu, hệ thống hiển thị giao diện nhập thông tin menu, người sử dụng nhập thông tin menu nếu thành công hệ thống đưa ra thông báo, nếu sai thực hiện dòng rẽ nhánh A1, lưu thông tin vào danh sách menu.
- Sửa thông tin menu: hệ thống hiển thị danh sách menu, chọn menu cần sửa, nhập các thông tin cần thay đổi, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách menu.
- Xóa thông tin menu: hệ thống hiển thị danh sách menu, chọn menu cần xóa, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách menu.

Use case kết thúc

Dòng sự kiện rẽ nhánh A1: hệ thống thông báo việc nhập dữ liệu không hợp lệ, nhập lại thông tin, quay lại bước 1 của dòng sự kiện chính.

Hậu điều kiện: các thông tin về nhân viên được cập nhật trong cơ sở dữ liệu.

### ❖ Biểu đồ Use-case quản lý đơn vị



Hình 3. 9: Biểu đồ use-case quản lý đơn vị

Tác nhân: Admin

Mô tả: use case cho phép xem, thêm, sửa, xóa thông tin đơn vị trong hệ thống.

Điều kiện trước: admin đã đăng nhập vào hệ thống.

Dòng sự kiện chính:

Người sử dụng chọn kiểu tác động: thêm, sửa, xóa thông tin đơn vị.

- Thêm khoa: chọn thêm đơn vị, hệ thống hiển thị giao diện nhập thông tin đơn vị, người sử dụng nhập thông tin đơn vị nếu thành công hệ thống đưa ra thông báo, nếu sai thực hiện dòng rẽ nhánh A1, lưu thông tin vào danh sách đơn vị.
- Sửa thông tin khoa: hệ thống hiển thị danh sách đơn vị, chọn đơn vị cần sửa, nhập các thông tin cần thay đổi, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách đơn vị.
- Xóa thông tin khoa: hệ thống hiển thị danh sách đơn vị, chọn đơn vị cần xóa, nếu việc thay đổi thông tin thành công hệ thống sẽ đưa ra thông báo, nếu sai thì thực hiện dòng rẽ nhánh A1, lưu thông tin danh sách đơn vị.

Use case kết thúc

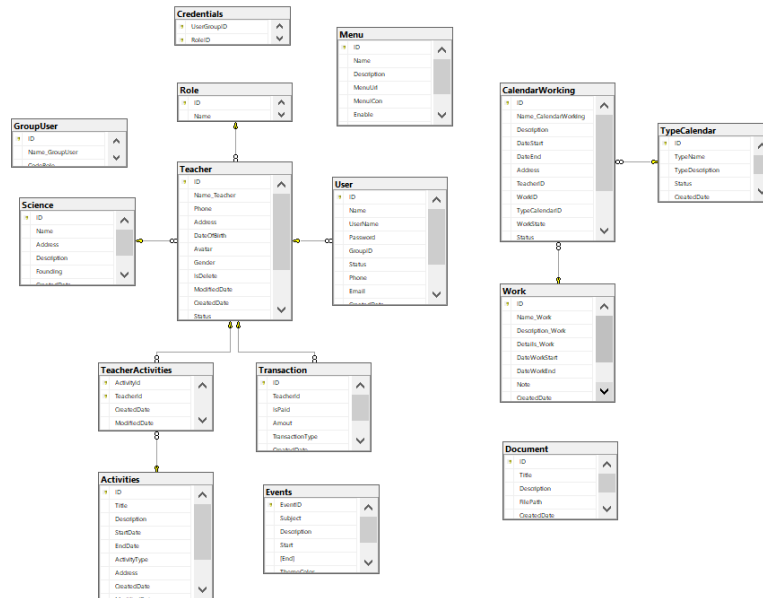
Dòng sự kiện rẽ nhánh A1: hệ thống thông báo việc nhập dữ liệu không hợp lệ, nhập lại thông tin, quay lại bước 1 của dòng sự kiện chính.



Hậu điều kiện: các thông tin về nhân viên được cập nhật trong cơ sở dữ liệu.

### 3.2.2 Biểu đồ lớp

#### ❖ Biểu đồ lớp tổng quát



Hình 3. 10: Biểu đồ lớp tổng quát

#### ❖ Danh sách các lớp đối tượng

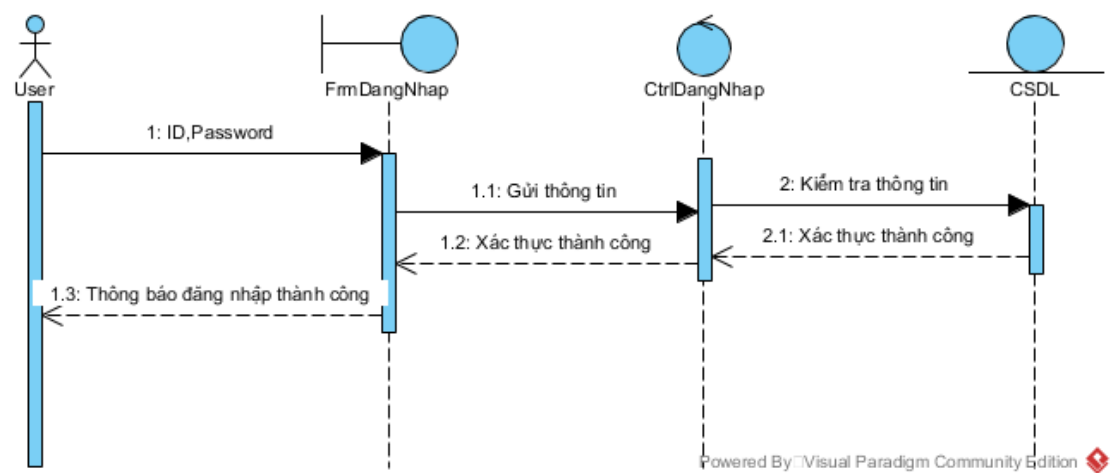
Bảng 3. 6: Danh sách các lớp

STT	Tên thực thể	Mô tả
1	Science	Mô tả các thuộc tính và các phương thức liên quan tới nghiệp vụ quản lý đơn vị
2	Activity	Mô tả các thuộc tính và các phương thức liên quan tới nghiệp vụ quản lý hoạt động
3	Teacher	Mô tả các thuộc tính và các phương thức liên quan tới nghiệp vụ quản lý công đoàn viên
4	Document	Mô tả các thuộc tính và các phương thức liên quan tới nghiệp vụ quản lý văn bản mẫu

STT	Tên thực thể	Mô tả
5	CalendarWorking	Mô tả các thuộc tính và các phương thức liên quan tới nghiệp vụ quản lý tin tức – sự kiện
6	Work	Mô tả các thuộc tính và các phương thức liên quan tới nghiệp vụ quản lý công việc
7	TypeCalendar	Mô tả các thuộc tính và các phương thức liên quan tới nghiệp vụ quản lý loại lịch công tác
8	Menu	Mô tả các thuộc tính và các phương thức liên quan tới nghiệp vụ quản lý menu
9	User	Mô tả các thuộc tính và các phương thức liên quan tới nghiệp vụ quản lý người dùng
10	Transaction	Mô tả các thuộc tính và các phương thức liên quan tới nghiệp vụ quản lý chi phí
11	TeacherActivities	Mô tả các thuộc tính và các phương thức liên quan tới nghiệp vụ quản lý công đoàn viên và hoạt động
12	Events	Mô tả các thuộc tính và các phương thức liên quan tới nghiệp vụ quản lý ghi chú công việc

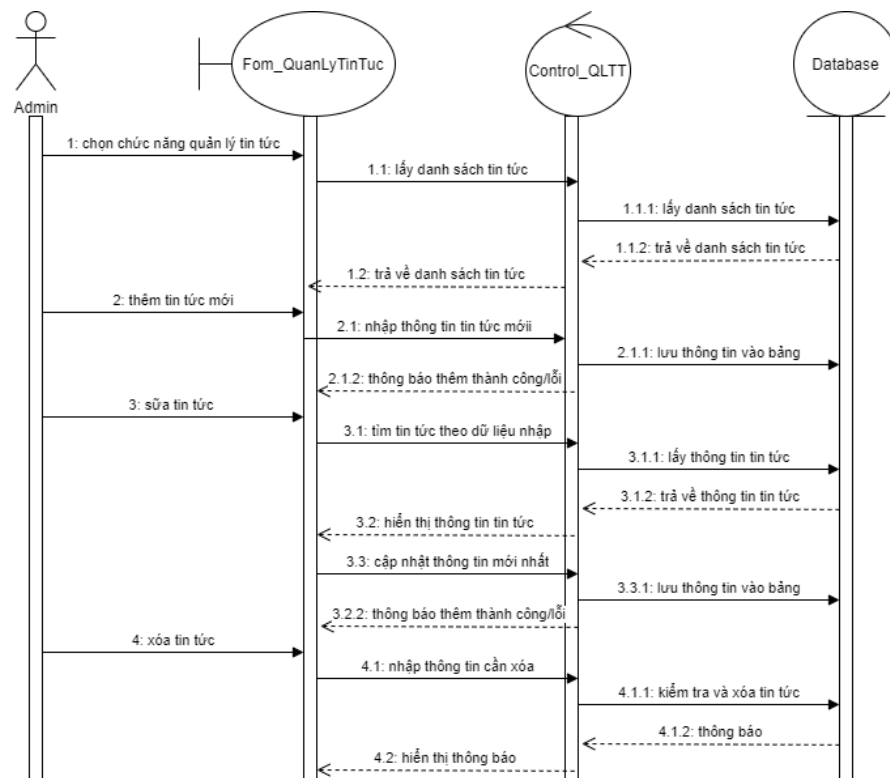
### 3.2.3 Biểu đồ tuần tự

#### ❖ Biểu đồ tuần tự chức năng “đăng nhập”



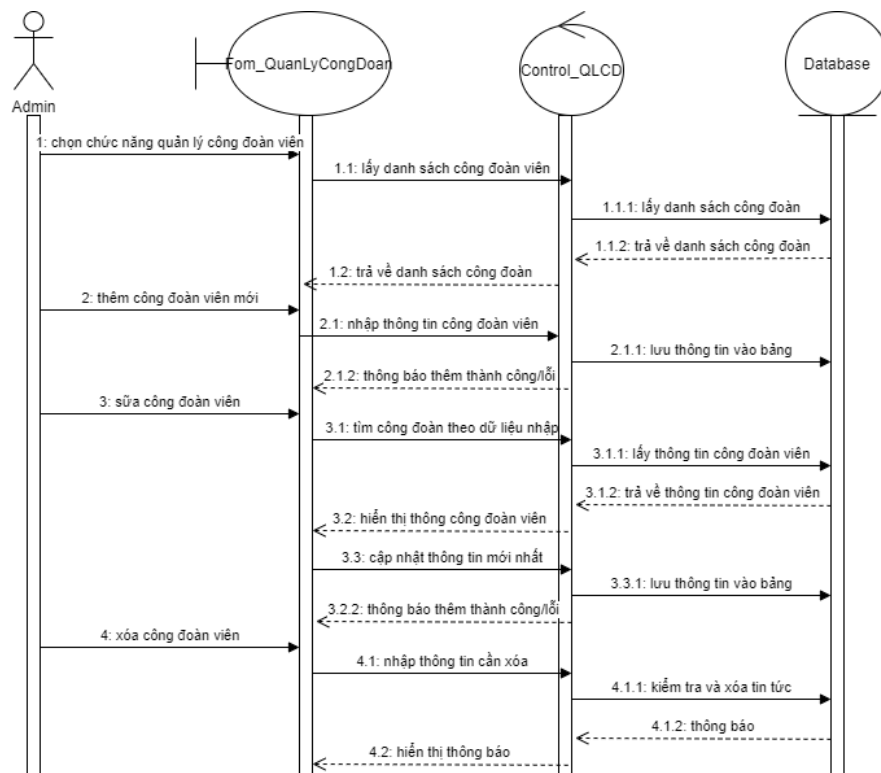
Hình 3 - 1: Biểu đồ tuần tự chức năng đăng nhập

