

ĐẠI HỌC ĐÀ NẴNG  
TRƯỜNG ĐẠI HỌC BÁCH KHOA  
KHOA CÔNG NGHỆ THÔNG TIN

**ĐỒ ÁN TỐT NGHIỆP**  
NGÀNH: CÔNG NGHỆ THÔNG TIN  
CHUYÊN NGÀNH: CÔNG NGHỆ THÔNG TIN

**ĐỀ TÀI:**

**HỆ THỐNG QUẢN LÝ ĐIỂM PHỤC VỤ CỘNG ĐỒNG VÀ  
SINH VIÊN 5 TỐT CỦA ĐẠI HỌC BÁCH KHOA ĐÀ NẴNG.**

Người hướng dẫn: TS. TRỊNH CÔNG DUY

Sinh viên thực hiện: TRẦN KIM HIẾU

Số thẻ sinh viên: 102210035

Lớp: 21TCLC\_Nhat1

Đà Nẵng, ...../2025

**ĐẠI HỌC ĐÀ NẴNG  
TRƯỜNG ĐẠI HỌC BÁCH KHOA  
KHOA CÔNG NGHỆ THÔNG TIN**

**ĐỒ ÁN TỐT NGHIỆP**  
**NGÀNH: CÔNG NGHỆ THÔNG TIN**  
**CHUYÊN NGÀNH: CÔNG NGHỆ THÔNG TIN**

**ĐỀ TÀI:**

**HỆ THỐNG QUẢN LÝ ĐIỂM PHỤC VỤ CỘNG ĐỒNG VÀ  
SINH VIÊN 5 TỐT CỦA ĐẠI HỌC BÁCH KHOA ĐÀ NẴNG.**

Người hướng dẫn: **TS. TRỊNH CÔNG DUY**  
Sinh viên thực hiện: **TRẦN KIM HIẾU**  
Số thẻ sinh viên: **102210035**  
Lớp: **21TCLC\_Nhat1**

**Đà Nẵng, ...../2025**

# TÓM TẮT

Tên đề tài: Hệ thống quản lý điểm phục vụ cộng đồng và sinh viên 5 tốt của Đại Học Bách Khoa Đà Nẵng

Sinh viên thực hiện: Trần Kim Hiếu

Số thẻ sinh viên: 102210035

Lớp: 21TCLC\_Nhat1

Đề tài “Hệ thống quản lý điểm phục vụ cộng đồng và sinh viên 5 tốt của Đại Học Bách Khoa Đà Nẵng” được thực hiện với mục tiêu xây dựng một hệ thống phần mềm hỗ trợ quản lý hiệu quả quá trình rèn luyện và tham gia hoạt động ngoại khóa của sinh viên. Trong bối cảnh các hoạt động tình nguyện, học thuật, thể chất và hội nhập ngày càng được đẩy mạnh, nhu cầu về một hệ thống thống nhất để ghi nhận, đánh giá và hỗ trợ sinh viên đạt danh hiệu “Sinh viên 5 tốt” trở nên cấp thiết.

Hệ thống tập trung vào hai chức năng chính: quản lý điểm phục vụ cộng đồng và theo dõi tiêu chí “Sinh viên 5 tốt”, nhằm cung cấp công cụ số hóa quy trình đánh giá, minh bạch hóa thông tin và hỗ trợ sinh viên tự theo dõi tiến trình rèn luyện của bản thân. Người quản trị có thể tạo và quản lý các sự kiện, phân loại hoạt động theo tiêu chí cụ thể, đồng thời theo dõi báo cáo điểm rèn luyện của từng sinh viên.

Hệ thống có tích hợp Chatbot, với hai chức năng chính:

- Tư vấn nội quy sinh viên: Chatbot có khả năng trả lời các câu hỏi phổ biến liên quan đến quy định nhà trường, điều kiện xét danh hiệu sinh viên 5 tốt,... giúp giảm tải cho bộ phận tư vấn và tăng cường sự chủ động của sinh viên.
- Tạo và truy vấn sự kiện: Chatbot cho phép người dùng tạo sự kiện mới bằng ngôn ngữ tự nhiên (ví dụ: “Tạo sự kiện hiến máu vào 8h sáng ngày 15/6 tại Hội trường A”), đồng thời cung cấp thông tin về các sự kiện hiện có.

Đề tài còn triển khai thành công các mô-đun chức năng chính của hệ thống, bao gồm quản lý tài khoản người dùng, quản lý sự kiện, hệ thống theo dõi điểm sinh viên. Những tính năng này giúp tạo ra một hệ thống hoàn chỉnh, hỗ trợ cả sinh viên và cán bộ nhà trường trong công tác tham gia và quản lý điểm phục vụ cộng đồng.

Với những tính năng trên, hệ thống góp phần tạo ra một môi trường học tập và rèn luyện năng động, giúp sinh viên phát triển toàn diện về học tập, kỹ năng và đạo đức. Đồng thời, đây cũng là công cụ hỗ trợ hiệu quả cho nhà trường trong công tác quản lý và khuyến khích phong trào thi đua giữa các sinh viên.

## NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: Trần Kim Hiếu

Số thẻ sinh viên: 102210035

Lớp: 21TCLC\_Nhat1 Khoa: Công nghệ thông tin Ngành: Công nghệ thông tin

- Tên đề tài đồ án:* Hệ thống quản lý điểm phục vụ cộng đồng và sinh viên 5 tốt của Đại Học Bách Khoa Đà Nẵng.
- Đề tài thuộc diện:*  *Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực Hiện.*
- Các số liệu và dữ liệu ban đầu:* Dữ liệu được tạo ra bởi sinh viên và tìm kiếm thông tin ở trên mạng.
- Nội dung các phần thuyết minh và tính toán:*

Mở đầu: Giới thiệu chung về đề tài, mục đích thực hiện, mục tiêu của đề tài, đối tượng nghiên cứu và phương pháp nghiên cứu của dự án.

Chương 1: Cơ sở lý thuyết và các công nghệ sử dụng, mô hình được sử dụng trong hệ thống.

Chương 2: Phân tích yêu cầu của hệ thống, mô tả các chức năng của từng chức năng và mô tả các chức năng bằng các biểu đồ.

Chương 3: Mô tả cách triển khai hệ thống và kết quả của hệ thống.

Kết luận: Tổng kết kết quả đạt được của hệ thống, định hướng phát triển trong tương lai.
- Các bản vẽ, đồ thị ( ghi rõ các loại và kích thước bản vẽ ): Không*
- Họ và tên người hướng dẫn : Ts. Trịnh Công Duy*
- Ngày giao nhiệm vụ đồ án: .../.../2025*
- Ngày hoàn thành đồ án: .../.../2025*

Đà Nẵng, ngày tháng năm 2025

Trưởng Bộ môn.....

Người hướng dẫn



## LỜI NÓI ĐẦU

Trước tiên, em xin gửi lời cảm ơn chân thành và sâu sắc đến thầy Trịnh Công Duy – người đã tận tình hướng dẫn, đồng hành và hỗ trợ em trong suốt quá trình thực hiện và hoàn thiện đề tài. Em cũng xin được bày tỏ lòng biết ơn đến quý thầy cô trong Khoa Công nghệ Thông tin, những người đã không chỉ truyền đạt cho em nền tảng kiến thức chuyên môn vững chắc mà còn chia sẻ nhiều kinh nghiệm thực tiễn quý báu trong suốt thời gian học tập tại trường.

Trong bối cảnh chuyển đổi số đang diễn ra mạnh mẽ trên toàn cầu, lĩnh vực giáo dục – đào tạo cũng đang từng bước chuyển mình, hướng tới việc ứng dụng công nghệ thông tin nhằm nâng cao chất lượng giảng dạy và quản lý. Một trong những nhiệm vụ quan trọng của các cơ sở giáo dục hiện nay là hiện đại hóa công tác quản lý rèn luyện sinh viên, đặc biệt là việc theo dõi các tiêu chí phục vụ cộng đồng và xét chọn danh hiệu “Sinh viên 5 tốt”.

Thực tế tại Trường Đại học Bách Khoa – Đại học Đà Nẵng cho thấy, quá trình đánh giá và quản lý các hoạt động rèn luyện hiện vẫn còn mang tính thủ công, thiếu sự đồng bộ, gây khó khăn trong việc tổng hợp dữ liệu, theo dõi tiến độ và hỗ trợ sinh viên tiếp cận thông tin. Xuất phát từ nhu cầu thực tiễn đó, em đã lựa chọn thực hiện đề tài:

“Hệ thống quản lý điểm phục vụ cộng đồng và sinh viên 5 tốt của Đại Học Bách Khoa Đà Nẵng”.

Hệ thống được xây dựng với mục tiêu hỗ trợ nhà trường và sinh viên trong việc số hóa toàn bộ quá trình đánh giá rèn luyện, quản lý sự kiện, theo dõi tiêu chí 5 tốt và thúc đẩy sự tham gia vào các hoạt động tình nguyện. Đặc biệt, hệ thống tích hợp chatbot thông minh có khả năng tư vấn nội quy, hỗ trợ tự động tạo và truy vấn thông tin sự kiện, từ đó nâng cao tính tương tác, cải thiện trải nghiệm người dùng và giảm thiểu gánh nặng cho bộ phận quản lý.

Em hy vọng rằng dự án này không chỉ góp phần nâng cao hiệu quả trong công tác quản lý rèn luyện sinh viên, mà còn đặt nền móng cho một môi trường học tập hiện đại, năng động – nơi sinh viên có thể chủ động tiếp cận thông tin và tham gia hoạt động ngoại khóa một cách thuận tiện và hiệu quả.

Với việc ứng dụng các công nghệ hiện đại như chatbot và quy trình đánh giá số hóa, hệ thống không chỉ mang lại giá trị thiết thực cho cộng đồng sinh viên, mà còn góp phần thúc đẩy sự phát triển kỹ năng cá nhân và chuyên môn của mỗi người học. Sự thành công của dự án sẽ là bước đệm quan trọng, mở ra cơ hội tiếp tục đổi mới và phát triển các giải pháp công nghệ trong lĩnh vực giáo dục và quản lý sinh viên trong tương lai.

## CAM ĐOAN

Tôi xin cam đoan rằng đề tài “Hệ thống quản lý điểm phục vụ cộng đồng và sinh viên 5 tốt của Đại Học Bách Khoa Đà Nẵng” là kết quả của quá trình nghiên cứu và làm việc nghiêm túc của bản thân tôi, được thực hiện dưới sự hướng dẫn của thầy Trịnh Công Duy. Toàn bộ nội dung, số liệu, kết quả nghiên cứu và các thông tin trình bày trong báo cáo này đều là trung thực và chưa từng được sử dụng để bảo vệ bất kỳ học vị hay công trình khoa học nào khác.

Tôi cũng xin cam đoan rằng mọi sự hỗ trợ trong quá trình thực hiện báo cáo này đã được ghi nhận và cảm ơn một cách đầy đủ, các tài liệu tham khảo đã được trích dẫn đúng quy định.

Nếu có bất kỳ hành vi gian lận hoặc vi phạm quy định nào, tôi xin hoàn toàn chịu trách nhiệm trước nhà trường và trước pháp luật.

