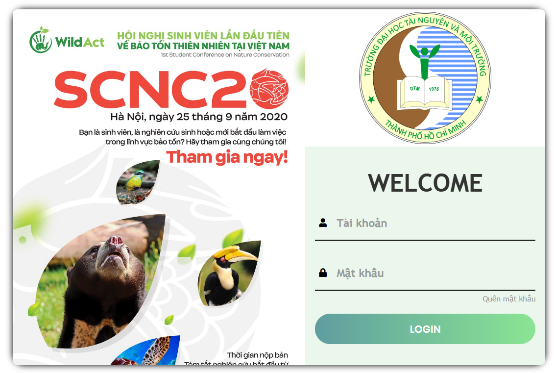
**TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG TP.HCM**

**KHOA HỆ THỐNG THÔNG TIN VÀ VIỄN THÁM**

**BÁO CÁO THỰC TẬP TỐT NGHIỆP NGÀNH CÔNG NGHỆ THÔNG TIN**

**XÂY DỰNG WEBSITE BLOG SINH VIÊN HCMUNRE**

Giảng viên hướng dẫn: **Th.S Phạm Trọng Huynh**

Sinh viên thực hiện: **Nguyễn Thanh Ý**

Lớp: **08\_ĐH\_CNPM**

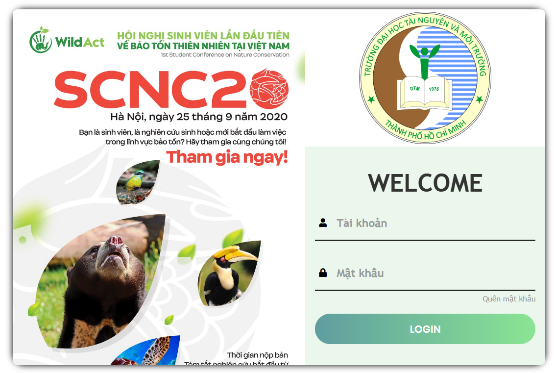
Khoá: **08**

**TP. Hồ Chí Minh, tháng … năm 2023**

**TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG TP.HCM**

**KHOA HỆ THỐNG THÔNG TIN VÀ VIỄN THÁM**

**BÁO CÁO THỰC TẬP TỐT NGHIỆP NGÀNH CÔNG NGHỆ THÔNG TIN**

**XÂY DỰNG WEBSITE BLOG SINH VIÊN HCMUNRE**

Giảng viên hướng dẫn: **Th.S Phạm Trọng Huynh**

Sinh viên thực hiện: **Nguyễn Thanh Ý**

Lớp: **08\_ĐH\_CNPM**

Khoá: **08**

**TP. Hồ Chí Minh, tháng … năm 2023**

# **MỞ ĐẦU**

Sự ra đời và phát triển của công nghệ thông tin là một trong những thành tựu vĩ đại nhất của con người. Nhịp độ phát triển công nghệ thông tin đang là một vấn đề rất được các nghành khoa học, giáo dục, kinh tế,.. quan tâm. Nó hiện hữu với một tầm vóc hết sức mạnh mẽ, to lớn và ngày càng lớn mạnh thêm. Tin học giữ vai trò đặc biệt trong hoạt động của con người, trong đời sống xã hội, trong khoa học kỹ thuật, trong sản xuất kinh doanh, trong quản lý về giáo dục, trong các hoạt động mua bán hàng hoá....

Ứng dụng website trong việc quảng cáo, cập nhật thông tin không còn xa lạ với các nghành nghề hiện nay. Lợi ích mà các chương trình quản lý mang lại khiến người ta không thể phủ nhận tính hiệu quả của nó.

Trong lĩnh vực giáo dục như hiện nay việc tin học hoá các hoạt động trao đổi thông tin , liên hệ, quản lý,…là rất cần thiết. Việc áp dụng các trang mạng xã hội lớn như Facebook, Tiktok, Intagram, LinkedIn, WhatsApp,… vào trong lĩnh vực giáo dục cũng đang được các cá nhân, tập thể triển khai rộng rãi. Tuy nhiên việc sử dụng 1 trang mạng xã hội quá lớn, quá nhiều chức năng không cần thiết, không đáp ứng được chức năng thật sự cần thiết, không tập trung vào vấn đề chính là sử dụng cho giáo dục, dẫn đến việc áp dụng vẫn chưa đạt hiệu quả cao.

Từ những yêu cầu trên việc lên ý tưởng xây dựng 1 website, một môi trường mà sinh viên, học sinh, giáo viên, giảng viên có thể tự do trao đổi, nhắn tin, đặt câu hỏi, thắc mắc về quá trình học tập tại trường, truyền tải kinh nghiệm cho nhau,… là điều thật sự cần thiết. Tuy nhiên, đó là một ý tưởng to lớn, áp dụng cho nhiều môi trường, cần sự chuẩn bị kĩ lưỡng trong thời gian dài và một mô hình phát triển đủ lớn, nhưng thời gian của em thì ngắn hạng nên em chỉ có thể thu nhỏ ý tưởng của mình và áp dụng nó cho chính ngôi trường mình đang theo học là trường Đại học Tài Nguyên và Môi Trường TP.HCM, nên em quyết định chọn dự án “ Xây dựng website Blog sinh viên HCMUNRE”.

# **LỜI CẢM ƠN**

Để hoàn thành báo cáo này, em xin gửi lời cảm ơn chân thành đến:

Ban giám hiệu Trường đại học Tài nguyên và Môi trường TP.HCM vì đã tạo điều kiện về cơ sở vật chất với hệ thống thư viện hiện đại, đa dạng các loại sách, tài liệu thuận lợi cho việc tìm kiếm, nghiên cứu thông tin.

Đơn vị Trung tâm tin học thống kê khu vực II nói chung, và anh trưởng phòng phát triển ứng dụng Lều Văn Hoan nói riêng vì đã hướng dẫn, hỗ trợ, em một cách tận tính, truyền đạt kinh nghiệm làm việc thực tế bổ ích cho em, và cho sự nghiệp của em sau này.

Giáo viên hướng dẫn thực tập là Ths. Phạm Trọng huynh đã bỏ ra thời gian chỉ dạy nhiệt tình, hướng dẫn chi tiết các điều mà em cần phải làm và học hỏi khi thực tập tại dơn vị. Ngoài ra còn hướng dẫn em hoàn thành bài cáo cáo trọn vẹn nhất.

Do chưa có nhiều kinh nghiệm làm các dự án thực tế cũng như những hạn chế về thời gian nên trong bài báo cáo chắc chắn sẽ không tránh khỏi những thiếu sót. Rất mong nhận được sự nhận xét, ý kiến đóng góp, từ phía giảng viên phản biện để sau này bài báo cáo được hoàn thiện hơn.

Lời cuối cùng, em xin kính chúc ban giám hiệu nhà trường, đơn vị thực tập, thầy Phạm Trọng Huynh, giảng viên phản biện nhiều sức khỏe, thành công và hạnh phúc.

Em xin chân thành cảm ơn!

**NHẬN XÉT**

**(Của cán bộ hướng dẫn tại đơn vị thực tập)**

XÁC NHẬN CỦA CƠ QUAN

THỦ TRƯỞNG

*(ký tên, đóng dấu)*

………….., ngày….tháng….năm……

CÁN BỘ HƯỚNG DẪN

*(ký tên)*

**NHẬN XÉT**

**(Của giảng viên hướng dẫn tại khoa)**

**Kết luận: Đồng ý hoặc không đồng ý cho sinh viên nộp báo cáo thực tập**

………….., ngày….tháng….năm……

GIẢNG VIÊN HƯỚNG DẪN

*(ký tên)*

**NHẬN XÉT**

**(Của giảng viên phản biện)**

………….., ngày….tháng….năm……

GIẢNG VIÊN PHẢN BIỆN

*(ký tên)*

**LỊCH LÀM VIỆC**

Họ và tên sinh viên: NGUYỄN THANH Ý

Đơn vị thực tập: TRUNG TÂM TIN HỌC THỐNG KÊ KHU VỰC II

Họ và tên người hướng dẫn (đơn vị thực tập): LỀU VĂN HOAN

Thời gian thực tập từ ngày 28 tháng 11 năm 2022 đến ngày 15 tháng 01 năm 2023

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tuần | Nội dung công việc được giao | Tự nhận xét về mức độ hoàn thành | Nhận xét của CB hướng dẫn | Chữ ký của CB hướng dẫn |
| **1**  Từ ngày 28/01/2022 đến ngày 04/12/2022 | Xác định đề tài, đặc tả yêu cầu, phân tích thiết kế hệ thống. | 90% |  |  |
| **2**  Từ ngày 05/12/2022 đến ngày 11/12/2022 | Xây dựng cơ sở dữ liệu, thiết kế giao diện, xác định công nghệ sử dụng. | 95% |  |  |
| **3**  Từ ngày 12/12/2022 đến ngày 18/12/2022 | Xây dựng phía back-end của hệ thống, tiến hành nâng cấp giao diện của hệ thống, xây dụng. | 90% |  |  |
| **4**  Từ ngày 19/12/2022 đến ngày 25/12/2022 | Tiếp tục xây dựng phía back-end của hệ thống, xác định các chức năng mở rộng của hệ thống, kiểm thử hệ thống. | 85% |  |  |
| **5**  Từ ngày 26/12/2022 đến ngày 01/01/2023 | Tiếp tục xây dựng các chức năng của hệ thống (các chức năng chính và chức năng mở rộng) | 95% |  |  |
| **6**  Từ ngày 02/01/2023 đến ngày 08/01/2023 | Hoàn thành các chức năng và kiểm thử hệ thống. | 90% |  |  |
| **7**  Từ ngày 09/01/2023 đến ngày 15/01/2023 | Hoàn thiện sản phẩm | 100% |  |  |

………….., ngày….tháng….năm……

**Sinh viên**

*(ký tên)*

**MỤC LỤC**

[**MỞ ĐẦU** 3](#_Toc126348454)

[**LỜI CẢM ƠN** 4](#_Toc126348455)

[**Chương 1.** **Tổng quan** 1](#_Toc126348456)

[1.1. Giới thiệu cơ quan thực tập 1](#_Toc126348457)

[1.1.1. Lịch sử hình thành và phát triển 1](#_Toc126348458)

[1.1.2. Chức năng và nhiệm vụ 1](#_Toc126348459)

[1.1.3. Cơ cấu tổ chức 2](#_Toc126348460)

[1.1.4. Các cục thống kê khu vực II 2](#_Toc126348461)

[1.2. Nội dung thực tập 3](#_Toc126348462)

[1.2.1. Nội dung 3](#_Toc126348463)

[1.2.2. Vấn đề cần giải quyết 3](#_Toc126348464)

[1.3. Phạm vi của đề tài 4](#_Toc126348465)

[**Chương 2.** **Cơ sở lý luận** 5](#_Toc126348466)

[2.1. Lý thuyết 5](#_Toc126348468)

[2.2. Kỹ thuật 10](#_Toc126348469)

[**Chương 3.** **Cài đặc thử nghiệm** 12](#_Toc126348470)

[3.1. Phương pháp nghiên cứu 12](#_Toc126348472)

[3.2. Phương pháp thực hiện 12](#_Toc126348473)

[3.2.1. Khảo sát chức năng hệ thống 12](#_Toc126348474)

[3.2.2. Phân tích hệ thống 14](#_Toc126348475)

[3.2.3. Thiết kế cơ sở dữ liệu. 16](#_Toc126348476)

[3.2.5. Xây dựng chức năng 20](#_Toc126348477)

[3.2.6. Quản lý mã nguồn 31](#_Toc126348478)

[3.3. Kết quả đạt được 31](#_Toc126348479)

[3.3.1. Đăng nhập xác thực, quên mật khẩu 32](#_Toc126348480)

[3.3.2. Bảng tin trang chủ 33](#_Toc126348481)

[3.3.3. Tạo bài viết mới 34](#_Toc126348482)

[3.3.5. Bình luận, vote realtime 36](#_Toc126348483)

[3.3.6. Nhắn tin realtime 36](#_Toc126348484)

[3.3.7. Tìm kiếm 38](#_Toc126348485)

[3.3.8. Trang cá nhân 38](#_Toc126348486)

[3.3.9. Quản lý thông tin cá nhân 39](#_Toc126348487)

[3.3.10. Quản lý bài viết 40](#_Toc126348488)

[3.3.11. Quản lý sinh viên 40](#_Toc126348489)

[3.3.12. Thống kê hoạt động 42](#_Toc126348490)

[**Chương 4.** **Kết luận và hướng phát triển** 44](#_Toc126348491)

[4.1. Kết luận 44](#_Toc126348493)

[4.2. Hướng phát triển 45](#_Toc126348494)

[**DANH MỤC TÀI LIỆU THAM KHẢO** 46](#_Toc126348496)

# **DANH MỤC CÁC BẢNG, SƠ ĐỒ, HÌNH**

[Hình 1.1: Banner Trung tâm tin học thống kê khu vực II 1](#_Toc126348101)

[Hình 2.1: Visual Studio Community 2022 5](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348102)

[Hình 2.2: Mô hình MVC 9](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348103)

[Hình 2.3: Mô hình MVC 10](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348104)

[Hình 3.1: Lượt đồ use case của hệ thống 14](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348105)

[Hình 3.2: Database diagram 17](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348106)

[Hình 3.3: Layout 18](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348107)

[Hình 3.4: Cấu trúc 1 bài viết 18](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348108)

[Hình 3.5: Cấu trúc 1 bình luận 19](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348109)

[Hình 3.6: Giao diện của chức năng nhắn tin 19](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348110)

[Hình 3.7: Cấu trúc giao diện của chức năng quản lý thông tin cá nhân 19](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348111)

[Hình 3.8: Cấu trúc giao diện chức năng quản lý sinh viên 20](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348112)

[Hình 3.9: Cấu trúc giao diện của chức năng thống kê hoạt động 20](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348113)

[Hình 3.10: Code nhận yêu cầu và xác nhận tài khoản 21](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348114)

[Hình 3.11: Code gửi Cookies 21](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348115)

[Hình 3.12: Code Ajax để gửi thông tin email vào controller 21](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348116)

[Hình 3.13: Code gửi mã xác nhận đến email 22](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348117)

[Hình 3.14: Đoạn code Javascript để hiển thị hình ảnh trước khi đăng 22](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348118)

[Hình 3.15: Code mẫu để lưu một danh sách các hình ảnh của bài viết 23](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348119)

[Hình 3.16: Code JavaScript để mở kết nối giữa client và server 23](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348120)

[Hình 3.17: Hàm JavaScript để bắt sự kiện khi người dùng comment 23](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348121)

[Hình 3.18: Hàm phía server xử lí dữ liệu bình luận từ client gọi về 24](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348122)

[Hình 3.19: Code JavaScript nhận kết quả từ server và hiển thị lên cho tất cả người dùng thấy. 24](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348123)

[Hình 3.20: Hàm bắt sự kiện like và hàm hiển thị số lượng like cho người dùng thấy khi client nhận dữ liệu từ server 24](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348124)

[Hình 3.21: Hàm phía server xử lí dữ liệu vote từ client gọi về 25](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348125)

[Hình 3.22: Hàm hiển thị lên khi cho người 2 người trong đoạn hội thoại thấy tin nhắn khi client nhận dữ liệu từ server 25](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348126)

[Hình 3.23: Code hiển hiện thị thêm tin nhắn. 26](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348127)

[Hình 3.24: Thuật toán tìm kiếm 26](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348128)

[Hình 3.25: Bộ thư viện để xử lí vấn đề datetime picker 27](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348129)

[Hình 3.26: Giao diện khi sài bộ thư viện datetime picker 27](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348130)

[Hình 3.27: Action tải mẫu Excel 28](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348131)

[Hình 3.28: Action thực hiện thêm sinh viên vào cơ sở dữ liệu với file Excel 28](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348132)

[Hình 3.29: Code JavaScript để sử dụng thư viện Datatables. 29](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348133)

[Hình 3.30: Code khai báo thư viện và cấu hình thẻ canvas thành một đồ thị 29](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348134)

[Hình 3.31: Code C# để lấy được mật độ hoạt động của hệ thống trong vòng 7 ngày. 30](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348135)

[Hình 3.32: Code xử lý khi số lượng cột thống kê quá nhiều(1) 30](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348136)

[Hình 3.33: Code xử lý khi số lượng cột thống kê quá nhiều(2) 31](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348137)

[Hình 3.34: Khai báo dịch vụ Authentication 31](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348138)

[Hình 3.35: Giao diện đăng nhập 32](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348139)

[Hình 3.36: Chức năng quên mật khẩu (1) 32](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348140)

[Hình 3.37: Chức năng quên mật khẩu (2) 32](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348141)

[Hình 3.38: Chức năng quên mật khẩu (3) 32](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348142)

[Hình 3.39: Giao diện trang chủ 33](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348143)

[Hình 3.40: Giao diện đầy đủ của một bài viết 33](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348144)

[Hình 3.41: Cấu trúc đầy đủ của phần bình luận 33](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348145)

[Hình 3.42: Báo cáo và ghim một bài viết 34](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348146)

[Hình 3.43: Tạo mới một bài viết 34](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348147)

[Hình 3.44: Thông báo đăn bài thành công 34](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348148)

[Hình 3.45: Tổ chức lưu trữ tệp của các bài đăng 35](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348149)

[Hình 3.46: Giao diện bài đăng cá nhân 35](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348150)

[Hình 3.47: Trạng thái và chức năng xoá bài viết cá nhân 35](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348151)

[Hình 3.48: Thực hiện bình luận ở 2 tài khoản khác nhau 36](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348152)

[Hình 3.49: Bình luận được sắp xếp theo số lượng vote từ lớn đến bé 36](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348153)

[Hình 3.50: Giao diện chính của chức năng nhắn tin 37](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348154)

[Hình 3.51: Nhắn tin trong thời gian thực 37](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348155)

[Hình 3.52: Thông báo có tin nhắn mới 37](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348156)

[Hình 3.53: Giao diện tìm kiếm 38](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348157)

[Hình 3.54: Giao diện trang cá nhân của một người dùng 38](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348158)

[Hình 3.55: Giao diện quản lý thông tin cá nhân 39](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348159)

[Hình 3.56: Chỉnh sửa thông tin cá nhân 39](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348160)

[Hình 3.57: Đổi mật khẩu 39](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348161)

[Hình 3.58: Đổi ảnh đại diện 39](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348162)

[Hình 3.59: Giao diện quản lý bài viết 40](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348163)

[Hình 3.60: Trạng thái, duyệt bài, xoá bài và số lượng tố cáo của bài viết 40](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348164)

[Hình 3.61: Giao diện quản lý sinh viên 41](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348165)

[Hình 3.62: Giao diện thêm sinh viên 41](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348166)

[Hình 3.63: Giao diện chỉnh sửa thông tin sinh viên 41](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348167)

[Hình 3.66: Giao diện chức năng thống kê hoạt động 42](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348168)

[Hình 3.64: Tìm kiếm trên table 42](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348169)

[Hình 3.65: Mẫu Excel 42](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348170)

[Hình 3.67: Thống kê theo tháng 43](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348171)

[Hình 3.68: Xuất báo cáo dưới dạng file Excel 43](file:///C:\Users\ASUS\Downloads\ThucTapTotNghiep.docx#_Toc126348172)

# **DANH MỤC CÁC CỤM TỪ VIẾT TẮT**

**CNTT**: Công nghệ thông tin.

**Realtime**: Thời gian thực.

**TP.HCM**: Thành phố Hồ Chí Minh.

**Database**: Cơ sở dữ liệu.

**HCMUNRE**: Trường đại học Tài nguyên và Môi trường thành phố Hồ Chí Minh.

**Code:** Mã

**Source code:** Mã nguồn

**Format:** Định dạng

**Request:** Yêu cầu

**Post request:** Là một phương thức HTTP để gửi yêu cầu kèm theo thông tin

**Blog:** Một nơi mà một nhóm người sử dụng blog để chia sẻ kiến thức, kinh nghiệm trong học tập, cuộc sống của mình.

**Table:**  Hiển thị danh sách dữ liệu dưới dạng bảng.

# **Tổng quan**

## Giới thiệu cơ quan thực tập

### Lịch sử hình thành và phát triển

Trung tâm Tin học Thống kê Khu vực II - tên giao dịch COSIS - tiền thân là Công ty Dịch vụ Thông tin Thống kê được thành lập từ năm 1988.