❖ Biểu đồ tuần tự cho chức năng “quản lý tin tức – sự kiện”



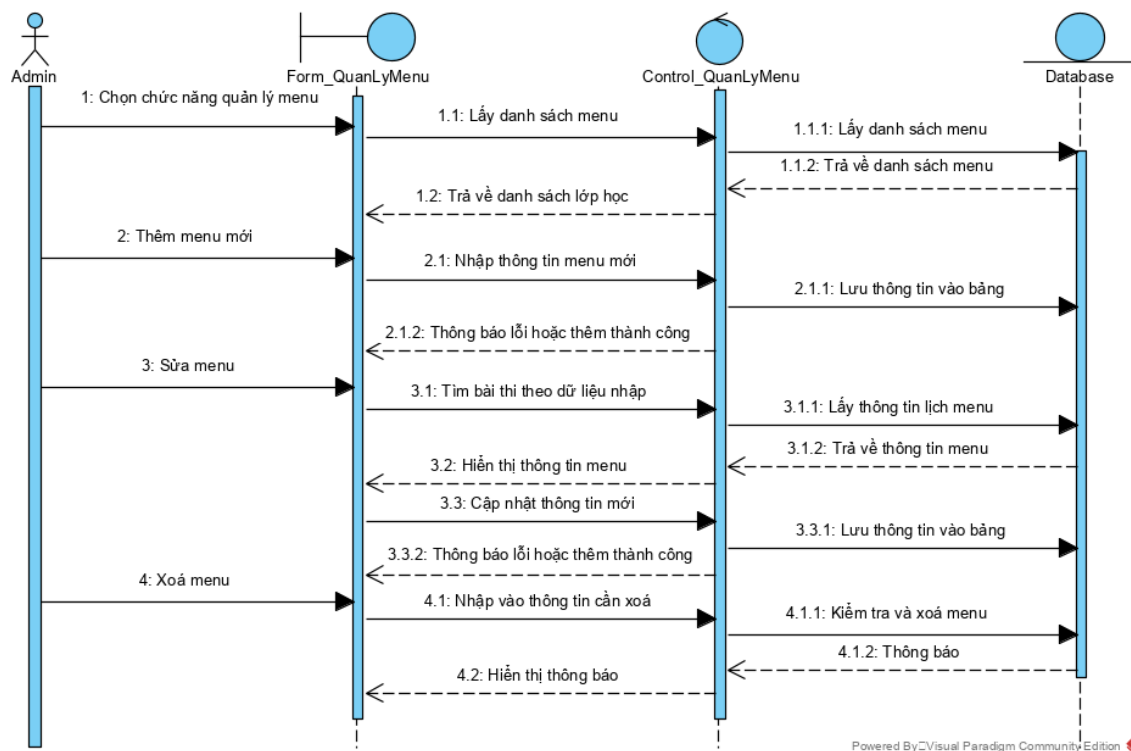
Hình 3. 11: Biểu đồ tuần tự cho chức năng quản lý tin tức – sự kiện

❖ Biểu đồ tuần tự cho chức năng “quản lý công đoàn viên”



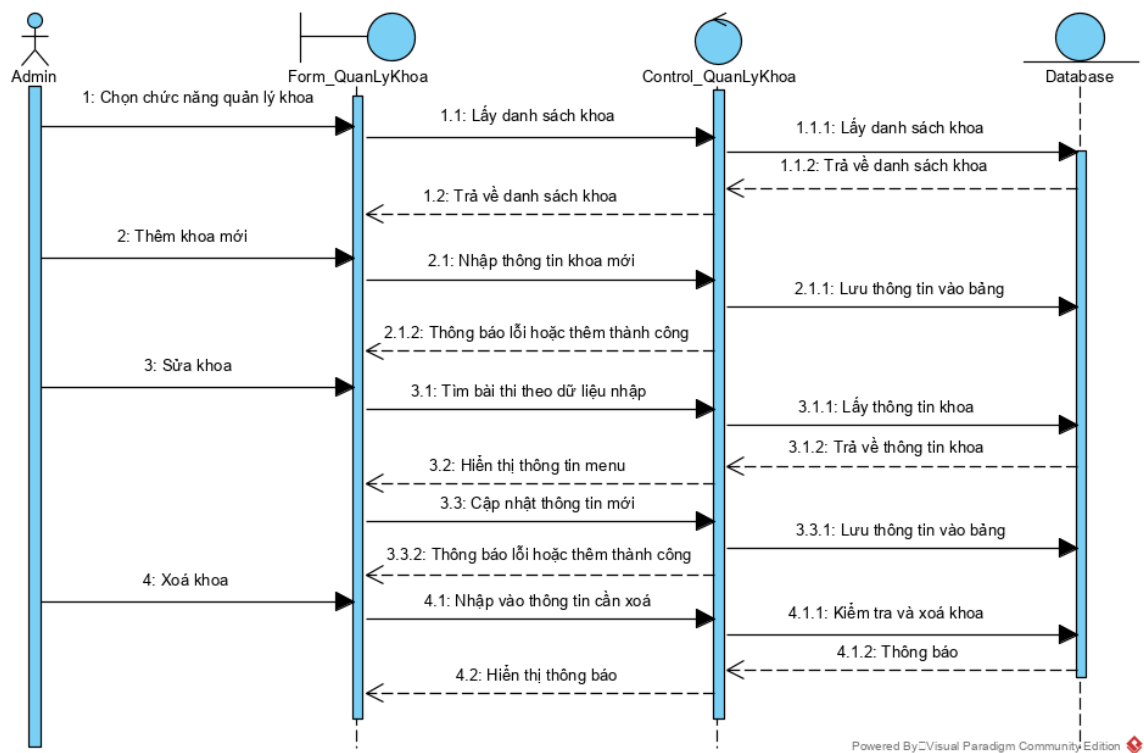
Hình 3. 12: Biểu đồ tuần tự cho chức năng quản lý lớp học

❖ Biểu đồ tuần tự cho chức năng “quản lý menu”



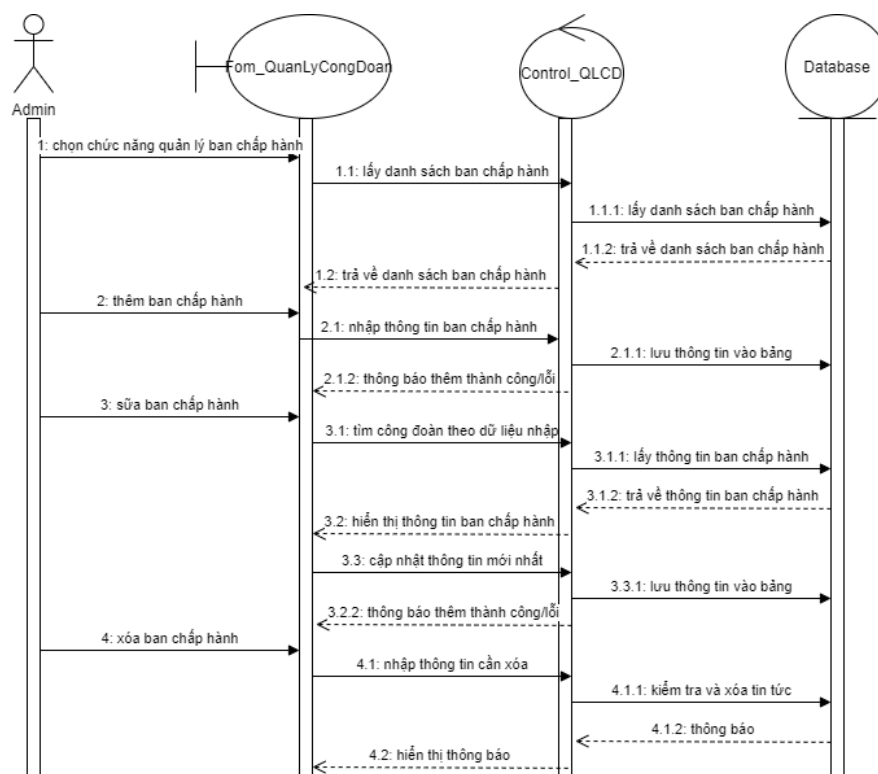
Hình 3. 13: Biểu đồ tuần tự cho chức năng thêm mới menu

❖ Biểu đồ tuần tự cho chức năng “thêm mới khoa”