Sinh viên thực hiện

Trần Kim Hiếu

# MỤC LỤC

<b>NHẬN XÉT CỦA NGƯỜI HƯỚNG DẪN</b>	i
<b>NHẬN XÉT CỦA NGƯỜI PHẢN BIỆN</b>	ii
<b>TÓM TẮT</b>	iii
<b>LỜI NÓI ĐẦU</b>	1
<b>CAM ĐOAN</b>	2
<b>MỤC LỤC</b>	3
<b>DANH SÁCH CÁC BẢNG</b>	6
<b>DANH SÁCH HÌNH VẼ</b>	7
<b>DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT</b>	9
1.    Mục đích thực hiện	1
2.    Mục tiêu đề tài	1
3.    Phạm vi nghiên cứu	2
4.    Đối tượng nghiên cứu	2
5.    Phương pháp nghiên cứu	2
<b>CHƯƠNG 1: CƠ SỞ LÝ THUYẾT</b>	4
1.1    Mô hình Khách-Chủ (Client – Server)	4
1.2    Git và Github	6
Quy trình làm việc với Git (Git flow):	7
1.3    Tổng quan về Spring Boot	7
Các đặc trưng của Spring Boot:	7
Ưu điểm của Spring Boot:	8
Nhược điểm của Spring Boot:	8
1.4    Hệ quản trị cơ sở dữ liệu PostgreSQL	9
Lý do lựa chọn PostgreSQL	9
1.5    Cloudinary	9
Lý do lựa chọn Cloudinary	9
1.6    Supabase	10
1.7    Docker	11
Đặc trưng cơ bản của Docker	11
So sánh Docker và Virtual Machines:	11
1.8    Render	12
Lý do lựa chọn Render:	13
1.9    n8n	13
Giới thiệu về n8n	13
Những đặc điểm nổi bật của :	13
Lý do lựa chọn n8n:	13

1.9.1	ReactJs	14
1.9.2	TailwindCss	14
	Các điểm chú ý của tailwinds css	14
1.10	HTML, CSS và TypeScript	15
1.11	Kiến trúc RAG	16
	Tổng quan về RAG	16
	Tóm tắt ngắn gọn quá trình của RAG như sau:	17
	Ưu điểm của RAG:	17
1.12	Pinecone	17
1.13	Kết luận chương 1	18
<b>CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG</b>		19
2.1	yêu cầu chức năng	19
	Sinh viên	19
	Liên chi đoàn	19
	Hội sinh viên	19
	Cộng tác sinh viên	20
	Ban thời vụ đoàn trường	20
2.2	Yêu cầu về phi chức năng	21
2.2.1	Hiệu suất (Performance)	21
2.2.2	Khả năng mở rộng (Scalability)	21
2.2.3	Bảo mật (Security)	21
2.2.4	Trình duyệt (Browser)	22
2.2.5	Giao diện (Interfaces)	22
2.2.6	Giả định (Assumptions)	22
2.3	Biểu đồ phân rã chức năng	23
2.4	Biểu đồ ca sử dụng	27
2.4.1	Biểu đồ ca sử dụng tổng quát cho sinh viên	27
2.4.2	Biểu đồ ca sử dụng tổng quát cho liên chi đoàn	29
2.4.3	Biểu đồ ca sử dụng tổng quát cho hội sinh viên	30
2.4.4	Biểu đồ ca sử dụng tổng quát cho ban thời vụ đoàn trường	31
2.4.5	Biểu đồ ca sử dụng tổng quát cho cộng tác sinh viên	32
2.5	Lược đồ cơ sở dữ liệu	33
	Tổng quát các bảng trong lược đồ cơ sở dữ liệu	33
2.6	Sơ đồ hoạt động	35
2.7	Sơ đồ tuần tự	42
2.8	Kết luận chương 2	45
<b>CHƯƠNG 3: TRIỂN KHAI HỆ THỐNG</b>		46
3.1	Xây dựng chatbot hỏi đáp nội quy nhà trường	46

3.1.1 Lý do lựa chọn n8n và mô hình ngôn ngữ lớn (LLM) để xây dựng chatbot	46
3.1.2 Các bước xây dựng chatbot hỏi đáp cho từng khóa học	
Bước 1: Xác định Mục tiêu và Yêu cầu của Chatbot	47
Bước 2: Thu thập	47
Bước 3: Lựa chọn mô hình ngôn ngữ	47
Bước 4: Xây dựng Hệ thống Lưu trữ Dữ liệu với Pinecone	47
Chuyển đổi dữ liệu thành vector:	47
Bước 5: Xây dựng prompt	47
3.1.3 Kết quả	48
3.2 Xây dựng chatbot tư vấn và tự động tạo sự kiện	49
3.2.1 Lý do lựa chọn n8n để phát triển chatbot agent tạo sự kiện	49
3.2.2 Kết quả	50
3.3 Một số giao diện hệ thống	54
3.3.1 Trang đăng nhập	54
3.3.2 Màn hình trang quản lý sự kiện	54
3.3.3 Màn hình chi tiết sự kiện	55
3.3.4 Màn hình tất cả sự kiện	56
3.3.5 Màn hình tiêu chí hoạt động và sinh viên 5 tốt	56
3.3.6 Màn hình quản lý điểm phục vụ cộng đồng	57
3.4.6 Màn hình tạo sự kiện	58
3.4.7 Màn hình quản lý lớp học	58
3.4.8 Màn hình quản lý tài khoản	59
3.4.9 Màn hình thêm tài khoản	60
3.4.6 Màn hình quản lý tham gia sự kiện	60
3.4.10 Màn hình điểm danh sinh viên	61
3.4.11 Màn hình trang chủ bên phía sinh viên	62
3.4.12 Màn hình danh sách hoạt động	63
3.4.13 Màn hình danh sách menu phía sinh viên	63
3.5 Kết luận Chương 3	64
<b>KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN</b>	65
Kết luận	65
Hướng phát triển	65
<b>TÀI LIỆU THAM KHẢO</b>	67
PHỤ LỤC 1	69
PHỤ LỤC 2	70

## DANH SÁCH CÁC BẢNG

Bảng 1 Hiệu suất	21
Bảng 2 Khả năng mở rộng	21
Bảng 3 Bảo mật	21
Bảng 4 Trình duyệt	22
Bảng 5 Giao diện	22
Bảng 6 Giả định	22
Bảng 7 Tổng quát các bảng trong lược đồ cơ sở dữ liệu	33

## DANH SÁCH HÌNH VẼ

Hình 1 Mô hình client-server	4
Hình 2 Github	6
Hình 3 Git Flow	7
Hình 4 Kiến trúc Docker	12
Hình 5 Mô hình hoạt động của RAG	16
Hình 6 Biểu đồ phân rã chức năng cho sinh viên	23
Hình 7 Biểu đồ phân rã chức năng cho liên chi đoàn	24
Hình 8 Biểu đồ phân rã chức năng cho hội sinh viên	25
Hình 9 Biểu đồ phân rã chức năng cho ban thời vụ	26
Hình 10 Biểu đồ phân rã chức năng cho cộng tác sinh viên	27
Hình 11 Biểu đồ ca sử dụng tổng quát cho sinh viên	28
Hình 12 Biểu đồ ca sử dụng tổng quát cho giảng viên	29
Hình 13 Biểu đồ ca sử dụng tổng quát cho hội sinh viên	30
Hình 14 Biểu đồ ca sử dụng tổng quát cho ban thời vụ	31
Hình 15 Biểu đồ ca sử dụng tổng quát cho cộng tác sinh viên	32
Hình 16 Lược đồ cơ sở dữ liệu	33
Hình 17 Sơ đồ Activity đăng nhập	35
Hình 18 Sơ đồ hoạt động - Đăng ký sự kiện	36
Hình 19 Sơ đồ hoạt động - Gửi minh chứng	37
Hình 20 Sơ đồ hoạt động - tạo sự kiện	38
Hình 21 Sơ đồ hoạt động - quản lý sự kiện	39
Hình 22 Sơ đồ hoạt động - tìm kiếm sự kiện	40
Hình 23 Sơ đồ hoạt động - quản lý tài khoản	41
Hình 24 Sơ đồ tuần tự đăng nhập	42
Hình 25 Sơ đồ tuần tự đặt lại mật khẩu	43
Hình 26 Sơ đồ tuần tự tạo sự kiện	44
Hình 27 Sơ đồ tuần tự đăng ký sự kiện	45
Hình 28 Kết quả chatbot trả lời câu hỏi sinh viên	48
Hình 29 Kết quả chatbot	50
Hình 30 Kết quả chatbot	51
Hình 31 Kết quả chatbot	52
Hình 32 Kết quả chatbot	53
Hình 33 Màn hình đăng nhập	54
Hình 34 Màn hình trang chủ	54
Hình 35 Màn hình chi tiết sự kiện	55
Hình 36 Màn hình chi tiết sự kiện	55
Hình 37 Màn hình tất cả sự kiện	56

Hình 38 Màn hình tiêu chí hoạt động và sinh viên 5 tốt	56
Hình 39 Màn hình quản lý điểm phục vụ cộng đồng	57
Hình 40 Màn hình quản lý điểm phục vụ cộng đồng	57
Hình 41 Màn hình tạo sự kiện	58
Hình 42 Màn hình quản lý lớp học	58
Hình 43 Màn hình quản lý tài khoản	59
Hình 44 Màn hình thêm tài khoản	60
Hình 45 Màn hình quản lý khóa học	60
Hình 46 Màn hình quản lý khóa học	61
Hình 47 Màn hình điểm danh sinh viên	61
Hình 48 Màn hình trang chủ sinh viên	62
Hình 49 Màn hình trang chủ sinh viên	62
Hình 50 Màn hình danh sách hoạt động	63
Hình 51 Màn hình menu sinh viên	63

## DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

Ký hiệu/Chữ viết tắt	Diễn giải
AI	<i>Artificial Intelligence</i>
LLM	<i>Large Language Model</i>
API	<i>Application Programming Interface</i>
HTML	<i>HyperText Markup Language</i>
HTTP	<i>HyperText Transfer Protocol</i>
JSON	<i>JavaScript Object Notation</i>
UI	<i>User Interface</i>
n8n	<i>Nodenodenode</i>
URL	<i>Uniform Resource Locator</i>
HTTPS	<i>HyperText Transfer Protocol Secure</i>
RAG	<i>Retrieval-Augmented Generation</i>
SQL	<i>Structured Query Language</i>
XML	<i>Extensible Markup Language</i>
VM	<i>Virtual Machine</i>
OS	<i>Operating System</i>
CDN	<i>Content Delivery Network</i>

## MỞ ĐẦU

### 1. Mục đích thực hiện

Trong bối cảnh chuyển đổi số đang diễn ra mạnh mẽ, đặc biệt là với sự phát triển vượt bậc của công nghệ thông tin và trí tuệ nhân tạo, việc ứng dụng các công nghệ hiện đại vào lĩnh vực giáo dục và quản lý sinh viên trở thành xu hướng tất yếu. Các công cụ số không chỉ giúp nâng cao hiệu quả quản lý mà còn cải thiện đáng kể trải nghiệm của người học, mở ra hướng tiếp cận mới trong công tác giáo dục và rèn luyện.

Tại Trường Đại học Bách Khoa – Đại học Đà Nẵng, công tác theo dõi điểm phục vụ cộng đồng cũng như việc đánh giá danh hiệu “Sinh viên 5 tốt” hiện vẫn đang được thực hiện thủ công, gây tốn kém thời gian, công sức và tiềm ẩn sai sót trong quá trình tổng hợp, phân tích dữ liệu. Điều này đặt ra nhu cầu cấp thiết về một giải pháp công nghệ nhằm tự động hóa quá trình thống kê, đánh giá, giúp nhà trường giảm tải khối lượng công việc và đồng thời tạo điều kiện để sinh viên dễ dàng nắm bắt tiến độ, tiêu chí rèn luyện của bản thân.

Xuất phát từ thực tiễn đó, em lựa chọn thực hiện đề tài:

“Hệ thống quản lý điểm phục vụ cộng đồng và sinh viên 5 tốt của Đại Học Bách Khoa Đà Nẵng.”

