

TRƯỜNG ĐẠI HỌC KỸ THUẬT CÔNG NGHIỆP

KHOA ĐIỆN TỬ

BỘ MÔN: CÔNG NGHỆ THÔNG TIN



BÀI TIỂU LUẬN

MÔN HỌC

CÔNG NGHỆ PHẦN MỀM

**ĐỀ TÀI: XÂY DỰNG ỨNG DỤNG SỐ KẾT NỐI TÌNH
NGUYÊN VIÊN VỚI CÁC HOẠT ĐỘNG XÃ HỘI**

GIÁO VIÊN HƯỚNG DẪN : THS. NGUYỄN THỊ HƯƠNG
NHÓM SINH VIÊN THỰC HIỆN : 1. NGUYỄN TIỀN THẮNG
2. TRẦN CHIẾN THẮNG
3. NGUYỄN VĂN THÚ
4. HOÀNG THỊ XUÂN TRANG
LỚP : 58KTP

THÁI NGUYÊN - 2025

NHIỆM VỤ BÀI TẬP LỚN MÔN CÔNG NGHỆ PHẦN MỀM

Sinh viên:	1. Nguyễn Văn Thú	MSSV: K225480106062
	2. Nguyễn Tiên Thắng	MSSV: K225480106058
	3. Trần Chiến Thắng	MSSV: K215480106121
	4. Hoàng Thị Xuân Trang	MSSV: K225480106064

Lớp: K58KTP

Ngành: Kỹ thuật máy tính

Bộ môn: Công Nghệ Thông Tin

Giáo viên hướng dẫn: ThS. Nguyễn Thị Hương

1. Tên đề tài : **Xây dựng ứng dụng số kết nối tình nguyện viên với các hoạt động xã hội.**

2. Nội dung các phần thuyết minh và tính toán

- Khảo sát, phân tích hiện trạng của hoạt động tình nguyện hiện nay.
- Thiết kế cơ sở dữ liệu
- Xây dựng các tệp cơ sở dữ liệu
- Thiết kế chương trình
- Kiểm thử và triển khai chương trình

3. Các sản phẩm, kết quả :

- Thuyết minh báo cáo (1 quyển)
- Code chạy chương trình
- Video báo cáo

4. Ngày giao nhiệm vụ : / /2025

5. Ngày hoàn thành nhiệm vụ : **28 / 11 /2025**

GIÁO VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

TRƯỜNG ĐHKT&CN

KHOA ĐIỆN TỬ

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập – Tự do – Hạnh phúc

PHIẾU GHI ĐIỂM

HƯỚNG DẪN BÀI TẬP LỚN MÔN CÔNG NGHỆ PHẦN MỀM

Sinh viên:	1. Nguyễn Văn Thứ	MSSV: K225480106062
	2. Nguyễn Tiến Thắng	MSSV: K225480106058
	3. Trần Chiến Thắng	MSSV: K215480106121
	4. Hoàng Thị Xuân Trang	MSSV: K225480106064

Lớp: K58KTP

Ngành: Kỹ thuật máy tính

GVHD: ThS. Nguyễn Thị Hương

Đề tài: Xây dựng ứng dụng số kết nối tình nguyện viên với các hoạt động xã hội.

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

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

Xếp loại : Điểm :

Thái Nguyên, ngày tháng năm 2025

GIÁO VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

LỜI NÓI ĐẦU

Trong bối cảnh chuyển đổi số đang diễn ra mạnh mẽ, các hoạt động tình nguyện và công tác xã hội ngày càng đóng vai trò quan trọng trong việc gắn kết cộng đồng, lan tỏa tinh thần sẻ chia và trách nhiệm xã hội. Tuy nhiên, thực tế hiện nay cho thấy quá trình kết nối giữa tình nguyện viên và các tổ chức vẫn còn tồn tại nhiều bất cập như: khó khăn trong việc tìm kiếm hoạt động phù hợp, thiếu minh bạch trong quản lý, hạn chế về công cụ theo dõi và đánh giá kết quả hoạt động. Điều này không chỉ gây trở ngại cho tình nguyện viên mà còn ảnh hưởng đến hiệu quả tổ chức các chương trình.

Trước thách thức đó, việc xây dựng một ứng dụng số kết nối tình nguyện viên với các hoạt động xã hội hiện đại, tiện lợi và minh bạch là yêu cầu cấp thiết. Xuất phát từ thực tiễn này, nhóm chúng em lựa chọn thực hiện đề tài với mục tiêu phát triển một nền tảng trực tuyến thân thiện, dễ sử dụng, hỗ trợ tình nguyện viên trong việc tìm kiếm, đăng ký, theo dõi quá trình tham gia và tích lũy thành tích, đồng thời giúp các tổ chức tuyển chọn, quản lý và đánh giá hoạt động hiệu quả hơn.

Ứng dụng sẽ tích hợp các chức năng như: quản lý hồ sơ tình nguyện viên, tìm kiếm và đăng ký tham gia hoạt động, check-in/check-out điện tử, chứng nhận hoàn thành hoạt động, cũng như công cụ dành cho tổ chức trong việc tuyển chọn, quản lý sự kiện và báo cáo thống kê. Thông qua đó, hệ thống không chỉ giúp tiết kiệm thời gian, tăng tính minh bạch mà còn góp phần nâng cao chất lượng và hiệu quả của các hoạt động xã hội.

Đề tài mang tính ứng dụng thực tiễn cao, có thể triển khai rộng rãi cho các tổ chức xã hội, câu lạc bộ, đoàn thể, hoặc tích hợp với các nền tảng cộng đồng thông minh. Đây cũng là bước đi phù hợp với xu thế số hóa và phát triển bền vững trong thời đại công nghiệp 4.0.

Trong quá trình nghiên cứu và thực hiện, nhóm chúng em đã tiến hành khảo sát nhu cầu thực tế, phân tích yêu cầu, thiết kế mô hình hệ thống, xây dựng cơ sở dữ liệu và phát triển các chức năng chính của ứng dụng. Mặc dù còn một số hạn chế về kỹ thuật và phạm vi thử nghiệm, nhưng nhóm tin rằng sản phẩm đã thể hiện được tinh thần sáng tạo, khả năng vận dụng kiến thức vào thực tiễn và định hướng phát triển lâu dài.

Nhóm xin chân thành cảm ơn cô đã tận tình hướng dẫn, đóng góp ý kiến quý báu và hỗ trợ trong suốt quá trình thực hiện đề tài. Chúng em hy vọng sản phẩm này sẽ là nền tảng cho việc tiếp tục nghiên cứu, hoàn thiện và triển khai trong thực tế, góp phần nâng cao hiệu quả hoạt động tình nguyện và tạo dựng một cộng đồng gắn kết, nhân ái hơn.

LỜI CẢM ƠN

Chúng em xin bày tỏ lòng biết ơn sâu sắc nhất tới **Cô Nguyễn Thị Hương**, giảng viên hướng dẫn của chúng em.

Trong suốt quá trình thực hiện đề tài "Xây dựng ứng dụng số kết nối tình nguyện viên với các hoạt động xã hội", Cô đã tận tâm hướng dẫn, truyền đạt những kiến thức quý báu, và cung cấp những góp ý chuyên môn vô cùng kịp thời và xác đáng. Sự động viên và định hướng của Cô không chỉ giúp chúng em vượt qua những khó khăn về mặt kỹ thuật và thiết kế hệ thống, mà còn khơi dậy niềm đam mê và tinh thần trách nhiệm trong việc ứng dụng công nghệ để giải quyết các vấn đề xã hội.

Chúng em cũng xin gửi lời cảm ơn chân thành tới **Bộ môn Công nghệ thông tin** đã tạo điều kiện về cơ sở vật chất và môi trường học tập tốt nhất.

Chúng em hiểu rằng, dù đã nỗ lực hết mình, sản phẩm và báo cáo này không thể tránh khỏi những thiếu sót. Kính mong Cô và Hội đồng đánh giá lượng thứ và tiếp tục cho chúng em những ý kiến đóng góp quý báu để chúng em hoàn thiện hơn trong tương lai.

Xin chân thành cảm ơn!

Thái Nguyên, ngày 28 tháng 11 năm 2025

MỤC LỤC

LỜI NÓI ĐẦU	2
LỜI CẢM ƠN	4
LỜI CAM ĐOAN	6
CHƯƠNG I. GIỚI THIỆU ĐỀ TÀI VÀ KHẢO SÁT	7
1.1. Thực trạng của hệ thống hiện tại.....	7
1.2. Yêu cầu của hệ thống mới.....	11
CHƯƠNG II. PHÂN TÍCH VÀ ĐẶC TẨY YÊU CẦU	14
2.1. Yêu cầu bài toán	14
2.2. Phân tích thông tin vào và ra của hệ thống	16
2.3. Phân tích hệ thống theo kỹ thuật hướng đối tượng	17
2.4. Sơ đồ hoạt động	23
CHƯƠNG III. THIẾT KẾ HỆ THỐNG	28
3.1. Thiết kế cơ sở dữ liệu thông tin.....	28
3.2. Thiết kế Hệ thống Phần mềm.....	30
CHƯƠNG IV. CÀI ĐẶT VÀ KIỂM THỬ	35
4.1. Môi trường phát triển	35
4.2. Cấu trúc chương trình	38
4.3. Cấu trúc thư mục dự án	38
4.4. Hướng dẫn cài đặt dự án (REAME.md	41
4.5. Kết quả kiểm thử.....	47
CHƯƠNG V. NHẬN XÉT VÀ ĐÁNH GIÁ HỆ THỐNG	51
5.1. Những kết quả đạt được	51
5.2. Những hạn chế và hướng cải tiến	52
5.3. Đánh giá tổng quan hệ thống	53
TÀI LIỆU THAM KHẢO	54

LỜI CAM ĐOAN

Kính gửi,

Chúng em – nhóm sinh viên thực hiện đề tài: "**Xây dựng ứng dụng số kết nối tình nguyện viên với các hoạt động xã hội**" – xin cam đoan:

1. Đây là công trình nghiên cứu và phát triển do nhóm chúng em tự thực hiện dưới sự hướng dẫn của **Cô Nguyễn Thị Hương**.
2. Tất cả các dữ liệu, kết quả nghiên cứu, mã nguồn và tài liệu tham khảo được sử dụng trong báo cáo này đều được trích dẫn và ghi nguồn một cách rõ ràng, trung thực, và tuân thủ các quy tắc về sở hữu trí tuệ.
3. Sản phẩm và báo cáo này chưa từng được công bố hoặc sử dụng để bảo vệ cho bất kỳ đề tài nghiên cứu khoa học hay đồ án tốt nghiệp nào trước đây.
4. Chúng em hoàn toàn chịu trách nhiệm về tính trung thực của các nội dung được trình bày trong báo cáo này.

Chúng em xin chân thành cảm ơn!

Thái Nguyên, ngày 28 tháng 11 năm 2025

Thành viên nhóm	Ký tên
Nguyễn Tiến Thắng	
Trần Chiến Thắng	
Nguyễn Văn Thứ	
Hoàng Thị Xuân Trang	

CHƯƠNG I. GIỚI THIỆU ĐỀ TÀI VÀ KHẢO SÁT

1.1. Thực trạng của hệ thống hiện tại

1.1.1. Khảo sát tính thực tế của đề tài

Mục tiêu đề tài: Xây dựng ứng dụng số kết nối tình nguyện viên với các hoạt động xã hội, cho phép người dùng tạo và quản lý tài khoản, tìm kiếm – đăng ký tham gia các hoạt động phù hợp, check-in/check-out điện tử, nhận chứng nhận tham gia; đồng thời hỗ trợ tổ chức quản lý tình nguyện viên, sự kiện, thống kê và báo cáo hiệu quả.

Tính thực tiễn và tính cấp thiết của đề tài :

- Nhu cầu xã hội: Phong trào tình nguyện ngày càng phát triển, đặc biệt trong học sinh, sinh viên và cộng đồng trẻ. Tuy nhiên, việc kết nối còn phân tán, thiếu hệ thống quản lý tập trung. Một ứng dụng trực tuyến sẽ giúp đồng bộ thông tin, nâng cao hiệu quả tổ chức và khuyến khích tinh thần cống hiến.
- Lợi ích cho tình nguyện viên: Dễ dàng tìm kiếm hoạt động phù hợp với thời gian – địa điểm, đăng ký nhanh chóng, theo dõi quá trình tham gia và tích lũy thành tích cá nhân. Hạn chế thủ tục giấy tờ và giúp chứng minh đóng góp trong học tập, xin học bổng hoặc việc làm.
- Lợi ích cho tổ chức: Quản lý danh sách tình nguyện viên hiệu quả; hỗ trợ tuyển chọn theo tiêu chí; điểm danh điện tử bằng mã QR; báo cáo và thống kê nhanh chóng; tăng tính minh bạch và độ tin cậy của chương trình.
- Tính khả thi về kỹ thuật: Công nghệ web, cơ sở dữ liệu, hệ thống xác nhận bằng mã QR, lưu trữ cloud, và báo cáo thống kê đều đã phổ biến, có thể triển khai bằng các framework hiện đại (Laravel, Django, Node.js, React Native...). Bảo mật dữ liệu có thể thực hiện với các công nghệ mã hóa, xác thực hai lớp.

- Tính khả thi về kinh tế: Chi phí phát triển ứng dụng ban đầu ở mức hợp lý, phù hợp với các dự án thử nghiệm trong trường học, đoàn thể hoặc tổ chức phi lợi nhuận. Khi triển khai rộng rãi, chi phí vận hành có thể được bù đắp từ tài trợ, hỗ trợ xã hội, hoặc tích hợp các mô hình gây quỹ và quảng bá.
- Tính pháp lý và an toàn: Ứng dụng cần tuân thủ quy định pháp luật về quản lý dữ liệu cá nhân và bảo mật thông tin. Việc xác nhận tham gia có thể được chuẩn hóa bằng chứng nhận điện tử, giúp minh bạch hóa hoạt động. Đồng thời, tổ chức có thể ban hành điều khoản sử dụng để đảm bảo quyền lợi cho cả tình nguyện viên và đơn vị tổ chức.

1.1.2. Khảo sát hiện trạng.

Hiện nay, việc tổ chức và quản lý hoạt động tình nguyện chủ yếu thực hiện thông qua các kênh truyền thống như thông báo trực tiếp tại trường học, cơ quan đoàn thể, hoặc đăng tải trên mạng xã hội (Facebook, Zalo). Ưu điểm của hình thức này là quen thuộc, dễ tiếp cận với nhiều đối tượng và cho phép người phụ trách xử lý trực tiếp các tình huống như thay đổi số lượng người tham gia, hoãn hoạt động. Tuy nhiên, nó tồn tại nhiều hạn chế như thông tin phân tán, khó tìm kiếm tập trung, quy trình đăng ký còn thủ công, thiếu công cụ theo dõi minh bạch và chưa ghi nhận đầy đủ đóng góp của tình nguyện viên. Vì vậy, hệ thống hiện tại chưa đáp ứng tốt nhu cầu kết nối, quản lý và phát triển bền vững phong trào tình nguyện.

Ưu điểm của phương thức quản lý truyền thống:

- Thói quen và tin cậy: Nhiều tình nguyện viên, đặc biệt là các nhóm nhỏ, vẫn quen với cách đăng ký trực tiếp qua cán bộ phụ trách hoặc qua mạng xã hội.
- Ít phụ thuộc vào công nghệ: Không cần smartphone hay internet; phù hợp với tình nguyện viên chưa quen dùng ứng dụng số.

- Quan hệ trực tiếp: Dễ trao đổi và xử lý ngay các trường hợp đặc biệt (bỏ sung người, thay đổi kế hoạch, hủy tham gia).

Nhược điểm của phương thức quản lý truyền thống:

- Tốn thời gian cho tình nguyện viên: Phải chờ xác nhận từ ban tổ chức, dễ nhầm lẫn hoặc bỏ sót thông tin khi đăng ký qua nhiều kênh.
- Hiệu suất thấp & dễ sai sót: Việc quản lý bằng giấy tờ hoặc file Excel thủ công dễ xảy ra trùng lặp, sai thông tin và khó theo dõi số lượng người tham gia chính xác.
- Thiếu tiện ích tra cứu thời gian thực: Thông tin về hoạt động thay đổi (địa điểm, thời gian) không được cập nhật kịp thời, tình nguyện viên dễ bị bỏ lỡ thông báo.
- Hạn chế về chứng nhận và minh bạch: Khó có cơ chế ghi nhận chính thức, tình nguyện viên không có hồ sơ tích lũy rõ ràng về các hoạt động đã tham gia.
- Quản lý báo cáo kém: Các đơn vị tổ chức khó tổng hợp, thống kê dữ liệu tham gia để đánh giá và lập kế hoạch cho các chương trình tiếp theo.