Hình 3. 14: Biểu đồ tuần tự cho chức năng quản lý khoa

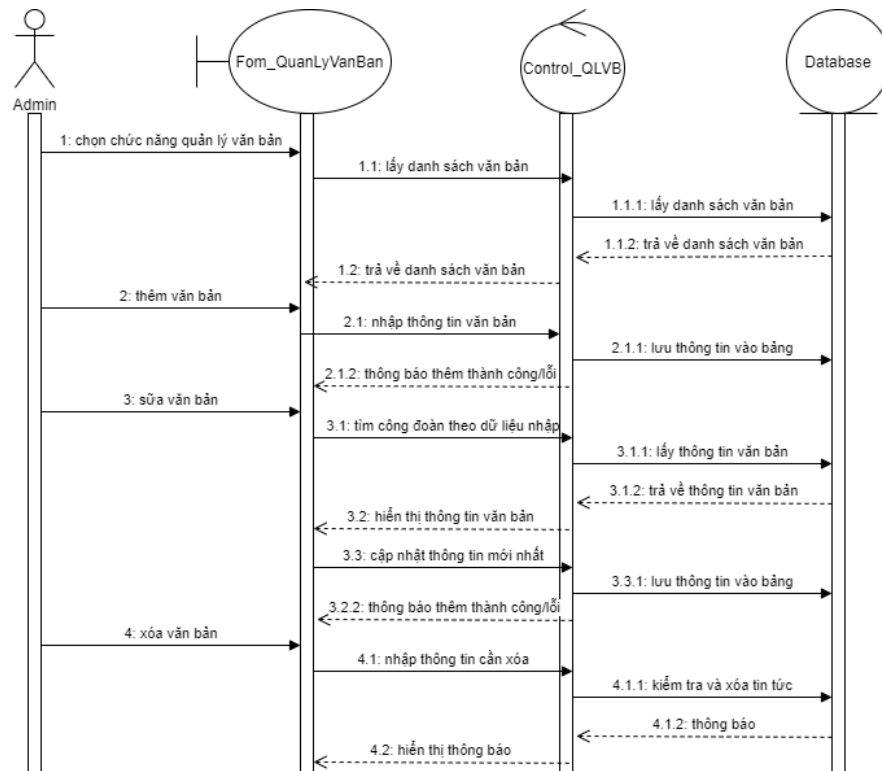
❖ Biểu đồ tuần tự cho chức năng “quản lý sinh viên”



Hình 3. 15: Biểu đồ tuần tự cho chức năng quản lý ban chấp hành

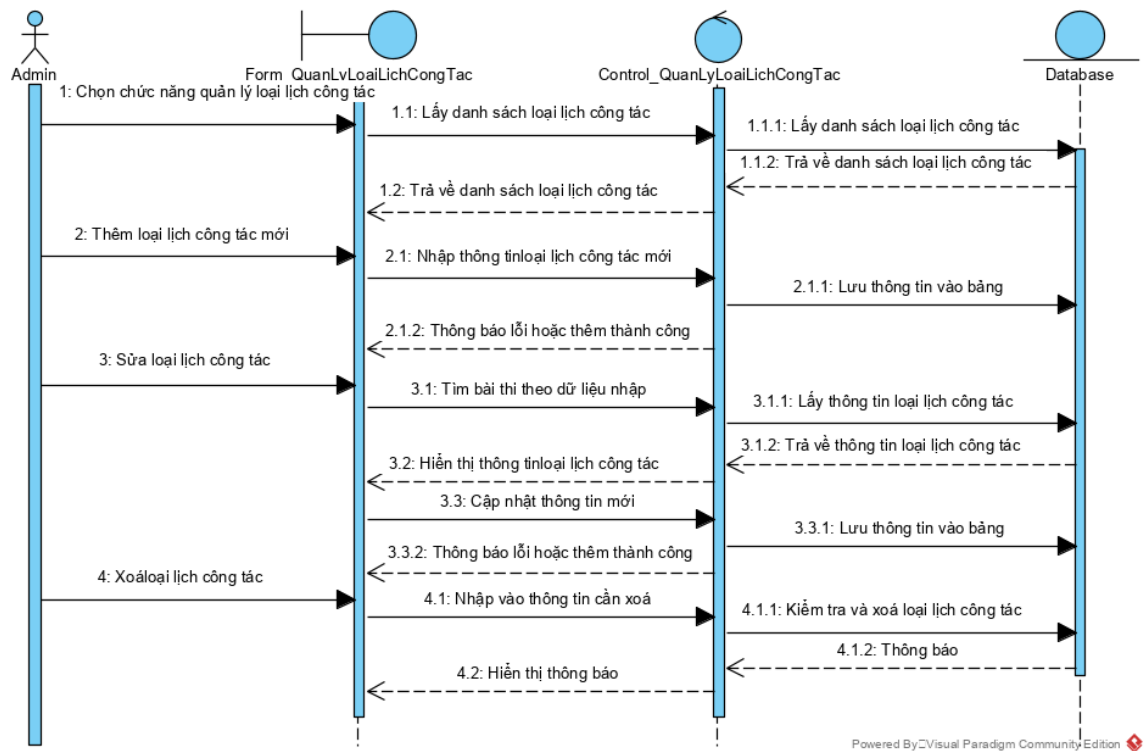
❖ Biểu đồ tuần tự cho chức năng “quản lý văn bản mẫu”





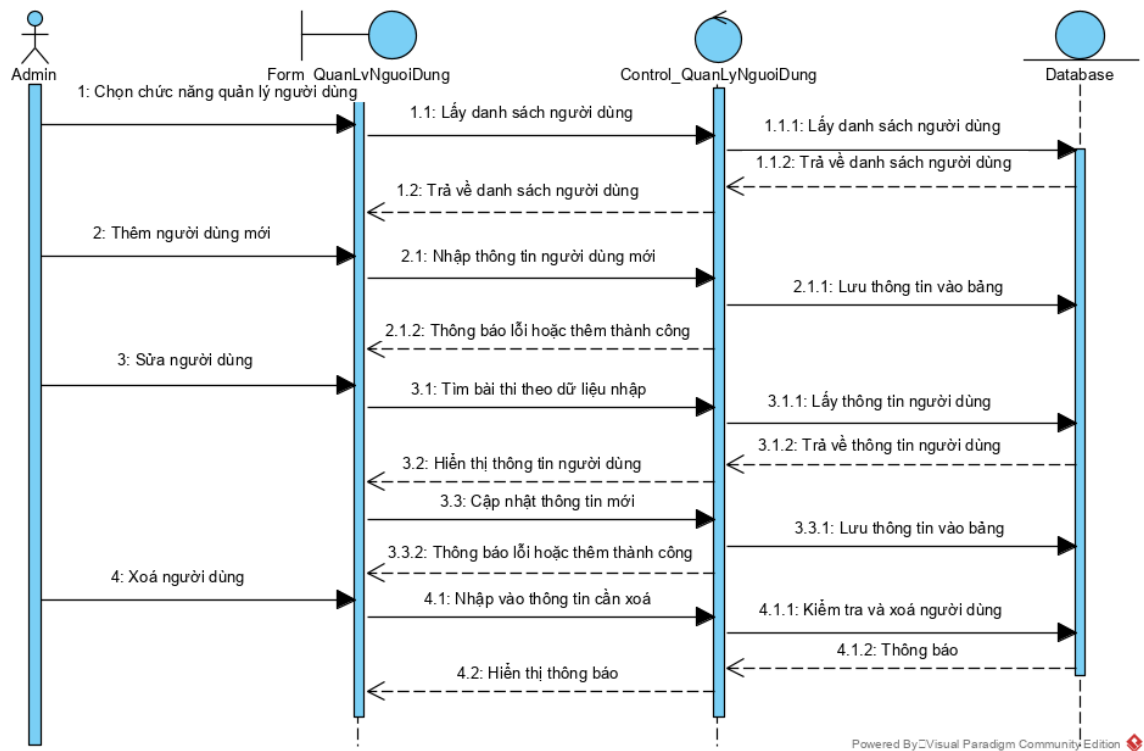
Hình 3. 16: Biểu đồ tuần tự cho chức năng quản lý văn bản

❖ Biểu đồ tuần tự cho chức năng “quản lý loại lịch công tác”



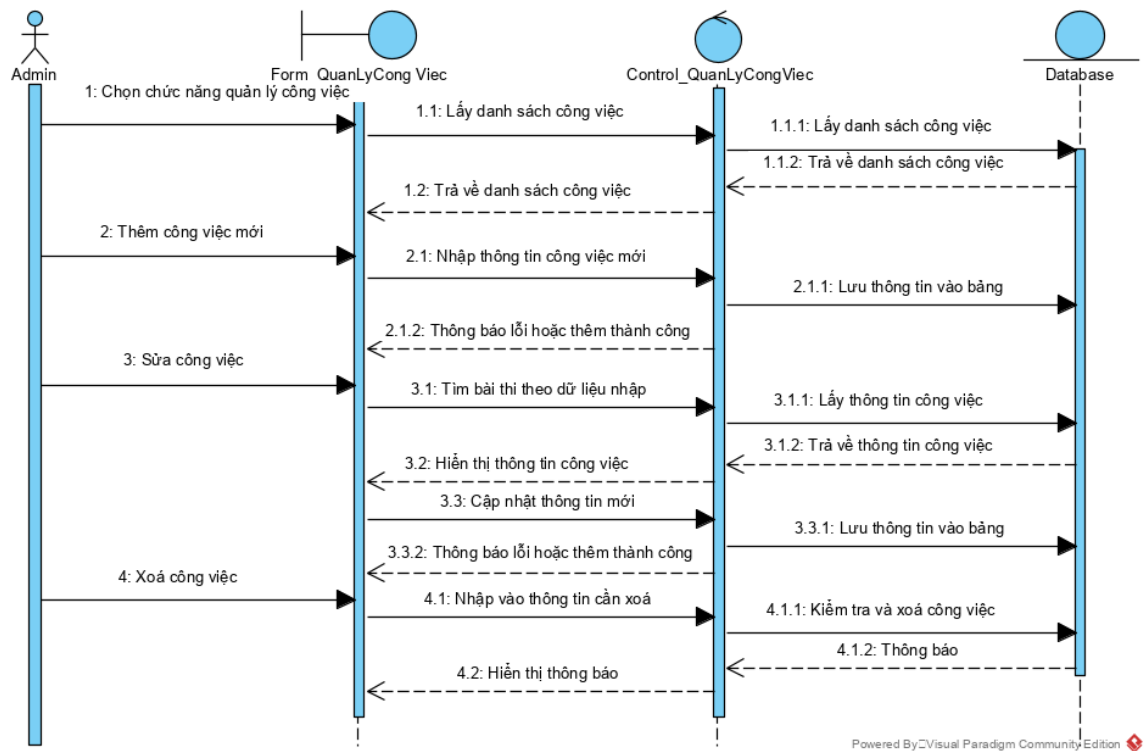
Hình 3. 17: Biểu đồ tuần tự cho chức năng quản lý loại lịch công tác

❖ Biểu đồ tuần tự cho chức năng “quản lý người dùng”



Hình 3. 18: Biểu đồ tuần tự cho chức năng quản lý người dùng

❖ Biểu đồ tuần tự cho chức năng “quản lý công việc”