Đề tài không chỉ hướng đến việc số hóa quá trình quản lý và đánh giá rèn luyện sinh viên, mà còn tích hợp các công nghệ hiện đại như chatbot hỗ trợ tư vấn nội quy, tạo và truy vấn sự kiện tình nguyện, qua đó tăng tính tương tác và nâng cao trải nghiệm người dùng. Bằng việc ứng dụng công nghệ thông tin và trí tuệ nhân tạo vào hoạt động giáo dục, hệ thống này hứa hẹn sẽ góp phần quan trọng vào việc xây dựng một môi trường học tập hiện đại, linh hoạt và hiệu quả, đồng thời tạo tiền đề cho việc mở rộng ứng dụng các giải pháp công nghệ vào các lĩnh vực khác trong nhà trường.

### 2. Mục tiêu đề tài

Đề tài đặt ra các mục tiêu cụ thể như:

- Xây dựng hệ thống quản lý tích hợp giúp theo dõi và đánh giá sinh viên theo tiêu chí "Sinh viên 5 tốt" và điểm phục vụ cộng đồng.
- Tối ưu hóa quy trình quản lý sự kiện phục vụ cộng đồng, giúp sinh viên đăng ký dễ dàng và nhà trường quản lý hiệu quả.
- Hỗ trợ sinh viên tra cứu điểm phục vụ cộng đồng nhanh chóng và minh bạch.
- Tích hợp AI chatbot để hỗ trợ sinh viên giải đáp thắc mắc về nội quy của trường.
- Phát triển AI Agent giúp nhà trường tự động tạo sự kiện theo yêu cầu.

- Giảm tải công việc thủ công, giúp ban quản lý dễ dàng theo dõi, đánh giá và tổng hợp dữ

### **3. Phạm vi nghiên cứu**

Đề tài tập trung xây dựng hệ thống quản lý điểm phục vụ cộng đồng và đánh giá “Sinh viên 5 tốt” dành cho sinh viên Đại học Bách Khoa Đà Nẵng. Hệ thống bao gồm các chức năng quản lý điểm, theo dõi tiêu chí, thông kê báo cáo và tích hợp chatbot tư vấn nội quy, tạo – truy vấn sự kiện tình nguyện. Mục tiêu là tự động hóa quy trình quản lý, giảm bớt thao tác thủ công và nâng cao trải nghiệm người dùng. Đề tài không mở rộng sang quản lý tài chính hay giảng dạy trực tuyến và chủ yếu áp dụng trong phạm vi trường Đại học Bách Khoa.

### **4. Đối tượng nghiên cứu**

Đối tượng nghiên cứu bao gồm thói quen và nhu cầu quản lý điểm phục vụ cộng đồng và tiêu chí “Sinh viên 5 tốt” của sinh viên Đại học Bách Khoa Đà Nẵng; quy trình tổ chức và tham gia các sự kiện phục vụ cộng đồng; các phương thức tương tác với chatbot hỗ trợ tư vấn nội quy và sự kiện; cùng với các công nghệ phát triển hệ thống quản lý trực tuyến và ứng dụng trí tuệ nhân tạo trong việc nâng cao hiệu quả quản lý và trải nghiệm người dùng.

### **5. Phương pháp nghiên cứu**

Đề tài được thực hiện dựa trên phương pháp nghiên cứu ứng dụng, kết hợp giữa khảo sát thực tế, phân tích yêu cầu nghiệp vụ và triển khai giải pháp công nghệ. Cụ thể, nhóm nghiên cứu đã:

- Khảo sát và thu thập thông tin từ sinh viên và ban quản lý để hiểu rõ quy trình hiện tại trong việc theo dõi điểm phục vụ cộng đồng và xét duyệt danh hiệu “Sinh viên 5 tốt”.
- Phân tích hệ thống thủ công hiện tại, xác định các bất cập, từ đó đưa ra các yêu cầu chức năng cần thiết cho hệ thống.

Trong đó, công nghệ để xây dựng và triển khai sản phẩm bao gồm:

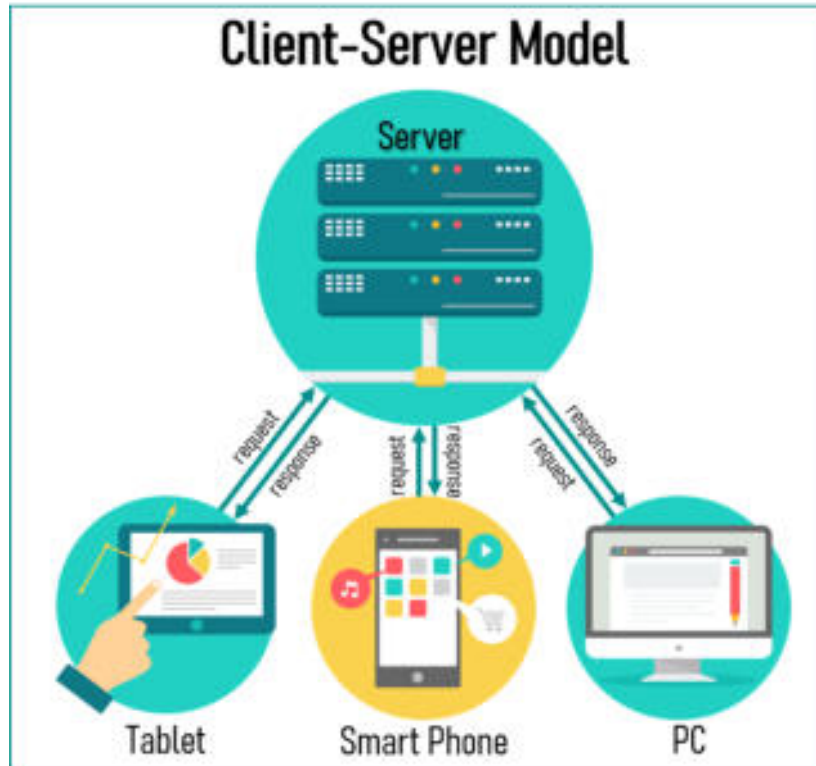
1. Giao diện sản phẩm (Frontend)
  - a. ReactJS
  - b. Tailwind CSS
  - c. Vite
2. Máy chủ (Backend)
  - a. Java Spring Boot
  - b. Spring security
  - c. Docker
3. Cơ sở dữ liệu
  - a. PostgreSQL

4. Triển khai sản phẩm
  - a. Render
5. Quản lý mã nguồn
  - a. GitHub
  - b. Git
6. Cấu trúc của đồ án
  - Mở đầu
  - Chương 1: Cơ sở lý thuyết
  - Chương 2: Phân tích và thiết kế hệ thống
  - Chương 3: Triển khai hệ thống
  - Kết luận

## CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

### 1.1 Mô hình Khách-Chủ (Client – Server)

Dự án lựa chọn mô hình máy khách - máy chủ (client-server) để triển khai.



*Hình 1 Mô hình client-server*

Mô hình Client Server[1] là một mô hình mạng máy tính gồm hai thành phần chính là client và server. Client sẽ là bên yêu cầu dịch vụ cài đặt cũng như lưu trữ tài nguyên từ server. Khi client gửi yêu cầu dữ liệu đến server qua Internet, server sẽ xử lý yêu cầu và gửi các gói dữ liệu cho client.

Mô hình Client Server[1] sử dụng cấu trúc phân tán để phân chia nhiệm vụ giữa client và server. Mô hình còn có một số ứng dụng phổ biến như email và World Wide Web (WWW)...

Ưu điểm của mô hình client-server:

- **Khả năng tập trung:** Client server có tích hợp Centralization (Tập trung hóa) giúp quản trị viên mạng dễ dàng quản lý và giải quyết vấn đề.
- **Tính bảo mật:** Dữ liệu được bảo vệ tốt hơn và tránh truy cập trái phép nhờ vào cấu trúc tập trung và các biện pháp hiện đại như firewall, mã hóa...
- **Hiệu quả cao:** Server có thể xử lý nhiều yêu cầu cùng lúc, giúp tăng hiệu suất tài nguyên cho client.
- **Khả năng mở rộng:** Người dùng có thể tăng số lượng tài nguyên như số lượng

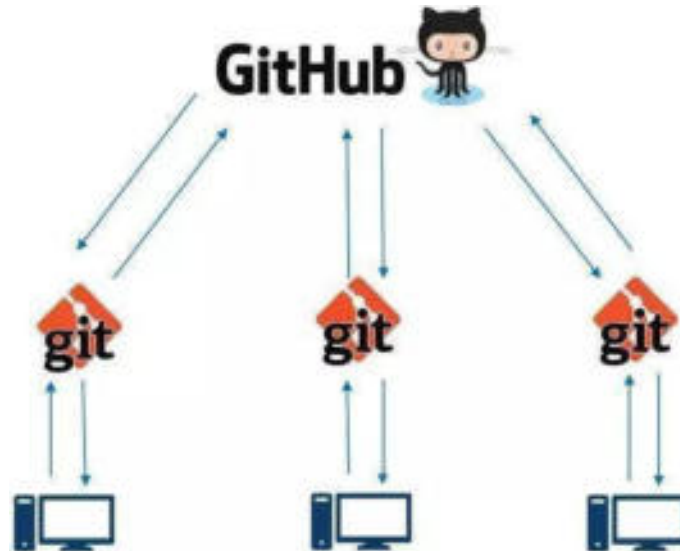
client hoặc server mà không bị gián đoạn nhờ khả năng mở rộng tốt của mô hình.

- **Khả năng truy cập:** Mô hình client-server không phân biệt mà cho phép người dùng truy cập thông tin từ bất kỳ vị trí hoặc nền tảng nào.
- Nhược điểm của mô hình client-server:
- **Tắc nghẽn lưu lượng:** Khi có quá nhiều client gửi request từ cùng 1 server sẽ dẫn đến kết nối chậm và gây tắc nghẽn.
- **Chi phí:** Bởi vì hệ thống mạng có sức mạnh lớn nên việc bảo trì và thiết lập thường khá cao mà không phải người dùng nào cũng chi trả được.
- **Độ bền:** Vì client – server là mạng tập trung nên server chỉ cần xảy ra sự cố hay bị nhiễu thì toàn bộ hệ thống mạng sẽ bị gián đoạn. Do vậy, các mạng client – server sẽ thiếu tính ổn định và độ bền.
- **Phức tạp:** Mô hình client-server có cấu trúc phức tạp và yêu cầu bảo trì thường xuyên. Vì vậy, luôn cần có quản trị viên mạng chuyên dụng để đảm bảo server hoạt động ổn định.
- **Tài nguyên:** Không phải tất cả tài nguyên trên server đều có thể truy cập từ client. Ví dụ, bạn không thể in tài liệu trực tiếp từ web hoặc chỉnh sửa bất kỳ thông tin nào trên ổ cứng của Client.

Tuy vẫn có những nhược điểm như đã nêu, nhưng nhìn chung mô hình này vẫn là mô hình phù hợp nhất cho đề tài khi nó cho phép tiếp cận được nhiều người dùng nhất trên internet nhưng vẫn giữ được tính bảo mật của hệ thống.