1.1.3. Khảo sát ứng dụng số kết nối.

Hiện nay, nhiều nền tảng quản lý và kết nối tình nguyện viên hiện đại đã được triển khai, đặc biệt ở các tổ chức phi lợi nhuận, trường đại học và các chương trình cộng đồng lớn. Các hệ thống này thường tích hợp những chức năng nổi bật như đăng ký hoạt động trực tuyến, quản lý hồ sơ tình nguyện viên, điểm danh bằng mã QR, chứng nhận điện tử và báo cáo thống kê chi tiết. Chúng mang lại sự tiện lợi, minh bạch cho tình nguyện viên, đồng thời giúp các tổ chức tối ưu hóa việc quản lý, tuyển chọn và đánh giá hiệu quả hoạt động. Tuy nhiên, vẫn còn tồn tại một số hạn chế như giao diện chưa thật sự thân thiện với mọi đối tượng, quy

trình xác nhận tham gia đôi khi phức tạp, hệ thống dễ quá tải khi số lượng người dùng lớn, và vấn đề bảo mật dữ liệu cá nhân cần được đảm bảo chặt chẽ hơn.

Đặc điểm chung của hệ thống tình nguyện hiện đại:

- Đăng ký tham gia hoạt động qua website hoặc ứng dụng di động.
- Quản lý hồ sơ tình nguyện viên, bao gồm thông tin cá nhân, kỹ năng và lịch sử tham gia.
- Cho phép tìm kiếm, đăng ký và theo dõi hoạt động trực tuyến.
- Điểm danh và xác nhận tham gia bằng mã QR hoặc check-in điện tử.
- Cung cấp chứng nhận điện tử sau khi hoàn thành hoạt động.
- Hệ thống quản lý tập trung, cập nhật số lượng tham gia theo thời gian thực.
- Tích hợp chức năng thông báo, nhắc lịch và gửi tin nhắn cho tình nguyện viên.

Hạn chế tồn tại của hệ thống tình nguyện hiện đại:

- Giao diện một số hệ thống chưa thân thiện, gây khó khăn cho người ít sử dụng công nghệ.
- Quy trình xác nhận, điểm danh hoặc cấp chứng nhận còn phức tạp, chưa tối ưu trải nghiệm người dùng.
- Hệ thống có thể bị quá tải khi tổ chức sự kiện lớn với số lượng tình nguyện viên đông.
- Vấn đề bảo mật dữ liệu cá nhân và quyền riêng tư cần được đảm bảo chặt chẽ hơn.
- Một bộ phận tình nguyện viên chưa quen sử dụng ứng dụng số, vẫn có xu hướng đăng ký truyền thống qua mạng xã hội hoặc trực tiếp.

Các nền tảng tình nguyện hiện nay được xem là đang ở mức **bán hiện đại – chuyển tiếp giữa mô hình quản lý truyền thống và hệ thống quản lý thông minh**. Tuy đã có sự cải tiến đáng kể về tính minh bạch và hiệu quả, nhưng vẫn chưa đáp ứng được nhu cầu tối ưu hóa trải nghiệm người dùng và khai thác triệt để lợi ích của công nghệ số trong công tác quản lý tình nguyện.

1.2. Yêu cầu của hệ thống mới

1.2.1. Những vấn đề cần thảo goř

Mặc dù nhu cầu tham gia các hoạt động xã hội và thiện nguyện ngày càng tăng cao, nhưng việc kết nối giữa tình nguyện viên và các tổ chức, chương trình vẫn còn nhiều bất cập. Một số vấn đề nổi bật cần giải quyết bao gồm:

Thứ nhất, khó khăn trong việc tìm kiếm thông tin. Hiện nay, các hoạt động tình nguyện thường được thông báo rầm rộ qua mạng xã hội, website riêng lẻ hoặc truyền miệng. Điều này khiến tình nguyện viên gặp khó khăn trong việc tiếp cận thông tin đầy đủ, kịp thời và chính xác.

Thứ hai, thiếu kênh kết nối tập trung. Các tổ chức xã hội chưa có một nền tảng thống nhất để đăng tải hoạt động, tuyển tình nguyện viên và quản lý quá trình tham gia. Sự phân tán này dẫn đến lãng phí nguồn lực và khó theo dõi hiệu quả.

Thứ ba, chưa có hệ thống quản lý hồ sơ tình nguyện viên. Nhiều bạn trẻ tham gia nhiều hoạt động khác nhau nhưng không được ghi nhận, không có nơi lưu trữ thông tin, kinh nghiệm và thành tích, dẫn đến khó khăn trong việc đánh giá mức độ cống hiến.

Thứ tư, hạn chế trong tương tác và phản hồi. Việc liên lạc, thông báo thay đổi lịch trình hoặc xác nhận tham gia thường thực hiện thủ công, dễ gây nhầm lẫn, bỏ sót hoặc chậm trễ.

Thứ năm, thiếu tính minh bạch và an toàn. Một số tình nguyện viên lo ngại thông tin hoạt động không chính thống hoặc bị lừa đảo. Đồng thời, các tổ chức cũng gặp khó trong việc xác minh thông tin của tình nguyện viên.

Cuối cùng, vấn đề động lực và gắn kết cộng đồng. Nhiều tình nguyện viên chưa có môi trường để giao lưu, chia sẻ kinh nghiệm hay được ghi nhận đóng góp, dẫn đến việc giảm hứng thú và duy trì sự gắn bó lâu dài.

1.2.2. Yêu cầu phát sinh hệ thống mới

Trong quá trình khảo sát và phân tích nhu cầu, hệ thống không chỉ dừng lại ở việc kết nối cơ bản giữa tình nguyện viên và tổ chức, mà còn phát sinh thêm nhiều yêu cầu nhằm nâng cao tính tiện lợi, minh bạch và hiệu quả:

Đăng ký và quản lý tài khoản

- Cho phép tình nguyện viên và tổ chức tạo tài khoản riêng.
- Hỗ trợ đăng nhập qua email, số điện thoại hoặc mạng xã hội.
- Quản lý hồ sơ cá nhân (thông tin, kỹ năng, kinh nghiệm, lịch sử tham gia).

Đăng tải và quản lý hoạt động

- Tổ chức có thể đăng thông tin chi tiết về chương trình: tên hoạt động, thời gian, địa điểm, số lượng cần tuyển, yêu cầu kỹ năng.
- Cho phép chỉnh sửa, hủy, hoặc gia hạn hoạt động khi cần.

Tìm kiếm và gợi ý thông minh

- Hỗ trợ tình nguyện viên tìm kiếm hoạt động theo vị trí địa lý, lĩnh vực (môi trường, giáo dục, nhân đạo...), thời gian rảnh.
- Hệ thống có thể gợi ý hoạt động phù hợp dựa trên sở thích và kỹ năng đã lưu.

Quản lý đăng ký và xác nhận tham gia

- Tình nguyện viên có thể đăng ký tham gia trực tuyến.
- Hệ thống gửi thông báo xác nhận và cập nhật danh sách tham gia theo thời gian thực.
- Cho phép hủy đăng ký hoặc thay đổi thông tin tham gia dễ dàng.

Thông báo và tương tác

- Gửi thông báo qua email, SMS hoặc ứng dụng về thay đổi lịch trình, địa điểm, hoặc nhắc nhở trước khi diễn ra.
- Hỗ trợ nhắn tin, trao đổi giữa tình nguyện viên và tổ chức.

Ghi nhận và đánh giá

- Sau mỗi hoạt động, tổ chức có thể xác nhận sự tham gia và đánh giá tình nguyện viên.
- Tình nguyện viên có thể nhận được chứng nhận online hoặc điểm tích lũy để ghi nhận đóng góp.

Tính minh bạch và an toàn

- Xác thực tổ chức trước khi cho phép đăng hoạt động.
- Bảo mật thông tin cá nhân và mã hóa dữ liệu giao dịch.

Báo cáo và thống kê

- Hệ thống cung cấp báo cáo về số lượng hoạt động, số tình nguyện viên tham gia, thống kê theo lĩnh vực/thời gian.
- Hỗ trợ quản lý hiệu quả cho tổ chức và minh chứng cho thành tích của tình nguyện viên.

CHƯƠNG II. PHÂN TÍCH VÀ ĐẶC TẨY YÊU CẦU

2.1. Yêu cầu bài toán

2.1.1. Mô tả tổng quan bài toán

Hệ thống được thiết kế để kết nối tình nguyện viên với các hoạt động xã hội thông qua một ứng dụng web hoặc di động đơn giản. Mục tiêu là giúp tình nguyện viên dễ dàng tìm kiếm và đăng ký tham gia các hoạt động, trong khi các tổ chức có thể quản lý sự kiện và tình nguyện viên một cách hiệu quả. Đây là một dự án học tập, vì vậy hệ thống tập trung vào các chức năng cơ bản, dễ triển khai, và sử dụng các công nghệ phổ biến phù hợp với sinh viên (ví dụ: HTML/CSS/JavaScript, PHP, hoặc Python với cơ sở dữ liệu MySQL).

Mục tiêu chính:

- Kết nối tình nguyện viên và tổ chức qua một nền tảng tập trung.
- Hỗ trợ đăng ký, điểm danh, và cấp chứng nhận đơn giản.
- Đảm bảo giao diện thân thiện và dễ sử dụng.
- Phù hợp với môi trường học tập, sử dụng công nghệ cơ bản.

Đối tượng người dùng:

- Tình nguyện viên: Sinh viên, cá nhân muốn tham gia các hoạt động xã hội.
- Tổ chức: Đoàn trường, câu lạc bộ, hoặc tổ chức phi lợi nhuận nhỏ.

2.1.2. Yêu cầu chức năng

Hệ thống được thiết kế với các chức năng cơ bản, dễ triển khai trong phạm vi bài tập môn học:

a. Đối với tình nguyện viên

- Đăng ký/Đăng nhập: Tạo tài khoản bằng email và mật khẩu; đăng nhập đơn giản.
- Quản lý hồ sơ: Cập nhật thông tin cá nhân (tên, email, số điện thoại, kỹ năng cơ bản).
- Tìm kiếm hoạt động: Tìm kiếm sự kiện theo tên, địa điểm, hoặc ngày.
- Đăng ký tham gia: Chọn sự kiện và gửi yêu cầu tham gia.

- Điểm danh: Sử dụng mã QR hoặc nhập mã sự kiện để điểm danh.
- Nhận chứng nhận: Xem và tải chứng nhận tham gia (PDF đơn giản).
- Nhận thông báo: Xem thông báo về xác nhận đăng ký hoặc thay đổi sự kiện.

b. Đối với tổ chức

- Đăng ký/Đăng nhập: Tạo tài khoản tổ chức với thông tin cơ bản.
- Quản lý sự kiện: Tạo, chỉnh sửa, hoặc xóa sự kiện (tên, ngày, địa điểm, số lượng tình nguyện viên cần).
- Quản lý đăng ký: Xem danh sách tình nguyện viên đăng ký, duyệt hoặc từ chối.
- Điểm danh: Tạo mã QR cho sự kiện hoặc xác nhận thủ công.
- Cấp chứng nhận: Gửi chứng nhận tham gia cho tình nguyện viên.
- Báo cáo cơ bản: Xem danh sách tham gia và số liệu đơn giản (số lượng tình nguyện viên).

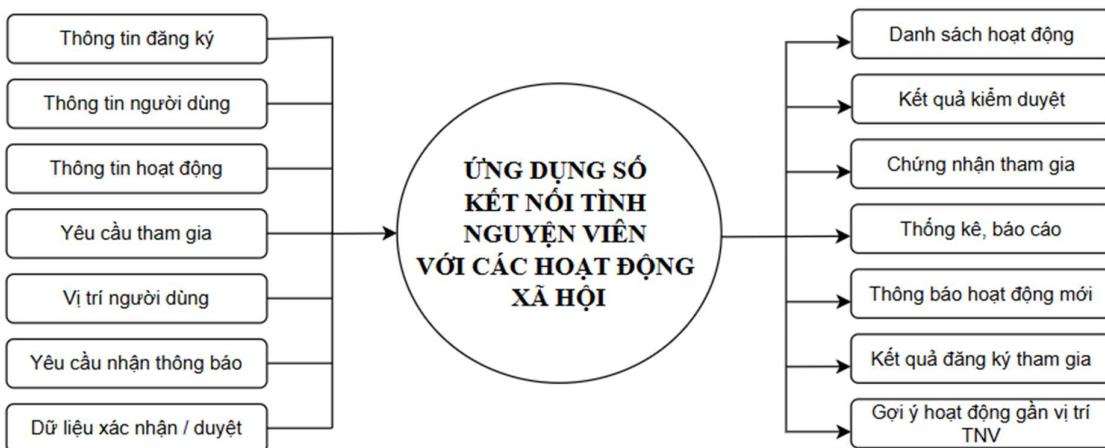
c. Quản trị viên hệ thống

- Quản lý tài khoản: Xem và khóa/mở tài khoản người dùng.
- Quản lý sự kiện: Kiểm duyệt các sự kiện do tổ chức đăng tải.

2.1.3. Yêu cầu phi chức năng

- Hiệu suất: Hỗ trợ tối đa 100 người dùng đồng thời (phù hợp với dự án nhỏ).
- Bảo mật: Mã hóa mật khẩu cơ bản (ví dụ: MD5 hoặc bcrypt); đăng nhập an toàn.
- Tính thân thiện: Giao diện đơn giản, hỗ trợ tiếng Việt, dễ sử dụng trên web hoặc di động.
- Tính khả dụng: Hệ thống hoạt động ổn định trên máy chủ localhost hoặc hosting miễn phí.
- Công nghệ: Sử dụng các công nghệ cơ bản như PHP/Laravel, Python/Django, hoặc Node.js với MySQL; hỗ trợ tạo mã QR bằng thư viện đơn giản (như QRCode.js).
- Tính pháp lý: Đảm bảo thông tin người dùng được lưu trữ an toàn, có điều khoản sử dụng cơ bản.

2.2. Phân tích thông tin vào và ra của hệ thống



Hình 2.1. Sơ đồ luồng vào ra của hệ thống

2.2.1. Thông tin vào (Input)

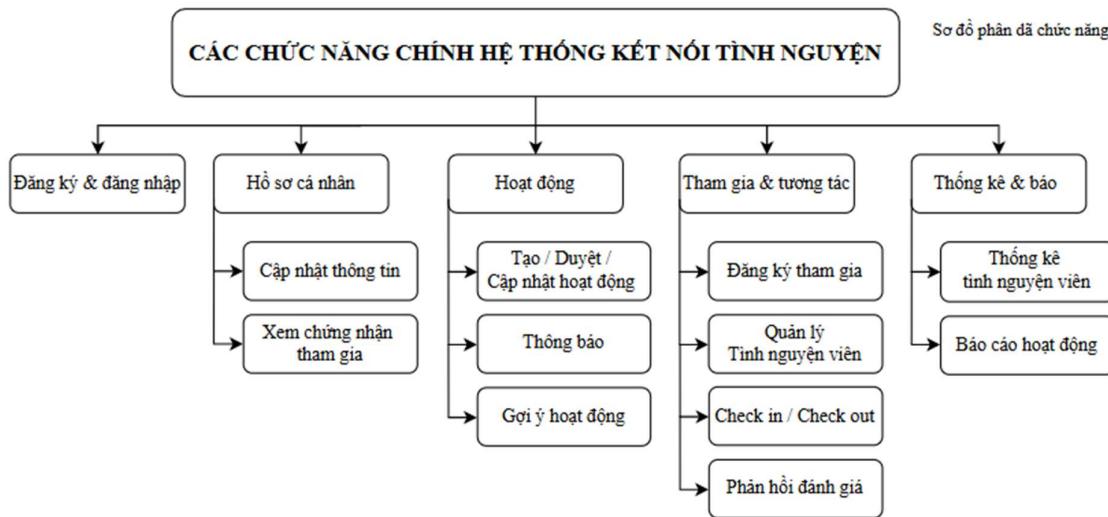
- **Từ tình nguyện viên:**
 - Thông tin đăng ký: email, mật khẩu, tên, số điện thoại.
 - Thông tin hồ sơ: kỹ năng, địa chỉ.
 - Yêu cầu tìm kiếm: từ khóa (tên sự kiện, địa điểm, ngày).
 - Dữ liệu đăng ký: chọn sự kiện, gửi yêu cầu.
 - Dữ liệu điểm danh: mã QR hoặc mã sự kiện.
- **Từ tổ chức:**
 - Thông tin đăng ký: tên tổ chức, email, số điện thoại.
 - Thông tin sự kiện: tên, ngày, địa điểm, số lượng cần tuyển.
 - Dữ liệu quản lý: duyệt/từ chối đăng ký, xác nhận điểm danh.
- **Từ hệ thống bên ngoài:**
 - Dữ liệu từ thư viện mã QR (tạo mã cho điểm danh).
 - Dữ liệu từ Google Maps API (nếu có, để hiển thị địa điểm).

2.2.2. Thông tin ra (Output)

- **Cho tình nguyện viên:**
 - Danh sách sự kiện: kết quả tìm kiếm.

- Xác nhận đăng ký: thông báo qua giao diện hoặc email.
- Mã QR điểm danh: hiển thị trên giao diện.
- Chứng nhận: file PDF hoặc hình ảnh đơn giản.
- Thông báo: xác nhận đăng ký, thay đổi sự kiện.
- Cho tổ chức:
 - Danh sách đăng ký: thông tin tình nguyện viên, trạng thái.
 - Mã QR sự kiện: để điểm danh.
 - Báo cáo: danh sách tham gia, số lượng tình nguyện viên.
- Cho hệ thống:
 - Nhật ký: log các thao tác (đăng nhập, đăng ký, điểm danh).