Hình .: Banner Trung tâm tin học thống kê khu vực II

Năm 1992 - đổi thành Trung tâm Xử lý Thông tin Thống Kê thuộc Cục thống kê TP.Hồ Chí Minh.

Năm 2004 - đổi thành Trung tâm Tin học Thống kê Khu vực II, đơn vị sự nghiệp trực thuộc Tổng cục Thống kê - Bộ Kế hoạch & Đầu tư.

Trụ sở tại : 54A Nơ Trang Long, P.14, Q.Bình Thạnh, TP.HCM - Điện thoại: 355 154 33 - Fax: 355 154 32 - Website: www.cosis.vn

### Chức năng và nhiệm vụ

Quản lý, vận hành hệ thống hạ tầng CNTT– truyền thông tại khu vực II của Tổng cục Thống kê, bao gồm:

* Trung tâm máy chủ khu vực II;
* Trung tâm sao lưu dữ liệu của Tổng cục Thống kê tại TPHCM;
* Hệ thống thư điện tử khu vực II của Tổng cục Thống kê;
* Thực hiện sao lưu định kỳ dữ liệu của Tổng cục Thống kê;
* Thực hiện bảo trì hệ thống hạ tầng CNTT tại khu vực II của Tổng cục Thống kê.

Quản lý, vận hành, bảo trì hệ thống mạng của đơn vị; bảo đảm kết nối thông suốt hệ thống mạng các đơn vị được phân công phụ trách với hệ thống mạng của cơ quan Tổng cục Thống kê.

Xử lý dữ liệu điều tra, tổng điều tra và báo cáo thống kê; xây dựng cơ sở dữ liệu, kho dữ liệu thống kê.

Phát triển và hướng dẫn các phần mềm ứng dụng; phối hợp lựa chọn giải pháp công nghệ mới áp dụng trong ngành Thống kê.

Thực hiện bảo trì và hỗ trợ, tư vấn kỹ thuật về vận hành đối với hệ thống mạng, thiết bị tin học của các đơn vị khu vực II.

Thực hiện đào tạo, bồi dưỡng về CNTT.

Hợp tác, liên kết với với các tổ chức, cơ quan, đơn vị trong nước và ngoài nước nghiên cứu và phát triển CNTT vào công tác thống kê.

Thực hiện công nghiệp phần mềm, dịch vụ về lĩnh vực CNTT. Cung cấp các dịch vụ:

* Xây dựng phần mềm.
* Xử lý số liệu.
* Thiết kế và thi công hệ thống mạng.
* Đào tạo tin học.

### Cơ cấu tổ chức

Gồm có Ban Giám đốc và 6 phòng:

* Phòng Tổ chức – Hành chánh.
* Phòng Kế hoạch – Kinh doanh.
* Phòng Phát triển ứng dụng.
* Phòng Xử lý thông tin.
* Phòng Kỹ thuật và quản trị hệ thống.
* Phòng Đào tạo.

### Các cục thống kê khu vực II

* Cục thống kê Tỉnh Bình Thuận
* Cục thống kê Tỉnh Tây Ninh
* Cục thống kê Tỉnh Trà vinh
* Cục thống kê Tỉnh Đồng Tháp
* Cục thống kê Tỉnh Bà rịa-Vũng Tàu
* Cục thống kê TP.Hồ Chí Minh
* Cục thống kê Tỉnh Bến Tre
* Cục thống kê Tỉnh An Giang
* Cục thống kê Tỉnh Đồng Nai
* Cục thống kê Tỉnh Long An
* Cục thống kê Tỉnh Sóc Trăng
* Cục thống kê Tỉnh Kiên Giang
* Cục thống kê Tỉnh Bình Dương
* Cục thống kê Tỉnh Tiền Giang
* Cục thống kê TP.Cần Thơ
* Cục thống kê Tỉnh Bạc Liêu
* Cục thống kê Tỉnh Bình Phước
* Cục thống kê Tỉnh Vĩnh Long
* Cục thống kê Tỉnh Hậu Giang
* Cục thống kê Tỉnh Cà Mau

## Nội dung thực tập

### Nội dung

Trong quá trình thực tập tại công ty, em được tìm hiểu và thực hành mảng lập trình  
website với C# cụ thể là ASP.NET trên nền tảng Visual Studio cụ thể như:

* Học hỏi cách tổ chức, phát triển và kĩ thuật triển khai một dự án thực tế tại công ty là như thế nào. Nắm được các kĩ năng cần thiết khi tham gia xây dựng và thiết kế một dự án thực tế. Kĩ năng phân tích yêu cầu của khách hàng, nắm bắt được tất cả các yêu cầu có thể có của khách hàng.
* Học hỏi kinh nghiệm, kiến thức, thái độ làm việc cách quản lý nhân sự từ trưởng phòng phát triển ứng dụng để nâng cao kĩ năng, tay nghề của bản thân qua các nghiệm vụ, yêu cầu được giao.
* Sử dụng ngôn ngữ C# về mảng xây dựng website với asp.net core mvc 3.1.
* Từ những kiến thức học hỏi được từ quá trình thực tập, phải áp dụng và hoàn thành được mục tiêu dự án đề ra từ đầu khi tham gia thực tập.

### Vấn đề cần giải quyết

Với dự án là xây dựng website blog sinh viên HCMUNRE thì đây là một ý tưởng không mới, tuy nhiên các blog như kenhsinhvien.com, sinhviendanang.com, blogsinhvien123.blogspot.com,…thì lại thiên hướng về kênh thông tin, tin tức, việc làm, không đáp ứng được nhu cầu đề ra là mang tính cục bộ trong phạm vi 1 ngôi trường hay một trung tâm giáo dục. Vì vậy vấn đề ở đây là một vấn đề không mới, nhưng ý tưởng lại hoàn toàn mới để đáp ứng được nhu cầu trao đổi thông tin, kiến thức, tài liệu, kinh nghiệm, thắc mắc giữa sinh viên – sinh viên, sinh viên – giáo viên, sinh viên – người quản trị một cách dễ dàng nhanh chóng, tiện lợi. Ngoài ra còn xây dựng một hệ thống đánh giá mức độ, tần xuất hoạt động của từng cá nhân từ đó có thể dễ dàng dựa vào đó để đánh giá được điểm rèn luyện hoặc độ tích cực.

Ở dự án này có nhiều vấn đề cần được giải quyết, trong đó việc đáp ứng được tốc độ xử lí của hệ thống khi triển khai trên dữ liệu lớn là rất quan trọng, ngoài ra việc đáp ứng được tính realtime của hệ thống, chức năng thống kê thông tin hoạt động của tất cả các người dùng và xuất ra báo cáo dưới dạng file excel. Đó là các vấn đề lớn cần được giải quyết của dự án, ngoài ra còn có các vấn đề nhỏ khác như: đáp ứng được các nhu cầu cơ bản như đăng bài gồm nhiều file đính kèm, có thể tải ảnh, xem video, tải file. Có thể tự do bình luận, đồng ý với các câu trả lời chính xác hơn để người đặc câu hỏi có thể biết được chính xác câu trả lời. Ngoài ra còn có các chức năng thay đổi thông tin cá nhân, đổi hình đại diện tuỳ thích, tìm kiếm người dùng, bài viết,…Nhắn tin trao đổi trực tiếp với các người dùng cần liên hệ.

## Phạm vi của đề tài

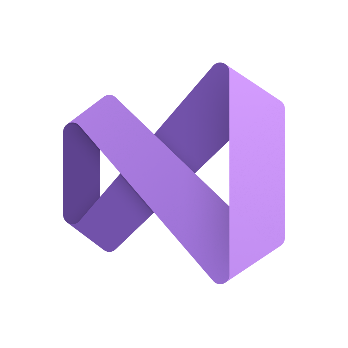
Trong đồ án của em dựa trên sự góp ý cũng như sự hướng dẫn của giảng viên hướng dẫn, cán bộ hướng dẫn và tham khảo một số website blog của sinh viên có phạm vi như sau:

* Sử dụng nền tảng Visual Studio 2022 để xây dựng giao diện và các chức năng cho người dụng (Quản lý, giáo viên, sinh viên).
* Đối tượng sử dụng là sinh viên, học sinh của các ngôi trường cấp bậc trung học cơ sở, trung học phổ thông, đại học, cao đẳng, các trung tâm giáo dục thưởng xuyên, các trung tâm giáo dục trên toàn quốc.
* Là hệ thống website mang tính cục bộ của mỗi đơn vị, dữ liệu chỉ mang tính cục bộ, không liên kết giữa các trường với nhau.

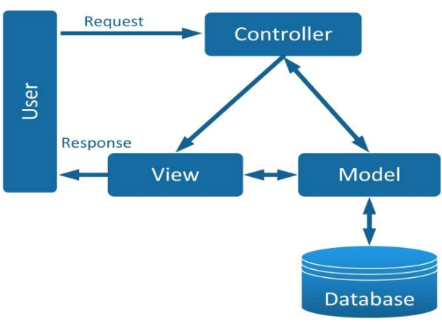
1. **Cơ sở lý luận**

## Lý thuyết

Để xây dựng một website đáp ứng nhu cầu đề ra một cách hoàn chỉnh cần phải hiểu rõ các lý thuyết cơ bản của ngôn ngữ lập trình C# trên nền ASP.NET CORE MVC 3.1, đây là những kiến thức lý thuyết em đã tìm hiểu được:

* **Visual Studio Community 2022:** Visual studio là một phần mềm hỗ trợ đắc lực hỗ trợ công việc lập trình website. Công cụ này được tạo lên và thuộc quyền sở hữu của ông lớn công nghệ Microsoft. Đây là một phiên bản miễn phí tuy nhiên nó cũng đã có đầy đủ các chức năng cần thiết để xây dựng một website vì vậy nó là là một lựa chọn tuyệt vời dành cho các lập trình viên. Đặc biệt là khi công cụ này còn sở hữu những ưu điểm vượt trội hơn hẳn bất kỳ IDE nào khác như:

Hình 2.1: Visual Studio Community 2022

* Ít dung lượng
* Tính năng phong phú, đa dạng và mạnh mẽ
* Hỗ trợ nhiều nền tảng khác nhau (Linux, Windows, Mac)
* Hỗ trợ đa ngôn ngữ (HTML, CSS, C#, F#, C/C++, JSON, JavaScript,…)
* Tính năng Intellisense chuyên nghiệp
* Giao diện thân thiện dễ sử dụng
* Cập nhật đa dạng phiên bản cho nhiều đối tượng sử dụng khác nhau,…
* **Theme:** Đây là một tập hợp các tập tin có chứa hình ảnh, âm thanh, code,.. giúp hình thành nên giao diện của trang web.
* **Layout:** Dạng xem bố cục cho phép chúng ta xác định một mẫu trang chung, mẫu này có thể được kế thừa trong nhiều dạng xem để cung cấp giao diện nhất quán trong nhiều trang của website. Chế độ xem bố cục loại bỏ CODE trùng lặp và nâng cao tốc độ phát triển cũng như bảo trì dễ dàng.
* **Hosting và Domain**: hosting đây là **nơi lưu trữ toàn bộ dữ liệu cho website của bạn**. Như kiểu website của bạn là 1 ngôi nhà thì hosting là mảnh đất. tên miền như địa chỉ nhà, có đuôi .com, .net, .org, .tk, .ga …..
* **HTML** là xây dựng cấu trúc siêu văn bản trên một website, hoặc khai báo các tập tin kỹ thuật số (media) như hình ảnh, video, nhạc. HTML thường được dùng để phân chia các đoạn văn, heading, links, blockquotes,…
* **CSS** sẽ giúp webmaster xác định styles và định nghĩa nhiều loại nội dung của website. Css kết hợp với ngôn ngữ markup HTML để tạo phong cách cho trang web
* **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… là một bộ sưu tập miễn phí của các mã nguồn mở và công cụ dùng để tạo ra một mẫu webiste 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.
* **C#:** (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000. C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java. C# với sự hỗ trợ mạnh mẽ của .NET Framework giúp cho việc tạo một ứng dụng Windows Forms hay WPF (Windows Presentation Foundation), phát triển game, ứng dụng Web, ứng dụng Mobile trở nên rất dễ dàng.
* **JavaScript:** JavaScript là ngôn ngữ lập trình được nhà phát triển sử dụng để tạo trang web tương tác. Từ làm mới bảng tin trên trang mạng xã hội đến hiển thị hình ảnh động và bản đồ tương tác, các chức năng của JavaScript có thể cải thiện trải nghiệm người dùng của trang web. Là ngôn ngữ kịch bản phía máy khách, JavaScript là một trong những công nghệ cốt lõi của World Wide Web. Ví dụ: khi duyệt internet, bất cứ khi nào bạn thấy quảng cáo quay vòng dạng hình ảnh, menu thả xuống nhấp để hiển thị hoặc màu sắc phần tử thay đổi động trên trang web cũng chính là lúc bạn thấy các hiệu ứng của JavaScript.
* **Jquery:** jQuery không phải là một ngôn ngữ lập trình riêng biệt mà hoạt động liên kết với JavaScript. Với jQuery, bạn có thể làm được nhiều việc hơn mà lại tốn ít công sức hơn. jQuery cung cấp các API giúp việc duyệt tài liệu HTML, hoạt ảnh, xử lý sự kiện và thao tác AJAX đơn giản hơn. jQuery hoạt động tốt trên nhiều loại trình duyệt khác nhau. Một trong những đối thủ nặng ký của jQuery đó là JS Framework. Các module phổ biến của jQuery bao gồm:
* Ajax – xử lý Ajax
* Atributes – Xử lý các thuộc tính của đối tượng HTML
* Effect – xử lý hiệu ứng
* Event – xử lý sự kiện
* Form – xử lý sự kiện liên quan tới form
* DOM – xử lý Data Object Model
* Selector – xử lý luồng lách giữa các đối tượng HTML
* **SignalR:** là một thư viện cho các lập trình viên Asp.Net đơn giản hóa quá trình thêm chức năng web realtime trong phát triển ứng dụng. SignalR có thể sử dụng trong bất kì chức năng web real-time nào. Trong đó ứng dụng chat trên web là một ví dụ điển hình. SignalR cung cấp một API đơn giản cho việc tạo server-to-client remote procedure call (RPC) để gọi những hàm javascript trong trình duyệt (và những nền tảng khác) từ code .Net của server-side. SignalR cũng bao gồm API cho việc quản lý kết nối (connect và disconnect events) và những kết nối nhóm. SignalR xử lý quản lý kết nối một cách tự động, và cho bạn truyền đi thông điệp tới tất cả các client đã được kết nối một cách đồng loạt, giống như một chat room. Bạn cũng có thể gửi những thông điệp tới những client được xác định. Kết nối giữa client và server là liên tục, không giống như kết nối HTTP cổ điển, cái mà sẽ thành lập lại kết nối cho mỗi lần giao tiếp. SignalR hỗ trợ chức năng "server push", trong server code có thể gọi tới client code trong trình duyệt bởi "Remote Procedure Calls" (RPC), hơn là sử dụng Service Bus, SQL Server hay Redis. SignalR là mã nguồn mở, có thể truy cập thông qua GitHub.
* **SQL Server:** hay còn được gọi là Microsoft SQL Server, nó từ viết tắt của MS SQL Server. Được phát triển bởi Microsoft và nó được sử dụng để có thể dễ dàng lưu trữ cho những dữ liệu dựa theo tiêu chuẩn RDBMS. Từ đó, người ta sẽ lưu trữ dữ liệu dựa vào tiêu chuẩn RDBMS và nó cũng là một trong những hệ quản trị cơ sở dữ liệu dạng quan hệ đối tượng. SQL Server có khả năng cung cấp đầy đủ các công cụ cho việc quản lý từ giao diện GUI đến sử dụng ngôn ngữ cho việc truy vấn SQL. Điểm mạnh của SQL điểm mạnh của nó là có nhiều nền tảng được kết hợp cùng như: ASP.NET, C# để xây dựng Winform cũng chính nó có khả năng hoạt động độc lập. Tuy nhiên, SQL Server thường đi kèm với việc thực hiện riêng các ngôn ngữ SQL, T-SQL,...
* **Enity FrameWork:** Microsoft đã cung cấp một O/RM framework đặt tên là “Entity Framework” để tự động hóa các hoạt động liên quan đến CSDL giúp bạn xây dựng ứng dụng đơn giản hơn. Entity Framework là một khung ORM(là một công cụ tự động tạo ra các Objects trong ứng dụng từ tables, views, store procedures của một Relational Database.) mã nguồn mở cho các ứng dụng .NET được Microsoft hỗ trợ, là 1 phần của .NET Framework, là một Open source ORM Framework. Emntity Framework giúp các nhà phát triển Web tương tác với dữ liệu quan hệ theo phương pháp hướng đối tượng với ít mã hơn so với các ứng dụng truyền thống. Lợi ích lớn nhất của nó là giúp lập trình viên giảm thiểu việc lập trình mã nguồn để thực hiện truy cập và tương tác với cơ sở dữ liệu.
* **Git và GitHub:** GitHub là một dịch vụ nổi tiếng cung cấp kho lưu trữ mã nguồn Git cho các dự án phần mềm. Github có đầy đủ những tính năng của Git, ngoài ra nó còn bổ sung những tính năng về social để các developer tương tác với nhau. Vài thông tin về GIT: Là công cụ giúp quản lý source code tổ chức theo dạng dữ liệu phân tán, Giúp đồng bộ source code của team lên 1 server, Hỗ trợ các thao tác kiểm tra source code trong quá trình làm việc (diff, check modifications, show history, merge source, …). GitHub có 2 phiên bản: miễn phí và trả phí. Với phiên bản có phí thường được các doanh nghiệp sử dụng để tăng khả năng quản lý team cũng như phân quyền bảo mật dự án. Còn lại thì phần lớn chúng ta đều sử dụng Github với tài khoản miễn phí để lưu trữ source code. Github cung cấp các tính năng social networking như feeds, followers, và network graph để các developer học hỏi kinh nghiệm của nhau thông qua lịch sử commit. Nếu một comment để mô tả và giải thích một đoạn code. Thì với Github, commit message chính là phần mô tả hành động mà bạn thực hiện trên source code.
* **Mô hình MVC:** Mô hình Model-View-Controller (MVC) là một mẫu kiến ​​trúc phân tách một ứng dụng thành ba thành phần logic chính Model, View và Controller. Do đó viết tắt MVC. Mỗi thành phần kiến ​​trúc được xây dựng để xử lý khía cạnh phát triển cụ thể của một ứng dụng. MVC tách lớp logic nghiệp vụ và lớp hiển thị ra riêng biệt.