## 1.2 Git và Github

Đề tài sử dụng Git và Github để quản lý mã nguồn của sản phẩm. Sau đây là một vài đặc điểm của 2 công nghệ này:



*Hình 2 Github*

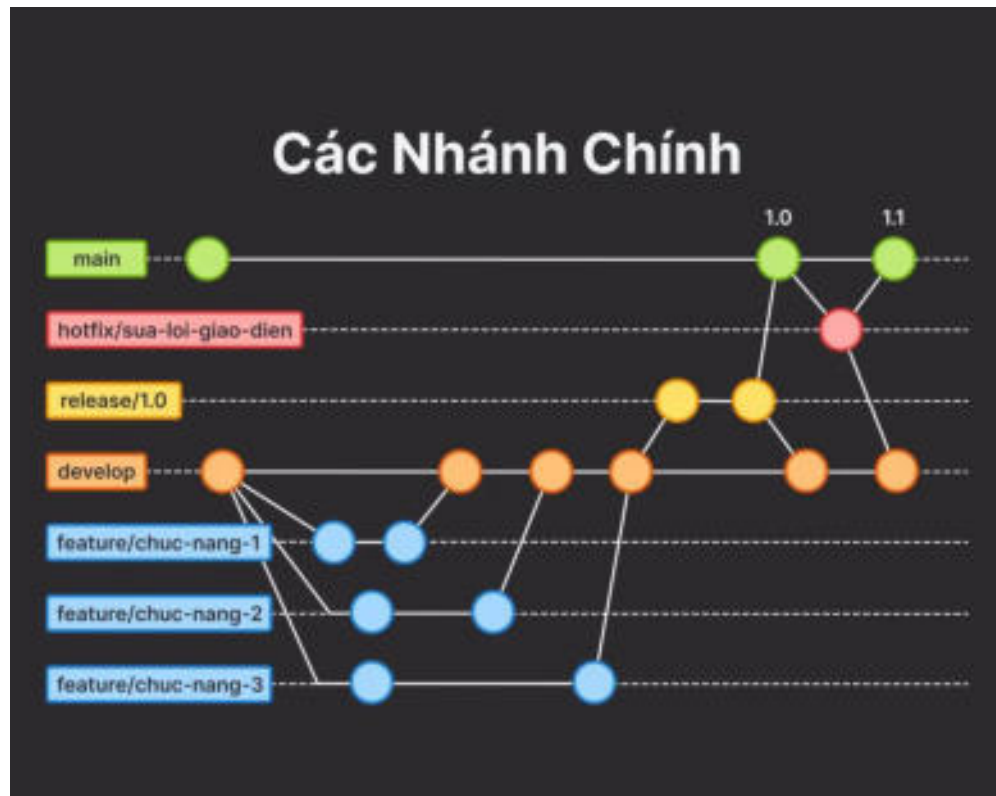
**Git**[2] là một hệ thống quản lý phiên bản phân tán (Distributed Version Control System – DVCS) rất được ưa chuộng, được sử dụng để theo dõi và quản lý các sự thay đổi trong mã nguồn trong suốt quá trình phát triển phần mềm. Được tạo ra bởi Linus Torvalds vào năm 2005, Git đã nhanh chóng trở thành một trong những công cụ quản lý phiên bản phổ biến và quan trọng nhất trên toàn cầu.

Mục đích chính của Git[2] là giúp đội ngũ phát triển làm việc cùng nhau trên cùng một dự án mà không gặp khó khăn về việc ghi đè lên công việc của nhau. Nó cho phép mọi người làm việc trên các phiên bản riêng của mã nguồn và sau đó hợp nhất các thay đổi này thành phiên bản chính thức, được gọi là "nhánh chính" (thường được gọi là "master" hoặc "main").

**GitHub**[3] là một nền tảng và dịch vụ đám mây giúp phát triển phần mềm và quản lý nhiều phiên bản mã nguồn của dự án bằng Git, cho phép các nhà phát triển lưu trữ, quản lý cùng lúc nhiều phiên bản mã nguồn và cộng tác với những người khác.

GitHub[3] được phát triển từ tháng 10/2007 và được ra mắt vào tháng 4/2008 bởi các nhà sáng lập Tom Preston-Werner, Chris Wanstrath, P. J. Hyett và Scott Chacon.

## Quy trình làm việc với Git (Git flow):



*Hình 3 Git Flow*

**Git Flow**[4] là một mô hình quản lý quy trình làm việc với Git được Vincent Driessen đề xuất, và hiện đang được áp dụng rộng rãi nhờ tính hệ thống và tổ chức rõ ràng. Mô hình này đặc biệt phù hợp với các dự án có chu kỳ phát hành phức tạp, giúp phân tách rõ ràng giữa các giai đoạn phát triển, kiểm thử và triển khai.

Mặc dù trong hình minh họa ta có thể thấy quy trình Git Flow[4] được áp dụng một cách tổng quát, nhưng khi triển khai vào đề tài này, quy trình đã được điều chỉnh linh hoạt để phù hợp với quy mô và đặc thù riêng của dự án. Tuy vậy, các nguyên tắc cốt lõi của Git Flow như quản lý nhánh phát triển, nhánh tính năng, nhánh phát hành và nhánh hotfix vẫn được đảm bảo và duy trì xuyên suốt quá trình phát triển phần mềm.

### 1.3 Tổng quan về Spring Boot

Spring Boot[5] là một framework được xây dựng dựa trên Spring Framework, được thiết kế để đơn giản hóa việc phát triển các ứng dụng Java bằng cách tự động hóa các cấu hình phức tạp và cung cấp các thiết lập mặc định tối ưu.

#### Các đặc trưng của Spring Boot:

**Tự động cấu hình:** Tự động cấu hình là một phương pháp làm việc trong Spring Boot giúp giảm số bước mà nhà phát triển phải thực hiện. Nó tự động định cấu hình ứng dụng Spring dựa trên các phần phụ thuộc đã được thêm trước đó. Theo mặc định, cấu hình tự động của Spring Boot cung cấp một số tính năng mạnh mẽ trong khi vẫn duy trì tính linh hoạt cao.

**Người phụ thuộc có ý kiến:** Có ý kiến có nghĩa là Spring Boot tự xác định một tập

hợp các Bean được cấu hình mặc định mà bạn có thể ghi đè nếu cần. Hơn nữa, khung này chọn các gói để cài đặt dựa trên các phụ thuộc mà bạn cần. Do đó, các nhà phát triển Spring Boot ngay lập tức bắt đầu xây dựng ứng dụng của họ, chú ý nhiều hơn đến logic nghiệp vụ, vì hầu hết công việc đều được thực hiện bởi chính framework.

**Máy chủ nhúng:** Máy chủ nhúng là một phần của ứng dụng. Điều này có nghĩa là bạn không cần cài đặt sẵn nó trong môi trường triển khai của mình. Spring Boot cung cấp máy chủ Tomcat tích hợp theo mặc định, nhưng bạn có thể thay đổi nó thành Jetty hoặc Undertow. Máy chủ nhúng cho phép triển khai hiệu quả hơn và giảm thời gian khởi động lại ứng dụng.

**Làm việc ngoại tuyến:** Spring Boot cho phép các nhà phát triển định cấu hình và chạy các ứng dụng Spring độc lập ở cấp độ sản xuất mà không cần triển khai chúng lên máy chủ web. Để chạy một ứng dụng Java thông thường, nó phải được đóng gói, chọn, tải và định cấu hình như một máy chủ web và được triển khai. Một ứng dụng Java được xây dựng trong Spring Boot chỉ cần được đóng gói và sau đó nó sẵn sàng chạy bằng các lệnh đơn giản.

#### **Ưu điểm của Spring Boot:**

- Phát triển ứng dụng nhanh chóng và dễ dàng bằng Spring.
- Tự động định cấu hình tất cả các thành phần cho ứng dụng Spring cấp sản xuất.
- Máy chủ nhúng sẵn dùng (Tomcat, Jetty và Undertow) để triển khai ứng dụng nhanh hơn, hiệu quả hơn.
- Điểm cuối HTTP cho phép bạn nhập các chức năng ứng dụng nội bộ như số liệu, trạng thái sức khỏe và các chức năng khác.
- Không có cấu hình XML.
- Một lượng lớn các plugin giúp nhà phát triển làm việc dễ dàng hơn với cơ sở dữ liệu nhúng và trong bộ nhớ.
- Dễ dàng truy cập vào cơ sở dữ liệu và các dịch vụ xếp hàng như MySQL, Oracle, MongoDB, Redis, ActiveMQ, v.v.
- Tích hợp liền mạch với hệ sinh thái Spring.
- Cộng đồng lớn và nhiều chương trình đào tạo giúp giai đoạn nhập môn trở nên dễ dàng hơn.

#### **Nhược điểm của Spring Boot:**

- Thiếu kiểm soát. Spring Boot tạo ra rất nhiều dependency không được sử dụng, dẫn đến file triển khai có dung lượng lớn.
- Quá trình phức tạp và tốn thời gian để chuyển đổi một dự án Spring cũ hoặc hiện có thành các ứng dụng Spring Boot.
- Không phù hợp cho các dự án quy mô lớn. Theo nhiều nhà phát triển, mặc dù không có vấn đề gì khi làm việc với microservice nhưng Spring Boot không phù

hợp để tạo các ứng dụng nguyên khối

## 1.4 Hệ quản trị cơ sở dữ liệu PostgreSQL

PostgreSQL[6] là một hệ thống quản trị cơ sở dữ liệu quan hệ-đối tượng (object-relational database management system) mã nguồn mở. Nó được phát triển dựa trên POSTGRES 4.2 tại phòng khoa học máy tính Berkeley, Đại học California và được thiết kế để chạy trên các nền tảng tương tự UNIX nhưng cũng có thể chạy trên các nền tảng khác như Windows và Mac OS X

### ***Lý do lựa chọn PostgreSQL***

- Cộng đồng lớn và chuyên nghiệp: PostgreSQL được phát triển và bảo trì bởi một cộng đồng lớn và đầy nhiệt huyết. Những người tham gia trong cộng đồng này đều là những chuyên gia về cơ sở dữ liệu và có kinh nghiệm thực tiễn.
- Bảo mật cao: Cung cấp nhiều tính năng bảo mật như hệ thống kiểm soát truy cập mạnh mẽ, bảo mật cấp độ cột và hàng, xác thực đa dạng và khả năng mã hóa dữ liệu.
- Tính linh hoạt: Cho phép định nghĩa các kiểu dữ liệu tùy chỉnh, xây dựng các hàm tùy chỉnh và viết mã từ các ngôn ngữ lập trình khác nhau. Điều này giúp cho các nhà phát triển có thể tùy chỉnh cơ sở dữ liệu cho phù hợp với nhu cầu của ứng dụng.
- Hỗ trợ trên nhiều hệ điều hành: PostgreSQL có thể chạy trên nhiều hệ điều hành khác nhau như Linux, Windows và macOS.
- Cập nhật thường xuyên: Được phát triển và cập nhật thường xuyên, với các phiên bản mới được phát hành để cải thiện tính năng, tăng hiệu suất và sửa các lỗi bảo mật.

Việc sử dụng PostgreSQL trong hệ thống học tập trực tuyến đã mang lại nhiều lợi ích về hiệu suất, tính ổn định và bảo mật. Đây là một lựa chọn phù hợp cho các ứng dụng yêu cầu quản lý dữ liệu phức tạp và lớn. PostgreSQL không chỉ đáp ứng được các yêu cầu hiện tại mà còn có khả năng mở rộng và phát triển trong tương lai.

## 1.5 Cloudinary

**Cloudinary**[7] là một nền tảng đám mây mạnh mẽ chuyên dùng để quản lý, lưu trữ, tối ưu hóa và phân phối hình ảnh cũng như video trên các ứng dụng web và di động. Với giao diện thân thiện và API linh hoạt, Cloudinary[7] giúp các nhà phát triển xử lý phương tiện một cách hiệu quả, giảm tải cho máy chủ và nâng cao trải nghiệm người dùng.