2.2.3. Sơ đồ phân dã chức năng hệ thống



Hình 2.2: Sơ đồ phân dã chức năng

2.3. Phân tích hệ thống theo kỹ thuật hướng đối tượng

2.3.1. Biểu đồ Use Case

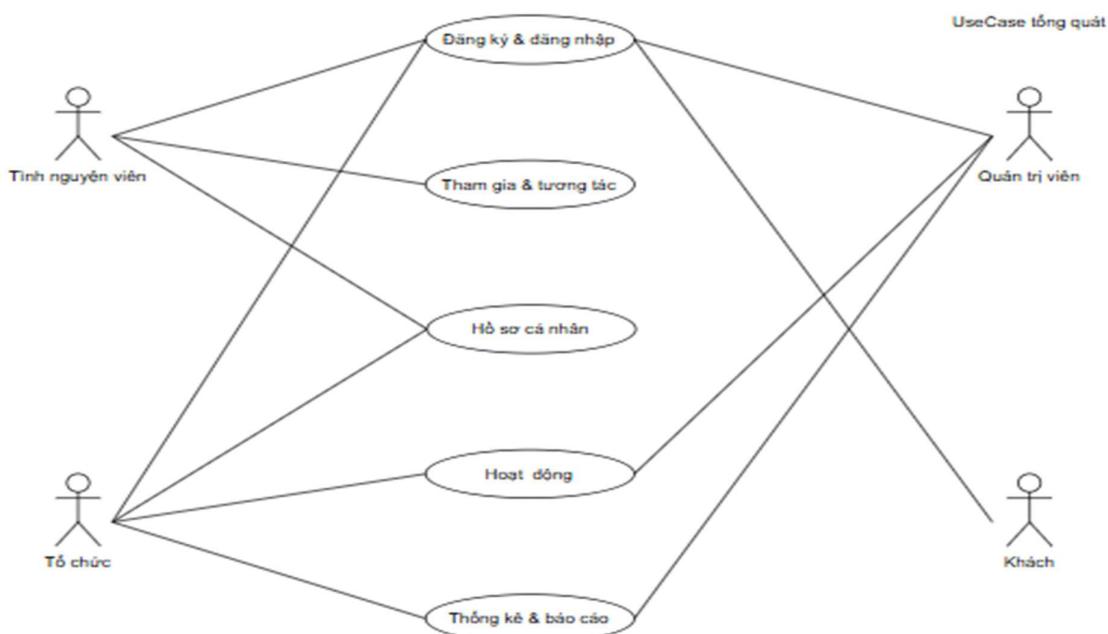
Biểu đồ Use Case mô tả các chức năng chính của hệ thống và tương tác giữa các tác nhân (Tình nguyện viên, Tổ chức, Quản trị viên).

Biểu đồ Use Case mô tả các chức năng chính của hệ thống và tương tác giữa các tác nhân (Tình nguyện viên, Tổ chức, Quản trị viên).

a. Các Use Case chính

1. Tình nguyện viên:

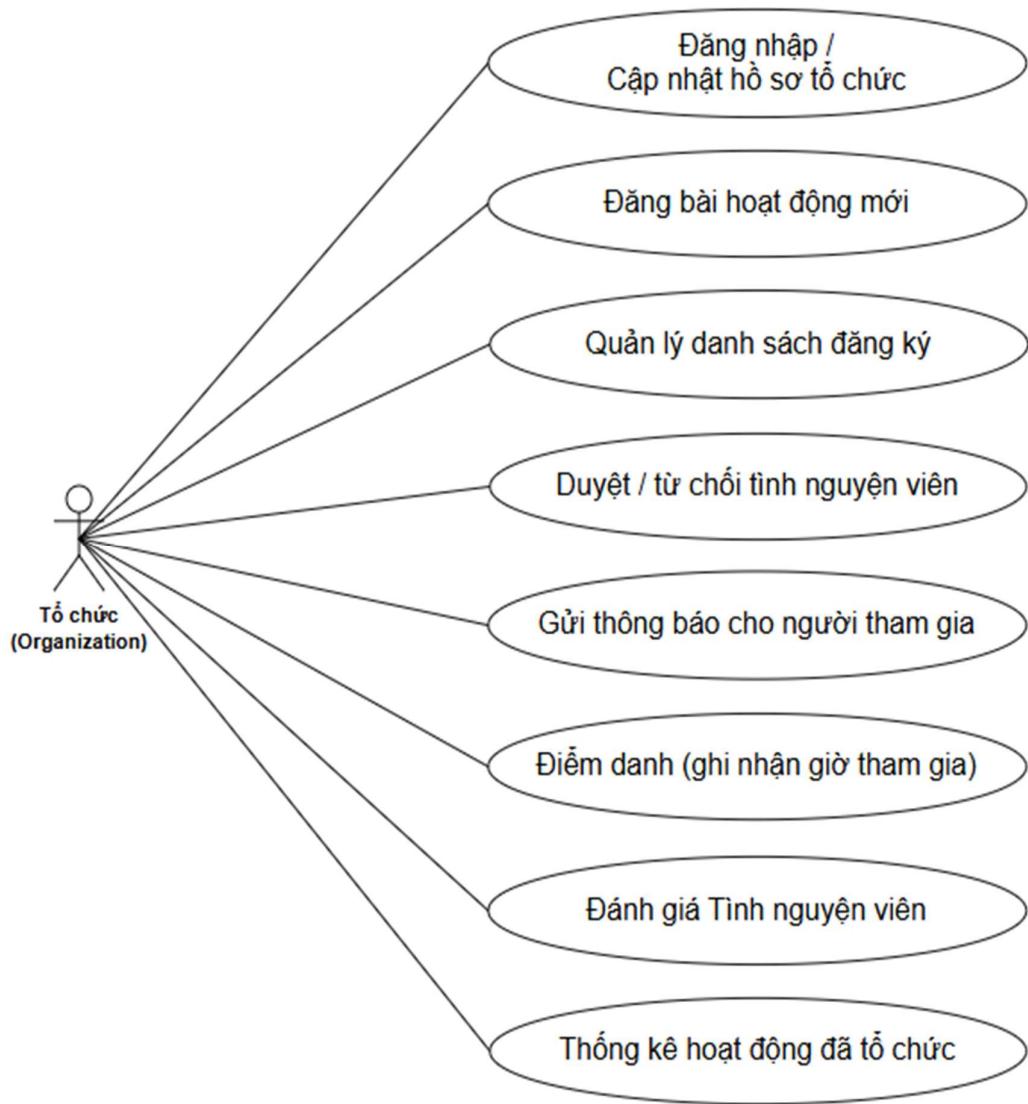
- o Đăng ký/Đăng nhập.
- o Quản lý hồ sơ.
- o Tìm kiếm sự kiện.
- o Đăng ký tham gia.
- o Điểm danh bằng mã QR.
- o Tải chứng nhận.
- o Xem thông báo.



Hình 2.3. Usecase tổng quát

2. Tổ chức:

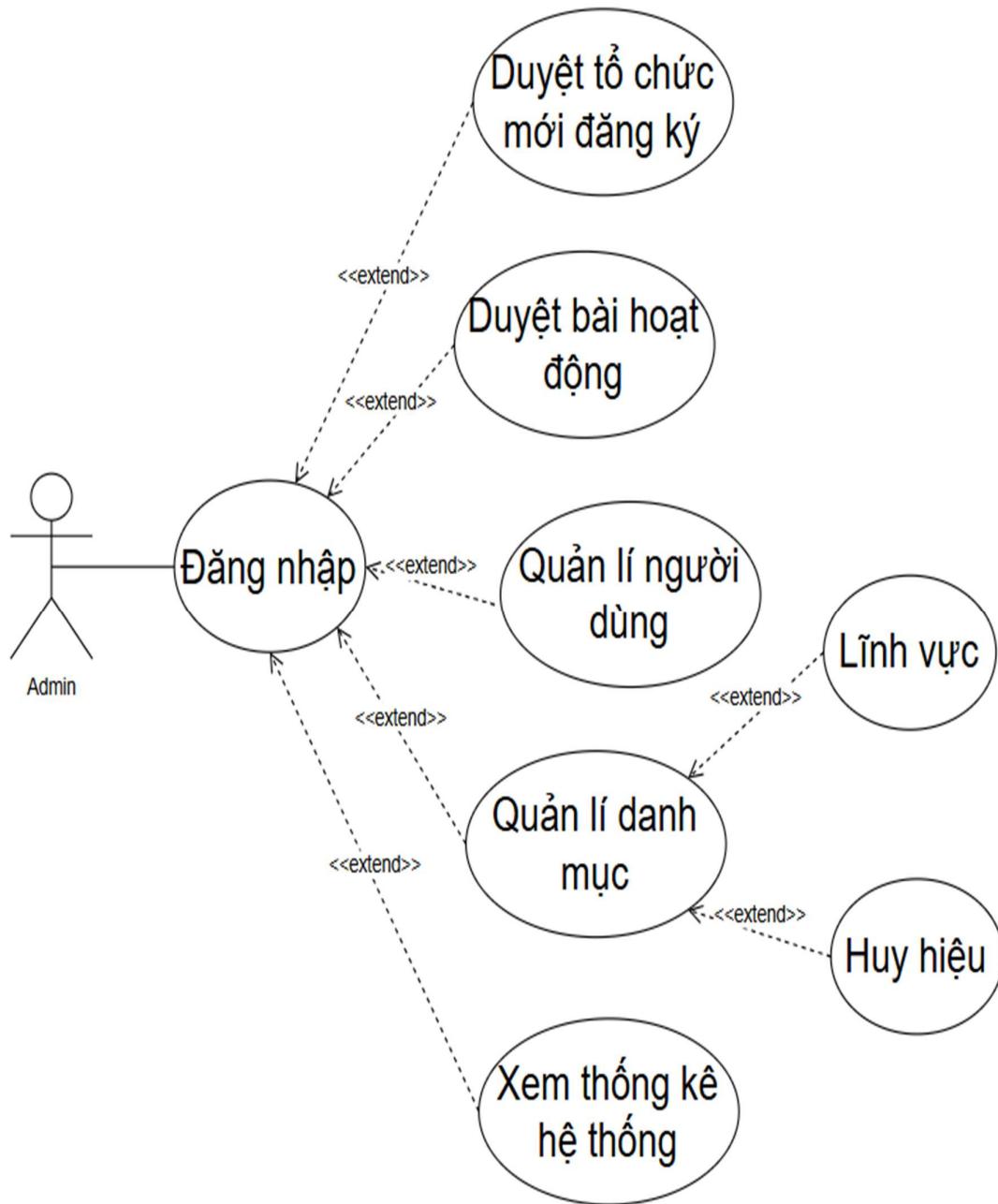
- o Đăng ký/Đăng nhập.
- o Tạo/Chỉnh sửa/Xóa sự kiện.
- o Quản lý đăng ký.
- o Tạo mã QR điểm danh.
- o Cấp chứng nhận.
- o Xem báo cáo.



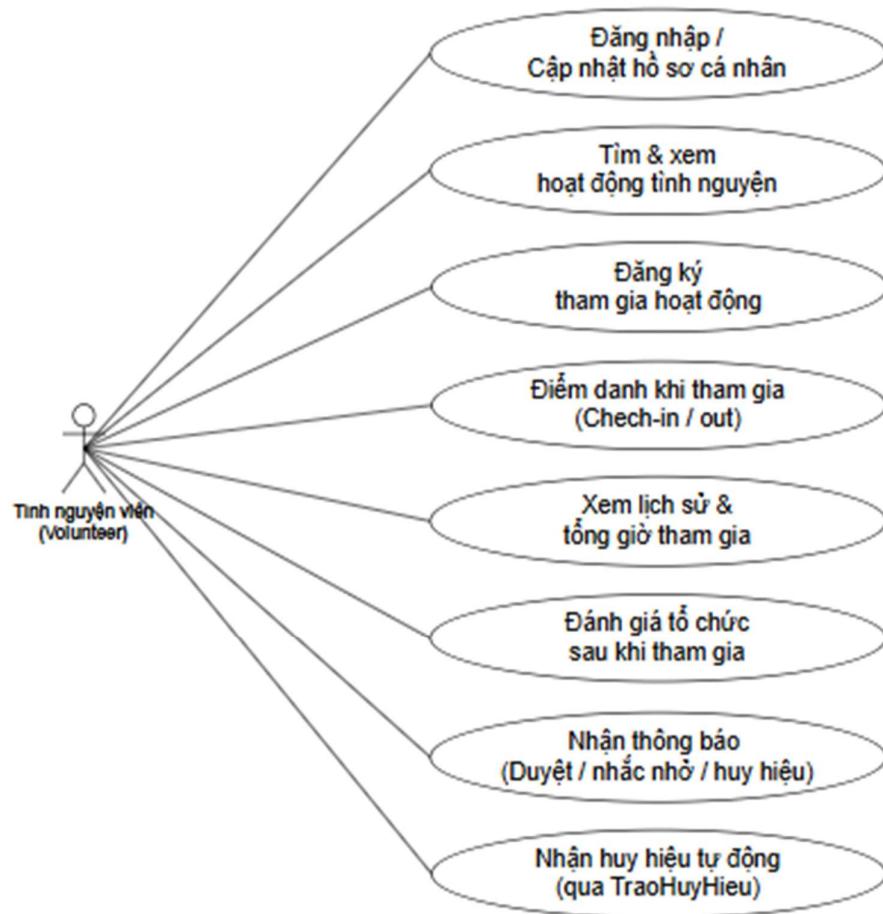
Hình 2.4. Usecase Tổ chức

3. Quản trị viên:

- Quản lý tài khoản.
- Kiểm duyệt sự kiện.



Hình 2.5. Usecase admin

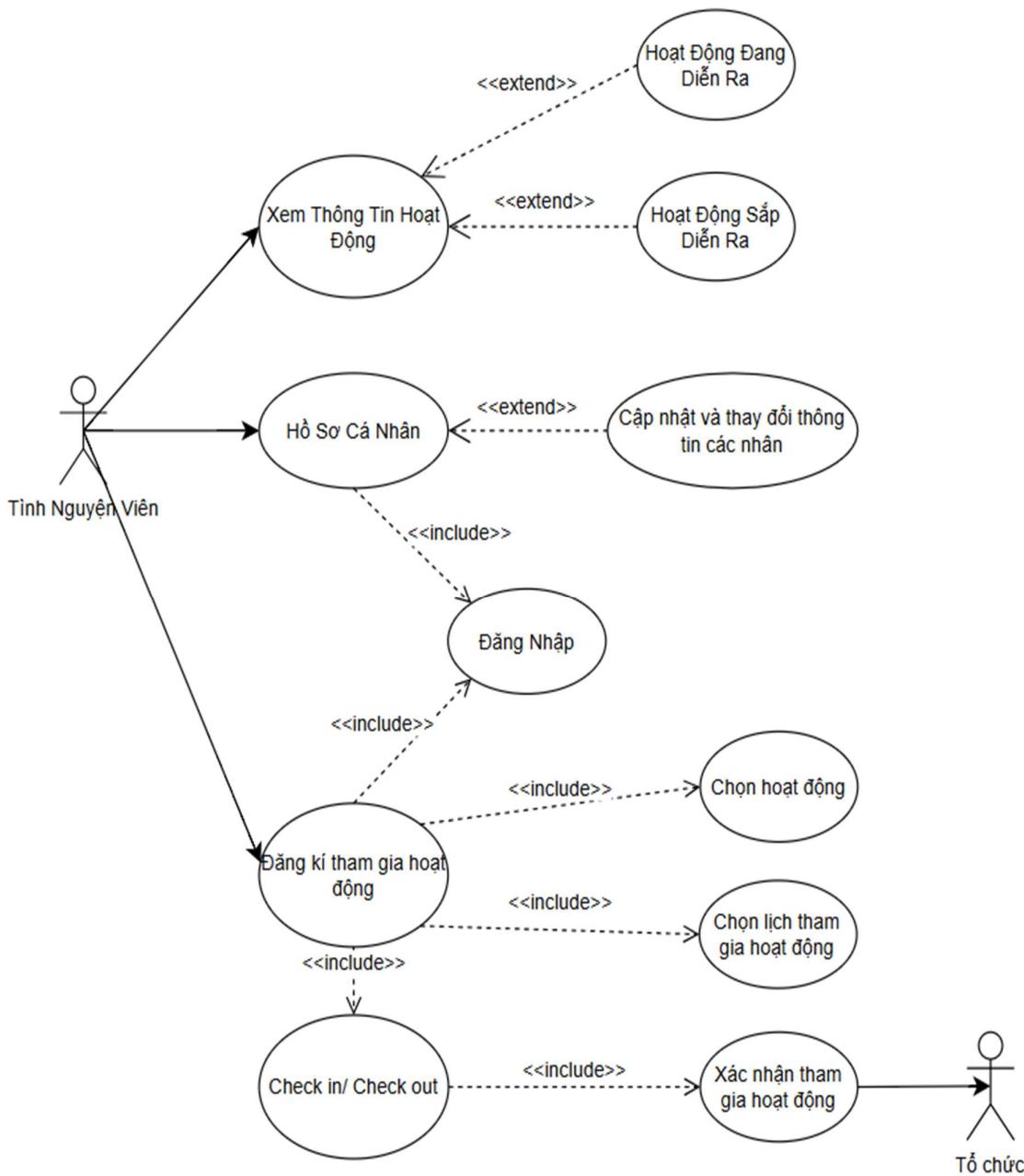


Hình 2.6. Usecase tình nguyện viên

c. Mô tả Use Case mẫu (Đăng ký tham gia)