Hình 3. 19: Biểu đồ tuần tự cho chức năng quản lý công việc

### 3.3 Thiết kế cơ sở dữ liệu

#### 3.3.1 Các bảng dữ liệu

##### ❖ Bảng role: Science

Bảng 3. 7: Bảng Science

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
1	ID	int	Not null	Khóa chính
2	NAME	Text	Not null	Tên khoa
3	ADDRESS	Text	Not null	Địa chỉ
4	DESCRIPTION	Text	Not null	Mô tả
5	FOUNDING	Datetime	Not null	Ngày thành lập
6	CREATEDDATE	Datetime	Not null	Ngày tạo

7	MODIFIEDDATE	Datetime	Not null	Ngày sửa
---	--------------	----------	----------	----------

❖ **Bảng CalendarWorking**

*Bảng 3. 8: Bảng CalendarWorking*

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
1	ID	Int	Not null	Khóa chính, tự động tăng
2	NAME_CALENDAR WORKING	Text	Not null	Tên nhân viên
3	DESCRIPTION	Text	Not null	Mô tả
4	DATESTART	Datetime	Not null	Ngày bắt đầu
5	DATEEND	Datetime	Not null	Ngày kết thúc
6	ADDRESS	Text	Not null	Địa chỉ
7	TEACHERID	Int	Not null	Khóa ngoại
8	WORKID	Int	Not null	Khóa ngoại
	TYPECALENDARID	Int	Not null	Khóa ngoại
8	CREATED_AT	Datetime	Not null	Ngày tạo
9	UPDATE_AT	Datetime	Not null	Ngày sửa
10	FILES	Text	Not null	Văn bản đính kèm

❖ **Bảng Activity**

*Bảng 3. 9: Bảng Activity*

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
1	ID	Int	Not null	Khóa chính, tự động tăng
2	TITLE	Text	Not null	Tên hoạt động
3	DESCRIPTION	Text	Not null	Mô tả
4	ACTIVITYTYPE	Int	Not null	Loại hoạt động
5	ADDRESS	Text	Not null	Địa chỉ
6	STARTDATE	Datetime	Not null	Ngày bắt đầu
7	ENDDATE	Datetime	Not null	Ngày kết thúc
8	ACTIVITYSTATUS	Int	Not null	Trạng thái

❖ **Bảng Credential**

*Bảng 3. 10: Bảng credential*

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
1	USERGROUPID	Int	Not null	Khóa chính
2	ROLEID	Int	Not null	Khoá chính

❖ **Bảng Event**

*Bảng 3. 11: Bảng Event*

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
1	EVENTID	Int	Not null	Khóa chính, tự động tăng
2	SUBJECT	Text	Not null	Tên sự kiện
4	DESCRIPTION	Text	Not null	Mô tả
5	START	Datetime	Not null	Ngày bắt đầu
6	END	Datetime	Not null	Ngày kết thúc
7	THEMCOLOR	Text	Not null	Màu
8	ISFULLDAY	Bit	Not null	Chọn cả ngày

❖ **Bảng GroupUser**

*Bảng 3. 12: Bảng GroupUser*

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
1	ID	Int	Not null	Khoá chính, tự tăng
2	NAME_GROUPUSER	Text	Not null	Tên nhóm
3	CODEROLE	Text	Not null	Mã HTML

❖ **Bảng Menu**

*Bảng 3. 13: Bảng Menu*

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
1	ID	Int	Not null	Khóa chính, tự động tăng
2	NAME	Text	Not null	Tên menu
3	DESCRIPTION	Text	Not null	Mô tả
4	MENURL	Text	Not null	Liên kết
5	MENUICON	Text	Not null	Css
6	ENABLE	Bit	Null	Trạng thái
5	CREATED_AT	Datetime	Not null	Ngày tạo
6	UPDATED_AT	Datetime	Not null	Ngày sửa

❖ **Bảng Role**

*Bảng 3. 14: Bảng Role*

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
1	ID	Int	Not null	Khóa chính, tự động tăng
2	NAME_ROLE	Text	Not null	Tên quyền



❖ **Bảng Transaction**

*Bảng 3. 15: Bảng Transaction*

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
1	ID	Int	Not null	Khóa chính, tự động tăng
2	TEACHERID	Int	Not null	Khóa ngoại
3	IsPaid	Bit	Not null	Trạng thái
4	AMOUNT	Decimal	Not null	Số tiền
5	TRANSACTIONTYPE	Int	Not null	Loại phí
6	CREATEDDATE	Datetime	Not null	Ngày Đóng
7	MODIFIEDDATE	Datetime	Not null	Ngày cập nhật

❖ **Bảng TypeCalendar**

*Bảng 3. 16: Bảng TypeCalendar*

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
1	ID	Int	Not null	Khóa chính, tự động tăng
2	TYPENAME	Text	Not null	Tên loại
3	TYPEDESCRIPTION	Text	Not null	Mô tả loại
4	STATUS	Text	Not null	Trạng thái

❖ **Bảng User**

*Bảng 3. 17: Bảng User*

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
1	ID	Int	Not null	Khóa chính, tự động tăng
2	Name	Text	Not null	Tên người dùng
3	USERNAME	Text	Not null	Tài khoản
4	PASSWORD	Text	Not null	Mật khẩu
5	GROUPIP	Text	Not null	Khoá ngoại
6	STATUS	Bit	Null	Trạng thái
7	PHONE	Text	Not null	Số điện thoại
8	EMAIL	Text	Not null	Email
9	CREATEDDATE	Datetime	Not null	Ngày tạo
10	MODIFIEDDATE	Datetime	Null	Ngày sửa

❖ **Bảng Work**

*Bảng 3. 18: Bảng Work*

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Mô tả
1	ID	Int	Not null	Khóa chính, tự động tăng
2	NAME_WORK	Int	Not null	Tên công việc
3	DESCRIPTION_WORK	Text	Not null	Mô tả
4	DETAILS_WORK	Text	Null	Chi tiết

5	DATEWORKSTART	Datetime	Not null	Ngày bắt đầu
6	DATEWORKEND	Datetime	Not null	Ngày kết thúc
7	NOTE	Text	Null	Ghi chú
8	CREATEDDATE	Datetime	Not null	Ngày tạo
9	MODIFIEDDATE	Datetime	Not null	Ngày sửa
10	STATUS	Bit	Not null	Trạng thái

### 3.4 Thiết kế đặc tả giao diện

#### 3.4.1 Chức năng hiển thị trang chủ

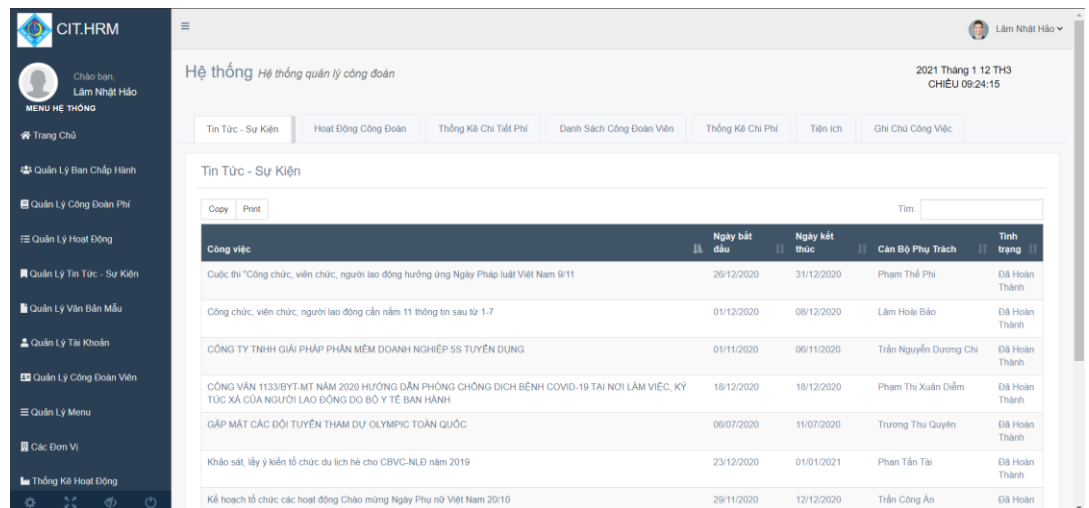
##### ❖ Mục đích

Giúp người quản lý nắm bắt được các thông tin về tin tức – sự kiện, danh sách hoạt động công đoàn, thông kê chi phí, tiện ích, ghi chú công việc.

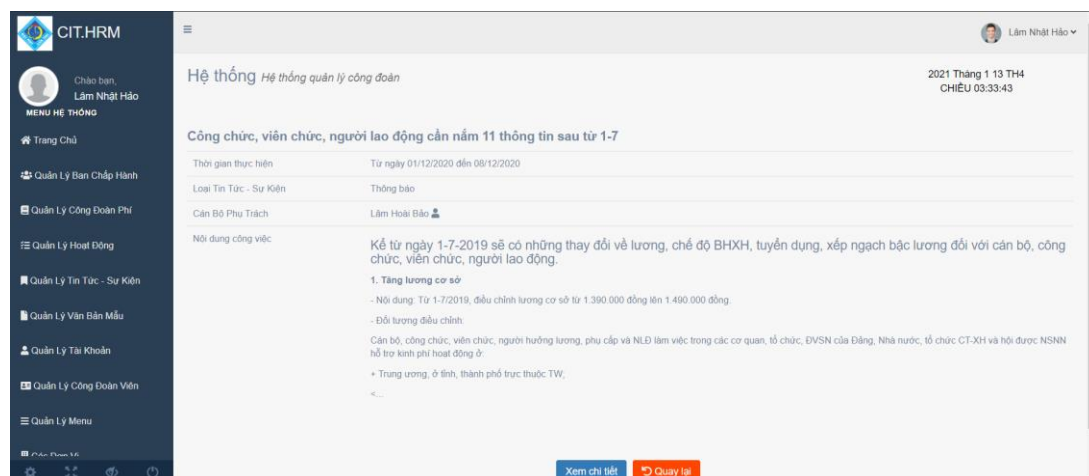
##### ❖ Phạm vi

Chức năng này sẽ cho phép người dùng truy cập vào website có thể sử dụng.