### **Lý do lựa chọn Cloudinary**

- **Tối ưu hiệu suất tải trang:** Cloudinary hỗ trợ nén ảnh, chuyên đổi định dạng thông minh (ví dụ: Web), giúp rút ngắn thời gian tải hình ảnh và video mà vẫn giữ chất lượng cao.

- **Quản lý media dễ dàng:** Cung cấp giao diện quản trị trực quan, hỗ trợ phân loại, tìm kiếm và theo dõi tài nguyên media hiệu quả.
- **Tự động xử lý ảnh và video:** Cho phép thay đổi kích thước, cắt, xoay, thêm watermark, tạo thumbnail, trích xuất frame từ video... thông qua các URL động hoặc API.  
**Phân phối qua CDN toàn cầu:** Tích hợp sẵn mạng phân phối nội dung (CDN), giúp truy cập ảnh/video nhanh chóng ở bất kỳ đâu trên thế giới.
- **Khả năng tích hợp cao:** Dễ dàng tích hợp với các framework và nền tảng như React, Vue, Angular, Node.js, .NET, v.v.

Việc sử dụng Cloudinary trong hệ thống quản lý điểm phục vụ cộng đồng và sinh viên 5 tốt giúp tối ưu hóa quá trình quản lý và hiển thị hình ảnh sự kiện:

- Tải ảnh sự kiện, ảnh minh chứng hoạt động nhanh chóng, giúp giảm tải cho server backend.
- Giảm băng thông và nâng cao trải nghiệm người dùng.
- Tăng khả năng mở rộng của hệ thống khi số lượng sự kiện ngày càng nhiều, đảm bảo hiệu suất tải hình ảnh và tài liệu vẫn ổn định và mượt mà.

## 1.6 Supabase

Supabase[8] là một nền tảng Backend as a Service (BaaS) mã nguồn mở, được tạo ra như một giải pháp thay thế cho Firebase nhưng với một triết lý hoàn toàn khác: tất cả đều dựa trên công nghệ mã nguồn mở và các tiêu chuẩn mở.

Trong dự án, Supabase[8] được lựa chọn làm giải pháp lưu trữ file nhằm tối ưu quản lý dữ liệu đa phương tiện như tài liệu minh chứng. Supabase[8] cung cấp dịch vụ lưu trữ đám mây đáng tin cậy với các ưu điểm nổi bật:

- **Dễ dàng tích hợp và sử dụng:** Supabase hỗ trợ API đơn giản, cho phép nhanh chóng upload, truy xuất và quản lý file trực tiếp từ ứng dụng, giúp giảm thiểu thời gian phát triển.
- **Bảo mật và phân quyền linh hoạt:** Supabase cung cấp hệ thống phân quyền chi tiết, đảm bảo quyền truy cập file được kiểm soát nghiêm ngặt theo từng nhóm người dùng và vai trò.
- **Khả năng mở rộng cao:** Khi lượng dữ liệu và số lượng người dùng tăng lên, Supabase vẫn duy trì hiệu suất ổn định, giúp hệ thống vận hành mượt mà, không bị nghẽn do lưu trữ file.
- **Chi phí hợp lý và tích hợp đa năng:** So với các giải pháp lưu trữ truyền thống, Supabase giúp tiết kiệm chi phí vận hành và dễ dàng kết nối với các dịch vụ backend khác trong dự án.

Nhờ sử dụng Supabase, hệ thống quản lý điểm phục vụ cộng đồng và sinh viên 5 tốt của trường Đại học Bách Khoa Đà Nẵng đã nâng cao được hiệu quả quản lý dữ liệu đa phương

tiện, đồng thời cải thiện trải nghiệm người dùng trong việc lưu trữ và truy cập các tài liệu cần thiết.

## **1.7 Docker**

Docker[9] là một nền tảng mở để phát triển, vận chuyển và chạy các ứng dụng. Docker cho phép bạn tách ứng dụng khỏi cơ sở hạ tầng để bạn có thể phân phối phần mềm một cách nhanh chóng. Với Docker, bạn có thể quản lý cơ sở hạ tầng của mình giống như cách bạn quản lý ứng dụng của mình. Bằng cách tận dụng các phương pháp của Docker để vận chuyển, thử nghiệm và triển khai mã, bạn có thể giảm đáng kể độ trễ giữa việc viết mã và chạy mã trong sản xuất [12].

Các container cung cấp môi trường nhẹ và nhất quán bao gồm một ứng dụng và các phần phụ thuộc của nó

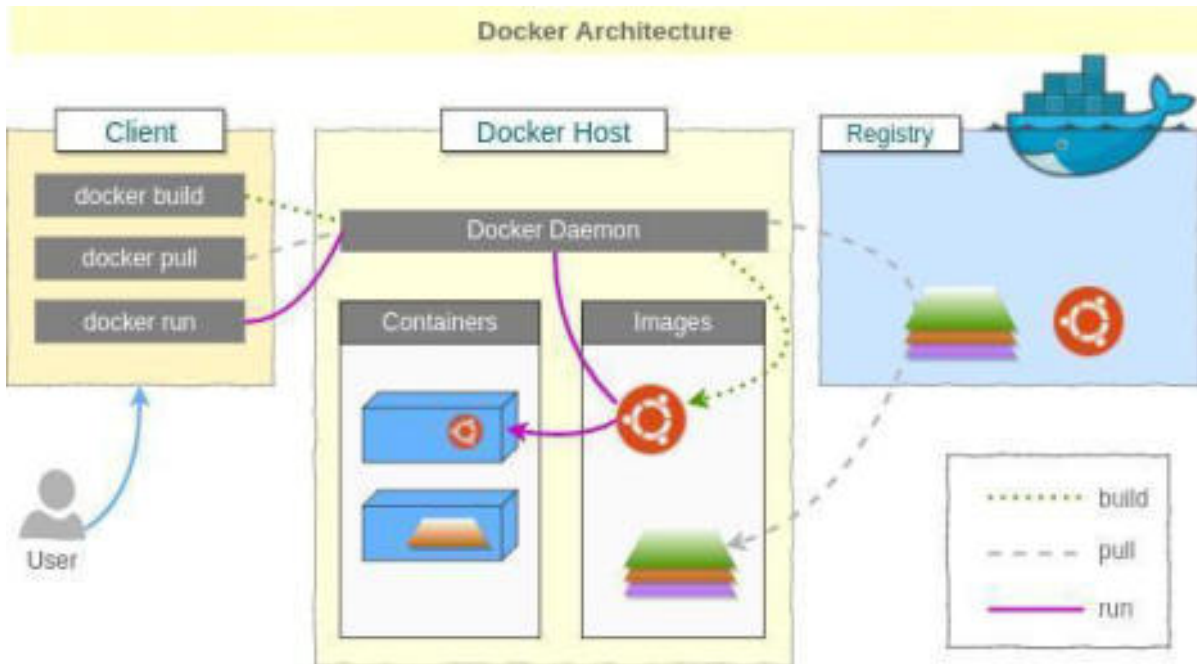
### ***Đặc trưng cơ bản của Docker***

- Consistency - Tính nhất quán: loại bỏ các sự cố "nó hoạt động được trên máy của tôi" bằng cách đảm bảo môi trường nhất quán: mọi dev đều có chung môi trường và chạy được project.
- Isolation - Tính cô lập, độc lập: Container độc lập với các ứng dụng, ngăn ngừa sự xung đột giữa các phần mềm.
- Portability - Tính di động, linh động: chạy các container trên bất kỳ hệ thống nào hỗ trợ docker, bất kể cơ sở hạ tầng bên dưới nào.
- Efficiency - Tính hiệu quả: Các container chia sẻ nhân hệ điều hành máy chủ, giảm chi phí tài nguyên.

### **So sánh Docker và Virtual Machines:**

+ Docker container nhẹ hơn so với VM truyền thống.

+ VM chạy trên các hệ điều hành guest riêng biệt, trong khi các container chia sẻ với hệ điều hành máy chủ, thông qua docker engine để ảo hóa một OS. Điều này dẫn đến thời gian khởi động nhanh hơn và sử dụng tài nguyên hiệu quả hơn với các container.



Hình 4 Kiến trúc Docker

Việc sử dụng Docker trong hệ thống học tập trực tuyến đã mang lại nhiều lợi ích về tính nhất quán, hiệu quả tài nguyên và khả năng quản lý. Docker không chỉ đảm bảo ứng dụng chạy mượt mà trong mọi môi trường mà còn giúp dễ dàng triển khai và mở rộng hệ thống. Docker là một công cụ mạnh mẽ và linh hoạt, phù hợp cho các ứng dụng yêu cầu tính nhất quán và khả năng mở rộng cao.

### 1.8 Render

Render[10] là một nền tảng lưu trữ ứng dụng đám mây dành cho các nhà phát triển triển khai các trang web tĩnh và các ứng dụng động. Nó cũng cung cấp một cơ sở dữ liệu được lưu trữ được quản lý hoàn toàn với kết nối nội bộ và bên ngoài. Điều này làm cho nó hữu ích cho cả nhà phát triển frontend và backend.

Các dịch vụ chính:

- Web Services: Cho phép deploy các ứng dụng web tĩnh hoặc động (React, Vue, Node.js, Python, Ruby, v.v.). Tự động build và deploy từ các repository GitHub, GitLab. Hỗ trợ HTTPS tự động, custom domain và CDN.
- Background Workers : Chạy các tác vụ không đồng bộ, xử lý công việc nền như gửi email, xử lý dữ liệu, cron jobs. Giúp tách biệt công việc nặng khỏi luồng chính của ứng dụng.
- Database: Cung cấp các dịch vụ cơ sở dữ liệu quản lý (Managed Databases) như PostgreSQL. Tự động sao lưu, mở rộng và bảo trì cơ sở dữ liệu.
- Cron Jobs : Cho phép lên lịch các tác vụ chạy định kỳ như gửi báo cáo, đồng bộ dữ liệu..
- Static Sites: Host các trang web tĩnh với hiệu năng cao, sử dụng CDN phân

phối toàn cầu. Thích hợp cho các blog, landing page, hoặc frontend tĩnh.

- Private Services: Cho phép deploy các dịch vụ backend, API không công khai, bảo vệ truy cập bằng mạng riêng.
- Docker Support: Hỗ trợ deploy ứng dụng sử dụng Docker container, giúp linh hoạt trong việc triển khai các môi trường phức tạp.

#### **Lý do lựa chọn Render:**

- Hỗ trợ triển khai tự động từ GitHub/GitLab, giúp tiết kiệm thời gian phát triển.
- Cung cấp nhiều dịch vụ như Web Services, Databases, Cron Jobs trong một nền tảng duy nhất.
- Dễ sử dụng, không cần cấu hình phức tạp, phù hợp với nhóm sinh viên.
- Có gói miễn phí phù hợp cho dự án học tập, tiết kiệm chi phí.
- Hỗ trợ HTTPS tự động và custom domain, đảm bảo bảo mật cho hệ thống.
- Khả năng mở rộng tốt, hoạt động ổn định khi số lượng người dùng tăng cao.
- Giao diện quản lý trực quan, dễ theo dõi và quản lý tài nguyên.

### **1.9 n8n**

#### ***Giới thiệu về n8n***

n8n[11] là một nền tảng automation mã nguồn mở, giúp developers dễ dàng tạo workflow để kết nối các ứng dụng, dịch vụ và API mà không cần phải viết quá nhiều code. Được thiết kế để giải quyết bài toán tích hợp và tự động hóa workflows, n8n nổi bật với tính linh hoạt và khả năng tùy chỉnh cao.