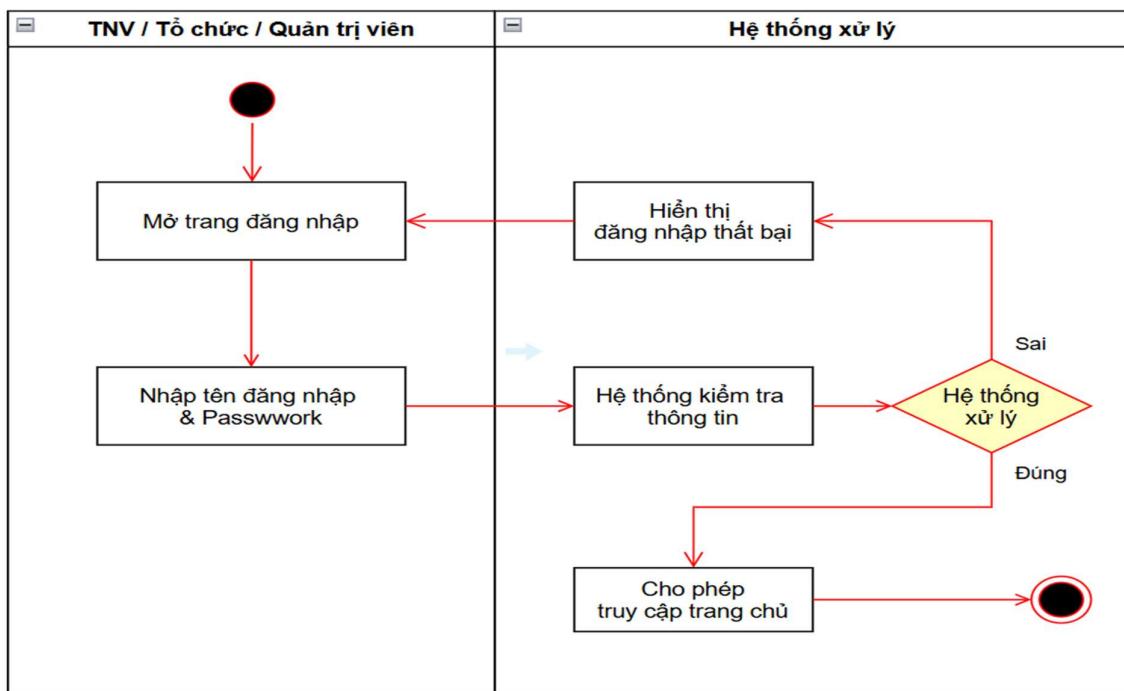
- **Tác nhân:** Tình nguyện viên.
- **Mô tả:** Tình nguyện viên tìm kiếm sự kiện và gửi yêu cầu tham gia.
- **Điều kiện tiên quyết:** Đã đăng nhập, sự kiện còn mở đăng ký.
- **Luồng chính:**
 1. Tình nguyện viên tìm kiếm sự kiện.
 2. Chọn sự kiện và nhấn "Đăng ký".
 3. Hệ thống kiểm tra số lượng còn trống.
 4. Hệ thống gửi thông báo xác nhận.

- **Luồng phụ:** Nếu sự kiện đầy, hiển thị thông báo và gợi ý sự kiện khác.
- **Kết quả:** Yêu cầu đăng ký được gửi đến tổ chức.

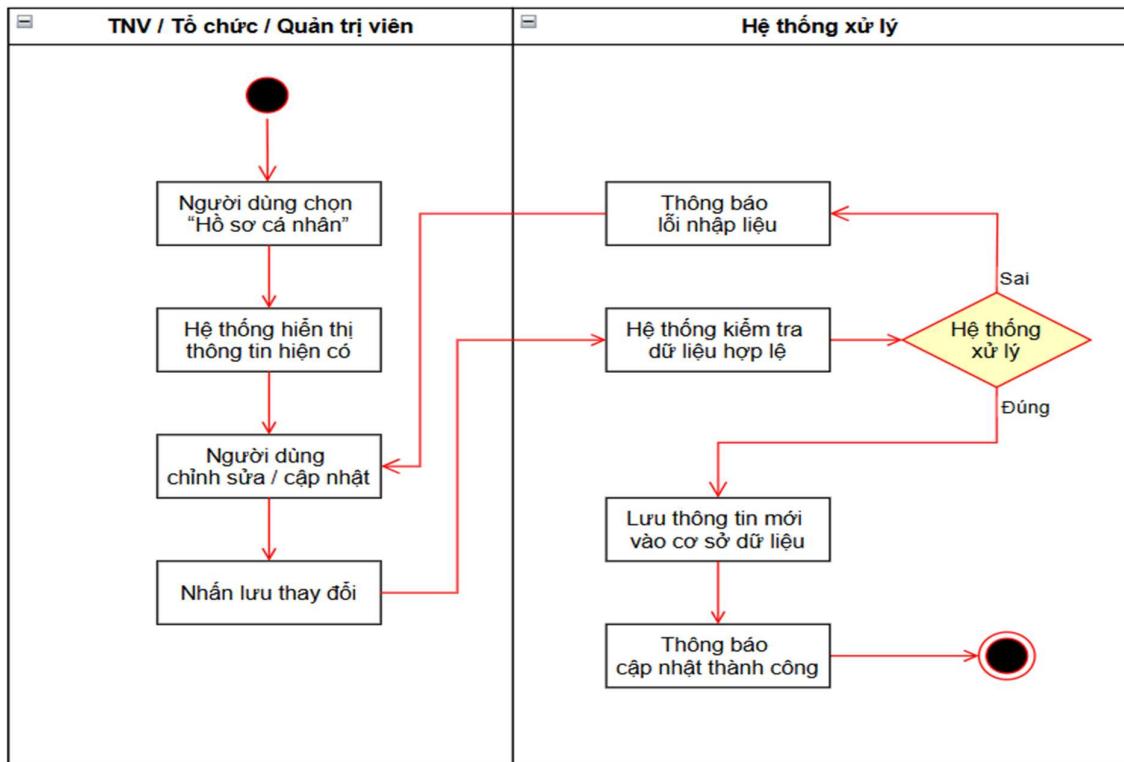


Hình 2.7. Usecase đăng kí tham gia hoạt động của tình nguyện viên

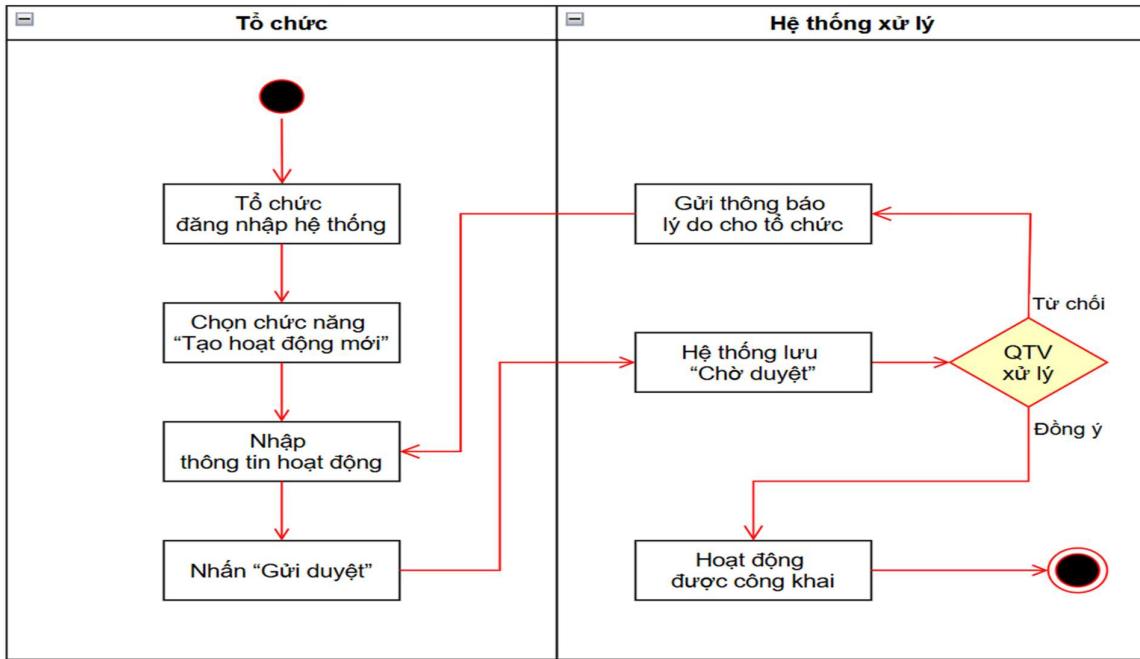
2.4. Sơ đồ hoạt động



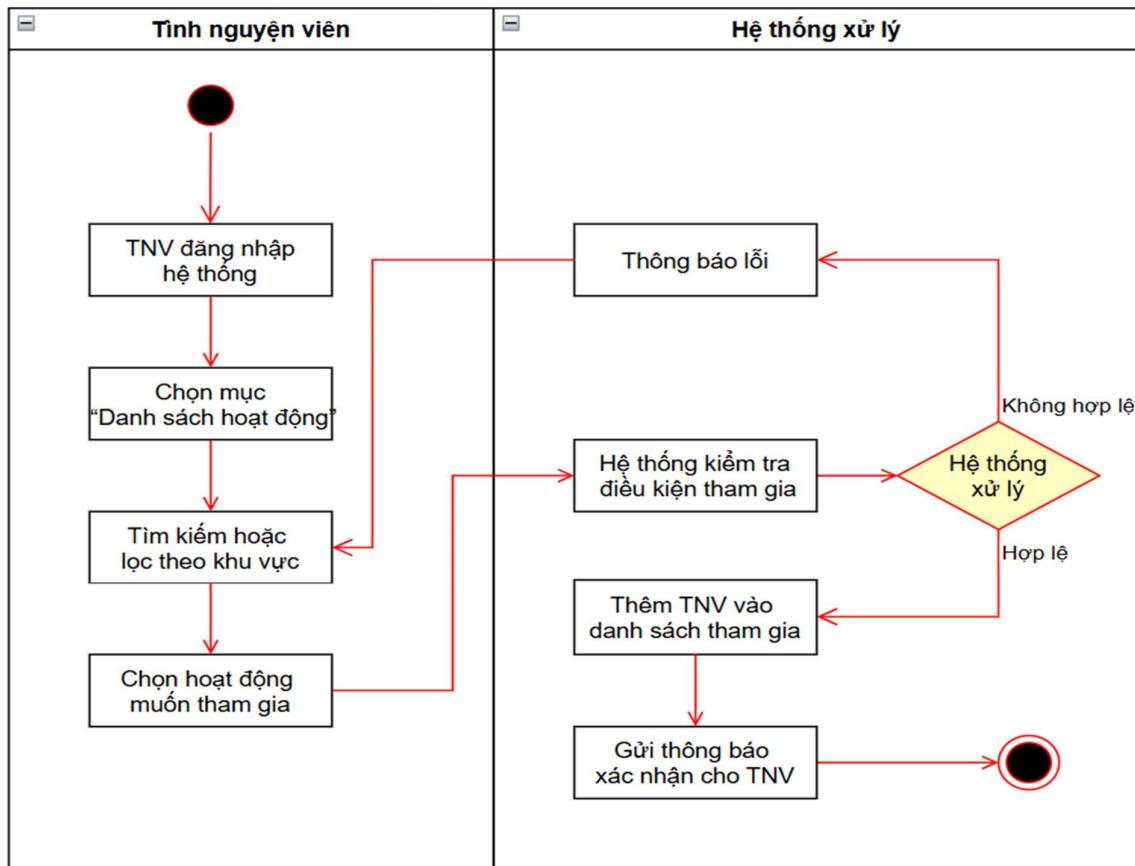
Hình 2.8. Sơ đồ hoạt động đăng nhập



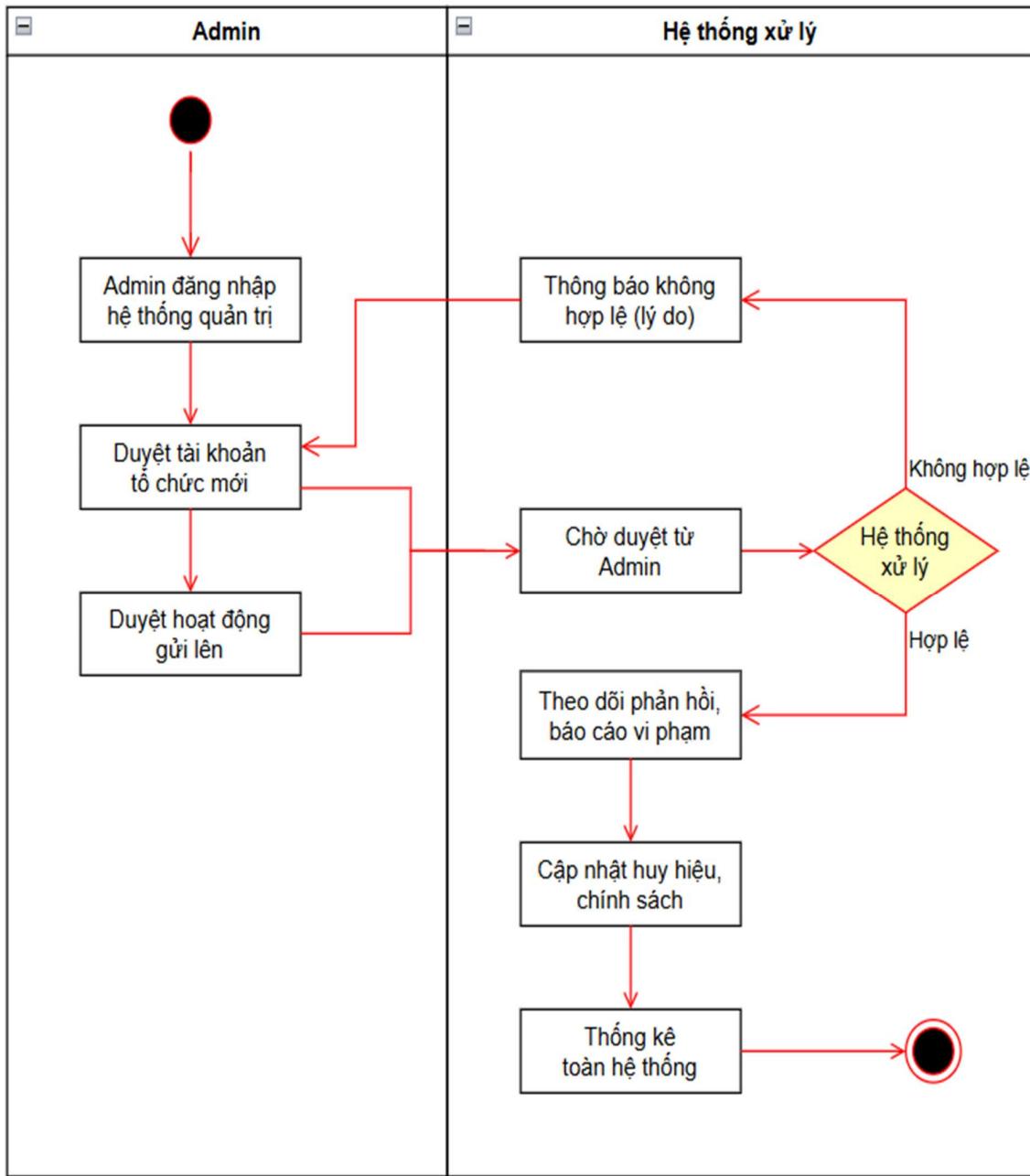
Hình 2.9. Sơ đồ hoạt động quản lý hồ sơ cá nhân