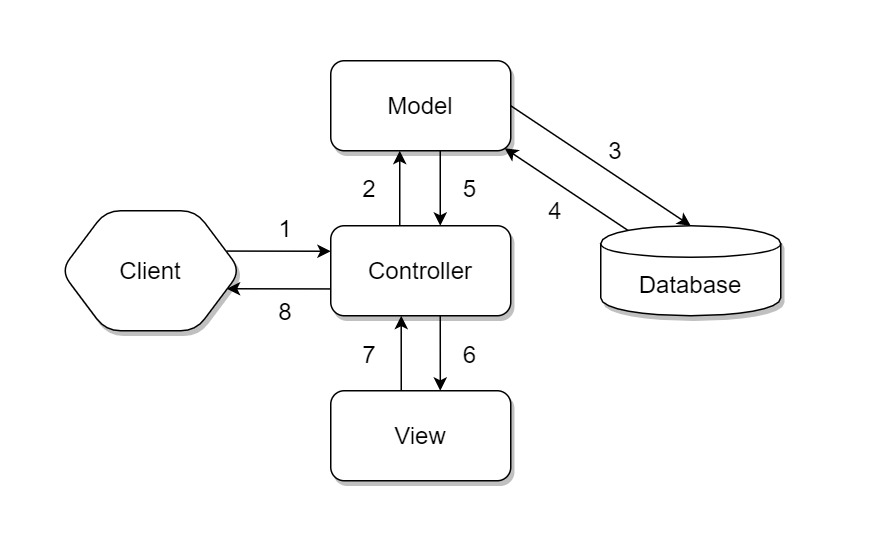
Hình 2.2: Mô hình MVC

* **ASP.NET CORE:** ASP.NET Core là một phiên bản mới của ASP.NET, được phát hành bởi Microsoft và là một mã nguồn mở trên GitHub. ASP.NET Core được sử dụng để phát triển khuôn khổ website và có thể thích ứng với nhiều trình duyệt khác nhau như Windows, Mac hoặc Linux kể cả trên nền tảng MVC. Những lợi ích khi sử dụng ASP.NET Core:
* Người dùng có thể tạo ra những ứng dụng web có thể thực hiện testing theo mô hình MVC.
* Xây dựng HTTP services hỗ trợ nhiều định dạng và đầy đủ những hỗ trợ cho nội dung của dữ liệu trả về.
* Razor cung cấp ngôn ngữ tạo Views hiệu quả.
* Tag Helper cho phép code server side tham gia vào quá trình tạo và render phần tử HTML.
* Model Binding có thể tự động ánh xạ dữ liệu từ HTTP request tới những tham số của method action.
* Model validation thực hiện validate client và server một cách tự động.
* ASP.NET Core được thiết kế tích hợp với nhiều client side frameworks một cách liên tục bao gồm AngularJS, Bootstrap và KnockoutJS.
* Là khung mẫu và gọn gàng hơn kiến trúc nhiều tầng.
* Là một khung công tác với mã nguồn mở.
* Dễ dàng tạo ứng dụng ASP.NET đa nền tảng trên Windows, Mac và Linux.
* Cấu hình là môi trường sẵn sàng cho đám mây.
* Khả năng lưu trữ trên: IIS, HTTP.sys, Kestrel, Nginx, Apache và Docker.

## Kỹ thuật

Với những kiến thức lý thuyết trên thì các công cụ và kỹ thuật được sử dụng trong dự án của em là:

* Phần mềm: Visual Studio Community 2022, Visual Studio Code, Microsoft Sql Server Managerment Studio 18
* Nền tảng: .NET Framework
* Công nghệ: ASP.NET CORE MVC 3.1
* Công cụ: Quản lý mã nguồn với GitHub, Google Chrome, Microsoft Edge.
* Thư viện: Enity FrameWorkCore, EPPlus, MailKit, AspNetCore Authentication, SignalR, Jquery, Bootstrap, Datatable, font-awesome, Bootstrap Datetimepicker, ChartJS.
  1. **Xây dựng mô hình ứng dụng**

Mô hình ứng dụng để xây dựng hệ thống dự án em dựa trên mô hình ba lớp MVC  
(Model – View -Controller).

Hình 2.3: Mô hình MVC

Dựa vào lý thuyết đã được tìm hiểu ở trên giờ ta sẽ đi vào tìm hiểu chi tiết cách tổ chức và luồng hoạt động của mô hình và ưu nhược điểm của nó.

* Đầu tiên, client sẽ gửi một request tới server thông qua Controller.
* Controller tiếp nhận request, và xử lý luồng đi tiếp theo của yêu cầu. Trong trường hợp chỉ chuyển từ trang này sang trang khác mà không yêu cầu gửi và nhận dữ liệu thì lúc này Controller sẽ redirect(chuyển hướng) lại cho browser và kết thúc luồng. Lúc này luồng xử lý sẽ đi theo thứ tự 1 → 6 → 7 → 8.
* Nếu request của người dùng yêu cầu phải xử lý dữ liệu thì Controller gọi xuống Model để lấy dữ liệu. Lúc này Model sẽ tương tác với database để lấy dữ liệu, dữ liệu được Model gửi về Controller, Controller gọi đến View phù hợp với request kèm theo dữ liệu cho View, View sẽ lắp dữ liệu tương ứng vào HTML và gửi lại một HTML cho Controller sau khi thực hiện xong nhiệm vụ của mình. Cuối cùng Controller sẽ trả kết quả về Browser. Lúc này luồng xử lý sẽ đi từ 1 → 2 → 3 → 4 → 5 → 6 → 7 → 8.
* Nếu request của người dùng chỉ yêu cầu dữ liệu nhưng không chuyển trang thì thay vì Controller sẽ trả dữ liệu về cho View thì Controller sẽ respone lại cho trình duyệt thông qua API, dữ liệu trả về thường sẽ là dạng JSON. Luồng xử lý lúc này sẽ là 1 → 2 → 3 → 4 → 5 → 8.

Ưu điểm của MVC

* Tạo mô hình chuẩn cho dự án, giúp cho việc tiếp cận với ứng dụng dễ dàng hơn.
* Trình tự xử lý rõ ràng, nhiệm vụ riêng biệt, độc lập với các thành phần khác và các thành phần có thể tương tác được với nhau.
* Source code được tách bạch rõ ràng nên việc bảo trì code sẽ dễ dàng hơn.

Nhược điểm của MVC

* Model sẽ phải xử lý rất nhiều nếu dữ liệu lớn và phức tạp.
* Đối với cái mô hình ứng dụng lớn và có độ phức tạp cao thì có lẽ MVC không còn khả dụng.

1. **Cài đặc thử nghiệm**

## Phương pháp nghiên cứu

Qua tìm hiểu và khảo sát các website blog sinh viên của các website trực tuyến trên mạng internet hiện nay như kenhsinhvienvn.com, sinhviendanang.com, gocnhosinhvien.edu.vn,…thì em đã rút ra được một số vấn đề như sau:

* Các website này đều cho phép mọi người dùng tham gia.
* Tất cả đều cho phép đăng các bài thảo luận. Truyền bá các thông tin giáo dục. Chia sẻ đời sống, việc làm sinh viên.
* Mang tính cộng đồng, liên kết với các trang mạng xã hội lớn.

Nhưng đối với dự án của em phạm vi sử dụng chỉ được áp dụng cục bộ, mang tính riêng biệt, với các đối tượng người dùng chỉ thuộc đơn vị sở hữu. Từ đó việc triển khai xây dựng ứng dụng sẽ được diễn ra một cách nhanh chóng, khó phát sinh và sửa đổi dễ dàng.

## Phương pháp thực hiện

Xây dựng website Blog sinh viên HCMUNRE bằng ASP.NET CORE MVC 3.1 qua quá trình khảo sát ban đầu em đã xác định yêu cầu đặt ra và phát triển trang web qua các bước như sau:

### Khảo sát chức năng hệ thống

Website Blog sinh viên HCMUNRE được sử dụng ngôn ngữ lập trình C# với nền tảng công nghệ là ASP.NET CORE 3.1 và hệ quản trị cơ sở dữ liệu SQL Server cho phép người dùng thực hiện các nhóm chức năng sau:

* Nhóm chức năng đăng nhập, bao gồm đăng nhập và quên mật khẩu.
* Nhóm chức năng xem bài viết: xem các bài viết đã đăng, tố cáo bài viết.
* Nhóm chức năng đăng bài viết: đăng bài với nội dung bài đăng, có thể đính kèm các tệp hình ảnh, file và video.
* Nhóm chức năng bình luận: Bình luận thời gian thực để trả lời một bài viết, xoá bình luận cá nhân, đồng ý với bình luận có tính chính xác ở thời gian thực.
* Nhóm chức năng quản lý bài đăng cá nhân: xem lại tất cả các bài viết cá nhân, xoá bài viết cá nhân.
* Nhóm chức năng nhắn tin: trò chuyện ở thời gian thực, tạo một đoạn hội thoại mới.
* Nhóm chức năng tìm kiếm thông tin: tìm tất cả các bài viết hoặc người dùng có liên quan tới từ khoá tìm kiếm, liên hệ với người dùng cần tìm.
* Nhóm chức năng quản lý thông tin cá nhân: tuỳ chỉnh thông tin cá nhân, đổi ảnh đại diện, đổi lại mật khẩu.
* Nhóm chứ năng xem trang cá nhân: xem được các thông tin cơ bản của người dùng, xem các bài viết của người dùng đó.
* Nhóm chức năng quản lý bài viết: Kiểm duyệt bài viết, xoá bài viết, xoá bình luận.
* Nhóm chức năng quản lý sinh viên: Thêm một sinh viên, thêm sinh viên từ file Excel, tải mẫu file Excel, chỉnh sửa thông tin sinh viên, vô hiệu hoá tài khoản, tìm kiếm sinh viên.
* Nhóm chức năng thống kê hoạt động: hiển thị số lượng bài đăng, bình luận, vote, trong một khoảng thời gian, hiển thị trong khoảng thời gian tìm kiếm, tải bản báo cáo tần xuất hoạt động của tất cả các sinh viên.

### Phân tích hệ thống

Hình 3.1: Lượt đồ use case của hệ thống

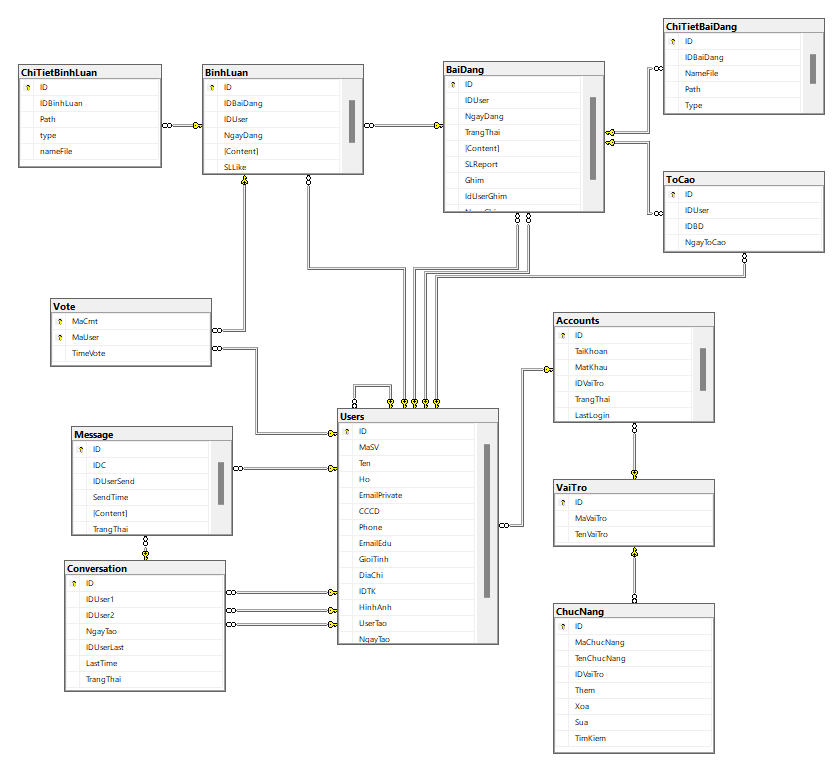
Hệ thống sẽ có 2 đối tượng chính sử dụng là người dùng và người quản lý với các chức năng như sau:

* **Người dùng:**
* **Chức năng đăng nhập và quên mật khẩu:** Mỗi người dùng để có thể truy cập vào hệ thống đều phải thực hiện đăng nhập với tài khoản và mật khẩu được cấp khi người quản lý tạo thông tin người dùng đó. Khi người dùng quên mật khẩu, người dùng có thể gửi mã xác nhận qua email của trường để đổi mật khẩu mới.
* **Chức năng đăng bài viết:** tất cả các người dùng đều có thể thực hiện thao tác đăng một bài viết với nội dung cần đăng và đính kèm nhiều hình ảnh, file và cả video. Sau khi đăng bài thành công, sẽ phải chờ người quản lý kiểm duyệt.
* **Chức năng xem bài viết:** có thể xem các bài viết đã xem và các bình luận của bài viết đó trên bảng tin, ngoài ra còn có thể tố cáo bài viết nếu cảm thấy nội dung bài viết không phù hợp.
* **Chức năng bình luận:** người dùng có thể bình luận với thời gian thực để trả lời các thắc mắc, các câu hỏi của tất cả các bài viết có trên bảng tin. Người dùng có thể xoá bình luận của chính mình. Ngoài ra còn có thể vote cho các câu trả lời mà người dùng cảm thấy hợp lý và đúng nhất.
* **Chức năng nhắn tin:** tạo một đoạn hội thoại với một người dùng bất kì có tài khoản trong hệ thống. Nhăn tin trao đổi trực tiếp trong thời gian thực với người dùng đó.
* **Chức năng quản lý bài đăng cá nhân:** người dùng có thể xem được tất cả các bài đăng cá nhân đã duyệt hoặc chưa được duyệt, có thể xoá bài đăng.
* **Chức năng quản lý thông tin cá nhân:** có thể xem thông tin cá nhân, chỉnh sửa thông tin cá nhân, đổi hình đại diện, đổi mật khẩu.
* **Chức năng xem trang cá nhân:** người dùng có thể vào trang cá nhân của người khác để có thể xem các thông tin cơ bản, xem các bài viết của người đó, nhắn tin riêng với người dùng đó.
* **Chức năng tìm kiếm thông tin:** người dùng có thể nhập một từ khoá bất kì vào ô tìm kiếm, sau đó hệ thống sẽ hiện ra tất cả các bài viết hoặc người dùng có liên quan đến từ tìm kiếm. Có thể liên hệ với các người dùng có liên quan.
* **Người quản lý:**
* **Quản lý bài viết:** kiểm duyệt hết tất cả các yêu cầu đăng bài viết mới của người dùng, kể cả người có vài trò là người quản lý đăng bài. Xoá một bài viết bất kì. Xem lượt tố cáo của từng bài viết. Xoá một bình luận bất kì. Quản lý cũng sẽ có một ô tìm kiếm riêng.
* **Quản lý sinh viên:** Tất cả các sinh viên sẽ được hiển thị lên dưới dạng bảng. Có thể vô hiệu hoá một sinh viên bất kì. Thêm một sinh viên mới, sau khi thêm một sinh viên mới, hệ thống sẽ tự động gửi một email có thông tin bao gồm tài khoảng và mật khẩu tới email trường của sinh viên đó. Ngoài ra có thể thêm sinh viên bằng cách thêm bằng file excel, mẫu file có thể được tải ở nút bên cạnh. Sửa thông tin sinh viên. Tìm kiếm sinh viên.
* **Thống kê:** Hiển thịSơ đồ thống kê mật độ hoạt động của hệ thống qua các ngày, có thể lựa chọn hiển thị thống kê qua khoảng thời gian mà ta lựa chọn. Có thể xuất ra báo cáo thông kê tần xuất hoạt động của tất cả các sinh viên có tài khoản trong hệ thống.

### Thiết kế cơ sở dữ liệu.

Để có thể xây dựng các chức năng đã đề ra của dự án, em sẽ xây dựng cơ sở dữ liệu với các bảng sau:

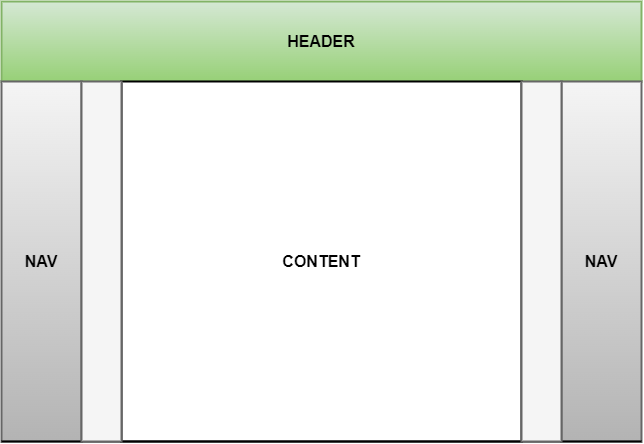
* Accounts (ID, TaiKhoan, MatKhau, IDVaiTro, TrangThai, LastLogin).
* BaiDang (ID, IDUser, NgayDang, TrangThai, Content, SLReport, Ghim, IDUserGhim, NgayGhim).
* BinhLuan (ID, IDBaiDang, IDUser, NgayDang, Content, SLLike).
* ChiTietBaiDang (ID, IDBaiDang, NameFile, Path, Type).
* ChiTietBinhLuan (ID, IDBinhLuan, Path, Type, NameFile).
* ChucNang (ID, MaChucNang, TenChucNang, IDVaiTro, Them, Xoa, Sua, TimKiem).
* Conversation (ID, IDUser1, IDUser2, NgayTao, IDUserLast, LastTime, TrangThai).
* Message (ID, IDC, IDUserSend, SendTime, Content, TrangThai).
* ToCao (ID, IDUser, IDBD, NgayToCao).
* Users (ID, MaSV, Ten, Ho, EmailPrivate, CCCD, Phone, EmailEdu, GioiTinh, DiaChi, IDTK, HinhAnh, UserTao, NgayTao, GhiChu).
* VaiTro (ID, MaVaiTro, TenVaiTro).
* Vote (MaCmt, MaUser, TimeVote).