#### **Những đặc điểm nổi bật của :**

- Mã nguồn mở: Mọi thứ đều công khai trên GitHub, cho phép bạn dễ dàng kiểm tra, tùy chỉnh, và đóng góp vào codebase.
- Self-hosted: Bạn có thể deploy n8n trên infrastructure của riêng mình để đảm bảo toàn quyền kiểm soát dữ liệu.
- Khả năng mở rộng: Hỗ trợ tạo custom nodes, giúp tích hợp các dịch vụ mà n8n chưa có sẵn.
- JavaScript/TypeScript native: Các node functions được viết bằng JavaScript hoặc TypeScript – một lợi thế lớn cho developers đã quen thuộc với những ngôn ngữ này.
- Expression support: Cho phép xử lý trực tiếp dữ liệu trong workflow với JavaScript expressions, giúp tăng sự linh hoạt khi xây dựng workflows.
- Tích hợp AI: Hỗ trợ tích hợp các chức năng AI như chatbot và phân tích dữ liệu để nâng cao khả năng tương tác và ra quyết định tự động.

#### ***Lý do lựa chọn n8n:***

- Hỗ trợ xây dựng quy trình tự động (workflow) trực quan bằng kéo thả, dễ dàng tích hợp AI vào hệ thống.

- Tích hợp sẵn với nhiều API phổ biến như OpenAI, Hugging Face, giúp triển khai AI nhanh chóng.
- Cho phép xử lý dữ liệu đầu vào/đầu ra linh hoạt, phù hợp với các tác vụ AI như tạo nội dung, phân tích văn bản,...
- Mã nguồn mở, có thể tùy chỉnh theo yêu cầu và triển khai nội bộ để đảm bảo bảo mật.
- Hỗ trợ chạy các script tùy chỉnh (JavaScript, Python), thuận tiện cho việc xử lý logic AI phức tạp.
- Kết nối dễ dàng với các hệ thống khác như cơ sở dữ liệu, email, chatbot, API,... để xây dựng ứng dụng AI hoàn chỉnh.
- Tiết kiệm thời gian và công sức phát triển nhờ khả năng tự động hóa toàn bộ quy trình AI.

### 1.9.1 ReactJs

React[12] là một thư viện UI phát triển tại Facebook để hỗ trợ việc xây dựng những thành phần (components) UI có tính tương tác cao, có trạng thái và có thể sử dụng lại được. React được sử dụng tại Facebook trong production, và [www.instagram.com](https://www.instagram.com) được viết hoàn toàn trên React.

Một trong những điểm hấp dẫn của React là thư viện này không chỉ hoạt động trên phía client, mà còn được render trên server và có thể kết nối với nhau.

React sử dụng khái niệm DOM ảo (Virtual DOM) để chọn lựa và render những phần tử của node dựa trên sự thay đổi trạng thái khiến cho ta chỉ cần thay đổi ít thành phần nhất có thể để giữ DOM update.

### 1.9.2 TailwindCss

Tailwind CSS[13] là một framework CSS tiện ích, công cụ này cung cấp các lớp tiện ích sẵn có. Điều này giúp người dùng dễ dàng tạo giao diện người dùng mà không cần viết CSS tùy chỉnh. Khác với các framework như Bootstrap, nơi sử dụng các thành phần đã được định nghĩa trước, Tailwind CSS cho phép bạn tạo kiểu linh hoạt và tùy chỉnh hoàn toàn theo ý muốn.

#### Các điểm chú ý của tailwinds css

- Xây dựng giao diện nhanh chóng: Chỉ cần khai báo các lớp CSS trực tiếp trong file HTML.
- Tối ưu hóa file CSS: Chỉ xuất ra các lớp CSS đã sử dụng trong giao diện, giúp giảm dung lượng file.
- Thiết lập Responsive dễ dàng: Có thể thực hiện ngay trên file HTML mà không cần thêm mã CSS riêng biệt.
- Hỗ trợ tương tác: Cung cấp các lớp cho hiệu ứng hover, focus và nhiều hiệu ứng tương tác khác ngay trong HTML.

- Tối ưu hóa cấu trúc lặp lại: Giúp giảm thiểu sự lặp lại trong HTML và CSS.
- Tạo lớp mới dễ dàng: Cho phép định nghĩa các lớp mới kết hợp từ các lớp có sẵn trong Tailwind. Hỗ trợ chế độ tối: Tích hợp Dark Mode để giao diện phù hợp với chế độ tối.
- Dễ dàng mở rộng và chỉnh sửa: Tailwind CSS cho phép bạn dễ dàng thay đổi và điều chỉnh cấu hình.
- Tăng tốc độ lập trình: Cung cấp các Extension thông minh trên VS Code để nâng cao hiệu quả công việc.

### 1.10 HTML, CSS và TypeScript

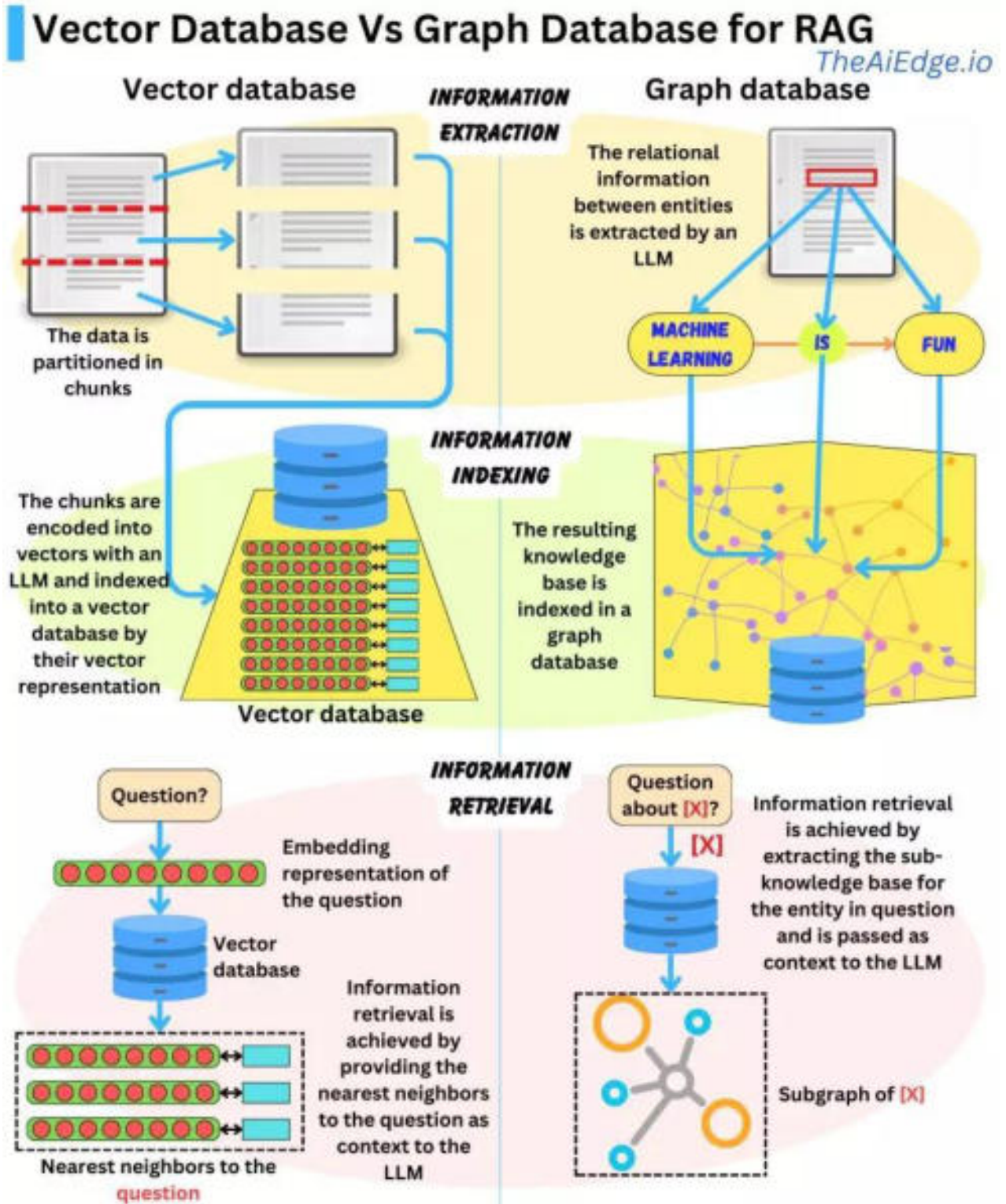
Về nền tảng cơ bản, giao diện của sản phẩm được xây dựng dựa trên ngôn ngữ đánh dấu siêu văn bản HTML - HyperText Markup Language, ngôn ngữ thiết kế CSS - Cascading Style Sheets và ngôn ngữ lập trình TypeScript[14].

Trong đó:

- **HTML** (HyperText Markup Language) là ngôn ngữ đánh dấu chuẩn để tạo và thiết kế các trang web. Nó sử dụng các phần tử và thẻ để mô tả cấu trúc của trang, bao gồm văn bản, hình ảnh, liên kết và các yếu tố đa phương tiện khác.
- **CSS** (Cascading Style Sheets) là ngôn ngữ thiết kế để mô tả cách các phần tử HTML hiển thị trên màn hình. Nó trang trí cho trang web bằng cách xác định các kiểu dáng như màu sắc, font chữ và bố cục.
- **TypeScript** là ngôn ngữ mã nguồn mở của Microsoft, mở rộng từ JavaScript bằng cách thêm kiểm tra kiểu tĩnh và tính năng hiện đại. Nó giúp phát triển JavaScript quy mô lớn dễ dàng và an toàn hơn bằng việc định rõ kiểu dữ liệu.
- **JavaScript** là ngôn ngữ phổ biến cho việc tạo các trang web động và tương tác. Hỗ trợ trên mọi trình duyệt, cũng như cho phát triển ứng dụng phía khách hàng và máy chủ.

## 1.11 Kiến trúc RAG

### Tổng quan về RAG



Hình 5 Mô hình hoạt động của RAG

Retrieval-Augmented Generation (RAG) [15] là một kỹ thuật giúp nâng cao khả năng của mô hình sinh (language model generation) kết hợp với tri thức bên ngoài (external knowledge)

Phương pháp này thực hiện bằng cách truy xuất thông tin liên quan từ kho tài liệu

(tri thức) và sử dụng chúng cho quá trình sinh câu trả lời dựa trên LLMs.

### **Tóm tắt ngắn gọn quá trình của RAG như sau:**

Create Vector database: Đầu tiên, convert toàn bộ dữ liệu tri thức thành các vector và lưu trữ chúng vào một vector database.

1. User input: User cung cấp 1 câu truy vấn (query) bằng ngôn ngữ tự nhiên nhằm tìm kiếm câu trả lời hoặc để hoàn thành câu truy vấn đó.
2. Information retrieval: Cơ chế retrieval quét toàn bộ vector trong database để xác định các phân đoạn tri thức (chính là paragraphs) nào có ngữ nghĩa tương đồng với câu truy vấn của người dùng. Các paragraphs này sau đó được vào LLM để làm tăng context cho quá trình sinh ra câu trả lời.
3. Combining data: Các paragraphs được lấy sau quá trình retrieval từ database được kết hợp với câu query ban đầu của user tạo thành 1 câu prompt.
4. Generate text: Câu prompt được bổ sung thêm context sau đó được đưa qua LLM để sinh ra câu phản hồi cuối cùng theo context bổ sung.

### **Ưu điểm của RAG:**

Với RAG, LLM có thể tận dụng dữ liệu bên ngoài để cung cấp tri thức cho nó.