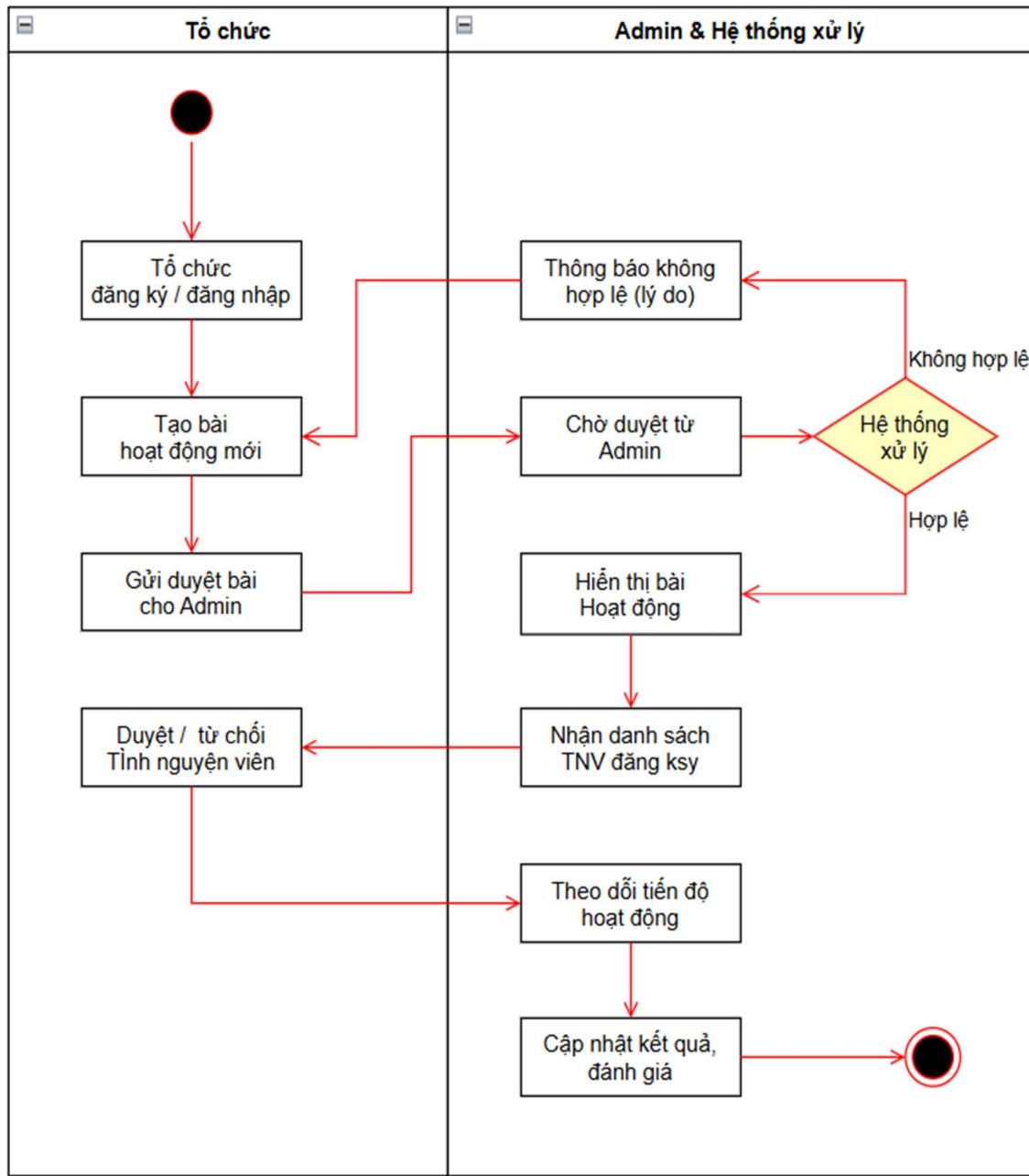
Hình 2.10. Sơ đồ đoạt động tạo và duyệt hoạt động



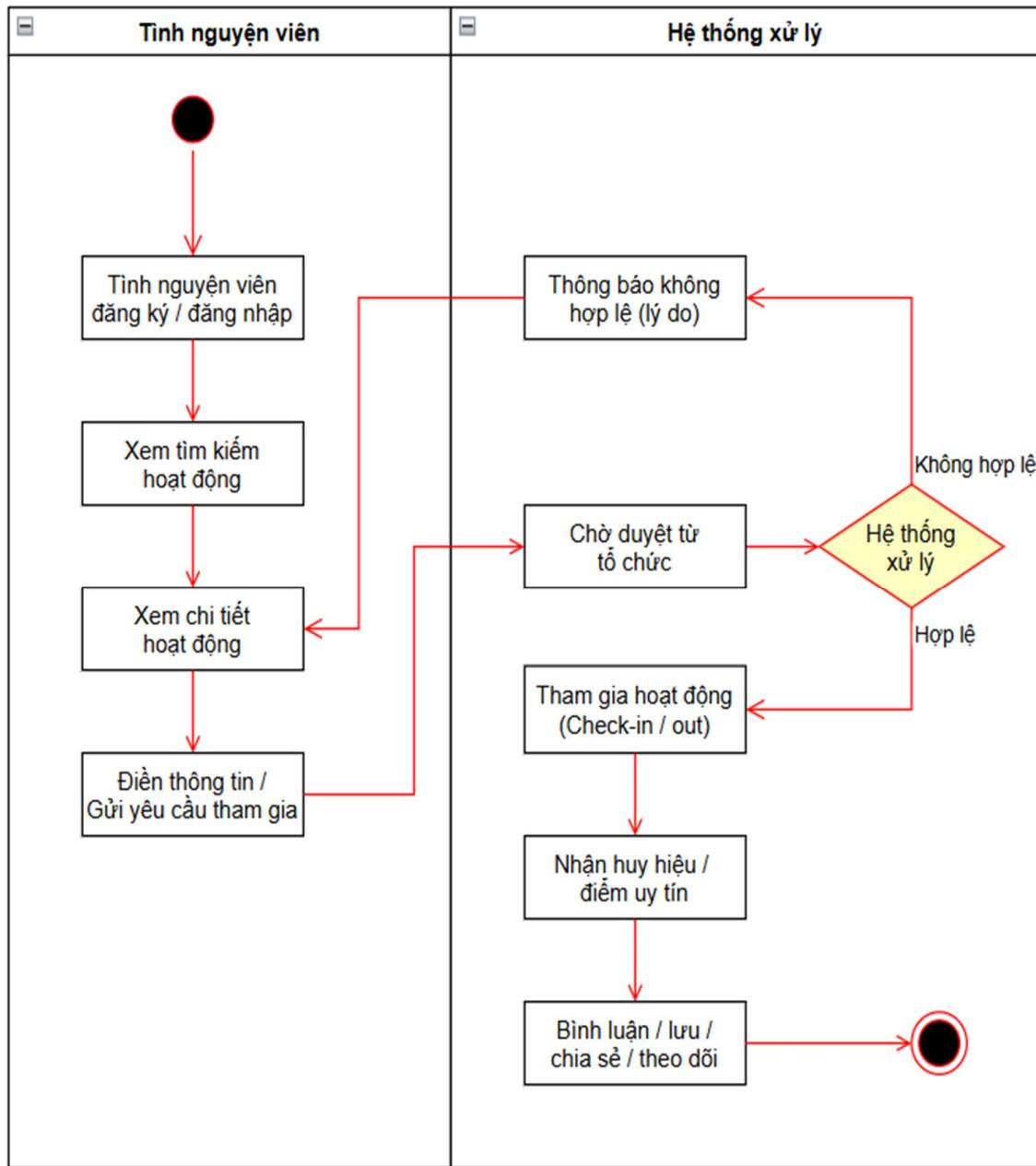
Hình 2.11. Sơ đồ hoạt động đăng ký tham gia hoạt động



Hình 2.12. Sơ đồ hoạt động của admin



Hình 2.13. Sơ đồ hoạt động của tổ chức



Hình 2.14. Sơ đồ hoạt động của tình nguyện viên

CHƯƠNG III. THIẾT KẾ HỆ THỐNG

3.1. Thiết kế cơ sở dữ liệu thông tin

3.1.1. Chuẩn hóa quan hệ

Để khắc phục các hiện tượng dư thừa dữ liệu, dữ liệu không nhất quán, dữ liệu lặp, nhập nhằng dữ liệu thì cần phải kiểm tra, rà soát, thanh lọc dữ liệu trước khi đưa vào bảng.

Quá trình đó gọi là quá trình chuẩn hóa mà nó sẽ được thực hiện qua ba bước lần lượt gọi là :

Dạng chuẩn 1: 1NF

Dạng chuẩn 2 : 2NF

Dạng chuẩn 3 : 3NF

Dạng chuẩn thứ nhất 1NF

Một quan hệ được coi là ở dạng chuẩn thứ nhất nếu tất cả các thuộc tính đều ở dạng đơn, tức là không tồn tại một tập hợp các thuộc tính giống nhau (thuộc tính lặp).

Theo định nghĩa phụ thuộc hàm thì nếu tồn tại 1 tập các thuộc tính lặp thì tại một thời điểm với mọi giá trị của khoá sẽ không thể có một giá trị duy nhất cho từng thuộc tính khác trong bảng. Vậy đưa về dạng chuẩn thứ nhất tức là loại bỏ nhóm thuộc tính lặp. Ta tách thành 2 thuộc tính là dòng đơn hàng xuất và dòng đơn hàng nhập

Dạng chuẩn thứ hai 2NF

Một quan hệ được gọi là ở dạng chuẩn thứ 2 (2NF) nếu nó là ở dạng 1NF và mọi phụ thuộc hàm giữa khoá với các thuộc tính đều là sơ đẳng, có nghĩa là mọi thuộc tính đều phải phụ thuộc hàm vào toàn bộ khoá chứ không phải một phần của khoá. Vậy để đưa một quan hệ về dạng 2NF là phải loại bỏ mọi phụ thuộc hàm bộ phận vào khoá. Mọi bảng (thực thể) với chỉ một thuộc tính làm khoá đều được xem như là ở dạng 2NF.

Dạng chuẩn thứ ba 3NF

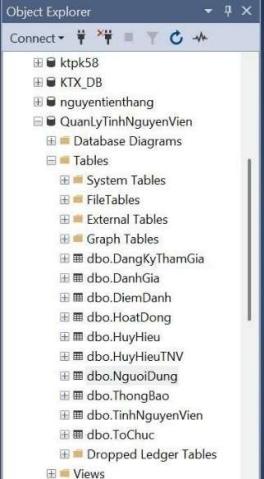
Một quan hệ được gọi là ở dạng chuẩn 3NF nếu nó là 2NF và các phụ thuộc hàm giữa khoá và các thuộc tính khác là trực tiếp hay nói cách khác là mỗi thuộc tính không phụ thuộc hàm vào bất kỳ thuộc tính nào trong quan hệ ngoài khoá. Do đó để đưa các quan hệ về dạng chuẩn 3NF ta phải loại bỏ các phụ thuộc hàm không phải khoá.

Do đó để đưa các quan hệ về dạng chuẩn 3NF ta phải loại bỏ các phụ thuộc hàm không phải khoá.

3.1.2. Mô tả chi tiết cơ sở dữ liệu

Dựa vào các yêu cầu của hệ thống và quá trình phân tích hệ thống thông tin từ chương II mà bây giờ ta có thể xây dựng được cơ sở dữ liệu.

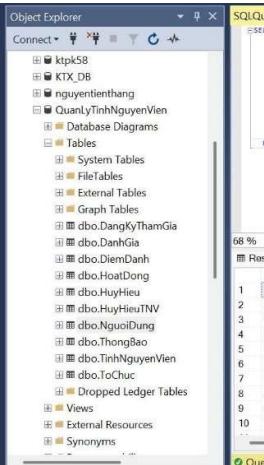
- Bảng NguoiDung



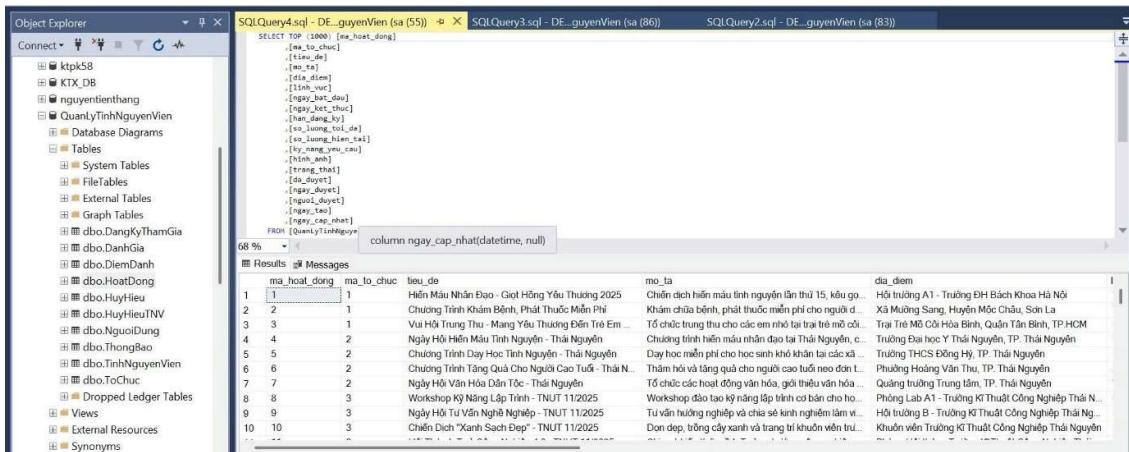
The screenshot shows the Microsoft SQL Server Object Explorer on the left and a grid view of the 'NguoiDung' table on the right. The table has 14 columns: ma_nguo..., email, mat_kha..., ho_ten, so_dien..., vai_tr..., anh_dai..., trang_th..., da_xac_t..., ngay_tao, and ngay_c... . There are 49 rows of data, including a header row and several sample entries like 'admin@...', 'chuathap...', and 'nguyenvn...'.

Hình 3.1. Bảng NguoiDung

- Bảng TinhNguyenVien

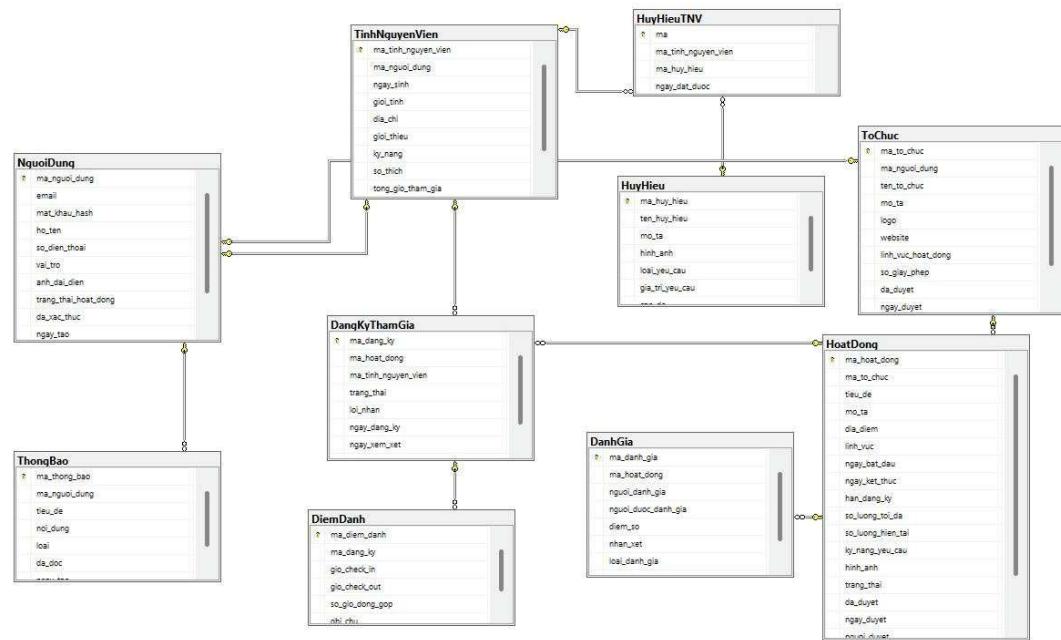


The screenshot shows the Microsoft SQL Server Object Explorer on the left and a results grid on the right. A query is being run against the 'NguoiDung' table to select columns [email], [mat_kha..._hash], [so_dien_thoai], [vai_tr...], [anh_dai...], [trang_th...], [da_xac_t...], [ngay_tao], and [ngay_c...]. The results grid shows 14 rows of data, including entries for 'Quản Trị Viên Hệ Thống', 'Hội Chủ Thập Đỗ Việt Nam', and 'Trần Thị B'.



Hình 3.2. Bảng TinhNguyenVien

3.1.3. Sơ đồ thực thể liên kết



Hình 3.3. Bảng sơ đồ liên kết thực thể

3.2. Thiết kế Hệ thống Phần mềm

3.2.1. Mục tiêu Thiết kế Phần mềm

Thiết kế phần mềm là giai đoạn chuyển hóa các yêu cầu đã được phân tích thành mô hình kỹ thuật cụ thể có thể triển khai và vận hành. Đối với ứng dụng số kết nối tình nguyện viên với các hoạt động xã hội, mục tiêu của thiết kế phần mềm là xây dựng một nền tảng web và/hoặc ứng dụng di động cho phép Tình nguyện

viên (TNV) có thể tìm kiếm hoạt động, đăng ký tham gia, theo dõi lịch sử giờ hoạt động, và cho phép Tổ chức/Đơn vị tạo, quản lý và phê duyệt hoạt động một cách nhanh chóng, tiện lợi và minh bạch.