Và dưới đây là sơ đồ database diagram của toàn bộ hệ thống, thể hiện mối quan hệ giữa các bảng với nhau thông qua các khoá chính và khoá ngoại.

Hình 3.2: Database diagram

* + 1. Xây dựng giao diện

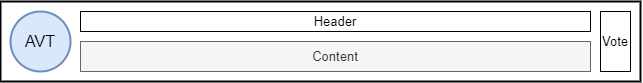
Bởi vì là đề tài dự án là xây dựng website Blog sinh viên HCMUNRE nên thiết kế khá giống với một trang mạng xã hội và không thể thiếu các tiêu chí phải dễ nhìn, trực quan, rõ ràng sẽ ảnh hưởng lớn đến trải nghiệm của người dùng nên việc thiết giao diện cũng cực kì quan trọng đối với dự án. Nên dự án sẽ có thiết kế giao diện của các chức năng chính như sau (vì có nhiều giao diện được lặp lại nên em chỉ thể hiện những giao diện chính và khác biệt):

* **Giao diện đăng nhập:** được đặt chính giữa khung hình, màu chủ đạo là xanh lá, có một tấm poster bên trái, một logo trường, 2 trường nhập tài khoản mật khẩu, nút quên mật khẩu, 1 form gửi email, 1 form xác nhận mã, 1 form nhập mật khẩu mới.
* **Giao diện hệ thống (layout)**: giao diện được áp dụng cho cả hệ thống sau khi đăng nhập. Content là nơi chứa các bài viết.

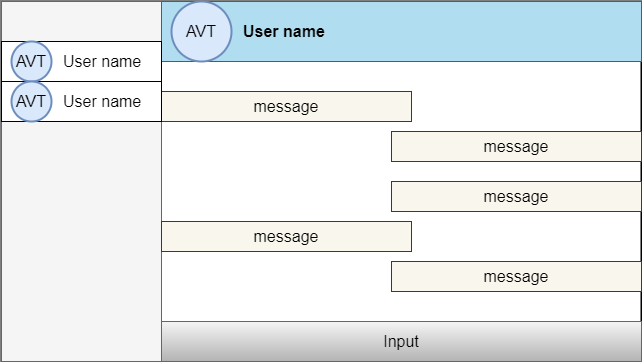
Hình 3.3: Layout

* **Cấu trúc bài viết:**

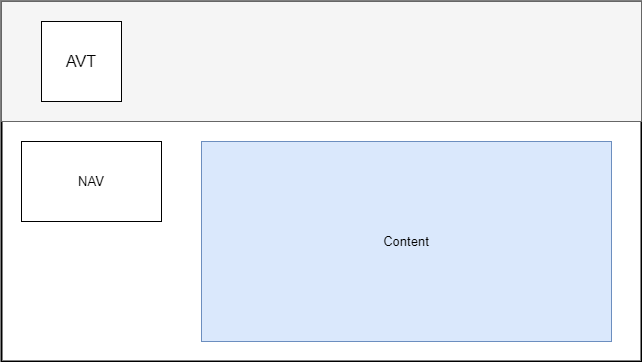
Hình 3.4: Cấu trúc 1 bài viết

* **Cấu trúc bình luận:**

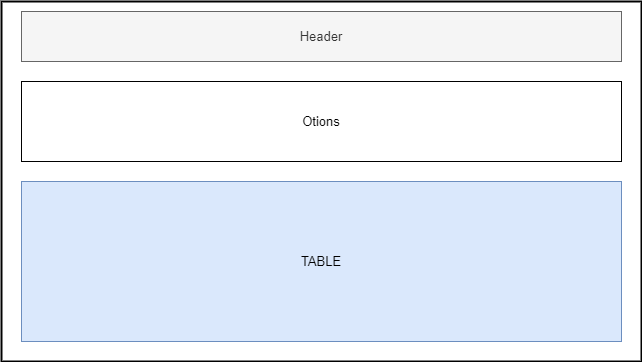
Hình 3.5: Cấu trúc 1 bình luận

* **Cấu trúc giao diện của chức năng nhắn tin:**

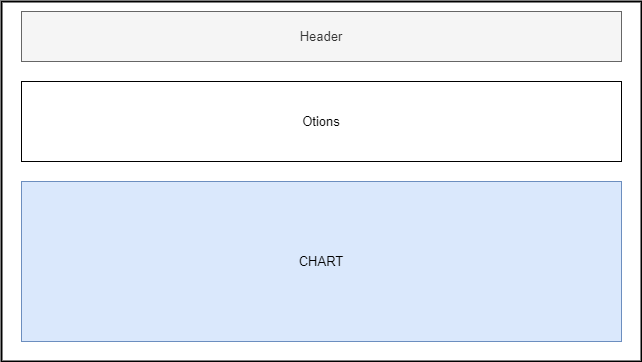
Hình 3.6: Giao diện của chức năng nhắn tin

* **Cấu trúc giao diện của chức năng quản lý thông tin cá nhân:**

Hình 3.7: Cấu trúc giao diện của chức năng quản lý thông tin cá nhân

* **Cấu trúc giao diện của chức năng quản lý sinh viên:**

Hình 3.8: Cấu trúc giao diện chức năng quản lý sinh viên

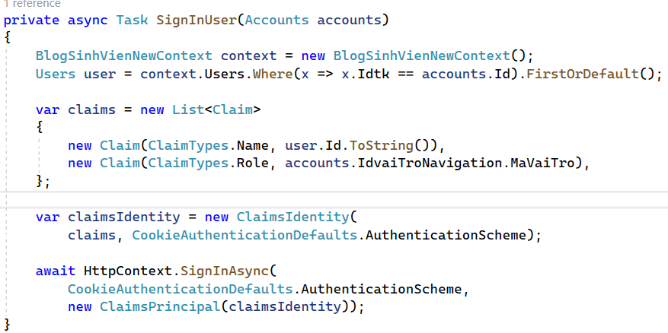
* **Cấu trúc giao diện của chức năng thống kê hoạt động:**

Hình 3.9: Cấu trúc giao diện của chức năng thống kê hoạt động

### Xây dựng chức năng

* **Tương tác với cơ sở dữ liệu:** Vì những kiến thức lý thuyết về Enity FrameWork ở mục cơ sở lý thuyết nên mọi tương tác với cơ sở dữ liệu của hệ thống này của em sẽ được tương tác qua công cụ Enity Framework. Ngoài ra, công cụ này còn hỗ trợ đắc lực trong việc tạo các models hoàn toàn tự động ánh xạ từ cơ sở dữ liệu với các quan hệ giữa chúng một cách chính xác và nhanh chóng chỉnh với 3 bộ thư viện: Enity FrameWorkCore SqlServer, Enity FrameWorkCore SqlServer Design, Enity FrameWorkCore Tools.
* **Đăng nhập:** vì hệ thống mang tính cục bộ nên việc bảo mật thông tin là hết sức cần thiết. Để xử lý vấn đề này ASP.NET Core 3.1 hỗ trợ cho chúng ta một bộ thư viện bao gồm: AspNetCore Authentication, AspNetCore Authentication Cookies. Sau khi nhập tài khoản và mật khẩu sau đó thực hiện đăng nhập, hệ thống sẽ xác nhận tính chính xác:

Hình 3.10: Code nhận yêu cầu và xác nhận tài khoản

Nếu thông tin tài khoản chính xác hệ thống phải gửi đi một Cookies gồm ID và Vai trò của người dùng vừa đăng nhập vào hệ thống sau đó gửi đến một Claims identity vào HttpContext để xác nhận với hệ thống là đã xác thực, ngược lại sẽ trả về thông báo đăng nhập thất bại.

Hình 3.11: Code gửi Cookies

* **Gửi email lấy lại mật khẩu:** sau khi người dùng nhập vào email của trường để lấy mã xác nhận, hệ thống không thể gửi về một post request trực tiếp từ form để tránh phải reload lại trang web, nên phải sử dụng 1 module của thư viện Jquery là Ajax để gửi 1 post request mà không phải reload lại trang. Sau đó hệ thống sẽ xác nhận tính chính xác và hợp lệ của email để gửi một mã xác nhận đến email đó. Thư viện để thực hiện chức năng gửi email này là MailKit.

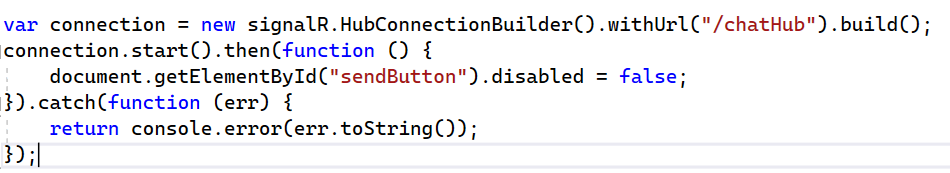
Hình 3.12: Code Ajax để gửi thông tin email vào controller

Hình 3.13: Code gửi mã xác nhận đến email

* **Đăng bài:** Cái khó của chức năng này có 2 điểm: thứ nhất phải hiển thị 1 bản xem trước cho tất cả các tệp đính kèm của người đăng, thứ hai là phải lưu hết tất cả các tệp đính kèm của bài viết vào folder của project theo từng ngày. Vì vậy để xử lí vấn đề hiển thị bản xem trước ta cần một đoạn code Javascript.

Hình 3.14: Đoạn code Javascript để hiển thị hình ảnh trước khi đăng

Để lưu tất cả các tệp đính kèm của bài viết, trước hết ta cần phải nhận hết tất cả các tệp vào trong controller, sau đó ta lưu vào thư mục lưu trữ chi tiết bài đăng với cấu trúc “yyyyMMdd” để hạn chế sự nỡ phình của dữ liệu sau này từ đó có thể tối ưu tốc độ truy cập.

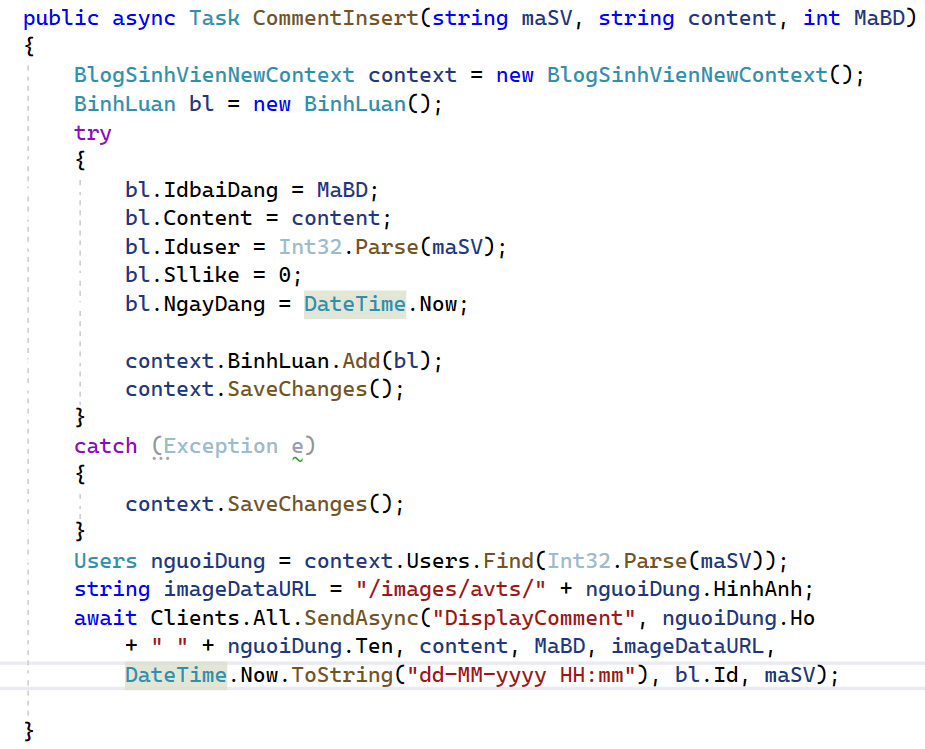
* **Bình luận:** để đáp ứng được tính thực tế của đề tài, việc viết một bình luận phải được hiển thị trong thời gian thực. Vì vậy để giải quyết vấn đề này ta cần phải dùng đến thư viện SignalR. Để sử dụng thư viện này, ta phải khai báo một file Javascript vào phần layout của dự án. Tiếp theo ta cần 1 kết nối giữa client với server bằng một đoạn JavaScript.

Hình 3.15: Code mẫu để lưu một danh sách các hình ảnh của bài viết

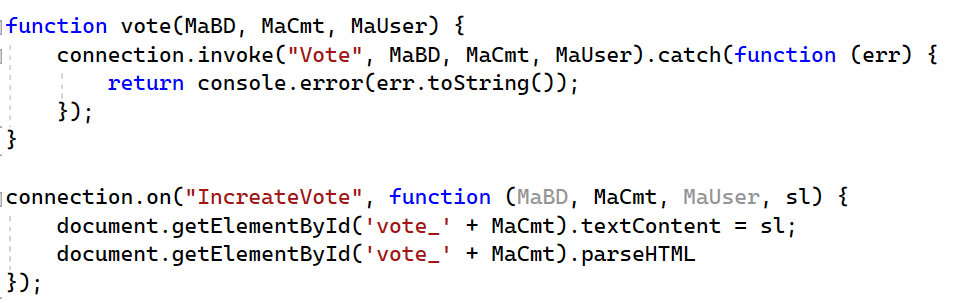
Ta sẽ có một lớp con kế thừa từ lớp cha là Hub, với 1 hàm để nhận dữ liệu từ phía client. Sau đó bắt sự kiện một người đăng bình luận thì ta mở connection và gọi đến server với các dữ liệu cần thiết.

Hình 3.17: Hàm JavaScript để bắt sự kiện khi người dùng comment

Hình 3.16: Code JavaScript để mở kết nối giữa client và server

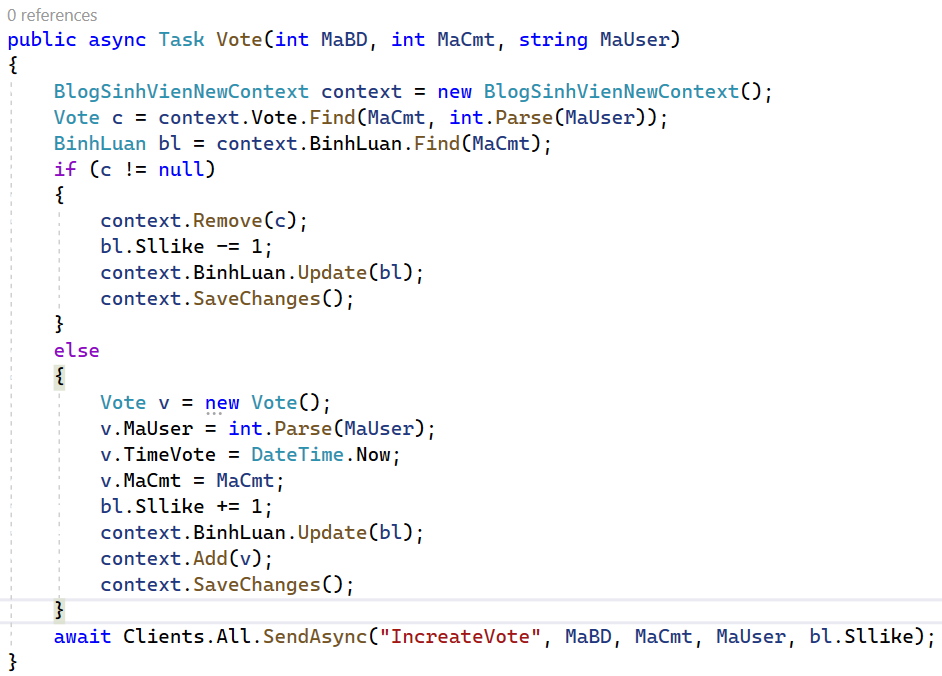
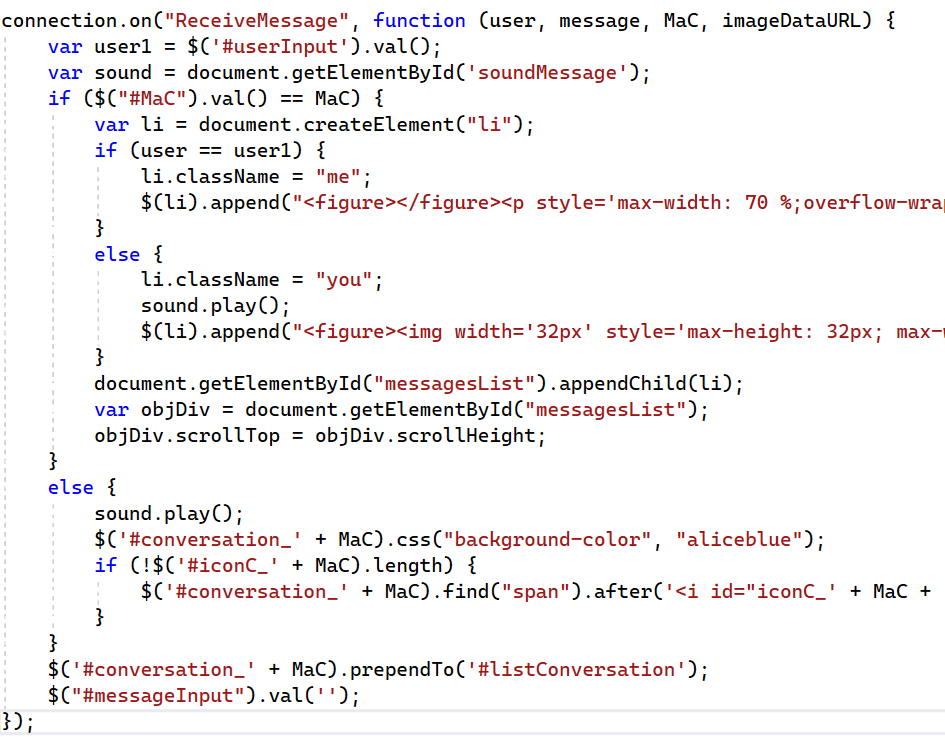
Sau khi tiến hành lưu vào cơ sỡ dữ liệu ta trả về kết quả với các biến cần thiết và hiển thị cho tất cả người dùng thấy.

Hình 3.18: Hàm phía server xử lí dữ liệu bình luận từ client gọi về

* ******Vote:** đối với vote cũng vậy, hiển thị số lượng người dùng đồng ý với bình luận trong thời gian thự cũng khá cần thiết, vì vậy tương tự với cách xử lý của bình luận, tuy nhiên đơn giản hơn ta chỉ cần hiển thị tăng giảm một con số.