- RAG không yêu cầu training lại mô hình, tiết kiệm thời gian và tài nguyên tính toán.
- Nó hiệu quả ngay cả với một lượng dữ liệu gán nhãn hạn chế.
- RAG phù hợp nhất cho các tình huống có nhiều dữ liệu chưa được gán nhãn (unlabeled-data) nhưng nguồn dữ liệu gán nhãn khan hiếm và lý tưởng cho các ứng dụng như trợ lý ảo cần truy cập theo thời gian thực vào thông tin cụ thể như hướng dẫn sử dụng phần mềm, tin tức, ...

### **1.12 Pinecone**

Pinecone[16] là một dịch vụ đám mây được quản lý hoàn toàn, chuyên về lưu trữ, quản lý và tìm kiếm các vector embeddings. Đây là một nền tảng giúp các nhà phát triển dễ dàng triển khai các ứng dụng dựa trên tìm kiếm vector mà không cần phải lo lắng về việc quản lý hạ tầng hoặc tối ưu hóa hiệu suất. Dưới đây là một số điểm chính về Pinecone:

Các Tính Năng Chính của Pinecone

1. **Tìm Kiếm Vector Hiệu Suất Cao:** Pinecone cung cấp các thuật toán và cơ sở dữ liệu tối ưu hóa để tìm kiếm các vector embeddings một cách nhanh chóng và chính xác.
2. **Quản Lý Hoàn Toàn:** Pinecone là một dịch vụ được quản lý hoàn toàn, nghĩa là bạn không cần phải lo lắng về việc triển khai, bảo trì hoặc mở rộng hạ tầng. Pinecone xử lý tất cả các khía cạnh này cho bạn.
3. **Khả Năng Mở Rộng:** Pinecone có thể mở rộng linh hoạt để xử lý từ hàng

triệu đến hàng tỷ vector embeddings, phù hợp với nhu cầu của các ứng dụng lớn.

4. **Dễ Dàng Sử Dụng:** Pinecone cung cấp các API dễ sử dụng để bạn có thể dễ dàng tích hợp với các ứng dụng hiện tại của mình. Nó hỗ trợ nhiều ngôn ngữ lập trình và framework phổ biến.
5. **Độ Tin Cậy Cao:** Với Pinecone, bạn có thể tin tưởng vào độ tin cậy và tính sẵn sàng cao của dịch vụ, giúp đảm bảo rằng các ứng dụng của bạn luôn hoạt động mượt mà.
6. **Tích Hợp AI:** Pinecone tích hợp tốt với các mô hình học sâu và các công cụ AI khác, giúp bạn dễ dàng sử dụng các vector embeddings từ các mô hình như OpenAI, TensorFlow, và nhiều hơn nữa.

### 1.13 Kết luận chương 1

Chương 1 cung cấp một cái nhìn tổng quan và chi tiết về các công nghệ và công cụ được sử dụng trong hệ thống. Các khái niệm như mô hình Khách-Chủ, Git và Spring Boot, PostgreSQL, Cloudinary, Supabase, Docker, HTML, CSS, TypeScript, ReactJS, Tailwind CSS, cấu trúc RAG và Pinecone đã được trình bày một cách rõ ràng và cụ thể. Những kiến thức này không chỉ giúp xây dựng nền tảng lý thuyết vững chắc mà còn định hướng cho việc phân tích và thiết kế hệ thống trong các chương sau.

## CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

### 2.1 yêu cầu chức năng

#### Sinh viên

Đăng nhập:

- Đăng nhập bằng mã số sinh viên và password
- Quản lý đăng nhập bằng token
- Có chức năng quên mật khẩu (gửi về email)
- Đăng ký tham gia sự kiện
- Hủy đăng ký sự kiện
- Xem thông tin chi tiết các sự kiện
- Đổi mật khẩu
- Xem các sự kiện đã đăng ký
- Xem các sự kiện đã tham gia
- Gửi các hoạt động bên ngoài trường
- Xem số điểm hoạt động cộng đồng của bản thân
- Chatbot tương tác với người dùng về nội quy nhà trường
- Tìm kiếm khóa học
- Xem thông tin cá nhân

#### Liên chi đoàn

Quản lý khóa học

- Xem thông tin tất cả sự kiện
- Xem các sự kiện do bản thân tạo ra
- Tạo sự kiện
- Chỉnh sửa sự kiện
- Xem danh sách các sinh viên tham gia sự kiện
- Xem thống kê sinh viên tham gia sự kiện
- Điểm danh các sinh viên tham gia sự kiện do bản thân tạo
- Xuất file excel danh sách tham gia sự kiện
- Chatbot truy xuất thông tin sự kiện
- Chatbot tự động tạo và hỗ trợ nội dung sự kiện

#### Hội sinh viên

- Xem thông tin tất cả sự kiện
- Xem các sự kiện do bản thân tạo ra

- Xem danh sách các sinh viên tham gia sự kiện
- Xem thống kê sinh viên tham gia sự kiện
- Tạo sự kiện
- Chỉnh sửa sự kiện
- Điểm danh các sinh viên tham gia sự kiện do bản thân tạo
- Xuất file excel danh sách tham gia sự kiện
- Chatbot truy xuất thông tin sự kiện
- Chatbot tự động tạo và hỗ trợ nội dung sự kiện
- Xem danh sách sinh viên 5 tốt và liên chi đoàn tốt
- Được quyền chỉnh sửa thông tin tiêu chí sinh viên 5 tốt
- Phê duyệt các hoạt động mà sinh viên tham gia bên ngoài trường

### **Cộng tác sinh viên**

- Xem thông tin tất cả sự kiện
- Xem các sự kiện do bản thân tạo ra
- Xem danh sách các sinh viên tham gia sự kiện
- Xem thống kê sinh viên tham gia sự kiện
- Tạo sự kiện
- Chỉnh sửa sự kiện
- Điểm danh các sinh viên tham gia sự kiện do bản thân tạo
- Xuất file excel danh sách tham gia sự kiện
- Chatbot truy xuất thông tin sự kiện
- Chatbot tự động tạo và hỗ trợ nội dung sự kiện
- Quản lý điểm phục vụ cộng đồng của sinh viên

### **Ban thời vụ đoàn trường**

- Xem thông tin tất cả sự kiện
- Xem các sự kiện do bản thân tạo ra
- Xem danh sách các sinh viên tham gia sự kiện
- Xem thống kê sinh viên tham gia sự kiện
- Tạo sự kiện
- Chỉnh sửa sự kiện
- Điểm danh các sinh viên tham gia sự kiện do bản thân tạo
- Xuất file excel danh sách tham gia sự kiện
- Chatbot truy xuất thông tin sự kiện
- Chatbot tự động tạo và hỗ trợ nội dung sự kiện
- Quản lý tiêu chí hoạt động liên chi đoàn
- Quản lý tài khoản

## **2.2 Yêu cầu về phi chức năng**

### **2.2.1 Hiệu suất (Performance)**

*Bảng 1 Hiệu suất*

<b>No.</b>	<b>Requirement</b>
1	Cho phép 1 lượng lớn truy cập bằng internet vào hệ thống
2	Thời gian phản hồi nhanh
3	Thời gian nhập/xuất dữ liệu nhanh
4	Thời gian tải video nhanh và có thể nhiều người xem cùng một lúc không bị lag

### **2.2.2 Khả năng mở rộng (Scalability)**

*Bảng 2 Khả năng mở rộng*

<b>No.</b>	<b>Requirement</b>
1	Khả năng lưu trữ dữ liệu lớn
2	Cho phép nhiều người dùng thực hiện các thao tác giống nhau trong cùng khoảng thời gian

### **2.2.3 Bảo mật (Security)**

*Bảng 3 Bảo mật*

<b>No.</b>	<b>Requirement</b>
1.	Hệ thống phải bảo vệ dữ liệu người dùng bằng cách sử dụng các phương pháp mã hóa mạnh mẽ cho dữ liệu lưu trữ và truyền tải.
2.	Đăng nhập để thực hiện các chức năng ứng với từng vai trò của người dùng
3.	Xác thực người dùng bằng tài khoản và mật khẩu
4.	Dữ liệu hệ thống được sao lưu hằng ngày và được lưu an toàn bên ngoài trang web

#### **2.2.4 Trình duyệt (Browser)**

*Bảng 4 Trình duyệt*

<b>No.</b>	<b>Requirement</b>
1.	IE6 and above (IE8 is not recommended).
2.	Chromium browser and Firefox

#### **2.2.5 Giao diện (Interfaces)**

*Bảng 5 Giao diện*

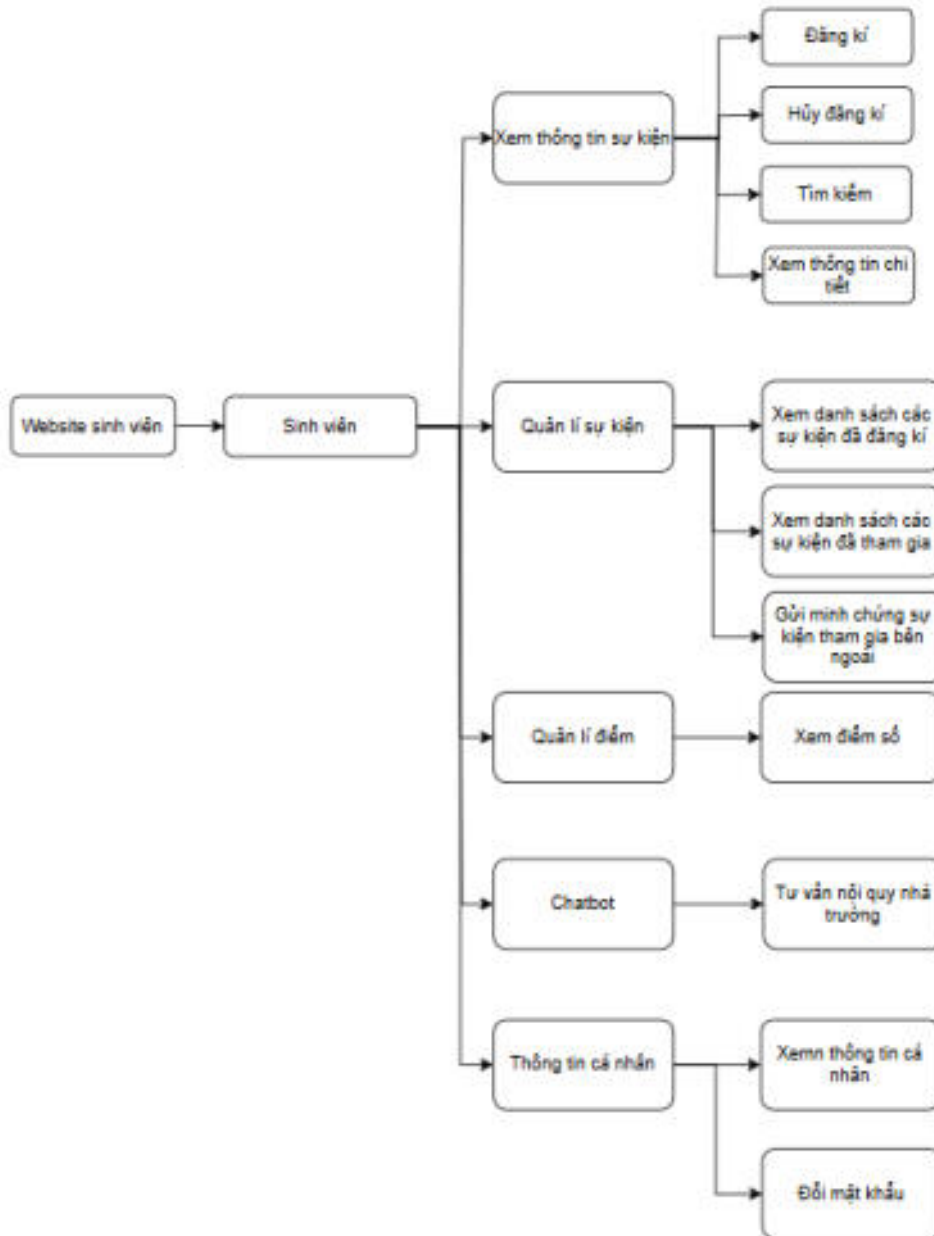
<b>No.</b>	<b>Requirement</b>
1.	Giao diện dễ dùng, tối ưu trải nghiệm người dung
2.	Không bị giật, chậm, không bị chớp nháy
3.	Màu sắc hài hòa, không gây đau mắt