##### ❖ Giao diện của chức năng



Hình 3. 20: Giao diện chức năng hiển thị trang chủ



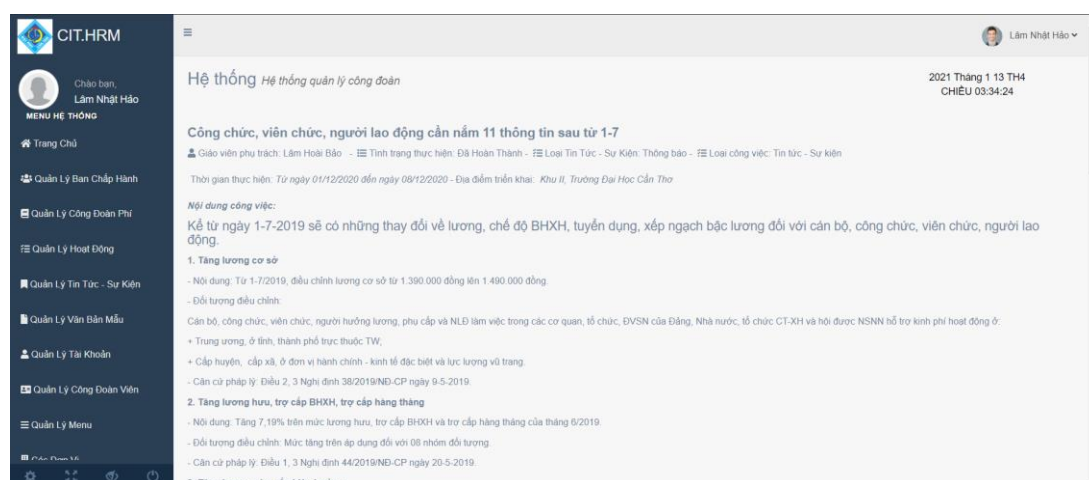
Hình 3. 21: Giao diện chức năng xem chi tiết tin tức sự kiện

### ❖ Mô tả chức năng

Khi người dùng click vào tiêu đề của một lịch công tác có trong danh sách thì hệ thống sẽ chuyển sang trang chi tiết tin tức – sự kiện cấp 1. Ở đây sẽ hiển thị một số thông tin cơ bản của lịch công tác.

### ❖ Phạm vi

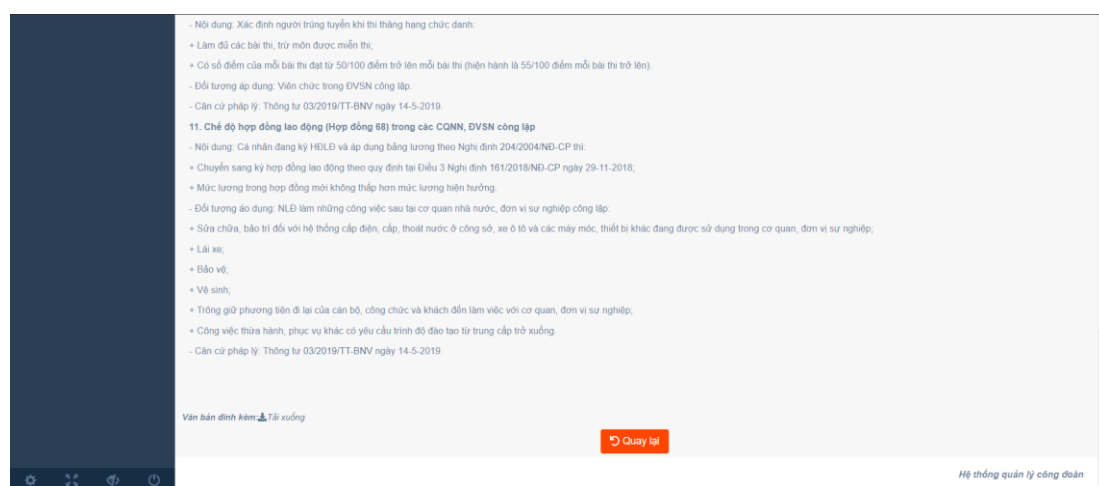
Người có tài khoản quản trị trong hệ thống.



Hình 3. 22: Giao diện chi tiết tin tức sự kiện cấp 2a



Hình 3. 23: Giao diện chi tiết tin tức sự kiện cấp 2b



Hình 3. 24: Giao diện chi tiết tin tức sự kiện cấp 2c

### ❖ Mô tả chức năng

Khi người dùng đã truy cập vào trang chi tiết tin tức – sự kiện cấp 1. Để xem tất cả các thông tin của tin tức – sự kiện giáo viên cần phải click vào button “xem chi tiết” hoặc không muốn xem thông tin có thể chọn “quay lại” để về trang trước.

### ❖ Phạm vi

Người có tài khoản quản trị trong hệ thống.

## 3.4.2 Chức năng đăng nhập

### ❖ Mục đích

Chức năng này sẽ kiểm tra tài khoản sử dụng hệ thống có hợp lệ hay không? Nhằm bảo mật các thông tin trong hệ thống. Và đảm bảo người dùng sử dụng đúng các chức năng được cấp.


❖ **Phạm vi**

Chức năng này sẽ kiểm tra tài khoản sử dụng hệ thống có hợp lệ hay không? Nhằm bảo mật các thông tin trong hệ thống.

❖ **Ràng buộc**

Khi người dùng kích hoạt hệ thống, một cửa sổ “Đăng nhập hệ thống” hiện lên, yêu cầu cung cấp các thông tin để đăng nhập sử dụng hệ thống.

❖ **Giao diện của chức năng**



The screenshot shows a login form titled "ĐĂNG NHẬP" (Login). It contains two input fields: "Tên Đăng Nhập" (Username) and "Mật Khẩu" (Password). Below the password field is a blue button labeled "Đăng nhập" (Login). At the bottom of the form, there is a link icon and the text "Hệ thống quản lý công đoàn" (Public Union Management System), followed by the copyright notice "© CIT CTU - 2020".

*Hình 3. 25: Giao diện chức năng đăng nhập*

❖ **Mô tả chức năng**

Hệ thống chạy nội bộ nên để sử dụng các chức năng của hệ thống thì bắt buộc phải đăng nhập. Hệ thống đăng nhập có validate các trường bắt buộc, kiểm tra thông tin đăng nhập của người dùng được cấp phép vào hệ thống.

### 3.4.3 Chức năng ghi chú sự kiện

❖ **Mục đích**

Giúp quan trị viên có thể ghi chú lại các công việc, sự kiện xảy ra trong tháng trực tiếp trên một giao diện lịch của năm.

❖ **Phạm vi**

Quản trị có quyền được phép truy cập chức năng này.

❖ **Ràng buộc**

Sau khi đăng nhập vào hệ thống người sử dụng muốn sử dụng chức năng chọn “Ghi chú công việc” trong menu ngang trên giao diện trang chủ.

❖ **Giao diện chức năng**



Hình 3. 26: Giao diện chức năng ghi chú công việc

❖ **Mô tả chức năng**

Ghi chú công việc giúp người dùng dễ dàng thêm các sự kiện chỉ bằng cách nhấp chọn vào ngày muốn thêm sự kiện, các sự kiện cũng có thể dễ dàng thay đổi bằng cách di chuyển sự kiện bằng giữ chuột.

### 3.4.4 Chức năng hiển thị danh sách công đoàn viên theo từng đơn vị

❖ **Mục đích**

Giúp người dùng dễ dàng tìm kiếm thông tin về công đoàn viên theo từng đơn vị.

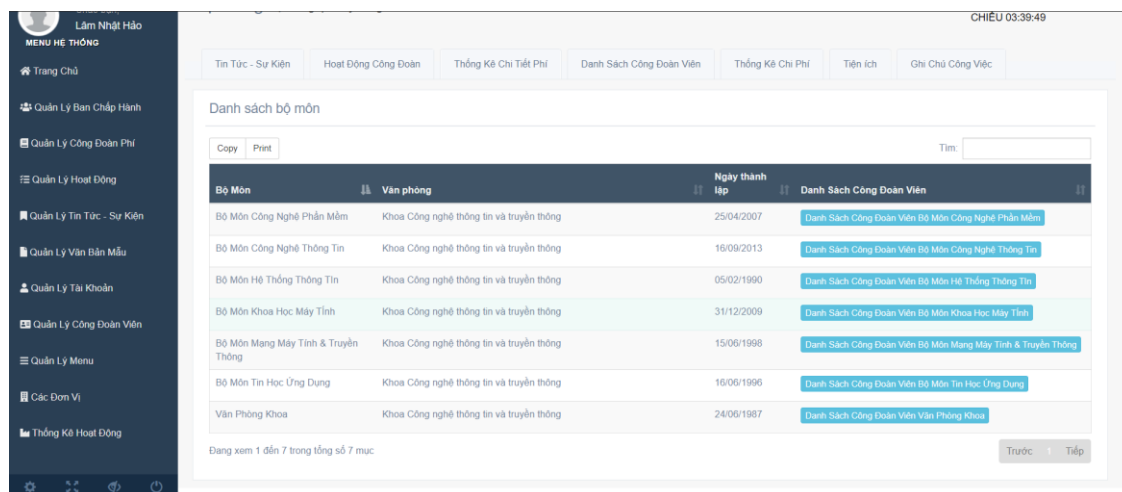
❖ **Phạm vi**

Người có tài khoản trong hệ thống được sử dụng chức năng này.

❖ **Ràng buộc**

Sau khi đăng nhập vào hệ thống người sử dụng muốn sử dụng chức năng chọn “Danh sách công đoàn viên” trong menu ngang ở trang chủ.

### ❖ Giao diện chức năng



Hình 3. 27: Giao diện chức năng hiển thị danh sách công đoàn viên theo đơn vị

### ❖ Mô tả chức năng

Người dùng khi muốn xem danh sách giáo viên có trong hệ thống theo các khoa trong trường thì click vào tên khoa, sau đó hệ thống sẽ trả về danh sách các giáo viên thuộc khoa đã chọn tương ứng. Khi click vào tên từng giáo viên thì hệ thống sẽ chuyển hướng người dùng sang trang chi tiết thông tin giáo viên.

### 3.4.5 Chức năng hiển thị danh sách công đoàn viên theo hoạt động

#### ❖ Mục đích

Giúp người dùng xem thông tin của các công đoàn viên theo hoạt động.

#### ❖ Phạm vi

Người dùng được cấp tài khoản trong hệ thống có thể truy cập.

#### ❖ Ràng buộc

Sau khi đăng nhập vào hệ thống người sử dụng muốn sử dụng chức năng chọn “Danh sách tham gia” trong menu ngang ở trang chủ.