Hình 3.20: Hàm bắt sự kiện like và hàm hiển thị số lượng like cho người dùng thấy khi client nhận dữ liệu từ server

Hình 3.19: Code JavaScript nhận kết quả từ server và hiển thị lên cho tất cả người dùng thấy.

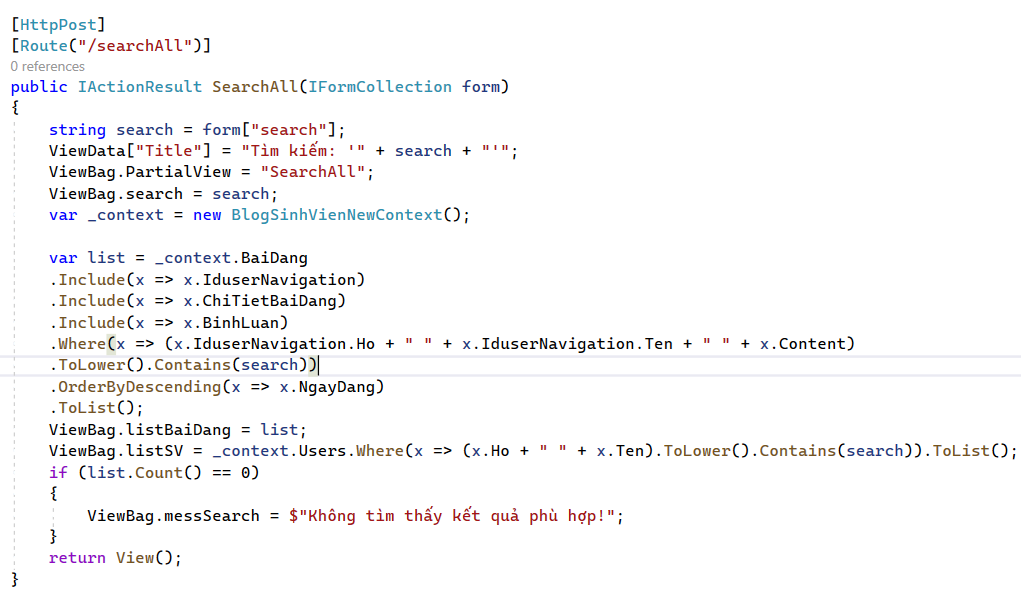
* **Nhắn tin:** Nhắn tin là một chức năng không thể thiếu điều kiện thời gian thực, nên việc tiếp tục sử dụng thư viện SignalR là cần thiết. Để sử dụng thì cũng tương tự như chức năng bình luận và vote. Tuy nhiên, lần này việc hiển thị cho tất cả các người dùng thấy là không đúng với nghiệp vụ, chỉ có người nhận được tin nhắn mới có thể thấy và ngược lại, ngoài ra còn phải đáp ứng được các tiêu chí kèm theo như âm thanh thông báo, tin nhắn mới nhận, tin nhắn đã đọc hoặc chưa đọc. Và quan trọng không kém đó tốc độ hiển xử lý của hệ thống, nên việc hiển thị tất cả các tin nhắn cùng lúc sẽ làm cho hệ thống bị chậm.

Hình 3.22: Hàm hiển thị lên khi cho người 2 người trong đoạn hội thoại thấy tin nhắn khi client nhận dữ liệu từ server

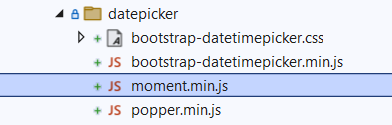
Hình 3.21: Hàm phía server xử lí dữ liệu vote từ client gọi về

Để xử lý tốc độ tải tin nhắn, thì em sẽ xử lí bằng cách chỉ hiển thị 25 tin nhắn mới nhất lúc đầu, nếu người dùng kéo đầu trang, hệ thống sẽ tự động hiển thị thêm 25 tin nhắn cũ hơn. Đây là code xử lý:

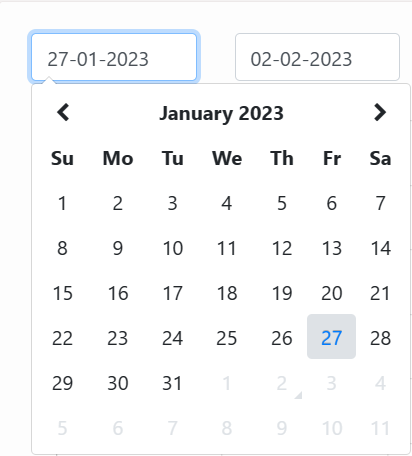
Hình 3.23: Code hiển hiện thị thêm tin nhắn.

* **Tìm kiếm:** Để tối ưu thuật toán tìm kiếm, em sẽ hiển thị tất cả các người dùng, các bài viết có nội dung liên quan đến từ, cụm từ cần tìm kiếm. Đây là thuật toán:

Hình 3.24: Thuật toán tìm kiếm

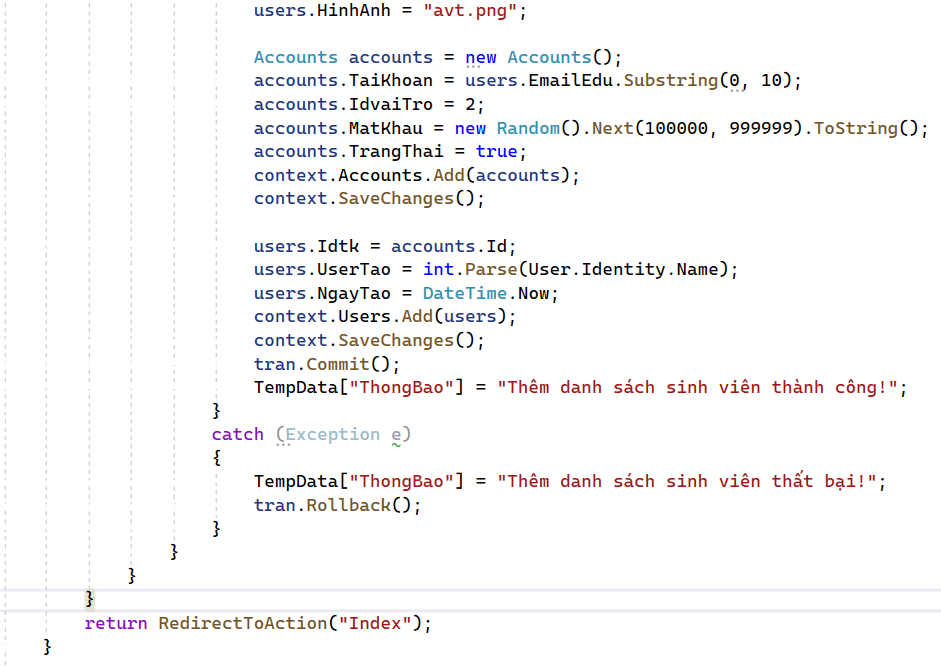
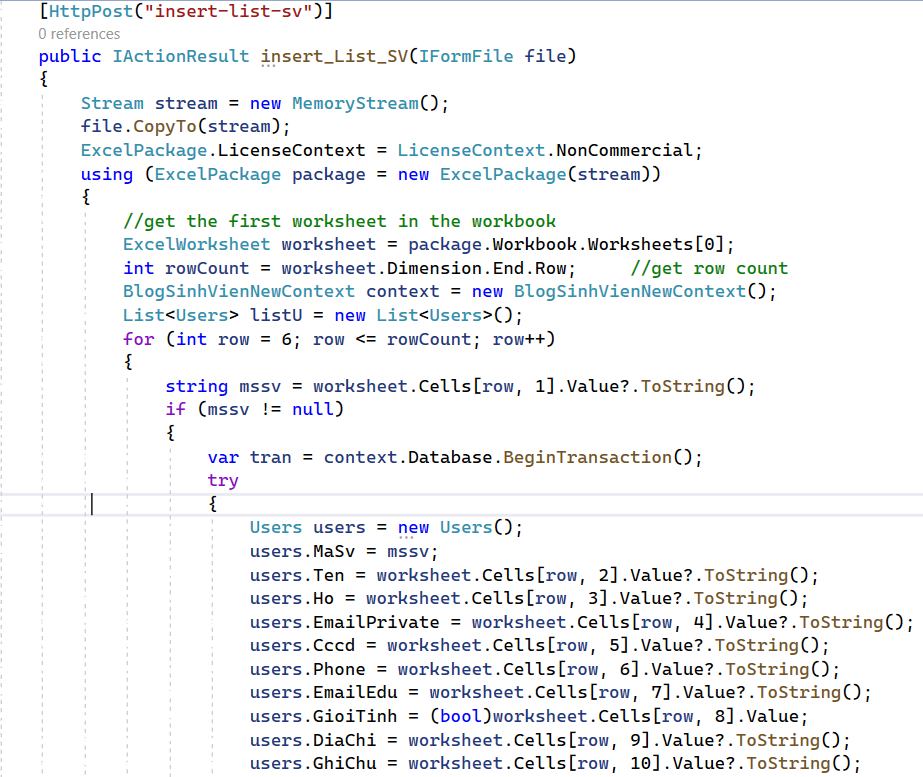
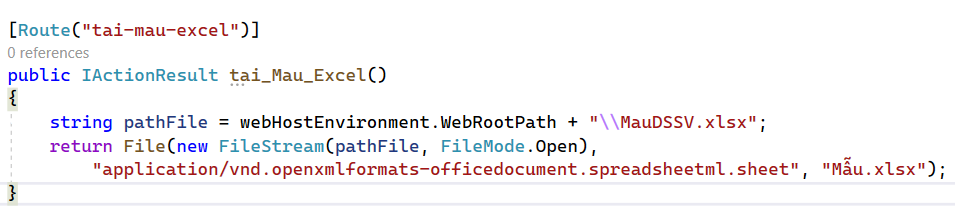
* **Datetime:** Bởi vì hệ thống sẽ được sử dụng bởi nhiều người dùng khác nhau, ở nhiều thiết bị khác nhau, nên việc xác định format ngày giờ từ đầu cho hệ thống là khá quan trọng. Nên em chọn quyết định format ngày giờ cho hệ thống toàn hệ thống của em là “dd-MM-yyyy”. Tuy nhiên việc hiển thị ngày giờ theo một format nhất định thì khá là dễ dàng, nhưng để một input type = “datetime” của HTML theo một format nhất định là không thể, bởi vì mặc định input type = “datetime” của HTML sẽ lấy theo quy định của thiết bị đang sử dụng. Để xử lí vấn đề này em dùng tới bộ thư viện gồm 4 file:

Hình 3.25: Bộ thư viện để xử lí vấn đề datetime picker

Với bộ thư viện này, ta có thể cài đặt format type date của hệ thống là “dd-MM-yyyy” một cách dễ dàng và bắt mắt chỉ với vài dòng code.

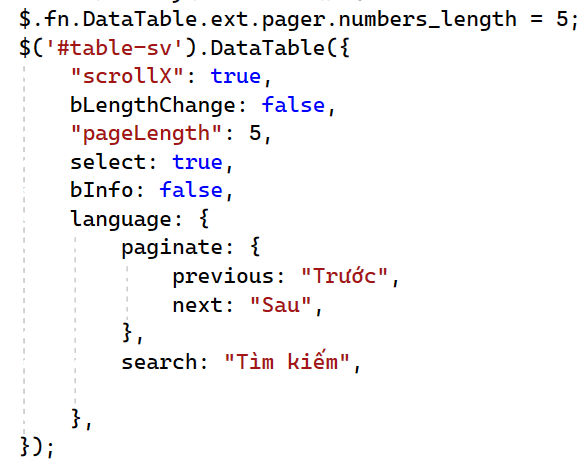
Hình 3.26: Giao diện khi sài bộ thư viện datetime picker

* **Thêm sinh viên với file Excel:** với một hệ thống việc thêm một sinh viên mới bằng tay là không hợp lí, nên việc thêm một danh sách sinh viên từ file Excel thì phù hợp hơn trong thực tế. Vậy để làm điều này em đã sử dụng một thư viện mã nguồn mở là EPPlus. Một nghiệp vụ không thể thiếu của chức năng này đó chính là tải về một mẫu Excel, việc hệ thống xử lí file Excel phải theo một khuông khổ nhất định là phù hợp với thực tế.

****

Hình 3.27: Action tải mẫu Excel

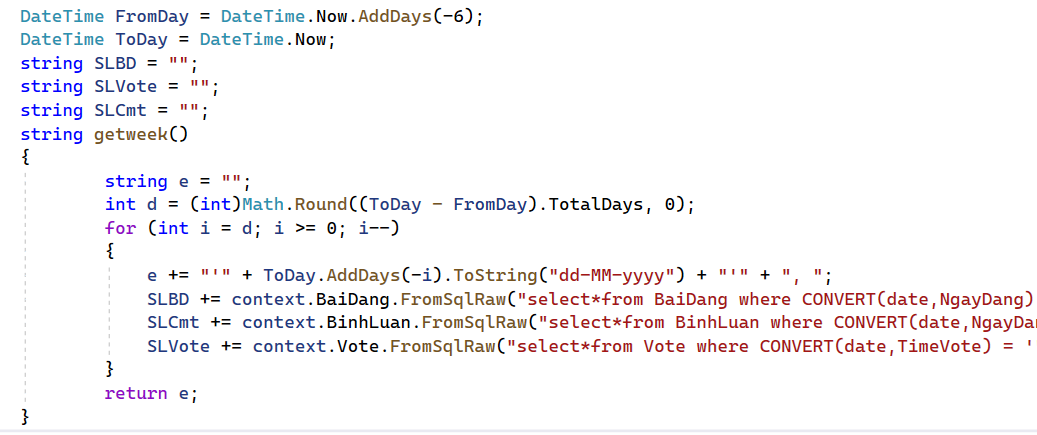
Hình 3.28: Action thực hiện thêm sinh viên vào cơ sở dữ liệu với file Excel

* **Datatable:** để hiển thị dữ liệu dưới dạng table một cách nhanh chóng và tiện lợi, bao gồm cả chức năng phân trang và tìm kiếm trên table, em sẽ lựa chọn sử dụng bộ thư viện mã nguồn mở của datatble bao gồm 2 file: datatables.css, datatables.min.js và một đoạn code JavaScript.

Hình 3.29: Code JavaScript để sử dụng thư viện Datatables.

* **Đồ thị thống kê:** thống kê mật độ hoạt động của hệ thống là một điều quan trọng và không thể thiếu. Vì vậy trước ta ta phải hiển thị được một đồ thị dạng cột, và để thực hiện chức năng này em sử dụng một thư viện mã nguồn mở của ChartJS. Để sử dụng thư viện này chỉ cần khai báo một file JavaScript và một đoạn code JavaScript để cấu hình.

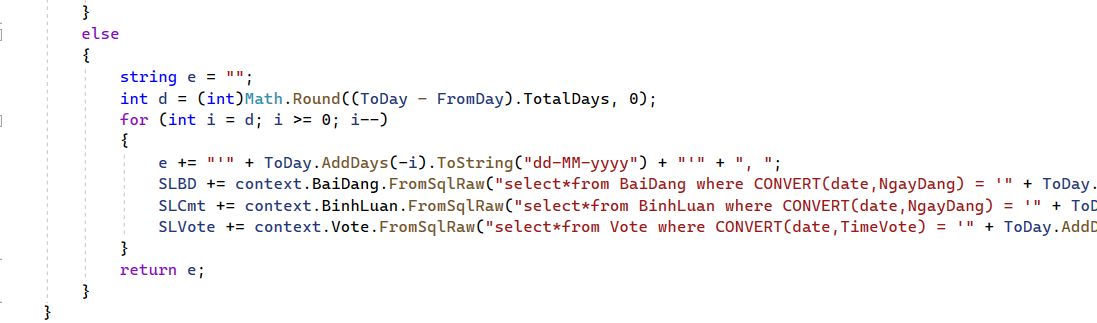
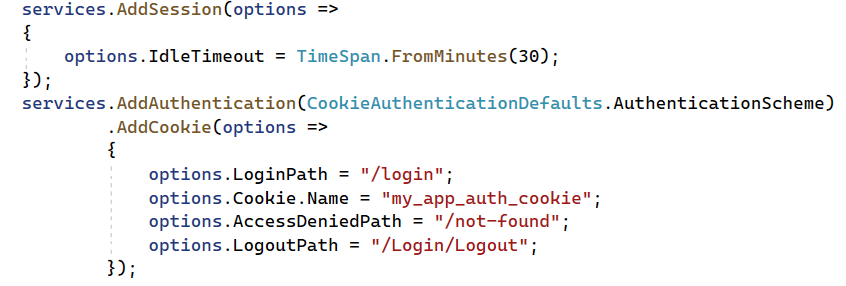
Hình 3.30: Code khai báo thư viện và cấu hình thẻ canvas thành một đồ thị

Tiếp theo để hiển thị được dữ liệu từ database. Trong Razor page của asp.net ta có 1 tag helper là html.Raw tag Helper này cho phép JavaScript nhận được dữ liệu từ biến C#. Đây là đoạn code trên Razor page để lấy được mật độ hoạt động của hệ thống trong vòng 7 ngày gần đây:

Hình 3.31: Code C# để lấy được mật độ hoạt động của hệ thống trong vòng 7 ngày.

Tuy nhiên nhu cầu hiển thị mật độ hoạt động của hệ thống trong vòng 7 ngày là chưa đủ. Người dùng cần phải chọn được số liệu thống kê trong một khoảng thời gian nào đó. Nhưng để thực hiện điều đó thì lại nảy sinh ra vấn đề là số lượng cột quá nhiều. Để giải quyết vấn đề này, em đã tạo một đoạn code tự động chuyển sang thống kê theo tháng khi thời gian yêu cầu thống kê lớn hơn 1 tháng thì có thể giảm được lượng cột của đồ thị. Đoạn code của em để xử lý vấn đề này là:

Hình 3.32: Code xử lý khi số lượng cột thống kê quá nhiều(1)

* **Bảo mật:** để bảo mật cho website, ta sẽ chia ra làm 2 đối tượng chính, đó là người dùng và người quản lý, người dùng và người quản lý muốn truy cập và sử dụng hệ thống phải thực hiện thao tác đăng nhập, sau khi đăng nhập thành công, người dùng chỉ có thể sử dụng được những chức năng mà hệ thống quy định, nếu người dùng gửi request vào đường dẫn không được phép, hệ thống sẽ trả về trang not found.