Hệ thống cần đảm bảo khả năng xử lý chính xác số lượng TNV đăng ký cho mỗi hoạt động, tránh tình trạng vượt quá chỉ tiêu, đồng thời hỗ trợ nhiều người dùng (cả TNV và Tổ chức) thao tác đồng thời. Ngoài ra, hệ thống phải dễ sử dụng, có tốc độ phản hồi nhanh, dễ bảo trì và mở rộng trong tương lai, đặc biệt trong bối cảnh nhu cầu tham gia và tổ chức hoạt động thiện nguyện trực tuyến ngày càng tăng.

3.2.2. Kiến trúc Tổng thể của Hệ thống

Hệ thống ứng dụng số kết nối tình nguyện viên được xây dựng dựa trên mô hình kiến trúc Client – Server kết hợp với mô hình Ba Lớp (Three-Tier Architecture) nhằm đảm bảo tính linh hoạt, khả năng bảo trì và mở rộng.

- **Lớp Giao diện (Presentation Layer):**
 - Sử dụng các công nghệ như **[HTML, CSS, JavaScript/React Native/Flutter]** để thiết kế giao diện trực quan, thân thiện với người dùng.
 - Lớp này chịu trách nhiệm tiếp nhận thao tác người dùng (đăng ký, tìm kiếm hoạt động, đăng ký tham gia) và gửi yêu cầu đến máy chủ thông qua các giao thức HTTP, sử dụng cơ chế gửi – nhận dữ liệu bằng JSON hoặc AJAX.
- **Lớp Xử lý Nghiệp vụ (Business Logic Layer):**
 - Được triển khai bằng **[Ngôn ngữ lập trình và Framework cụ thể, ví dụ: Python Django/Node.js Express]** để xây dựng API và xử lý dữ liệu.
 - Lớp này đóng vai trò trung tâm, thực hiện các logic nghiệp vụ quan trọng như xác thực tài khoản, xử lý đăng ký tham gia, kiểm tra điều kiện (ví dụ: TNV đã đủ kỹ năng yêu cầu chưa), thực hiện gửi thông

báo, cập nhật trạng thái hoạt động và phản hồi kết quả cho người dùng.

- **Lớp Dữ liệu (Data Layer):**

- Được triển khai trên hệ quản trị cơ sở dữ liệu [SQL Server/MySQL/PostgreSQL], nơi lưu trữ toàn bộ thông tin liên quan đến người dùng, hồ sơ TNV, hoạt động, đăng ký tham gia và phản hồi/đánh giá.
- Lớp này kết nối với CSDL thông qua thư viện/ORM để truy vấn, ghi nhận và cập nhật dữ liệu theo yêu cầu.

Việc tách biệt ba lớp giúp hệ thống dễ dàng nâng cấp, chỉnh sửa và đảm bảo tính an toàn dữ liệu, đồng thời cho phép mở rộng trong tương lai, ví dụ như phát triển thêm ứng dụng di động mà không cần thay đổi cấu trúc lõi của hệ thống.

3.2.3. Các Chức năng Chính của Hệ thống

Hệ thống được xây dựng nhằm hỗ trợ người dùng dễ dàng tra cứu thông tin hoạt động, đăng ký tham gia và quản lý hồ sơ tình nguyện một cách thuận tiện. Các chức năng của hệ thống được chia theo nhóm người dùng bao gồm Tình nguyện viên (TNV), Tổ chức Quản lý Hoạt động, và Quản trị viên Hệ thống.

Chức năng dành cho Tình nguyện viên:

- **Quản lý Tài khoản:** Tạo tài khoản, đăng nhập, cập nhật hồ sơ cá nhân (kỹ năng, địa chỉ, kinh nghiệm).
- **Tìm kiếm và Đăng ký:** Tìm kiếm hoạt động theo địa điểm, kỹ năng yêu cầu, thời gian. Xem chi tiết hoạt động và thực hiện đăng ký tham gia.
- **Quản lý Tham gia:** Theo dõi lịch sử các hoạt động đã và đang tham gia, xem trạng thái đăng ký (chờ duyệt, chấp nhận, từ chối).
- **Xem Hồ sơ Tình nguyện:** Xem tổng số giờ đã hoàn thành, nhận chứng nhận số và phản hồi/đánh giá từ Tổ chức.

Chức năng dành cho Tổ chức Quản lý Hoạt động:

- **Quản lý Hoạt động:** Tạo, chỉnh sửa và đăng tải các hoạt động mới, bao gồm mô tả chi tiết, số lượng TNV cần thiết và kỹ năng yêu cầu.

- **Quản lý Đơn đăng ký:** Xem danh sách TNV đăng ký, thực hiện **phê duyệt** hoặc từ chối đơn đăng ký.
- **Xác nhận Giờ:** Ghi nhận và xác nhận số giờ hoạt động thực tế của TNV sau khi hoạt động kết thúc.
- **Đánh giá:** Gửi phản hồi/đánh giá về hiệu suất của TNV.

Chức năng dành cho Quản trị viên Hệ thống:

- **Quản lý Chung:** Quản lý thông tin người dùng, Tổ chức, và kiểm duyệt nội dung hoạt động trước khi công khai.
- **Theo dõi Báo cáo:** Xem báo cáo thống kê về số lượng hoạt động, số lượng TNV, và các xu hướng tham gia.

Với việc tổ chức các chức năng theo từng vai trò rõ ràng, hệ thống giúp người dùng thuận tiện hơn trong thao tác, đồng thời hỗ trợ bộ phận vận hành (Tổ chức) quản lý hiệu quả hoạt động thiện nguyện.

3.2.4. Thiết kế Giao diện Người dùng (UI/UX)

Giao diện người dùng của ứng dụng được thiết kế theo hướng đơn giản, trực quan và dễ thao tác, phù hợp với nhiều nhóm người dùng. Hệ thống sử dụng các công nghệ hiện đại để xây dựng bối cảnh trang, hiệu ứng tương tác và xử lý các thao tác phía người dùng, đồng thời đảm bảo tính nhất quán về thiết kế trên toàn bộ hệ thống.

- **Trang Chủ và Tìm kiếm:** Ngay từ trang chủ, TNV có thể nhanh chóng truy cập các chức năng chính như tìm kiếm hoạt động, đăng nhập/đăng ký. Quá trình tìm kiếm hoạt động được thiết kế với các bộ lọc thông minh (theo kỹ năng, thời gian, địa điểm) để rút ngắn thời gian tra cứu.
- **Quy trình Đăng ký:** Quá trình đăng ký tham gia được thiết kế theo dạng từng bước rõ ràng, giúp TNV dễ dàng theo dõi. Các biểu mẫu nhập liệu được tối ưu với xác thực đầu vào và hiển thị lỗi rõ ràng để hạn chế sai sót.
- **Giao diện Quản trị Tổ chức:** Hệ thống cung cấp giao diện quản lý riêng với bối cảnh bảng dữ liệu, bộ lọc và chức năng thao tác nhanh (như phê duyệt

hàng loạt), giúp việc quản lý hoạt động và TNV được thuận tiện, rõ ràng và hiệu quả.

- **Tính Tương thích (Responsive Design):** Giao diện được thiết kế theo phong cách responsive, tương thích tốt với các thiết bị khác nhau như máy tính bàn, laptop, máy tính bảng và điện thoại di động. Điều này giúp TNV có thể đăng ký và theo dõi hoạt động mọi lúc, mọi nơi.

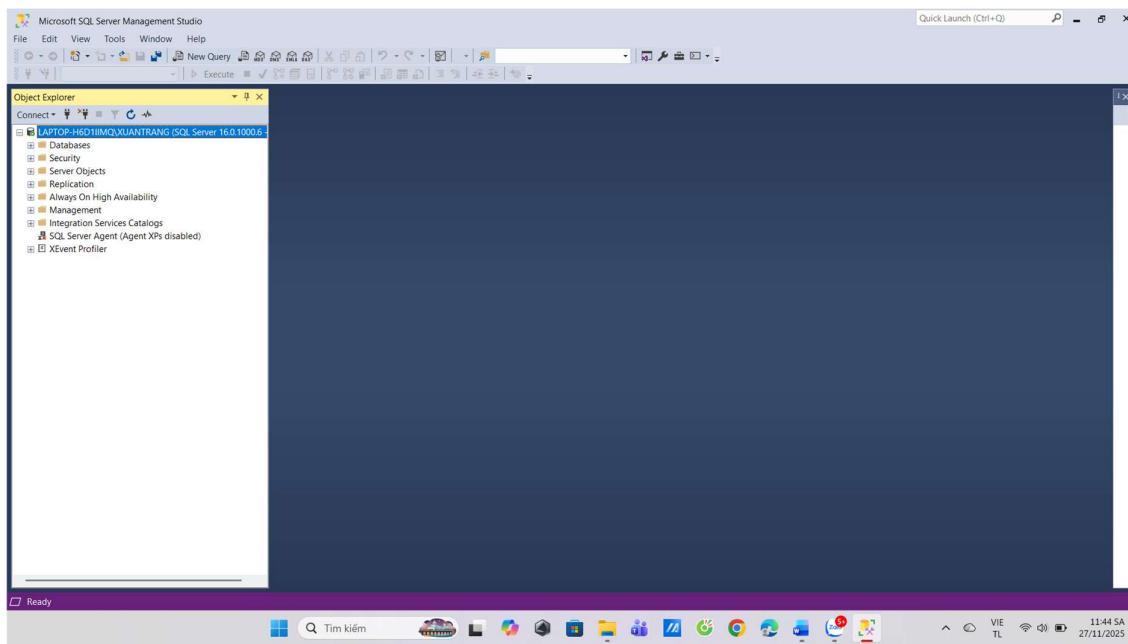
Nhờ cách tổ chức giao diện hợp lý, thân thiện và khoa học, hệ thống giúp người dùng thực hiện thao tác nhanh chóng và nâng cao trải nghiệm tổng thể trong quá trình kết nối và tham gia hoạt động xã hội.

CHƯƠNG IV. CÀI ĐẶT VÀ KIỂM THỬ

4.1. Môi trường phát triển

4.1.1. Giới thiệu về ngôn ngữ SQL

Ngôn ngữ SQL (Structured Query Language) là một trong những ngôn ngữ chuẩn và phổ biến nhất được sử dụng để thao tác và quản lý dữ liệu trong các hệ quản trị cơ sở dữ liệu quan hệ. Đây là công cụ quan trọng và không thể thiếu đối với các lập trình viên, kỹ sư dữ liệu, hay bất kỳ ai làm việc với hệ thống lưu trữ thông tin. Với SQL, người dùng có thể dễ dàng truy vấn dữ liệu, thêm mới, chỉnh sửa, hoặc xóa thông tin từ các bảng dữ liệu; đồng thời cũng có thể tạo ra cấu trúc cơ sở dữ liệu, phân quyền truy cập, và kiểm soát quá trình giao dịch một cách chính xác. SQL được chuẩn hóa bởi ANSI và ISO nên nó tương thích với hầu hết các hệ quản trị phổ biến như MySQL, PostgreSQL, Oracle, đặc biệt là Microsoft SQL Server. Với cú pháp rõ ràng, dễ học, SQL đã trở thành kỹ năng cơ bản mà bất kỳ ai làm việc trong lĩnh vực công nghệ thông tin đều cần phải nắm vững.



Hình 4.1: Giao diện phần mềm SQL

Trong số các hệ quản trị cơ sở dữ liệu hiện nay, Microsoft SQL Server là một trong những nền tảng mạnh mẽ và phổ biến nhất, đặc biệt trong môi trường

doanh nghiệp và hệ thống lớn. SQL Server 2022 là phiên bản mới nhất và tiên tiến nhất của dòng sản phẩm này, được tích hợp nhiều cải tiến vượt bậc cả về hiệu năng, tính bảo mật, khả năng mở rộng và khả năng tích hợp với điện toán đám mây. Phiên bản

này cho phép người dùng không chỉ lưu trữ và xử lý dữ liệu nhanh chóng hơn, mà còn có thể kết nối liền mạch với các dịch vụ của Azure để đồng bộ dữ liệu, sao lưu dự phòng, hay thực hiện phân tích dữ liệu lớn mà không cần thay đổi cơ sở hạ tầng cục bộ. SQL Server 2022 cũng tăng cường các cơ chế mã hóa và kiểm soát truy cập, giúp đảm bảo an toàn cho dữ liệu trong bối cảnh an ninh mạng ngày càng phức tạp. Ngoài ra, các tính năng như tối ưu hóa truy vấn thông minh, tự động sửa lỗi hiệu suất, và khả năng phân tích dữ liệu trong thời gian thực đã biến SQL Server 2022 trở thành một công cụ toàn diện cho việc quản lý, khai thác và phân tích dữ liệu doanh nghiệp.

4.1.2. Công nghệ sử dụng

Frontend (Giao diện người dùng):

Sử dụng HTML, CSS và JavaScript thuần để xây dựng toàn bộ giao diện web. HTML cung cấp cấu trúc trang, CSS được dùng để tạo bố cục, màu sắc, hiệu ứng và hỗ trợ responsive bằng Flexbox, Grid và Media Queries. JavaScript đảm nhiệm xử lý các tương tác trên trang như lấy dữ liệu từ API, hiển thị danh sách chuyến tàu, đánh dấu các ghế đã đặt và xử lý thông tin khi người dùng chọn mua vé. Mặc dù không sử dụng framework như Bootstrap hay React, giao diện vẫn đảm bảo sự đơn giản, tối ưu và tương thích trên mọi thiết bị.

Backend (Xử lý logic):

Backend của hệ thống Volunteer Management được xây dựng bằng Python 3.10+ và framework FastAPI, đảm bảo hiệu suất cao, hỗ trợ async và tự động sinh tài liệu API (Swagger UI). Toàn bộ logic nghiệp vụ, xác thực người dùng, phân quyền và tương tác với cơ sở dữ liệu đều được xử lý tại đây.

Backend sử dụng SQLAlchemy 2.0 kết hợp pyodbc để kết nối ổn định với Microsoft SQL Server. Xác thực người dùng dựa trên JWT (PyJWT) với mật khẩu được mã hoá bằng bcrypt, token có thời hạn 24 giờ và chứa thông tin vai trò (quản trị viên, tổ chức, tình nguyện viên). Phân quyền được thực hiện linh hoạt qua các dependency của FastAPI như get_current_user, get_current_admin, giúp bảo vệ các endpoint nhạy cảm.

Dữ liệu đầu vào được kiểm tra tự động bởi Pydantic v2, trả về thông báo lỗi chi tiết khi không hợp lệ. Các API được tổ chức theo module rõ ràng trong thư mục routers (admin, auth, organization, activity, volunteer, badge), dễ bảo trì và mở rộng. Hệ thống ghi log hành động admin vào bảng NhatKyHanhDong, hỗ trợ CORS (hiện cho phép tất cả origin trong môi trường dev) và sẵn sàng chuyển đổi cấu hình qua biến môi trường khi triển khai production.

Nhờ kiến trúc sạch, backend không chỉ đáp ứng nhanh các yêu cầu từ frontend tĩnh mà còn dễ dàng tích hợp thêm Docker, unit test và CI/CD trong tương lai.

- **Cơ sở dữ liệu:** Hệ thống sử dụng Microsoft SQL Server 2022 để lưu trữ và quản lý dữ liệu về tàu, tuyến đường, lịch chạy, số lượng ghé, thông tin khách hàng và vé đã đặt. SQL Server được lựa chọn vì hiệu năng cao, độ ổn định tốt và khả năng đảm bảo an toàn dữ liệu. Ngoài ra, SQL Server hỗ trợ tốt các truy vấn phức tạp, giúp hệ thống xử lý việc kiểm tra chỗ trống hay tìm kiếm chuyến tàu nhanh chóng và chính xác.
- **Công cụ hỗ trợ:** Visual Studio Code được sử dụng làm môi trường lập trình chính cho cả frontend và backend. Git được dùng để quản lý phiên bản mã nguồn, hỗ trợ làm việc nhóm và theo dõi quá trình phát triển hệ thống. Postman được sử dụng để kiểm thử API, giúp kiểm tra tính chính xác của dữ liệu trả về từ server. SQL Server Management Studio (SSMS) hỗ trợ quản lý cơ sở dữ liệu, kiểm tra bảng, thiết kế cấu trúc dữ liệu và thực thi các câu truy vấn.

4.2. Cấu trúc chương trình

Tổng quan kiến trúc: Dự án Volunteer Management được thiết kế theo mô hình tách biệt hoàn toàn backend và frontend (full-stack separation), giúp dễ bảo trì, mở rộng và triển khai độc lập.