### ❖ Giao diện chức năng





Tên Hoạt Động	Loại Hoạt Động	Ngày Bắt Đầu	Địa Điểm	Trạng Thái	Danh Sách
Đăng ký tham gia giải quần vợt Chào mừng Ngày Nhà giáo Việt Nam 20/11	Thể Dục - Thể Thao	18/12/2020 12:00:00 SA	Hội Trường 2 - Khoa CNTT	Đã Hoàn Thành	<a href="#">Danh Sách Tham Gia</a>
Hội thao chào mừng Ngày Nhà giáo Việt Nam 20/11	Thể Dục - Thể Thao	19/11/2020 12:00:00 SA	Sân Chính Trị	Đã Hoàn Thành	<a href="#">Danh Sách Tham Gia</a>
Hỗ Trợ Trẻ Em Nghèo Hiếu Học	Từ Thiện	26/12/2020 12:00:00 SA	Trường THCS Bùi Hữu Nghĩa	Đã Hoàn Thành	<a href="#">Danh Sách Tham Gia</a>
Kế hoạch tổ chức giải cầu lông CBVC NLD Trường ĐH CNTT&TT	Thể Dục - Thể Thao	23/12/2020 12:00:00 SA	Hội Trường Rùa, Khu II, Đại Học Cần Thơ	Đã Hoàn Thành	<a href="#">Danh Sách Tham Gia</a>
Kêu gọi đóng góp hỗ trợ đồng bào Miền Trung bị lũ lụt	Từ Thiện	12/12/2020 12:00:00 SA	Phòng CTSV	Đã Hoàn Thành	<a href="#">Danh Sách Tham Gia</a>
Kế hoạch tổ chức chương trình "Mùa Xuân Yêu Thương" lần VIII, năm 2021	Tình nguyện	11/12/2020 12:00:00 SA	Vinh Lôi, Bạc Liêu	Đã Hoàn Thành	<a href="#">Danh Sách Tham Gia</a>
Ủng hộ phòng chống dịch bệnh Covid-19	Từ Thiện	06/12/2020 12:00:00 SA	Văn Phòng Đoàn Khoa	Đã Hoàn Thành	<a href="#">Danh Sách Tham Gia</a>
Văn nghệ Chào mừng ngày Nhà giáo Việt Nam 20/11	Văn Nghệ	19/11/2020 12:00:00 SA	Hội Trường Rùa, Khu II, Đại Học Cần Thơ	Đã Hoàn Thành	<a href="#">Danh Sách Tham Gia</a>

Hình 3. 28: Giao diện trang danh sách công đoàn viên theo hoạt động

### ❖ Mô tả chức năng

Người khi muốn xem thông tin của sinh viên theo từng lớp thì click vào “Danh sách tham gia” ở trên menu ngang của hệ thống, sau đó chọn danh sách tham gia muốn xem. Hệ thống sẽ trả ra danh sách các công đoàn viên đã chọn tương ứng.

### 3.4.6 Chức năng thống kê chi tiết phí

#### ❖ Mục đích

Giúp người quản trị dễ dàng theo dõi và quản lý chi tiết phí theo từng tháng trong năm

#### ❖ Phạm vi

Quản trị viên được phép xem thông tin chi tiết phí

#### ❖ Ràng buộc

Sau khi đăng nhập vào hệ thống người sử dụng muốn sử dụng chức năng chọn “Thống kê chi tiết phí” trong menu ngang của hệ thống.

#### ❖ Giao diện chức năng



Hình 3. 29: Giao diện chức năng thống kê chi tiết phí

### ❖ Mô tả chức năng

Người dùng không có quyền quản trị thì sẽ nhìn thấy nút “Thêm báo cáo tháng” và nút “Thêm mẫu báo cáo” sẽ bị ẩn đi do chỉ có quyền thành viên. Khi người dùng muốn viết báo cáo tháng thì cần phải tải xuống mẫu báo cáo phù hợp để chỉnh sửa cho đúng mẫu. Sau đó click “Gửi báo cáo tháng” để gửi báo cáo về cho phòng công tác sinh viên.

### 3.4.7 Chức năng thống kê chi phí

#### ❖ Mục đích

Giúp người quản trị viên nắm tổng chi phí trong năm.

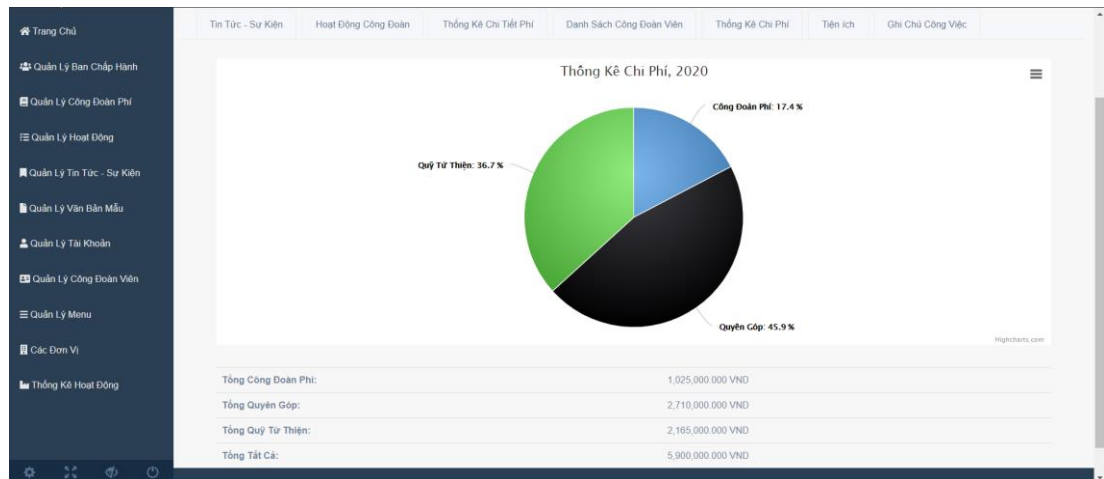
#### ❖ Phạm vi

Quản trị có quyền được phép truy cập chức năng thống kê chi phí.

#### ❖ Ràng buộc

Sau khi đăng nhập vào hệ thống người quản trị sử dụng muốn sử dụng chức năng chọn “Thống kê chi phí” trong menu ngang của hệ thống.

#### ❖ Giao diện chức năng



Hình 3. 30: Giao diện trang thống kê chi phí

### ❖ Mô tả chức năng

Người quản trị khi truy cập trang thống kê chi phí sẽ thấy được tổng phí trong năm với từng loại chi phí khác nhau

#### 3.4.8 Chức năng quản lý tin tức – sự kiện

### ❖ Mục đích

Giúp người quản lý quản lý thông tin về tin tức – sự kiện.

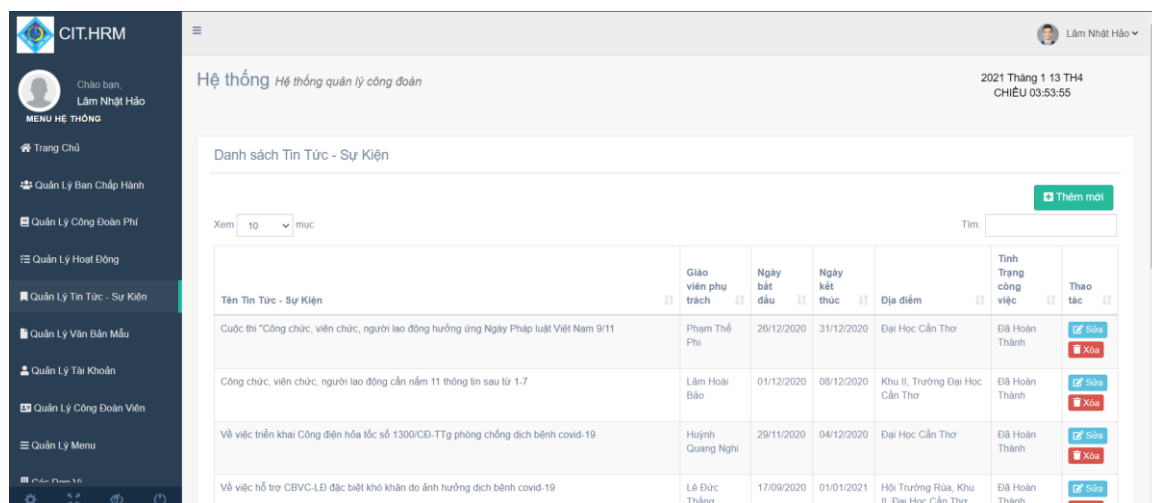
### ❖ Phạm vi

Quản trị có quyền được phép truy cập chức năng quản lý tin tức – sự kiện.

### ❖ Ràng buộc

Sau khi đăng nhập vào hệ thống người sử dụng muốn sử dụng chức năng chọn “Quản lý tin tức – sự kiện” trong ở menu bên trái.

### ❖ Giao diện chức năng



Hình 3. 31: Giao diện trang quản lý tin tức – sự kiện

### ❖ Mô tả chức năng

Các thao tác người dùng có thể thực hiện khi sử dụng chức năng “Quản lý tin tức – sự kiện”:

- Chức năng thêm: khi nút thêm “Thêm mới” và điền đầy đủ các thông tin theo yêu cầu thì ấn nút “Thêm mới” trong dialog. Hệ thống sẽ thêm thông tin vừa nhập vào database và hiển thị lên thông tin đã thêm trên danh sách.

- Chức năng sửa thông tin: khi ấn vào nút “Sửa” hệ thống sẽ hiển thị thông tin của lịch công tác đã chọn lên dialog, người dùng chỉnh sửa các thông tin cần thay đổi sau đó click “Cập nhật”. Hệ thống sẽ cập nhật các thông tin thay đổi vào trong database và hiển thị danh sách các thay đổi mới.

### 3.4.9 Chức năng quản lý công đoàn viên

#### ❖ Mục đích

Giúp người quản lý quản lý thông tin công đoàn viên.

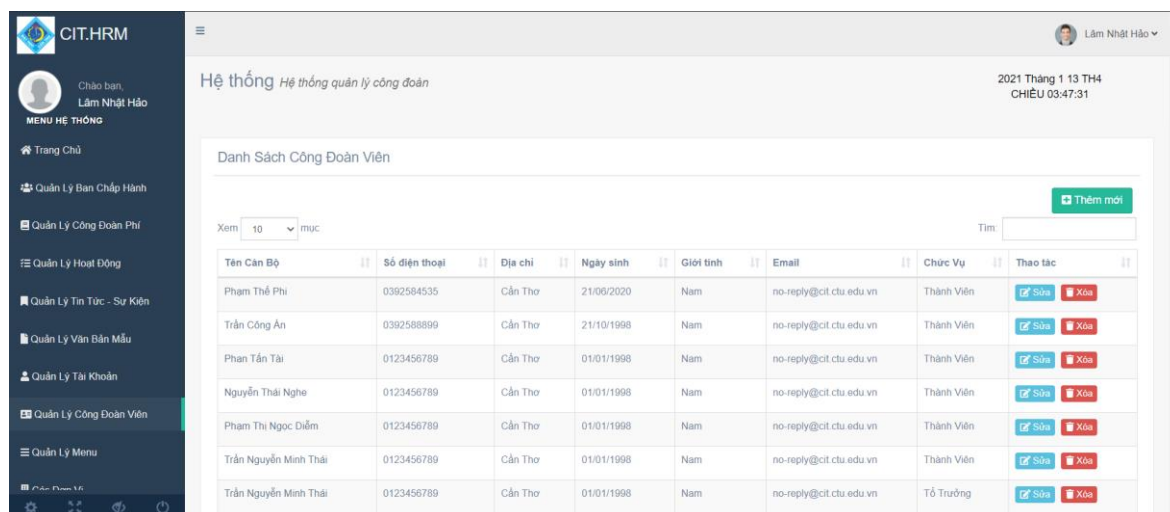
#### ❖ Phạm vi

Quản trị có quyền được phép truy cập chức năng quản lý công đoàn viên.