Hình 3.33: Code xử lý khi số lượng cột thống kê quá nhiều(2)

Hình 3.34: Khai báo dịch vụ Authentication

### Quản lý mã nguồn

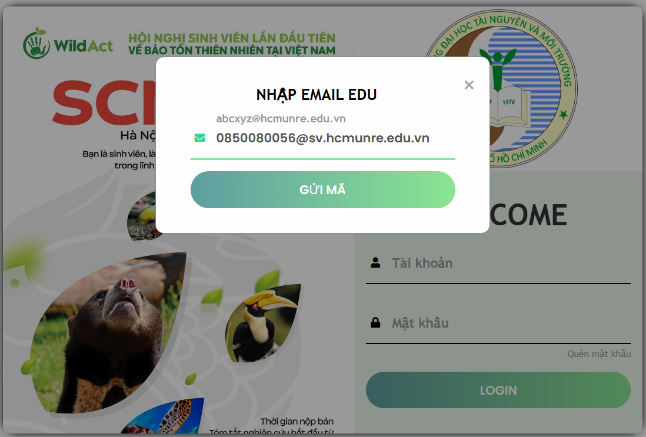
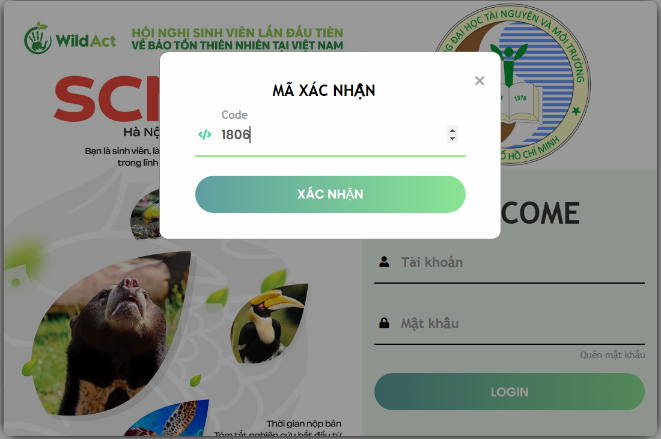
Để quản lý và bảo vệ tránh mất mát mã nguồn của hệ thống hoặc những trường hợp không may xảy ra, em sẽ sử dụng Git để lưu trữ các phiên bản khác nhau của hệ thống, giúp em dễ dàng phát triển nhiều chức năng khác nhau của hệ thống mà không sợ đụng phải một chức năng khác. Sau đó em lưu trữ Source Code trên GitHub để có thể dễ dàng chia sẽ và quản lý khi có bất kì trường hợp không may nào xảy ra.

## Kết quả đạt được

Hầu hết các chức năng và yêu cầu đề ra ban đầu của hệ thống đều đã được triển khai thành công, và thực hiện đúng với các yêu cầu nghiệp vụ. Tuy nhiên việc có những sai xót chưa được phát hiện ra là không thể tránh khỏi, mong hội đồng có thể bỏ qua cho em. Dưới đây là các kết quả đạt được của hệ thống:

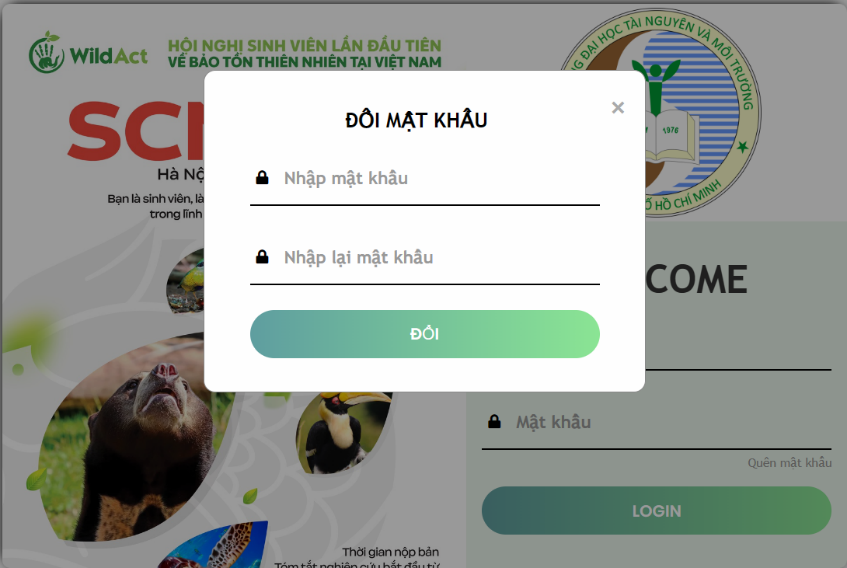
### Đăng nhập xác thực, quên mật khẩu

Hình 3.35: Giao diện đăng nhập

Nhấn vào quên mật khẩu sẽ hiển thị Modal nhập email, để lấy mã xác thực cho phép đặt lại mật khẩu.

Hình 3.37: Chức năng quên mật khẩu (2)

Hình 3.36: Chức năng quên mật khẩu (1)



Hình 3.38: Chức năng quên mật khẩu (3)

### Bảng tin trang chủ

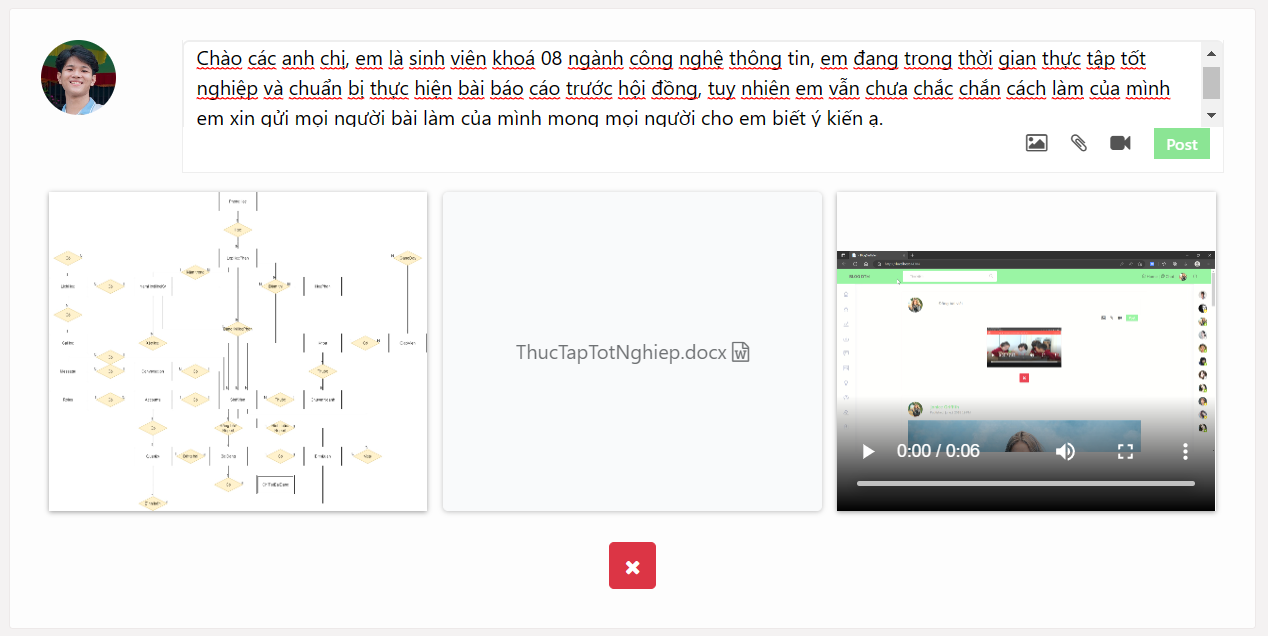
Hình 3.40: Giao diện đầy đủ của một bài viết

Hình 3.41: Cấu trúc đầy đủ của phần bình luận

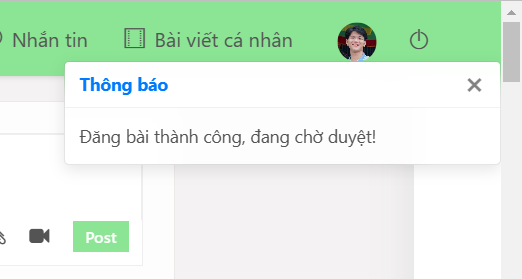
Hình 3.39: Giao diện trang chủ

### Tạo bài viết mới

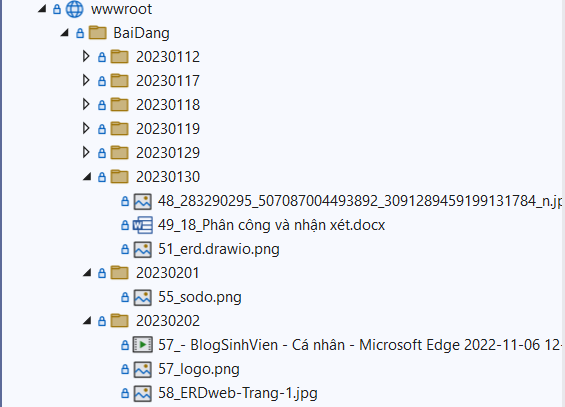
Hình 3.42: Báo cáo và ghim một bài viết

Có thể đính kèm nhiều tệp cùng lúc, ngoài ra còn có thể cho người dùng thấy bản xem trước của các tệp trước khi đăng.

Hình 3.43: Tạo mới một bài viết

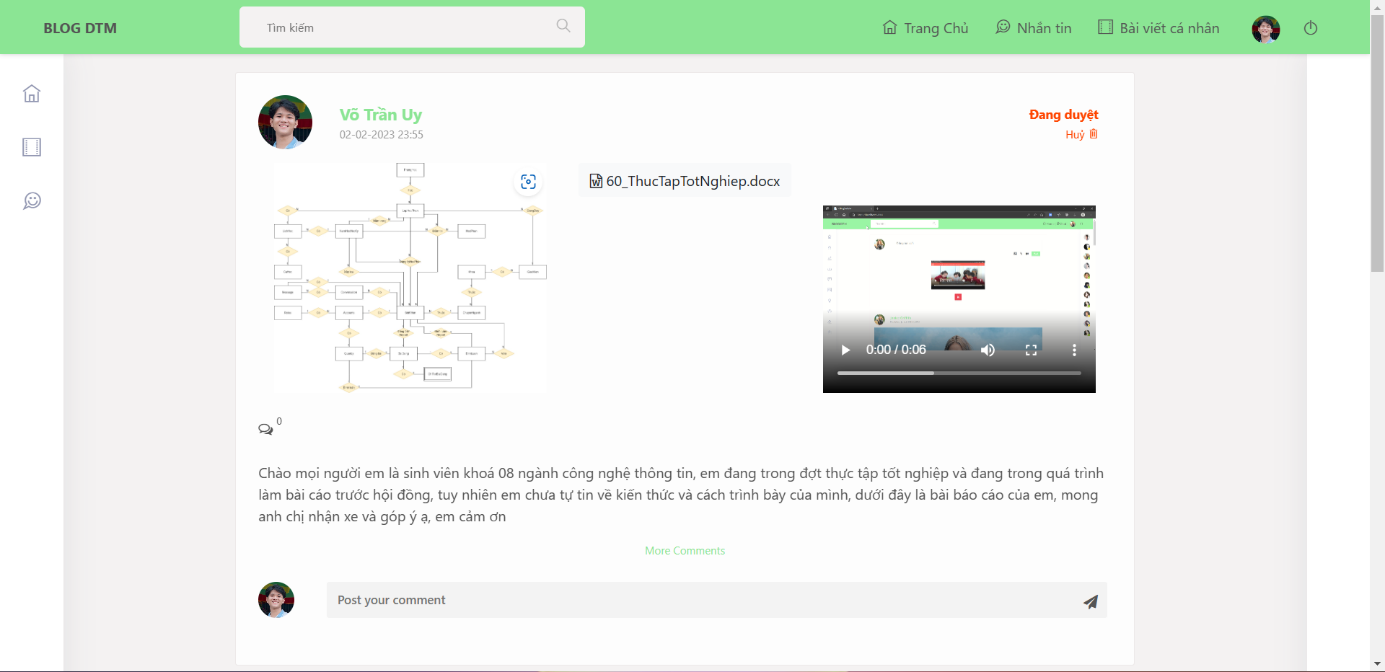
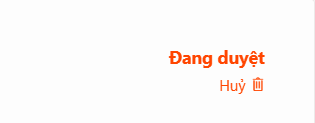
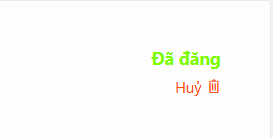
Thông báo khi đăng bài thành công:

Hình 3.44: Thông báo đăn bài thành công

Cách lưu trữ các tệp sau khi đăng bài thành công:

Hình 3.45: Tổ chức lưu trữ tệp của các bài đăng

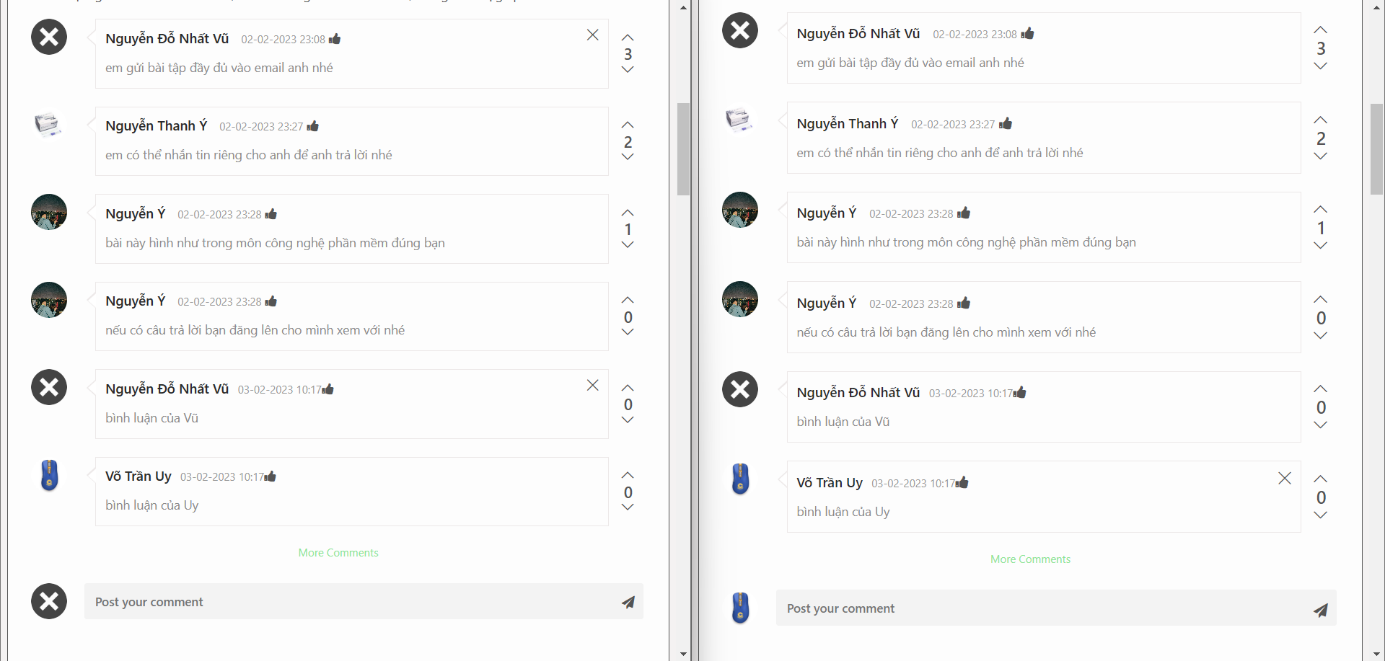
* + 1. Bài đăng cá nhân:

Khá giống với bảng tin của trang chủ, tuy nhiên chỉ hiển thị các bài đăng của cá nhân và trạng thái hoạt duyệt hoặc chưa duyệt của bài đăng, ngoài ra còn có thể xoá bài đăng bất kì của mình.

Hình 3.46: Giao diện bài đăng cá nhân

Hình 3.47: Trạng thái và chức năng xoá bài viết cá nhân

### Bình luận, vote realtime

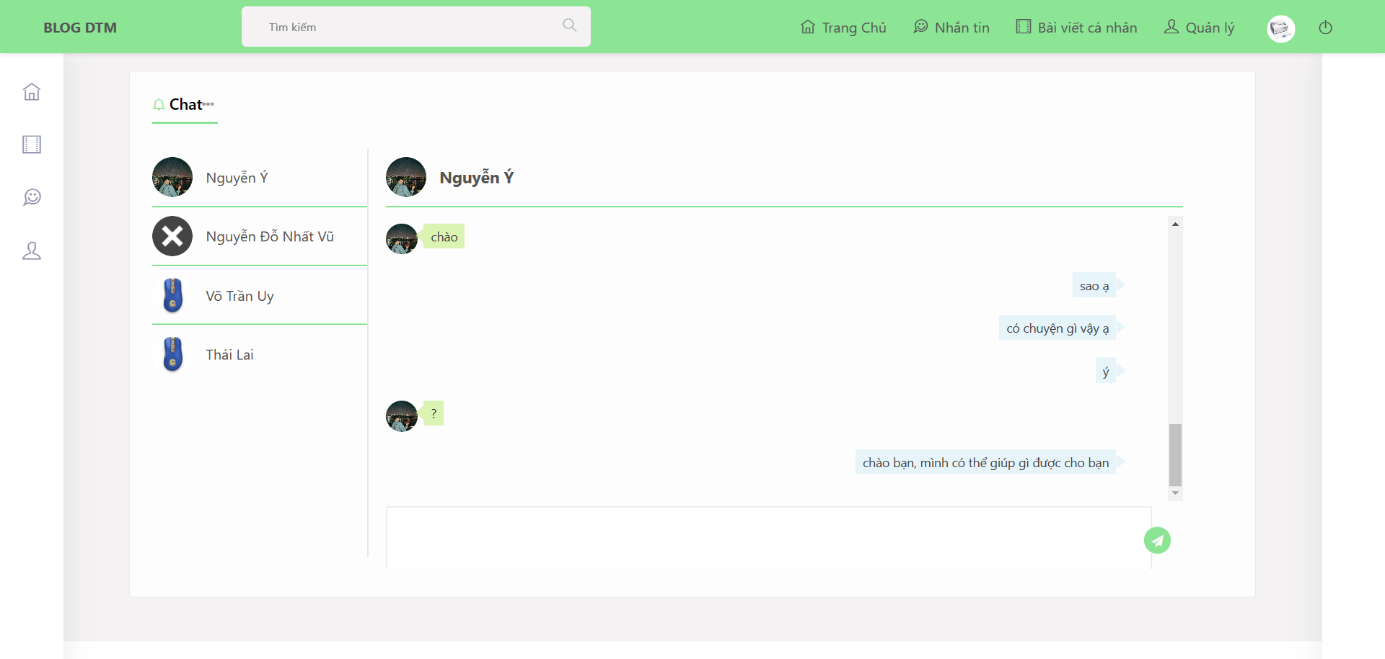
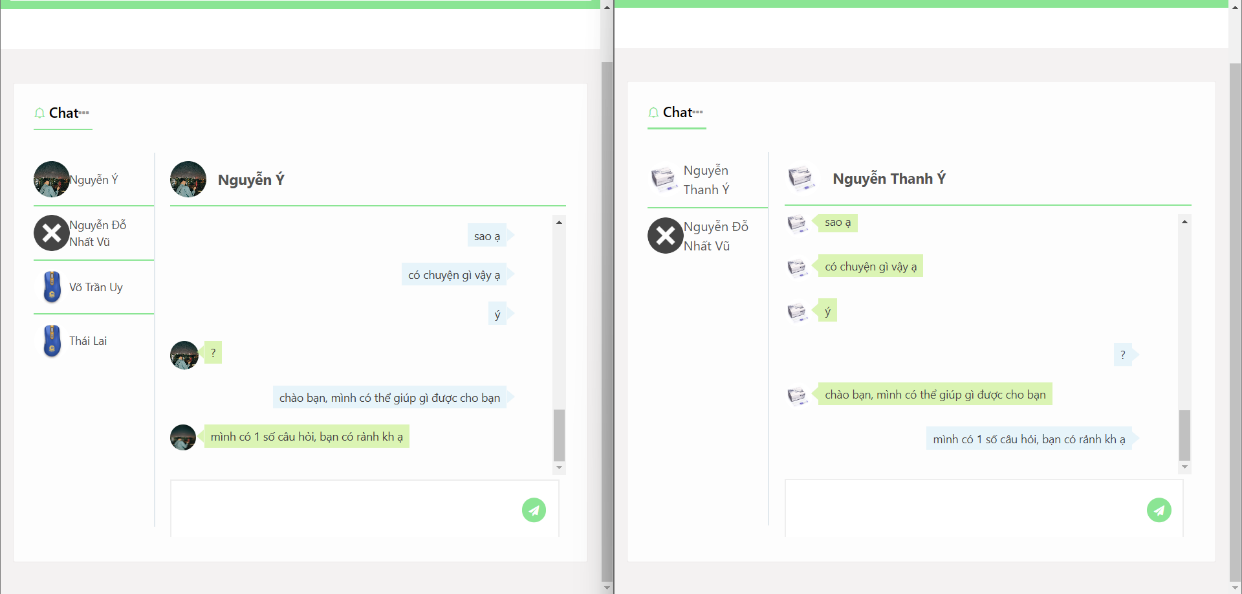
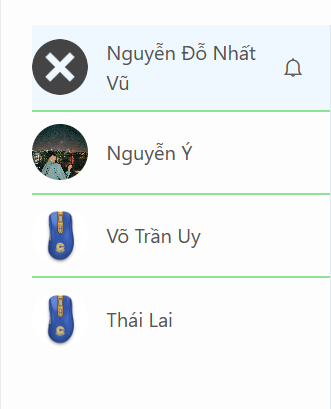
Có thể thực hiện bình luận và đồng ý độ chính xác của bình luận trong thời gian thực.