- **Backend:** FastAPI (Python)
- **Frontend:** HTML + CSS + Vanilla JavaScript + Bootstrap 5 (không dùng framework nặng)
- **Database:** Microsoft SQL Server
- **Phân quyền:** Role-based (quản trị viên, tổ chức, tình nguyện viên, khách)

4.3. Cấu trúc thư mục dự án

4.3.1 Thư mục

Volunteer-Management/

```
|── backend          # Mã nguồn server  
|── frontend         # Giao diện người dùng (file tĩnh)  
|── database         # Script khởi tạo DB và dữ liệu mẫu  
|── docs             # Tài liệu (bao gồm file này)  
|── .gitignore  
└── README.md
```

```

Volunteer-Management/
├ backend/                               # Mã nguồn server (FastAPI - Python)
│   ├ main.py                            # Entry point, khởi tạo app + middleware
│   ├ database.py                         # Kết nối SQL Server, Session factory
│   ├ config.py                           # Cấu hình (sẽ chuyển sang .env)
│   ├ auth.py                             # JWT, hash password, verify token
│   ├ dependencies.py                   # get_db, get_current_user, role_checker
│   ├ routers/                            # Các route theo chức năng & vai trò
│   │   ├ auth.py                          # Đăng nhập, đăng ký, đổi mật khẩu
│   │   ├ public.py                         # Endpoint không cần đăng nhập
│   │   └ admin/                            # Toàn bộ chức năng quản trị
│   │       ├ organizations.py           # Duyệt tổ chức
│   │       ├ activities.py              # Duyệt hoạt động
│   │       ├ volunteers.py             # Quản lý TNV
│   │       ├ badges.py                # Quản lý & trao huy hiệu
│   │       ├ stats.py                 # Thống kê dashboard
│   │       └ logs.py                  # Audit log admin
│   └ organization/                      # Chức năng dành cho tổ chức
│       └ volunteer/                  # Chức năng dành cho tình nguyện viên
└ models/                                # SQLAlchemy models (các bảng DB)
└ schemas/                               # Pydantic schemas (validate request/response)
└ crud/                                   # Hàm thao tác DB tái sử dụng
└ utils/                                  # Email, upload file, tính giờ TNV, v.v.
└ scripts/
    └ create_admin.py                  # Script tạo tài khoản admin (idempotent)

└ frontend/                               # Giao diện tĩnh (HTML + Vanilla JS + Bootstrap 5)
    ├ index.html                         # Trang chủ công khai
    ├ auth/                                # Đăng nhập, đăng ký
    ├ admin/                               # Trang quản trị viên
    ├ organization/                      # Trang tổ chức
    ├ volunteer/                          # Trang tình nguyện viên
    ├ assets/                                # CSS, JS, hình ảnh
    │   ├ css/                                # main.js + các file JS riêng từng trang
    │   ├ js/                                # main.js + các file JS riêng từng trang
    │   └ img/                                # Header, sidebar, footer tái sử dụng
    └ components/                          # Header, sidebar, footer tái sử dụng

└ database/                               # Database & dữ liệu khởi tạo
    └ init_db.sql                         # Schema + dữ liệu mẫu + admin mặc định

└ docs/                                    # Tài liệu chi tiết dự án
└ uploads/                                 # Thư mục lưu ảnh upload (activities, avatar...)
└ .gitignore
└ README.md                                # Hướng dẫn cài đặt & chạy nhanh

```

4.3.2. Một số luồng hoạt động cấu trúc

Duyệt Tổ chức

1. Tổ chức đăng ký → bảng ToChuc có trang_thai = 'pending'

2. Admin vào trang pending-orgs.html → gọi GET
`/api/admin/organizations?status=pending`
3. Admin nhấn Duyệt/Từ chối → POST
`/api/admin/organizations/{id}/approve`
4. Hệ thống cập nhật trạng thái + ghi log vào bảng AdminLog
5. Tổ chức nhận thông báo (email + dashboard)

Đăng ký hoạt động của Tình nguyện viên (TNV)

1. TNV xem danh sách hoạt động đã duyệt → Nhấn “Đăng ký”
2. Gửi POST `/api/tinh-nguyen-vien/hoat-dong/{id}/dang-ky`
3. Hệ thống:
4. Tạo bản ghi trong bảng DangKyHoatDong (trạng thái = đang chờ xác nhận)
5. Giảm 1 chỉ tiêu còn lại của hoạt động
6. Tổ chức nhận thông báo → vào Dashboard xem đơn mới
7. Tổ chức duyệt/từ chối → POST `/api/organization/dang-ky/{id}/xac-nhan`
8. Kết quả:
9. Duyệt → trạng thái = đã xác nhận
10. Từ chối → trạng thái = bị từ chối, hoàn lại chỉ tiêu, gửi lý do cho TNV
11. Hoạt động kết thúc → tự động cộng giờ TNV + kiểm tra trao huy hiệu

Đăng bài (tạo hoạt động tình nguyện) của Tổ chức

1. Tổ chức (đã được duyệt) → vào Dashboard → “Tạo hoạt động mới”
2. Điền form + upload ảnh → Gửi POST `/api/organization/hoat-dong`
3. Hệ thống tự động tạo hoạt động với trạng thái = **pending**
4. Admin nhận thông báo → vào trang “Hoạt động chờ duyệt”
5. Admin duyệt/từ chối → POST `/api/admin/hoat-dong/{id}/approve`

6. Kết quả:
7. Duyệt → trạng thái = **approved**, hoạt động hiển thị công khai, TNV có thể đăng ký ngay
8. Từ chối → trạng thái = **rejected**, gửi email lý do cho tổ chức
9. Tổ chức nhận email + xem trạng thái trong Dashboard

Tạo tài khoản Tình nguyện viên

1. Khách vào trang Đăng ký → chọn vai trò **Tình nguyện viên**
2. Điền form (họ tên, email, mật khẩu, sđt, ngày sinh, tỉnh/thành...)
3. Gửi POST /api/auth/dang-ky
4. Hệ thống: • Tạo tài khoản + hồ sơ TNV • Tự động kích hoạt ngay (trang_thai = active) • Gửi email chào mừng
5. TNV có thẻ **đăng nhập ngay** và sử dụng toàn bộ chức năng
6. → Không cần admin duyệt (khác với Tổ chức).

4.4. Hướng dẫn cài đặt dự án (REAME.md)

Tài liệu ngắn này hướng dẫn cách chạy nhanh backend và frontend trên môi trường local (Windows / PowerShell). Nội dung tập trung vào các bước cần thiết để khởi động, cấu hình database, seed admin và kiểm thử nhanh API.

Nội dung

- *Yêu cầu môi trường*
- *Chuẩn bị & chạy database*
- *Cài dependencies và chạy backend*
- *Seed admin*
- *Chạy frontend tĩnh*

- Tài khoản mẫu và kiểm thử nhanh
- Ghi chú bảo mật & khắc phục lỗi

Yêu Cầu Môi Trường

- Python 3.8+ (khuyến nghị 3.10+)
- SQL Server (local hoặc remote) với quyền chạy script và tạo DB
- Trình duyệt hiện đại

Nếu dùng conda/venv, kích hoạt môi trường trước (PowerShell):

```
```powershell
conda
conda activate <env>

hoặc venv PowerShell activation
.\venv\Scripts\Activate.ps1

```

```

Chuẩn Bị Database

- Schema và dữ liệu mẫu nằm ở `database/init_db.sql` (có sẵn một tài khoản admin mẫu `admin@volunteer.vn` / `admin123`).

Bạn có thể chạy file SQL bằng SQL Server Management Studio hoặc `sqlcmd`:

```
```powershell
sqlcmd -S <SERVER_NAME> -U <USER> -P <PASSWORD> -i
"database\init_db.sql"

```

```

Lưu ý: cấu hình kết nối cơ sở dữ liệu của ứng dụng nằm trong `backend/database.py`. Nếu bạn chạy DB ở server/credentials khác, chỉnh file đó hoặc quản lý bằng biến môi trường trước khi khởi động backend.

Cài Dependencies & Khởi Chạy Backend

1) Cài dependencies Python:

```
```powershell
```

```
pip install -r backend\requirements.txt
```

```
```
```

2) Chạy backend (development) bằng Uvicorn (từ thư mục dự án):

```
```powershell
```

```
python -m uvicorn backend.main:app --reload
```

```
```
```

Mặc định server chạy tại `http://localhost:8000`. OpenAPI/Swagger có tại `http://localhost:8000/docs`.

Seed Admin (tùy chọn)

- `database/init_db.sql` thường đã tạo sẵn admin mẫu. Để tạo admin khác dùng script:

```
```powershell
```

```
python backend\scripts\create_admin.py --email admin@local.test --password admin123 --name "Admin Local"
```

```
```
```

Script này idempotent (nếu email đã tồn tại sẽ không ghi đè).

Chạy Frontend

- Frontend nằm trong thư mục `frontend/`. Để nghị phục vụ bằng static server thay vì mở file trực tiếp.

```
```powershell
```

```
cd frontend
```

```
python -m http.server 8001
```

```
Truy cập http://localhost:8001/index.html
```

```
...
```

Hoặc dùng extension Live Server (VSCode) để phục vụ.

Tài Khoản Admin Mẫu

- Email: `admin1volunteer.vn`

- Mật khẩu: `123456`

Gửi `POST /api/auth/dang-nhap` để lấy token. Sau khi có token, thêm header `Authorization: Bearer <token>` để gọi các endpoint yêu cầu quyền.

\*\*Một số endpoint quan trọng (tóm tắt)\*\*

- `GET /api/admin/organizations?status=pending` — danh sách tổ chức chờ duyệt
- `POST /api/admin/organizations/{ma\_to\_chuc}/approve` — duyệt/tù chối tổ chức (body: `{ "approve": true|false, "reason": "..."}`)
- `GET /api/admin/hoat-dong?status=pending` — hoạt động chờ duyệt
- `POST /api/admin/hoat-dong/{ma\_hoat\_dong}/approve` — duyệt/tù chối hoạt động
- `GET /api/admin/stats` — thống kê (tổng TNV, tổ chức, hoạt động...)

- `GET /api/admin/logs?limit=50&offset=0` — nhật ký hành động admin (audit log)

Huy hiệu (Badges):

- `GET /api/huy-hieu` — toàn bộ huy hiệu
- `GET /api/tinh-nguyen-vien/huy-hieu` — huy hiệu của TNV (cân token)
- `POST /api/tinh-nguyen-vien/huy-hieu/check` — TNV trigger kiểm tra & trao huy hiệu
- `POST /api/admin/huy-hieu` — Admin tạo huy hiệu mới
- `POST /api/admin/huy-hieu/check/{ma\_tnv}` — Admin trigger trao huy hiệu cho TNV

\*\*Kiểm Thử Nhanh (PowerShell)\*\*

1) Khởi động backend:

```
```powershell
```

```
python -m uvicorn backend.main:app --reload
```

```
...
```

2) (Tùy chọn) Tạo admin mới:

```
```powershell
```

```
python backend\scripts\create_admin.py --email admin@local.test --password
admin123 --name "Admin Local"
```

```
...
```

3) Lấy token (ví dụ PowerShell):