#### ❖ Ràng buộc

Sau khi đăng nhập vào hệ thống người sử dụng muốn sử dụng chức năng chọn “Quản lý công đoàn viên” trong ở menu bên trái.

#### ❖ Giao diện chức năng



Hình 3. 32: Giao diện trang quản lý công đoàn viên

### ❖ Mô tả chức năng

Các thao tác người dùng có thể thực hiện khi sử dụng chức năng “Quản lý công đoàn viên”:

- Chức năng thêm: khi nút thêm “Thêm mới” và điền đầy đủ các thông tin theo yêu cầu thì ấn nút “Thêm mới” trong dialog. Hệ thống sẽ thêm thông tin vừa nhập vào database và hiển thị lên thông tin đã thêm trên danh sách.

- Chức năng sửa thông tin: khi ấn vào nút “Sửa” hệ thống sẽ hiển thị thông tin của sinh viên đã chọn lên dialog, người dùng chỉnh sửa các thông tin cần thay đổi sau đó click “Cập nhật”. Hệ thống sẽ cập nhật các thông tin thay đổi vào trong database và hiển thị danh sách các thay đổi mới.

### 3.4.10 Chức năng quản lý ban chấp hành

#### ❖ Mục đích

Giúp người quản lý quản lý thông tin về ban chấp hành.

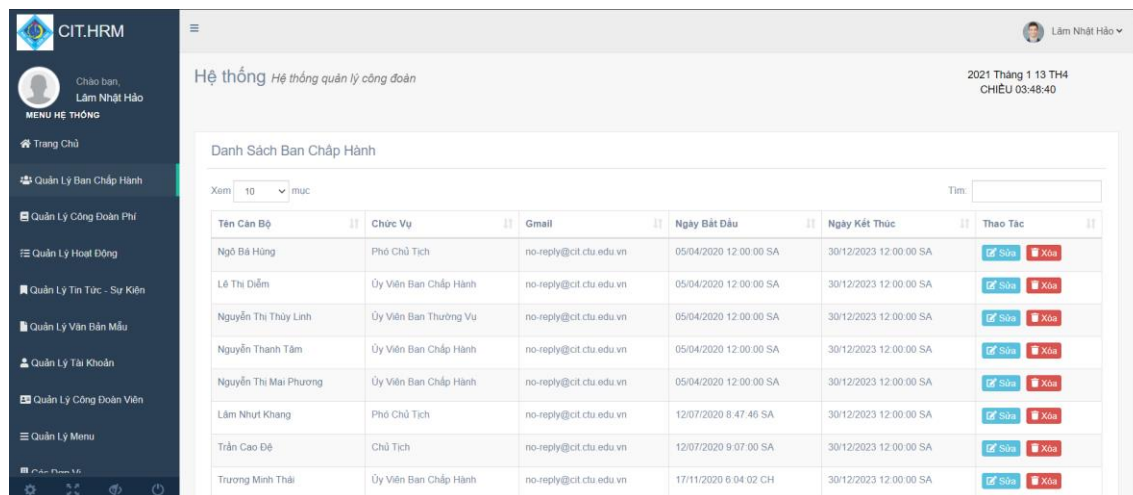
#### ❖ Phạm vi

Quản trị có quyền được phép truy cập chức năng quản lý ban chấp hành.

#### ❖ Ràng buộc

Sau khi đăng nhập vào hệ thống người sử dụng muốn sử dụng chức năng chọn “Quản lý ban chấp hành” trong ở menu bên trái.

### ❖ Giao diện chức năng



Hình 3. 33: Giao diện trang quản lý ban chấp hành

### ❖ Mô tả chức năng

Các thao tác người dùng có thể thực hiện khi sử dụng chức năng “Quản lý ban chấp hành”:

- Chức năng thêm: khi nút thêm “Thêm mới” và điền đầy đủ các thông tin theo yêu cầu thì ấn nút “Thêm mới” trong dialog. Hệ thống sẽ thêm thông tin vừa nhập vào database và hiển thị lên thông tin đã thêm trên danh sách.

- Chức năng sửa thông tin: khi ấn vào nút “Sửa” hệ thống sẽ hiển thị thông tin của giáo viên đã chọn lên dialog, người dùng chỉnh sửa các thông tin cần thay đổi sau đó click “Cập nhật”. Hệ thống sẽ cập nhật các thông tin thay đổi vào trong database và hiển thị danh sách các thay đổi mới.

### 3.4.11 Chức năng quản lý công đoàn phí

#### ❖ Mục đích

Giúp người quản lý quản lý thông tin về công đoàn phí.

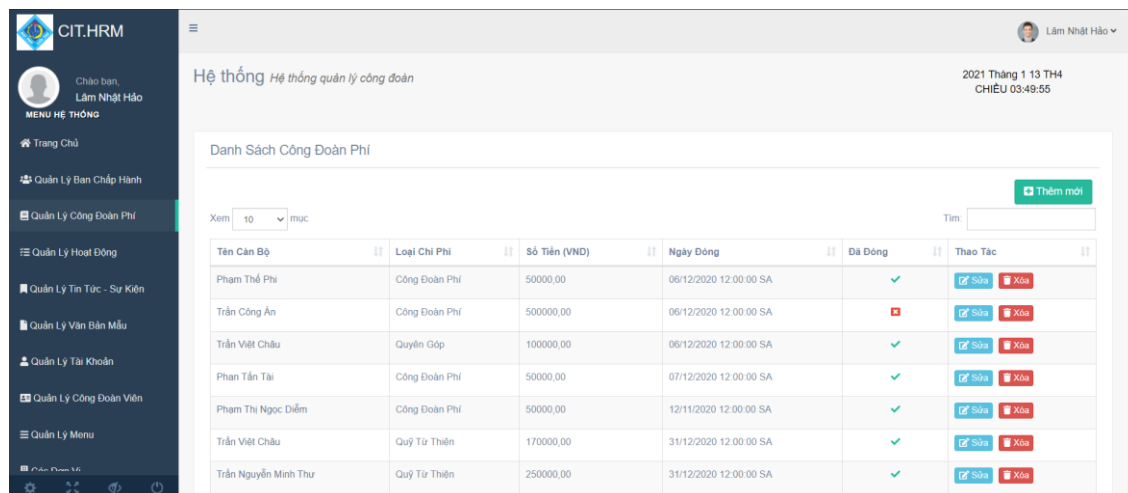
#### ❖ Phạm vi

Quản trị có quyền được phép truy cập chức năng quản lý công đoàn phí.

#### ❖ Ràng buộc

Sau khi đăng nhập vào hệ thống người sử dụng muốn sử dụng chức năng chọn “Quản lý công đoàn phí” trong ở menu bên trái.

### ❖ Giao diện chức năng



Hình 3. 34: Giao diện trang quản lý công đoàn phí

### ❖ Mô tả chức năng

Các thao tác người dùng có thể thực hiện khi sử dụng chức năng “Quản lý công đoàn phí”:

- Chức năng thêm: khi nút thêm “Thêm mới” và điền đầy đủ các thông tin theo yêu cầu thì ấn nút “Thêm mới” trong dialog. Hệ thống sẽ thêm thông tin vừa nhập vào database và hiển thị lên thông tin đã thêm trên danh sách.

- Chức năng sửa thông tin: khi ấn vào nút “Sửa” hệ thống sẽ hiển thị thông tin của lớp học đã chọn lên dialog, người dùng chỉnh sửa các thông tin cần thay đổi sau đó click “Cập nhật”. Hệ thống sẽ cập nhật các thông tin thay đổi vào trong database và hiển thị danh sách các thay đổi mới.

### 3.4.12 Chức năng quản lý đơn vị

#### ❖ Mục đích

Giúp người quản lý quản lý thông tin về đơn vị.

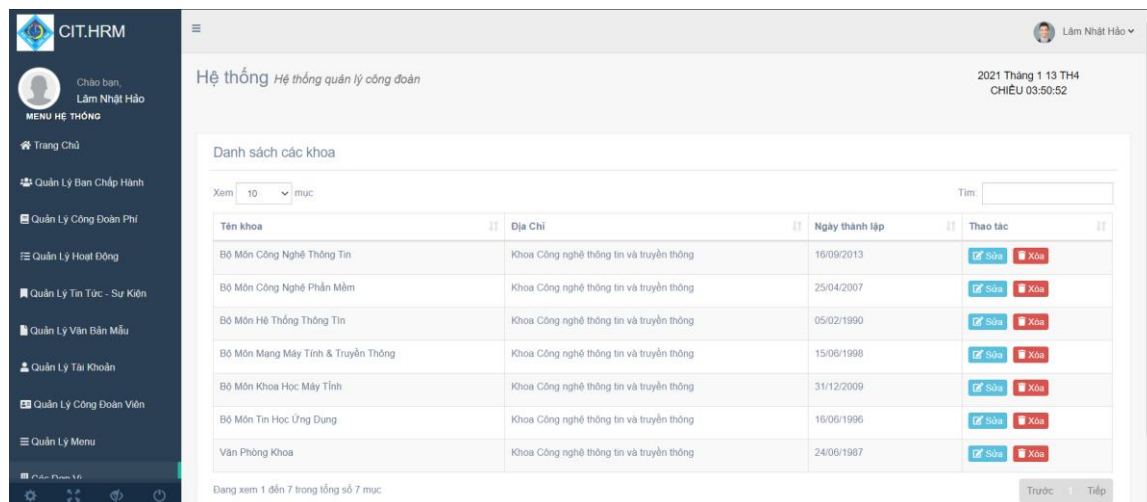
#### ❖ Phạm vi

Quản trị có quyền được phép truy cập chức năng quản lý đơn vị.