Hình 3.48: Thực hiện bình luận ở 2 tài khoản khác nhau

Hình 3.49: Bình luận được sắp xếp theo số lượng vote từ lớn đến bé

### Nhắn tin realtime

Ở phần chức năng này cho phép người dùng liên hệ trức tiếp với một người dùng khác thông qua tin nhắn ở thời gian thực, đã tối ưu thực toán tải dữ liệu, có các thông báo tin nhắn đến, đoạn hội thoại có tin nhắn chưa đọc.

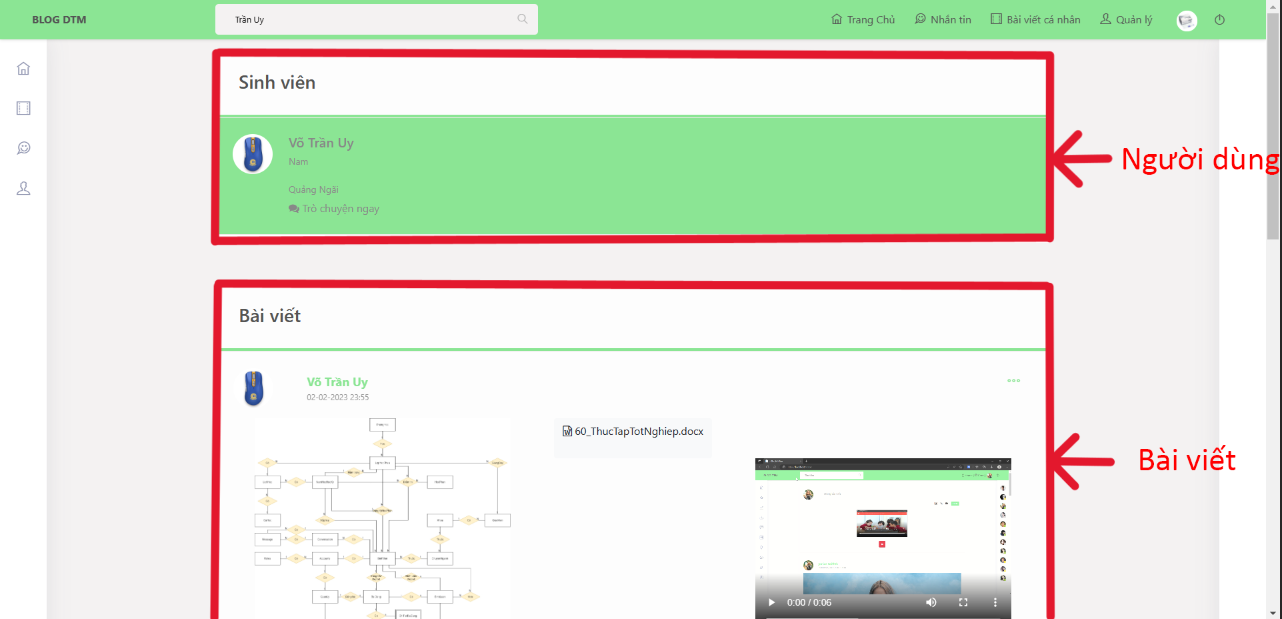


Hình 3.50: Giao diện chính của chức năng nhắn tin

Hình 3.51: Nhắn tin trong thời gian thực

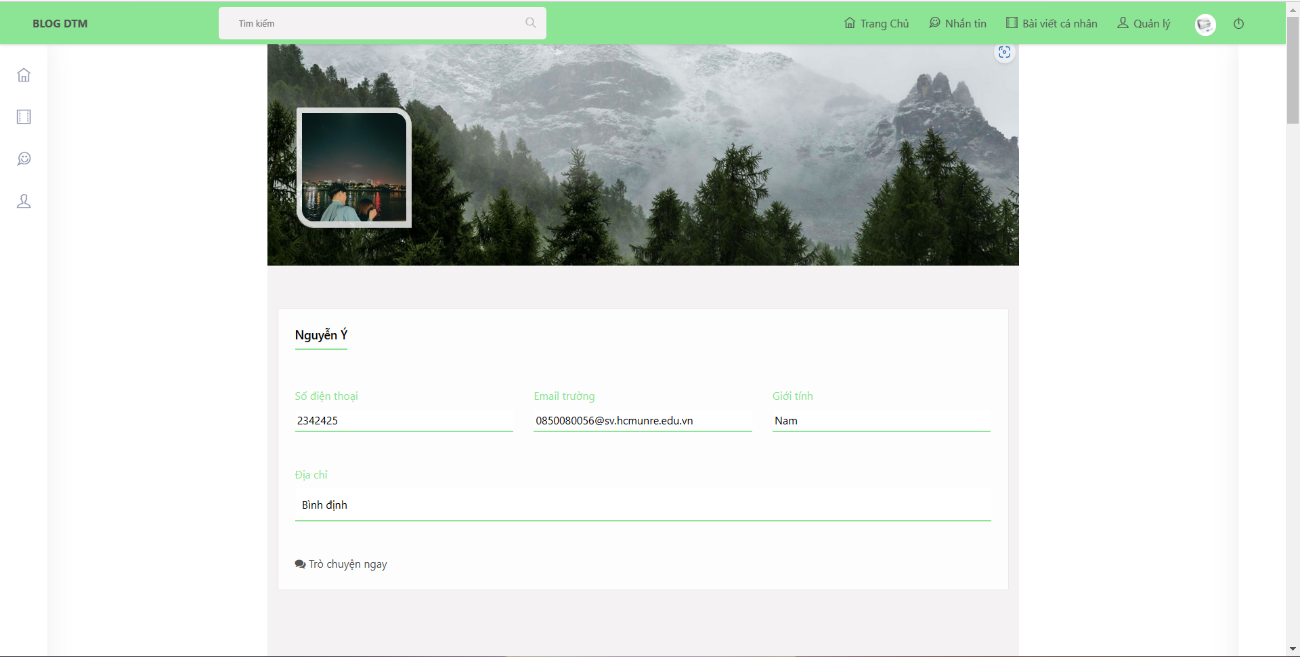
Hình 3.52: Thông báo có tin nhắn mới

### Tìm kiếm

Ở chức năng này, hệ thống sẽ hiển thị ra tất cả các bài viết và người dùng liên quan đến từ khoá hoặc cụm từ khoá tìm kiếm. Sau đó ta có liên hệ với người dùng liên quan.

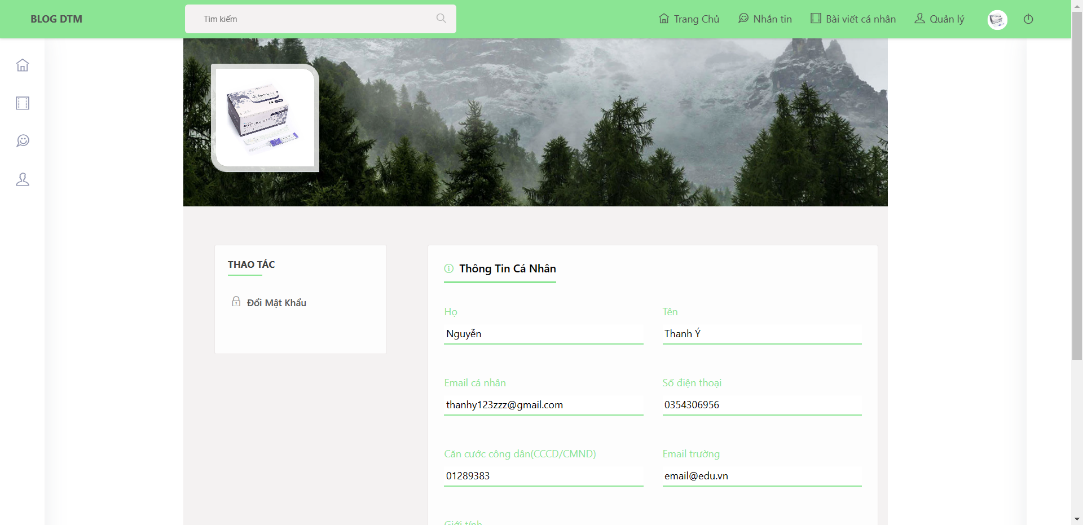
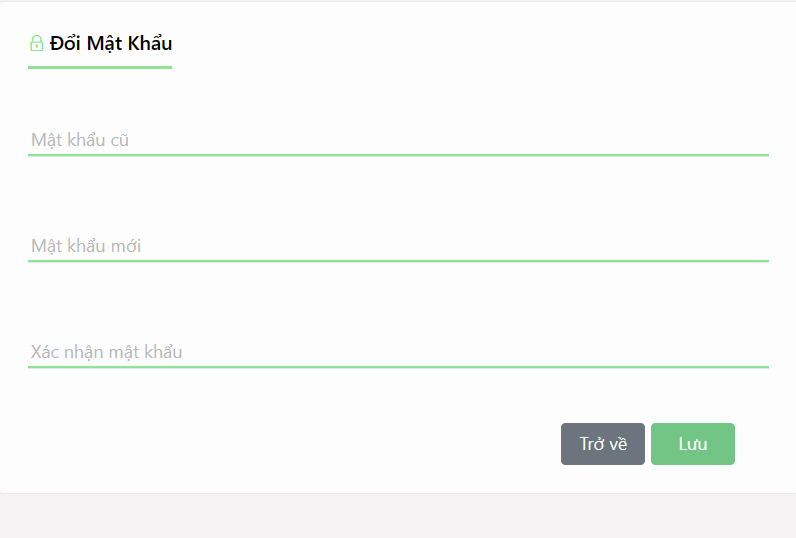
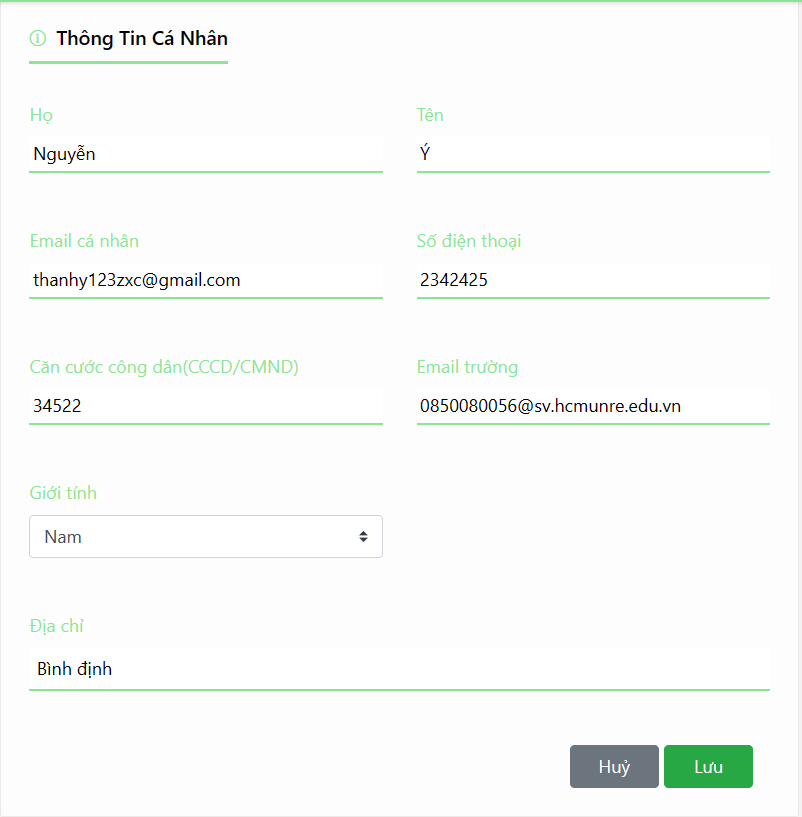
### Trang cá nhân

Hình 3.53: Giao diện tìm kiếm

Chức năng này sẽ được thực hiện khi ta nhấp vào tên của một người dùng bất kì trên bài viết, bình luận, hoặc trong chức năng tìm kiếm để có thể vào trang cá nhân của người dùng đó. Trang cá nhân bao gồm các thông tin cơ bản của người dùng và các bài viết mà người dùng đó từng đăng. Ngoài ra còn có thể tạo cuộc trò chuyện với người dùng đó.

Hình 3.54: Giao diện trang cá nhân của một người dùng

### Quản lý thông tin cá nhân

Ở chức năng này cho phép người dùng thay đổi các thông tin cá nhân, thay đổi ảnh đại diện, thay đổi mật khẩu đăng nhập.

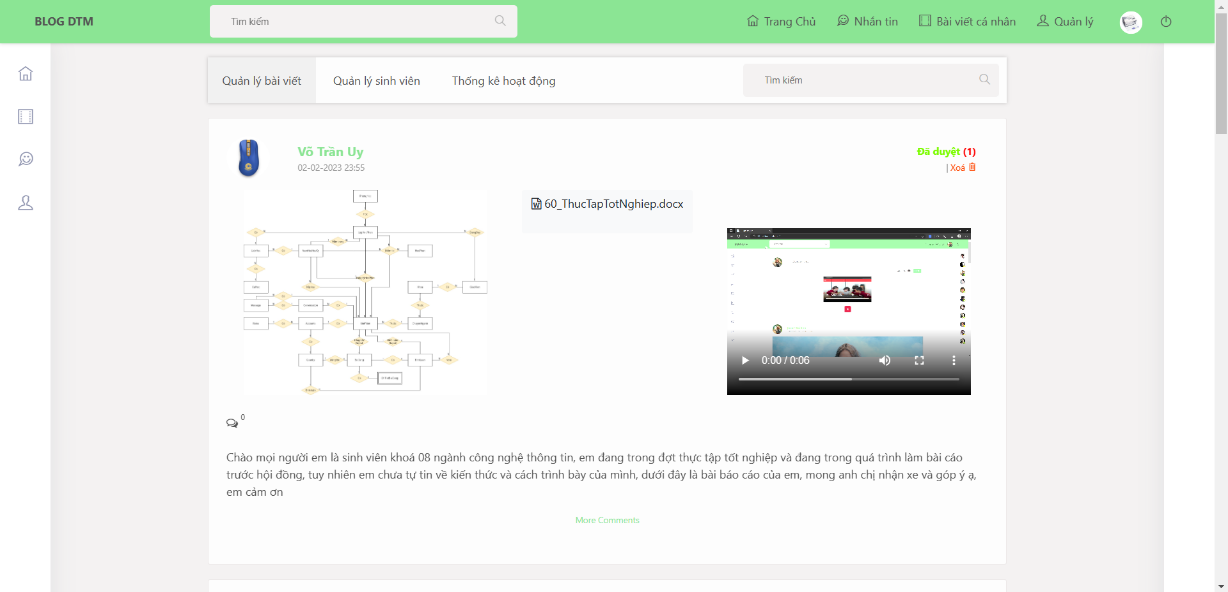
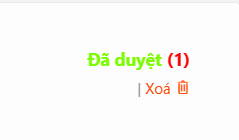
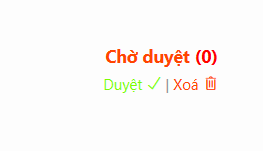
Hình 3.55: Giao diện quản lý thông tin cá nhân