```
```powershell
```

```
$body = '{"email":"admin@volunteer.vn","mat_khau":"admin123"}'
```

```
Invoke-RestMethod -Method Post -Uri http://localhost:8000/api/auth/dang-nhap  
-Body $body -ContentType 'application/json'
```

```
```
```

Sau khi nhận `access\_token`, thêm header `Authorization: Bearer <token>` để kiểm thử các endpoint admin.

**\*\*Ghi chú bảo mật & câu hình\*\***

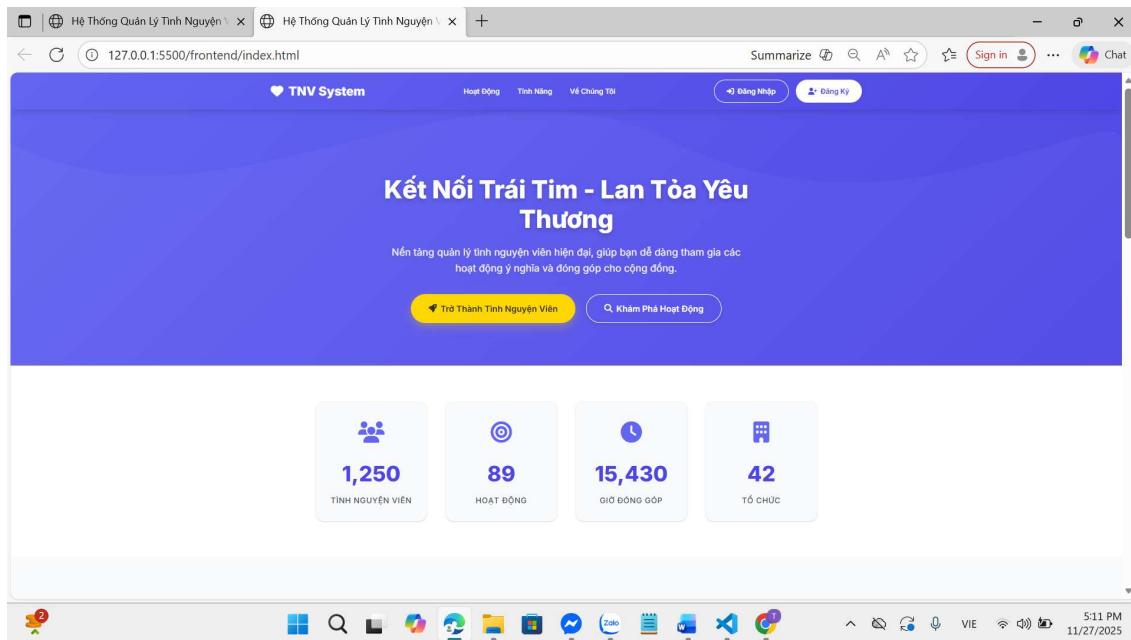
- `SECRET\_KEY` hiện đang hard-coded trong `backend/auth.py` / `backend/main.py`. Trong production dùng biến môi trường và giá trị mạnh.
- CORS hiện set `allow\_origins=["\*"]` cho dev; giới hạn domain khi deploy thực tế.
- DB credentials nằm trong `backend/database.py` — KHÔNG đẩy thông tin nhạy cảm lên kho công khai.
- Xem xét dùng cookie `httpOnly` cho token thay vì lưu client-side khi cần bảo mật cao.

**\*\*Troubleshooting nhanh\*\***

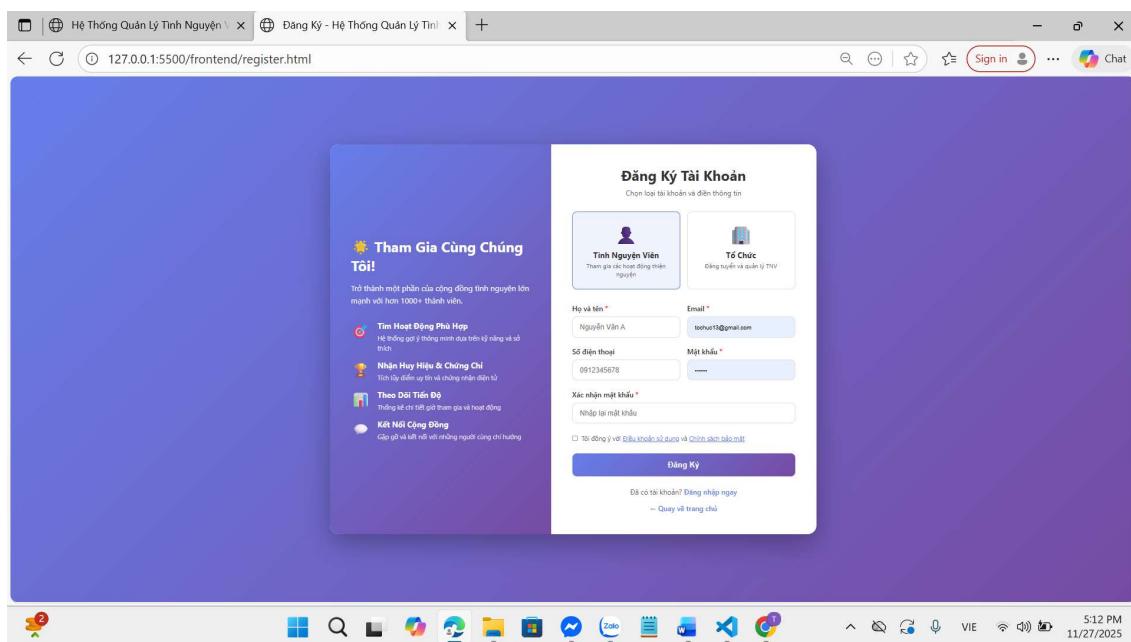
- Lỗi kết nối DB: kiểm tra cấu hình trong `backend/database.py`, SQL Server đang chạy, firewall và ODBC driver.
- Lỗi 401: kiểm tra token, header `Authorization` và thời hạn token.
- Lỗi 403: kiểm tra vai trò (`vai\_tro`) trong payload token (cần `quản trị viên`)

## 4.5. Kết quả kiểm thử

### Trang chủ



### Trang đăng kí



### Trang Admin

## Bài tiểu luận môn Công nghệ phần mềm

The screenshot shows the Admin Panel dashboard with the following key elements:

- Tổng Quan** sidebar with links: Người Dùng, Tính Nguyên Viên, Tổ Chức, Hoạt Động, Thông Kê, Đăng Xuất.
- Chào mừng quay lại, Admin**: Greet message and "Quản lý toàn diện hệ thống tinh nguyện".
- Statistics cards**:
  - TINH NGUYEN VIEN: 6
  - TỔ CHỨC: 3
  - HOẠT ĐỘNG: 109
  - TỔNG GIỚI ĐÓNG GÓP: 0
- Tổ Chức Chờ Duyệt**: Table of pending organization reviews.

| TÊN TỔ CHỨC       | EMAIL | GIẤY PHÉP | NGÀY ĐĂNG KÝ | HÀNH ĐỘNG                                        |
|-------------------|-------|-----------|--------------|--------------------------------------------------|
| Nguyễn Tiến Thắng | 9     | —         | Invalid Date | <button>Duyệt</button> <button>Tùy chẩn</button> |
| Nguyễn Tiến Thắng | 49    | —         | Invalid Date | <button>Duyệt</button> <button>Tùy chẩn</button> |
- Hoạt Động Chờ Duyệt**: Table of pending activity reviews.

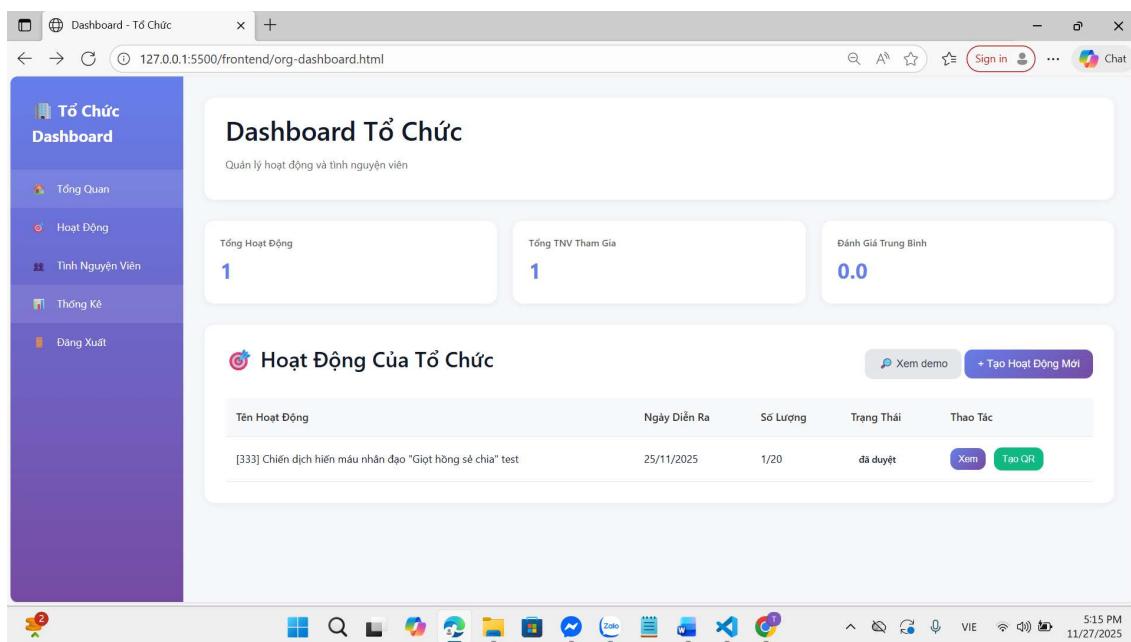
| TIỂU ĐỀ                                               | TỔ CHỨC | LĨNH VỰC | NGÀY TẠO | HÀNH ĐỘNG                                        |
|-------------------------------------------------------|---------|----------|----------|--------------------------------------------------|
| Chiến dịch hiến máu nhân đạo "Giọt hồng sẻ chia" 2025 | ...     | ...      | ...      | <button>Duyệt</button> <button>Tùy chẩn</button> |

## Trang tình nguyện viên

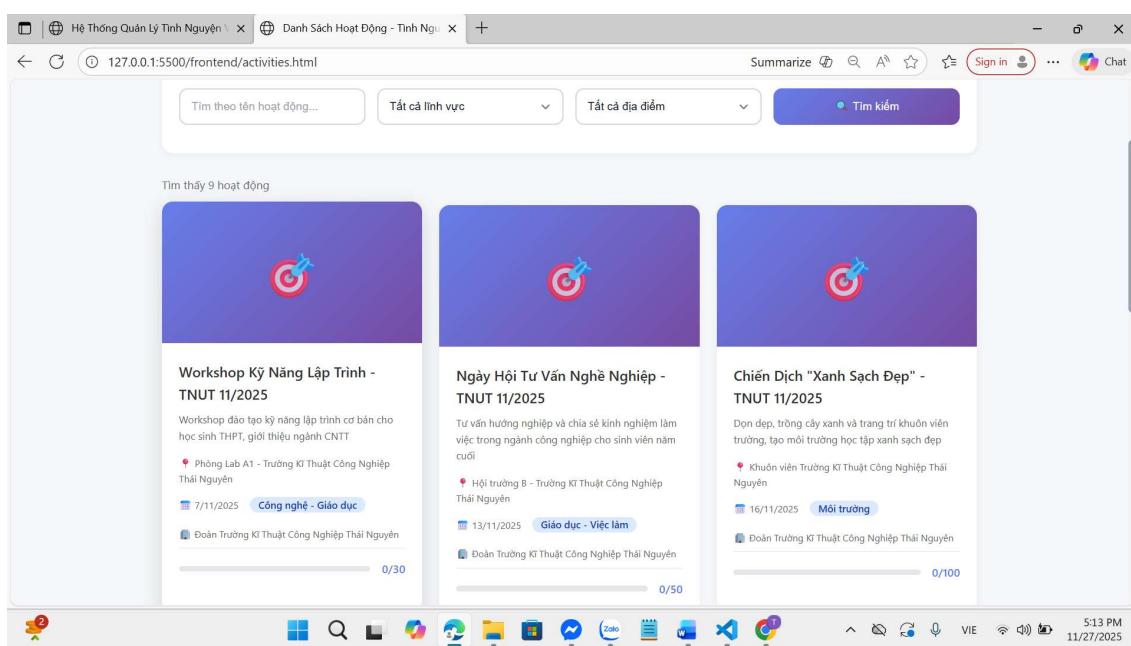
The screenshot shows the Volunteer Dashboard (TNV) with the following key elements:

- TNV Dashboard** sidebar with links: Nguyễn Tiến Thắng (Tình nguyện viên), Tổng Quan, Hoạt Động, Hoạt Động Của Tôi, Hồ Sơ Cá Nhân, Huy Hiệu, Thông Kê, Cài Đặt.
- Personal Stats**:
  - TỔNG GIỚI THAM GIA: 0 (0 giờ tháng này)
  - HOẠT ĐỘNG ĐÃ THAM GIA: 4 (0 hoạt động mới)
  - ĐIỂM UY TÍN: 0.0 (Xếp hạng: Mới bắt đầu)
  - HUY HIỆU: 0 (0 huy hiệu mới)
- Hoạt Động Của Tôi**: List of recent activities.
  - [333] Chiến dịch hiến máu nhân đạo "Giọt hồng sẻ chia" test
    - Hội trường tầng 1, Bệnh viện Đa khoa tỉnh XYZ, số 123 đường Lê Lợi, TP. Thái Nguyên
    - Nguyễn Tiến Thắng
    - 25/11/2025
    - ĐÃ DUYỆT
  - Hội Thảo IoT và Công Nghiệp 4.0 - TNUT 11/2025
    - Phòng Hội thảo - Trường Kỹ Thuật Công Nghiệp Thái Nguyên
    - Đoàn Trường Kỹ Thuật Công Nghiệp Thái Nguyên
    - 22/11/2025
    - CHỜ DUYỆT

## Trang tổ chức



### Trang xem hoạt động



### Trang đăng kí

## Ngày Hội Tư Vấn Nghề Nghiệp - TNUT 11/2025

Giáo dục - Việc làm

Hội trường B - Trường Kỹ Thuật Công Nghệ Thái Nguyên | 13/11/2025 | 13:30

### Tình Trạng Đăng Ký

0/50 người

Còn 50 chỗ trống

### Mô Tả Hoạt Động

Tư vấn hướng nghiệp và chia sẻ kinh nghiệm làm việc trong ngành công nghiệp cho sinh viên năm cuối

### Thông Tin Chi Tiết

THỜI GIAN BẮT ĐẦU  
13/11/2025 13:30

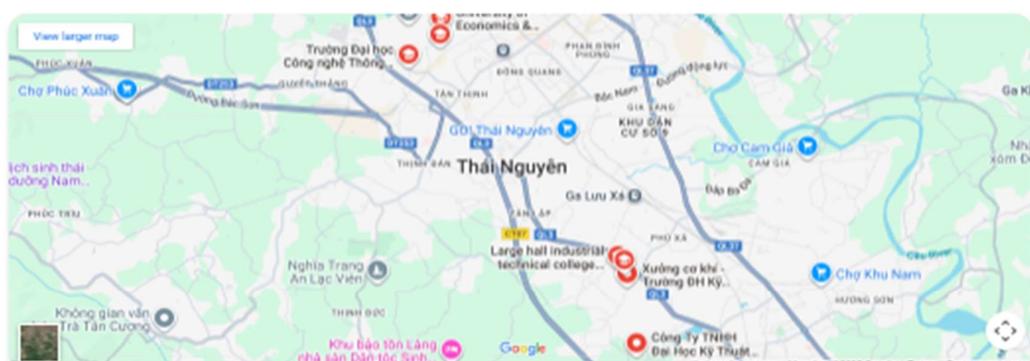
THỜI GIAN KẾT THÚC  
13/11/2025 17:30

HẠN ĐĂNG KÝ  
11/11/2025 23:59

TRẠNG THÁI  
đã duyệt

### Vị trí

Hội trường B - Trường Kỹ Thuật Công Nghệ Thái Nguyên



Mở trong Google Maps

### Kỹ Năng Yêu Cầu

Tư vấn

Truy cập thông tin

### Thông Tin Tổ Chức

## CHƯƠNG V. NHẬN XÉT VÀ ĐÁNH GIÁ HỆ THỐNG

### 5.1. Những kết quả đạt được

Hệ thống quản lý tình nguyện viên đã được thiết kế và triển khai thành công, đạt được nhiều kết quả tích cực, đáp ứng hầu hết các yêu cầu chức năng đề ra. Trước hết, hệ thống hỗ trợ đầy đủ các tác nhân chính với quyền hạn phù hợp. Khách (Guest) có thể dễ dàng truy cập danh sách hoạt động, tìm kiếm theo vị trí hoặc lĩnh vực quan tâm, xem thông tin chi tiết của các tổ chức và đăng ký tài khoản để trở thành thành viên.

Đối với tình nguyện viên (Volunteer), hệ thống cung cấp hành trình tham gia hoạt động một cách liền mạch. Người dùng có thể đăng nhập, cập nhật hồ sơ cá nhân, tìm kiếm và đăng ký tham gia các hoạt động, thực hiện điểm danh (check-in/check-out), nhận huy hiệu và điểm uy tín, đồng thời đánh giá các tổ chức sau khi kết thúc hoạt động. Tất cả các bước này đều được hệ thống quản lý tự động và minh bạch, giúp nâng cao trải nghiệm người dùng.

Đối với tổ chức (Organization), hệ thống hỗ trợ quản lý toàn diện các hoạt động. Các tổ chức có thể đăng bài hoạt động mới, quản lý danh sách tình nguyện viên đăng ký, duyệt hoặc từ chối tình nguyện viên, theo dõi tiến độ hoạt động, ghi nhận giờ công và đánh giá tình nguyện viên. Hệ thống cũng cho phép thống kê chi tiết các hoạt động đã tổ chức, hỗ trợ việc ra quyết định và đánh giá hiệu quả công tác tổ chức.

Về phía Admin, hệ thống đảm bảo chức năng giám sát và kiểm duyệt toàn diện. Admin có thể duyệt tài khoản tổ chức mới, phê duyệt các bài hoạt động, quản lý người dùng và danh mục hệ thống, đồng thời xem thống kê tổng thể về tình nguyện viên, tổ chức và hoạt động. Cơ chế phân quyền rõ ràng giúp bảo mật dữ liệu và kiểm soát quyền truy cập hiệu quả.

Bên cạnh đó, hệ thống thể hiện sự thân thiện và trực quan trong giao diện người dùng. Luồng thao tác được thiết kế hợp lý, giúp các tác nhân thực hiện công việc một cách dễ dàng, trên cả máy tính và thiết bị di động. Các tính năng tự động

thông báo, trao huy hiệu, nhắc nhở tham gia hoạt động đều hoạt động ổn định, tạo động lực tham gia cho tình nguyện viên và nâng cao hiệu quả quản lý của tổ chức.

### 5.2. Những hạn chế và hướng cải tiến

Mặc dù hệ thống quản lý tình nguyện viên đã đáp ứng tốt các mục tiêu chức năng cơ bản, vẫn tồn tại một số hạn chế cần được khắc phục nhằm nâng cao hiệu quả vận hành và trải nghiệm người dùng.

Trước tiên, khả năng xử lý đồng thời của hệ thống vẫn còn hạn chế. Khi số lượng tình nguyện viên và tổ chức tăng lên đáng kể, tốc độ phản hồi và hiệu suất tổng thể có thể bị ảnh hưởng. Để cải thiện, cần tối ưu cơ sở dữ liệu, nâng cấp hạ tầng máy chủ, đồng thời áp dụng các giải pháp phân tán và caching thông minh, đảm bảo hệ thống hoạt động ổn định ngay cả với lượng lớn người dùng.

Tiếp theo, trải nghiệm người dùng vẫn còn tiềm năng để hoàn thiện. Mặc dù giao diện trực quan và dễ thao tác, hệ thống chưa hỗ trợ các tính năng cá nhân hóa hoặc đề xuất hoạt động dựa trên sở thích, lịch sử tham gia hay kỹ năng của tình nguyện viên. Việc bổ sung các tính năng thông minh này sẽ giúp tăng mức độ gắn kết, khuyến khích người dùng tham gia chủ động và nâng cao sự hài lòng tổng thể.

Ngoài ra, các công cụ thống kê và báo cáo hiện tại còn đơn giản. Hệ thống mới chỉ cung cấp các số liệu cơ bản như giờ công, số lượng tình nguyện viên tham gia và hiệu quả hoạt động. Trong tương lai, việc triển khai các báo cáo nâng cao, phân tích dữ liệu chi tiết về hiệu quả từng hoạt động, đánh giá tổ chức và dự đoán xu hướng sẽ hỗ trợ người quản lý đưa ra quyết định chính xác và kịp thời hơn.

Về bảo mật, mặc dù hệ thống đã thực hiện phân quyền rõ ràng và bảo vệ dữ liệu người dùng, các cơ chế bảo mật nâng cao vẫn cần được tăng cường. Việc áp dụng mã hóa dữ liệu nhạy cảm, giám sát truy cập, phòng chống tấn công mạng và đảm bảo quyền riêng tư sẽ tạo nền tảng an toàn hơn, đặc biệt khi hệ thống mở rộng cho nhiều tổ chức và hàng ngàn tình nguyện viên.

### **5.3. Đánh giá tổng quan hệ thống**

Hệ thống quản lý tình nguyện viên sau khi triển khai đã chứng tỏ khả năng đáp ứng các yêu cầu chức năng và nghiệp vụ một cách hiệu quả. Nó mang lại trải nghiệm thân thiện, trực quan cho tất cả các tác nhân, đồng thời hỗ trợ tổ chức và Admin trong việc quản lý và giám sát hoạt động một cách minh bạch.

Các tính năng chính như đăng ký và quản lý hoạt động, điểm danh tình nguyện viên, trao huy hiệu tự động, gửi thông báo, cũng như thống kê và báo cáo cơ bản đều hoạt động ổn định, đáp ứng nhu cầu thực tế. Sự phối hợp giữa các tác nhân trong hệ thống diễn ra liền mạch, giúp tối ưu hóa luồng công việc và nâng cao hiệu quả vận hành.

Bên cạnh đó, hệ thống cũng cho thấy tiềm năng phát triển trong tương lai. Các hạn chế về hiệu suất xử lý, trải nghiệm cá nhân hóa, báo cáo phân tích nâng cao và bảo mật có thể được cải thiện thông qua các giải pháp kỹ thuật và bổ sung tính năng. Việc này không chỉ giúp hệ thống vận hành hiệu quả hơn mà còn nâng cao sự hài lòng và gắn kết của người dùng.

### **Kết luận:**

Nhìn chung, hệ thống quản lý tình nguyện viên đã hoàn thành mục tiêu ban đầu, cung cấp một nền tảng ổn định, trực quan và hiệu quả. Đây là cơ sở vững chắc để tiếp tục phát triển, mở rộng tính năng, tối ưu hóa hiệu suất và củng cố bảo mật, hướng tới một hệ thống toàn diện, thông minh và bền vững trong dài hạn.

### **TÀI LIỆU THAM KHẢO**

- [1]. ThS.Trần Thị Thanh, “Giáo trình Phân tích và Thiết kế Hệ thống Thông tin”, NXB Trường Đại học Kỹ Thuật Công Nghiệp Thái Nguyên.
- [2]. Tổng Công ty Đường Sắt Việt Nam – VNR. Hệ thống đặt vé tàu trực tuyến. Truy cập tại: <https://dsvn.vn> (Ngày truy cập: 20/12/2024).
- [3]. Lê Minh Hùng (2020). Nghiên cứu xây dựng hệ thống đặt vé xe khách trực tuyến. Luận văn Thạc sĩ, Đại học Công nghệ TP.HCM.