#### ❖ Ràng buộc

Sau khi đăng nhập vào hệ thống người sử dụng muốn sử dụng chức năng chọn “Các đơn vị” trong ở menu bên trái.

### ❖ Giao diện chức năng



Hình 3. 35: Giao diện trang quản lý đơn vị

### ❖ Mô tả chức năng

Các thao tác người dùng có thể thực hiện khi sử dụng chức năng “Các đơn vị”:

- Chức năng thêm: khi nút thêm “Thêm mới” và điền đầy đủ các thông tin theo yêu cầu thì ấn nút “Thêm mới” trong dialog. Hệ thống sẽ thêm thông tin vừa nhập vào database và hiển thị lên thông tin đã thêm trên danh sách.

- Chức năng sửa thông tin: khi ấn vào nút “Sửa” hệ thống sẽ hiển thị thông tin của khoa đã chọn lên dialog, người dùng chỉnh sửa các thông tin cần thay đổi sau đó click “Cập nhật”. Hệ thống sẽ cập nhật các thông tin thay đổi vào trong database và hiển thị danh sách các thay đổi mới.

### 3.4.13 Chức năng quản lý menu

#### ❖ Mục đích

Giúp người quản lý quản lý thông tin về menu.

#### ❖ Phạm vi

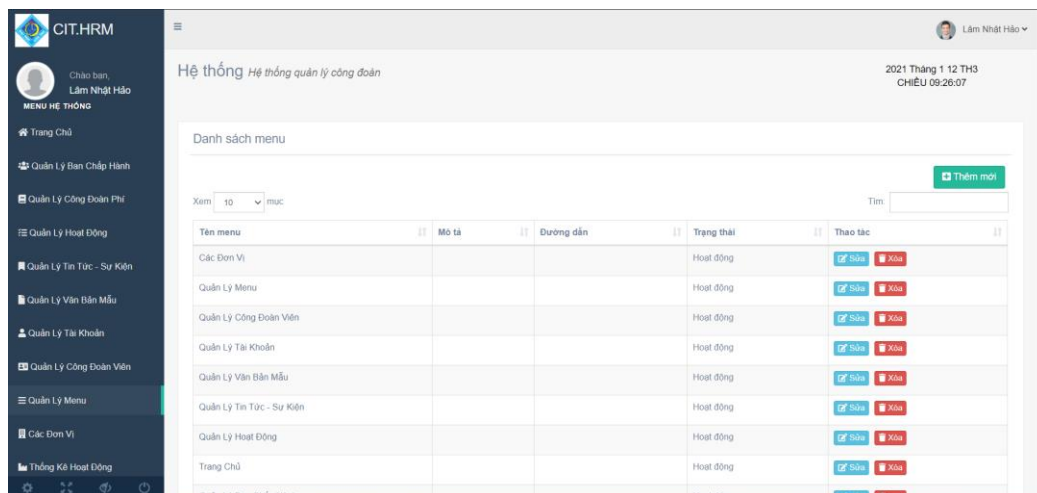
Quản trị có quyền được phép truy cập chức năng quản lý menu.

#### ❖ Ràng buộc



Sau khi đăng nhập vào hệ thống người sử dụng muốn sử dụng chức năng chọn “Quản lý menu” trong ở menu bên trái.

#### ❖ Giao diện chức năng



Hình 3. 36: Giao diện trang quản lý menu

#### ❖ Mô tả chức năng

Các thao tác người dùng có thể thực hiện khi sử dụng chức năng “Quản lý menu”:

- Chức năng thêm: khi nút thêm “Thêm mới” và điền đầy đủ các thông tin theo yêu cầu thì ấn nút “Thêm mới” trong dialog. Hệ thống sẽ thêm thông tin vừa nhập vào database và hiển thị lên thông tin đã thêm trên danh sách.

- Chức năng sửa thông tin: khi ấn vào nút “Sửa” hệ thống sẽ hiển thị thông tin của menu đã chọn lên dialog, người dùng chỉnh sửa các thông tin cần thay đổi sau đó click “Cập nhật”. Hệ thống sẽ cập nhật các thông tin thay đổi vào trong database và hiển thị danh sách các thay đổi mới.

#### 3.4.14 Chức năng quản lý văn bản mẫu

##### ❖ Mục đích

Giúp người quản lý quản lý thông tin về văn bản mẫu.

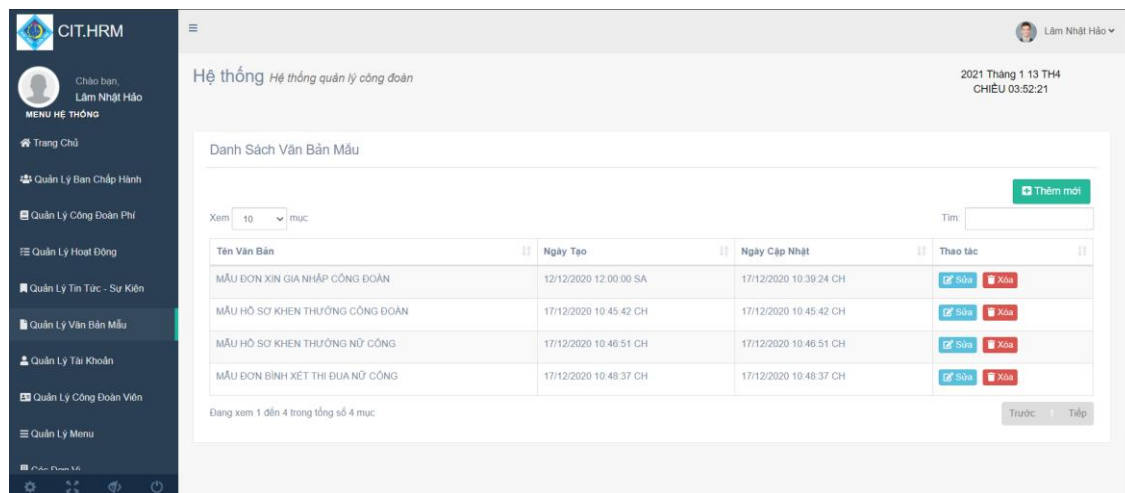
##### ❖ Phạm vi

Quản trị có quyền được phép truy cập chức năng quản lý văn bản mẫu.

##### ❖ Ràng buộc

Sau khi đăng nhập vào hệ thống người sử dụng muốn sử dụng chức năng chọn “Quản lý văn bản mẫu” trong ở menu bên trái.

### ❖ Giao diện chức năng



Hình 3. 37: Giao diện trang quản lý văn bản mẫu

### ❖ Mô tả chức năng

Các thao tác người dùng có thể thực hiện khi sử dụng chức năng “Quản lý văn bản mẫu”:

- Chức năng thêm: khi nút thêm “Thêm mới” và điền đầy đủ các thông tin theo yêu cầu thì ấn nút “Thêm mới” trong dialog. Hệ thống sẽ thêm thông tin vừa nhập vào database và hiển thị lên thông tin đã thêm trên danh sách.
- Chức năng sửa thông tin: khi ấn vào nút “Sửa” hệ thống sẽ hiển thị thông tin của công việc đã chọn lên dialog, người dùng chỉnh sửa các thông tin cần thay đổi sau đó click “Cập nhật”. Hệ thống sẽ cập nhật các thông tin thay đổi vào trong database và hiển thị danh sách các thay đổi mới.

## 3.5 Kết luận chương 3.

Trong chương 3 của báo cáo đã thực hiện việc Xây dựng Website quản lý công đoàn khoa CNTT & TT. Từ các kết quả đã thực hiện, có thể tóm tắt quá trình xây dựng hệ thống đã cài đặt như sau:

- Tiến hành được quá trình khảo sát và phân tích yêu cầu bài toán quản lý công tác giáo viên. Chỉ ra được yêu cầu của hệ thống khảo sát và mô tả chi tiết các yêu cầu.
- Đã sử dụng ULM để phân tích thiết kế hệ thống với các biểu đồ use case tổng quát, use case chi tiết cho từng chức năng. Xác định được các lớp cần quản lý tương ứng với các bảng trong cơ sở dữ liệu. Và sử dụng biểu đồ tuần tự để mô tả luồng đi của hệ thống trong từng sự kiện.
- Thiết kế được cơ sở dữ liệu từ các yêu cầu của khách hàng cần quản lý.
- Đã Thiết kế và đặc tả được giao diện của website.

## CHƯƠNG 4: KẾT LUẬN

### 4.1. Kết quả đạt được của đề tài

Sau một thời gian tìm hiểu và nghiên cứu đề tài “Xây dựng *Website quản lý công đoàn khoa CNTT & TT*” đề tài đã đạt được một số kết quả sau:

- Thông qua nghiệp vụ kết hợp với mục đích thực hiện tin học hóa, mô tả chức năng của từng hệ thống, xây dựng các biểu đồ UML và cơ sở dữ liệu thông qua công cụ Visual Diagram.
- Xây dựng chương trình thông qua kết quả thu thập được qua quá trình khảo sát, phân tích hệ thống và đặc tả giao diện

Ngoài những kết quả trên, qua quá trình thực hiện đề tài “Xây dựng *Website quản lý lịch công tác giáo viên chủ nhiệm*” em đã có được:

- Những kinh nghiệm quý báu về quy trình và nghiệp vụ quản lý website quản lý công tác.
- Trau dồi kiến thức về C# ASP.NET và MVC Framework, Entity Framework.
- Rèn luyện kỹ năng làm việc cá nhân, khả năng tư duy và sáng tạo.

### 4.2. Hạn chế của đề tài

Do thời gian nghiên cứu chưa được nhiều, bên cạnh những kết quả đạt được còn tồn tại một số vấn đề hạn chế:

- Hiện thị trong trang chủ chưa được tối ưu hóa nhất.
- Chưa triển khai hệ thống thực tế được.

### 4.3. Hướng phát triển của đề tài

Với mục đích ngày càng hoàn thiện để đáp ứng tốt nhất cho việc tin học hóa quản lý công tác trong môi trường giáo dục. Cài đặt hệ thống hoàn chỉnh hơn, để chương trình thực sự hữu ích hơn, đáp ứng những nghiệp vụ phức tạp hơn, dữ liệu lớn hơn.

## TÀI LIỆU THAM KHẢO

- [1] <https://dotnet.microsoft.com/apps/aspnet>,
- [2] <https://www.dammio.com/2016/11/04/entity-framework-phan-1-gioi-thieu-entity-framework>,
- [3] <https://www.w3schools.com/>,
- [4] <https://stackoverflow.com/>,
- [5] <https://www.favicon-generator.org/>,
- [6] <https://topdev.vn/blog/from-mvc-to-modern-web-frameworks/>