Hình 3.56: Chỉnh sửa thông tin cá nhân

Hình 3.57: Đổi mật khẩu

Hình 3.58: Đổi ảnh đại diện

### Quản lý bài viết

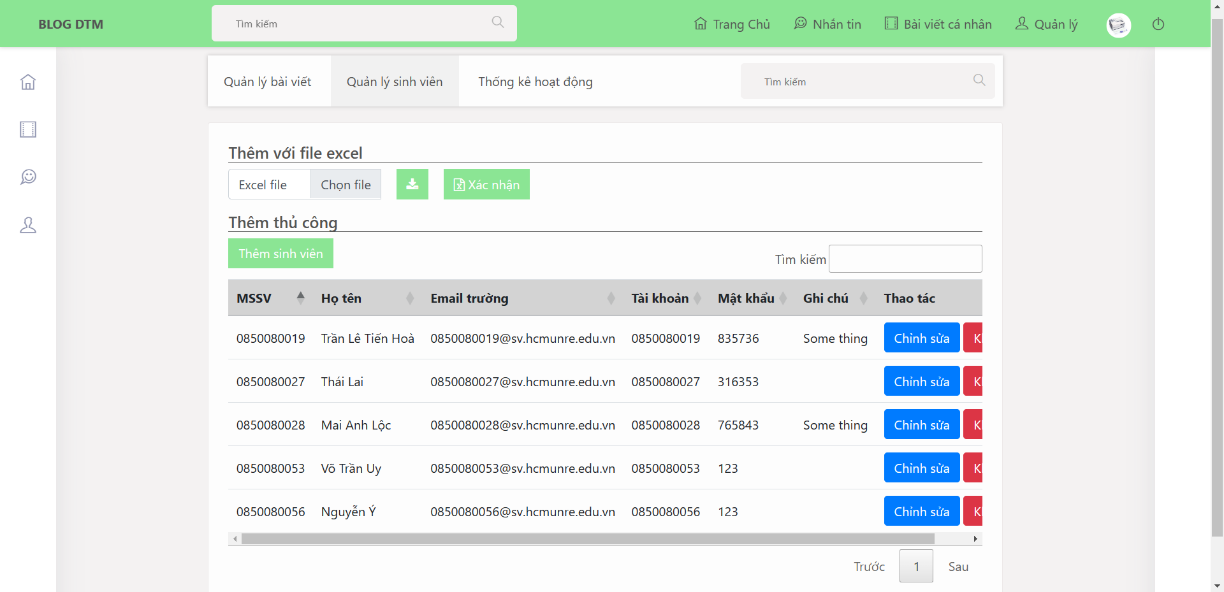
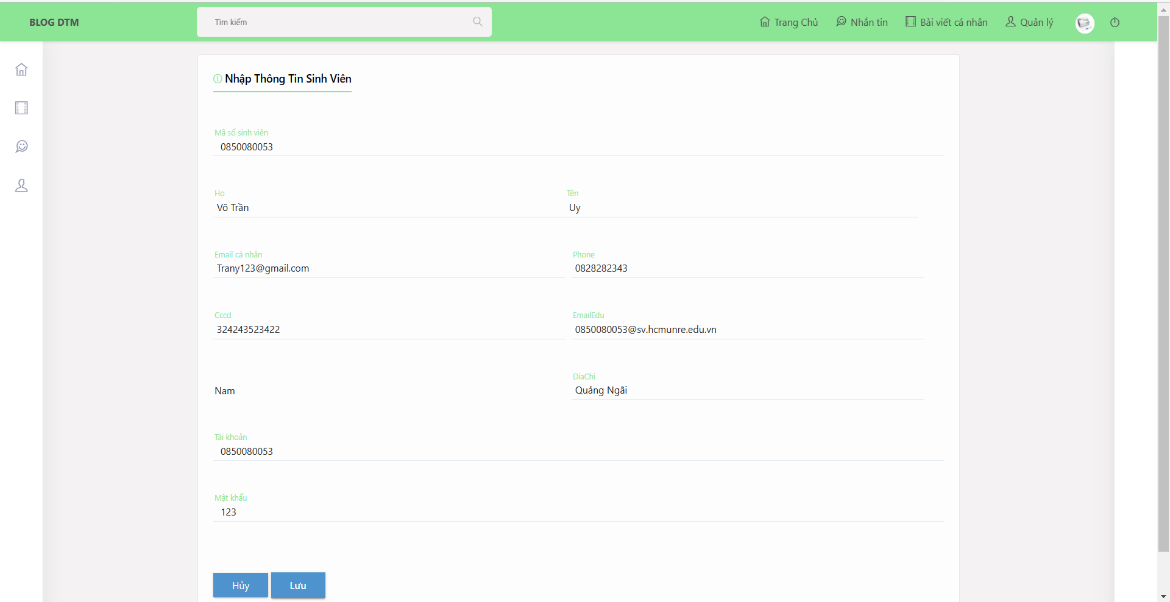
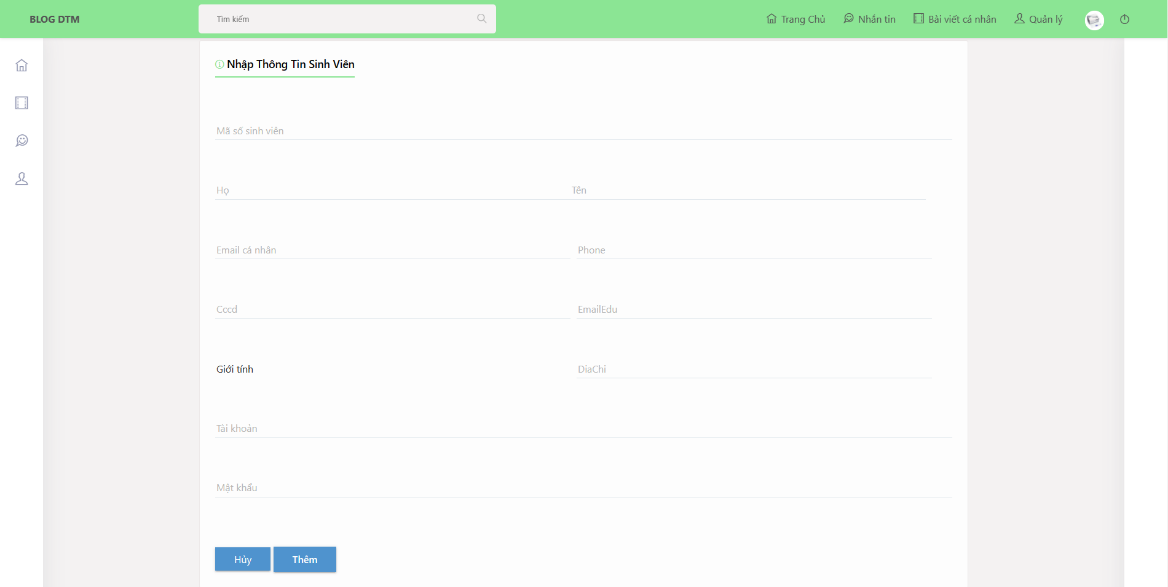
Cũng khá là tương dự với bảng tin trang chủ, tuy nhiên nó sẽ hiển thị tất cả các bài viết đã duyệt và chưa duyệt của hệ thống. Người quản lý có thể duyệt các yêu cầu đăng bài viết mới, xoá một bài viết bất kì, xem được số lượng tố cáo của bài viết đó, ngoài ra còn có thể xoá một bình luận bất kì.

Hình 3.59: Giao diện quản lý bài viết

Hình 3.60: Trạng thái, duyệt bài, xoá bài và số lượng tố cáo của bài viết

### Quản lý sinh viên

Ở chức năng này sẽ hiển thị danh sách các sinh viên của hệ thống dưới dạng table, table cho phép tìm kiếm, phân trang, sắp xếp theo cột. Ngoài ra có thể thêm một sinh viên mới bằng tay, chỉnh sửa thông tin một sinh viên, khoá tài khoản. Chức năng thêm sinh viên bằng file excel, và có thể tải mẫu excel trước thêm sinh viên bằng danh sách.



Hình 3.61: Giao diện quản lý sinh viên

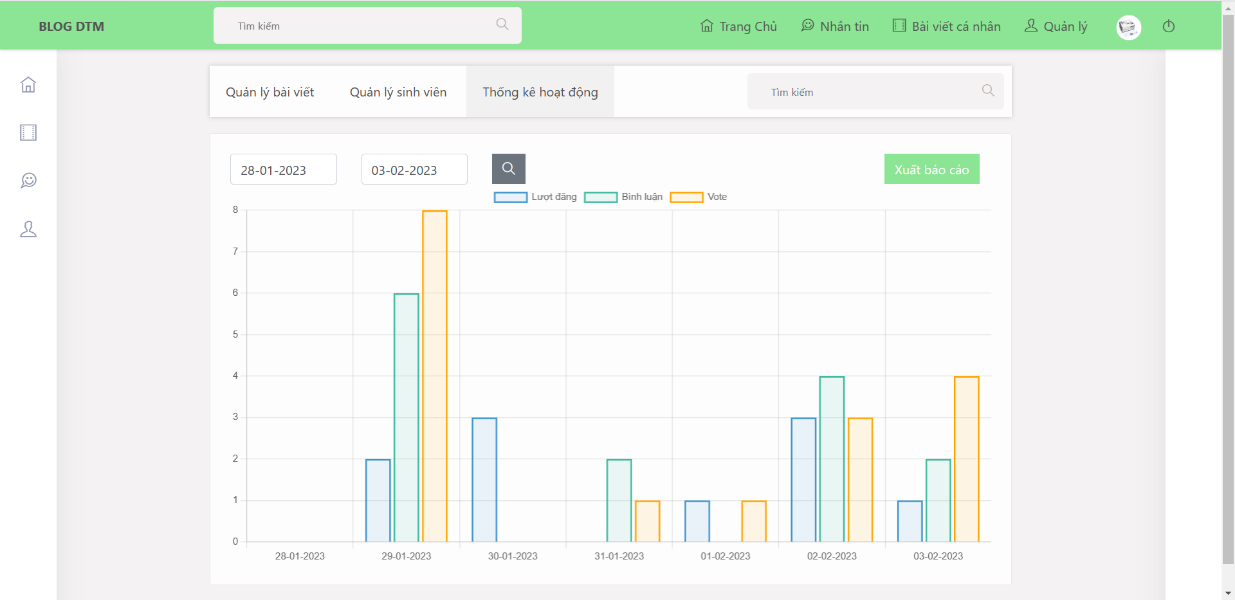
Hình 3.62: Giao diện thêm sinh viên

Hình 3.63: Giao diện chỉnh sửa thông tin sinh viên

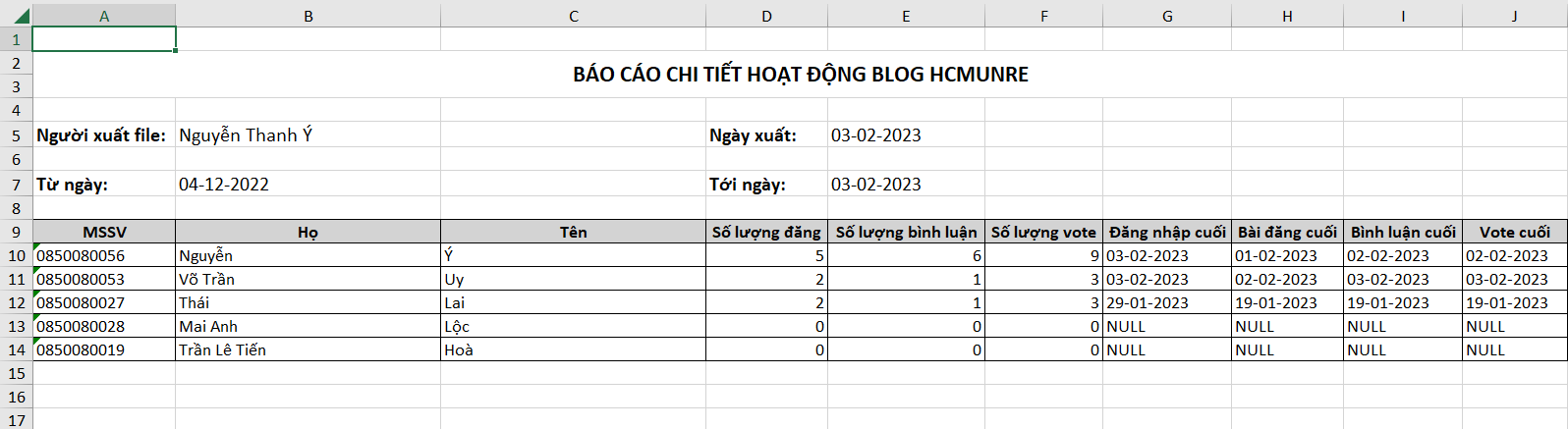
### Thống kê hoạt động

Hình 3.64: Tìm kiếm trên table

Hình 3.65: Mẫu Excel

Chức năng này thống kê được mật độ hoạt động của tất cả các người dùng có trong hệ thống, ngoại trừ người dùng với vai trò là người quản lý. Thông qua 3 cột số lượng bài đăng, số lượng bình luận, số lượng vote. Có thể thống kê trong một khoảng thời gian nhất định. Ngoài ra còn có thể xuất báo cáo thống kê hoạt động của các sinh viên.

Hình 3.66: Giao diện chức năng thống kê hoạt động



Hình 3.67: Thống kê theo tháng

Hình 3.68: Xuất báo cáo dưới dạng file Excel

1. **Kết luận và hướng phát triển**

## Kết luận

Đồ Qua quá trình thực tập em đã nghiên cứu và học hỏi được rất nhiều về cả kiến thức chuyên ngành cũng như các kỹ năng lập trình. Em đã củng cố cho bản thân nhiều kiến thức quan trọng như là: Sử dụng thành thạo ngôn ngữ lập trình C#, HTML, CSS…, thành thạo hơn về cơ sở dữ liệu SQL Server, các kỹ thuật lập trình trình, lập trình theo mô hình MVC. Cách để xây dựng một trang web chạy trên thời gian thực với thư viện SignalR.

Cùng với đó, em cũng học hỏi thêm được Javascript, Bootstrap biết sử dụng các công cụ như Github, Visual Studio Code các gói thư viện hữu ích như jQuery, Datatables, ChartJS và được ThS Phạm Trọng Huynh hướng dẫn về cách xử lý các tình huống thực tế và dữ liệu lớn. Ngoài ra em còn biết được thêm cách tự nghiên cứu tài liệu, biết được nhiều website cung cấp các tài liệu uy tín.

Bên cạnh nội dung báo cáo trên, trong quá trình thực tập em còn được mentor hướng xây dựng một website để phục vụ việc điều tra – tính giá trị sản phẩm các ngành chủ yếu, với dự án này em cũng đã được học thêm kiến thức về điều tra số liệu kinh tế, làm việc với kiểu dữ liệu Json, thống kê tiến độ nhiều lớp, làm việc với .NET Core, cơ sở dữ liệu SQL Server và biết được, để xây dựng một website có thể đưa vào thực tế sử dụng là vô cùng khó khăn. Thực tập không cung cấp kiến thức chuyên ngành mà em còn học được cách giao tiếp, trao đổi kết nối với các anh chị trong công ty, phát triển những kỹ năng mềm.

Để nói về những gì đóng góp được cho cơ quan thực tập thì chưa thực sự chưa có gì đáng kể ngoài việc hoàn thành các công việc được giao. Em vẫn đang nô lực rèn luyện thêm để có thể tham gia vào quy trình phát triển ứng dụng của công ty sau này. Những điều mà em có thể rút ra được sau quá trình thực tập đó là luôn cần có thái độ tốt, ham học hỏi, nhiệt tình trong công việc, tự tin và tăng khả năng giao tiếp, quan tâm tới mọi người.

## Hướng phát triển

Sau khi hoàn thành báo cáo thực tập tốt nghiệp ở trường, em nghĩ mình sẽ cố gắn nhiên cứu và tiếp tục các chức năng mới của sản phẩm để có thể phát triển hệ thống tiện lợi hơn, tối ưu các thuật toán để có thể xử lí dữ liệu lớn và hướng đến bài báo cáo tốt nghiệp vào cuối năm nay.

Các hướng và chức năng có thể sẽ phát triển trong tương lai:

* Tạo ra một phiên bản Android để có thể chạy cùng lúc song song 2 bản trên Website và Android.
* Mở rộng đối tượng người dùng như giáo viên, giảng viên, nhân viên của trường.
* Cho phép người dùng tạo một nhóm chat, nhóm hoạt động.
* Tối ưu thuật toán để xử lí dữ liệu lớn
* Cho phép nhắn tin và bình luận kèm theo tệp đính kèm.
* Xử lí dữ liệu cho đúng nghiệp vụ và phù hợp với thực tế.

# **DANH MỤC TÀI LIỆU THAM KHẢO**

[1] **Mark J. Price,** *C# 8.0 and .NET Core 3.0*.Packt Publishing, 2019.

[2] **Phạm Quang Huy, Vũ Trọng Luật**, *Lập trình với C#,* Thanh Niên.

[3] **Adam Freeman,** *Pro ASP.NET Core 3*.Apress, 2020.

[4] **Rick Anderson,** *ASP.NET Core MVC*, 03/02/2023, [Trực tuyến]. Link: <https://learn.microsoft.com/en-us/aspnet/core/tutorials/first-mvc-app/start-mvc?view=aspnetcore-3.1&tabs=visual-studio>. [Đã truy cập 03/02/2023].

[5] **Bradygaster,** *ASP.NET Core SignalR,* 05/01/2023, [Trực tuyến]. Link: https://learn.microsoft.com/en-us/aspnet/core/signalr/introduction?view=aspnetcore-3.1. [Đã truy cập 03/02/2023].

[6] **Jon Duckett,** *JavaScript and jQuery: Interactive Front-End Web Development*. Wiley, 2014.

[7] **Ben Frain**, *Responsive Web Design with HTML5 and CSS*. Packt Publishing, 2022.

[8] **Jon Duckett**, *HTML and CSS: Design and Build Websites*. John Wiley & Sons, 2011.

[9] **Walter Shields**, *SQL QuickStart Guide: The Simplified Beginner's Guide to Managing, Analyzing, and Manipulating Data With SQL*. 2019.

[10] **learn.microsoft.com,** *Entity Framework documentation,* [Trực tuyến]. Links: <https://learn.microsoft.com/en-us/ef/>. [Đã truy cập 03/02/2023].

[11] **getbootstrap.com**, *Bootstrap 4*, [Trực tuyến]. Links: <https://getbootstrap.com/docs/4.1/getting-started/introduction/>. [Đã truy cập 03/02/2023].

[12] **datatables.net**, *Datatables,* [Trực tuyến]. Links: <https://datatables.net/manual/>. [Đã truy cập 03/02/2023].

[13] **chartjs.org**, *Chart.JS*, [Trực tuyến]. Links: <https://www.chartjs.org/docs/latest/getting-started/>. [Đã truy cập 03/02/2023].

[14] **Getdatepicker.com**, *Datepicker*, [Trực tuyến]. Links: <https://getdatepicker.com/6/>. [Đã truy cập 03/02/2023].

[15] **epplussoftware.com**, *EPPlus*, [Trực tuyến]. Links: <https://github.com/EPPlusSoftware/EPPlus/wiki/Getting-Started>. [Đã truy cập 03/02/2023].

[16] **mimekit.net**, *MimeKit and MailKit are popular fully-featured email frameworks for .NET*, [Trực tuyến]. Links: <http://www.mimekit.net/docs/html/Introduction.htm>. [Đã truy cập 03/02/2023].

[17] **learn.microsoft.com,** *Visual Studio documentation*, [Trực tuyến]. Links: <https://learn.microsoft.com/en-us/visualstudio/windows/?view=vs-2022> . [Đã truy cập 03/02/2023].

[18] **code.visualstudio.com**, *Visual Studio Code*, [Trực tuyến]. Links: <https://code.visualstudio.com/docs> . [Đã truy cập 03/02/2023].

[19] **Jennifer Robbins**, *Learning Web Design*, 2014.

[20] **Jeffrey Zeldman,** *Designing with Web Standards*. 2003.