#### **2.2.6 Giả định (Assumptions)**

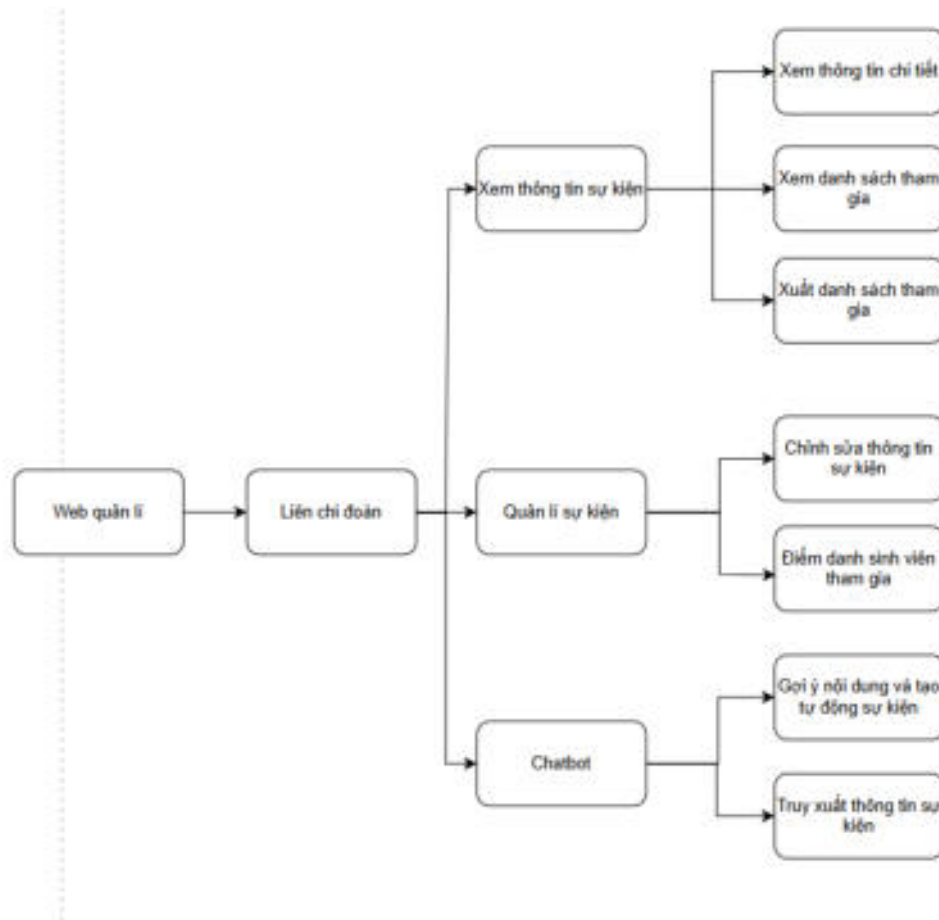
*Bảng 6 Giả định*

<b>No.</b>	<b>Requirement</b>
1.	Có thể tạm ngưng hệ thống nếu cần phải nâng cấp

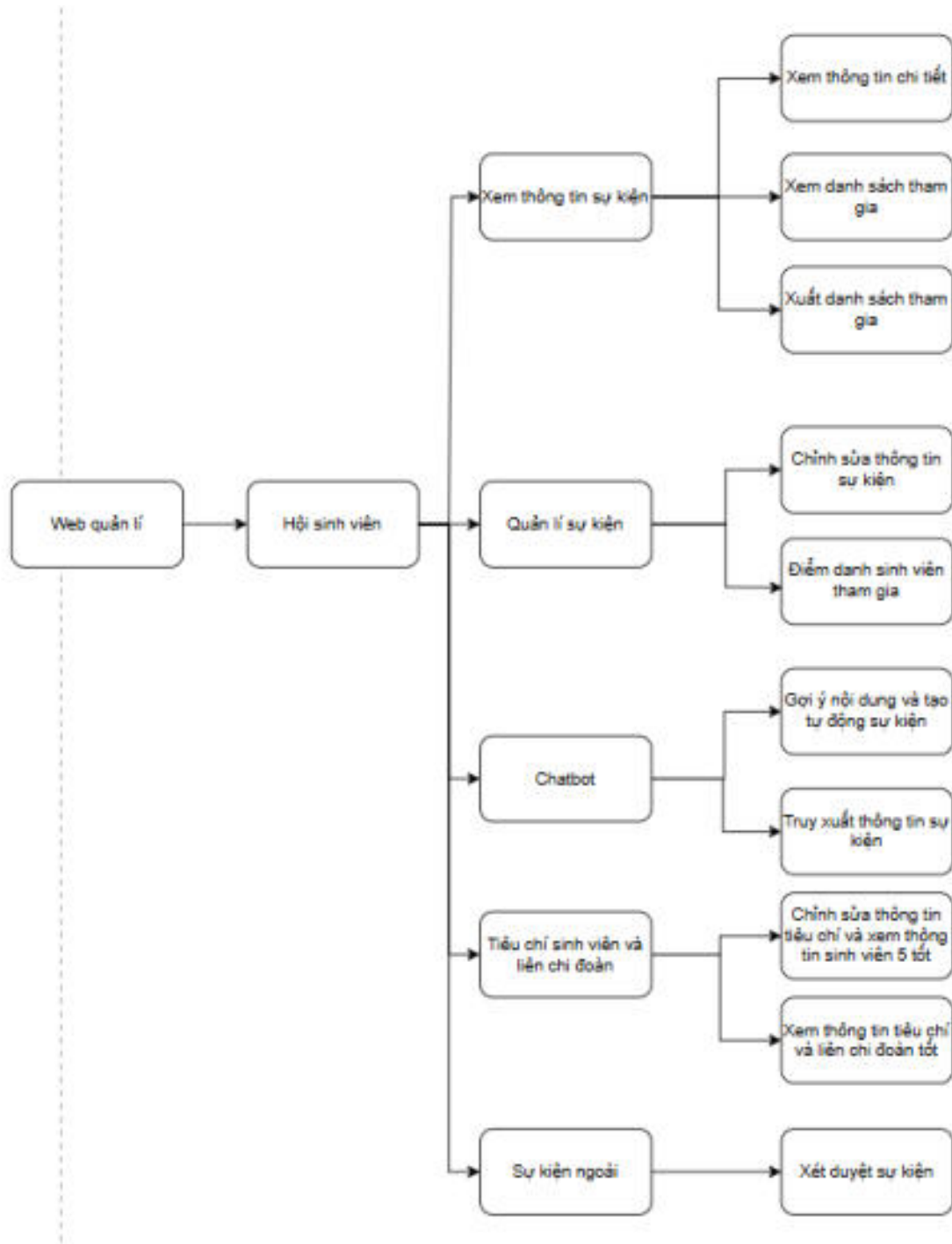
### 2.3 Biểu đồ phân rã chức năng



*Hình 6 Biểu đồ phân rã chức năng cho sinh viên*



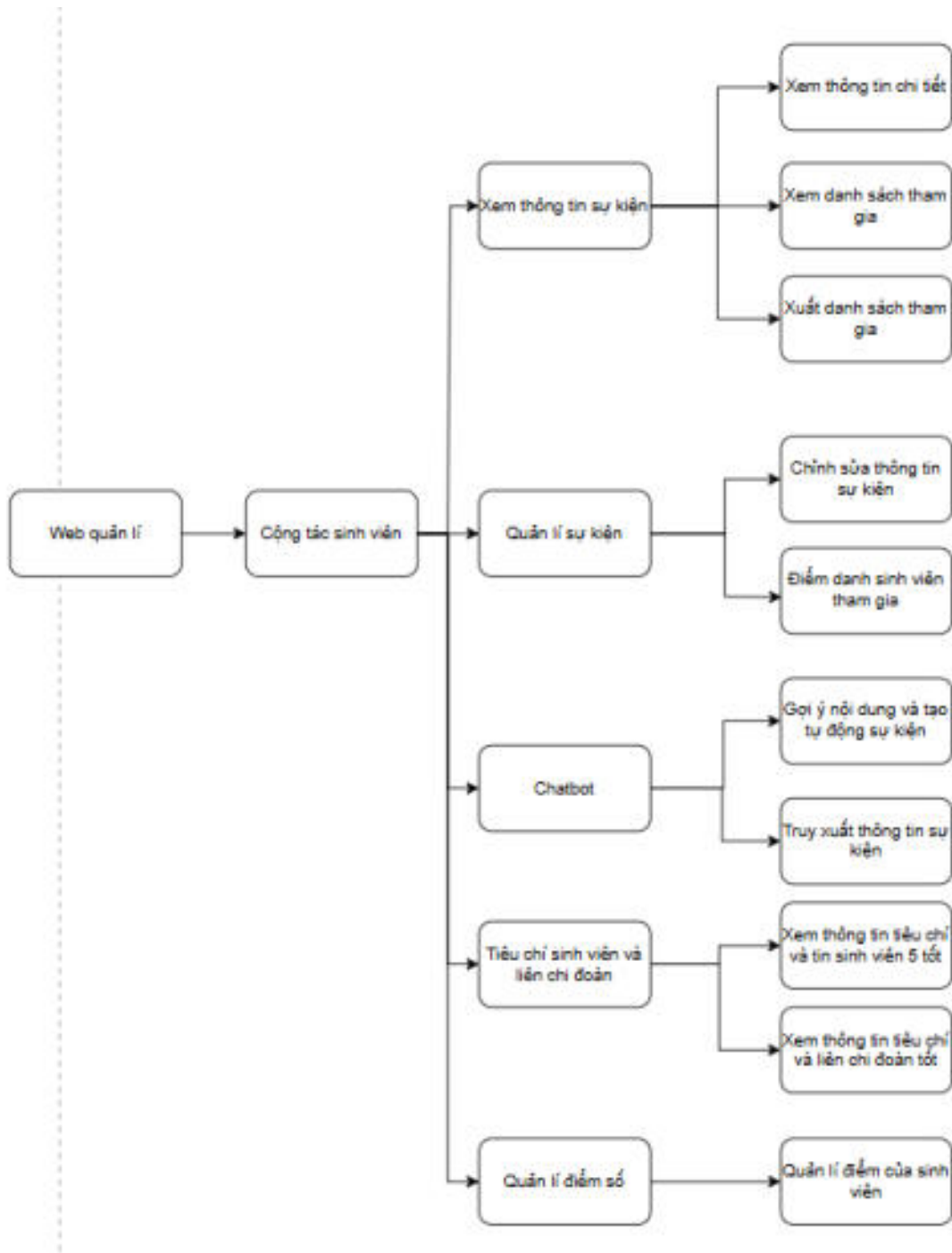
Hình 7 Biểu đồ phân rã chức năng cho liên chi đoàn



Hình 8 Biểu đồ phân rã chức năng cho hội sinh viên



Hình 9 Biểu đồ phân rã chức năng cho ban thời vụ



*Hình 10 Biểu đồ phân rã chức năng cho cộng tác sinh viên*

## **2.4 Biểu đồ ca sử dụng**

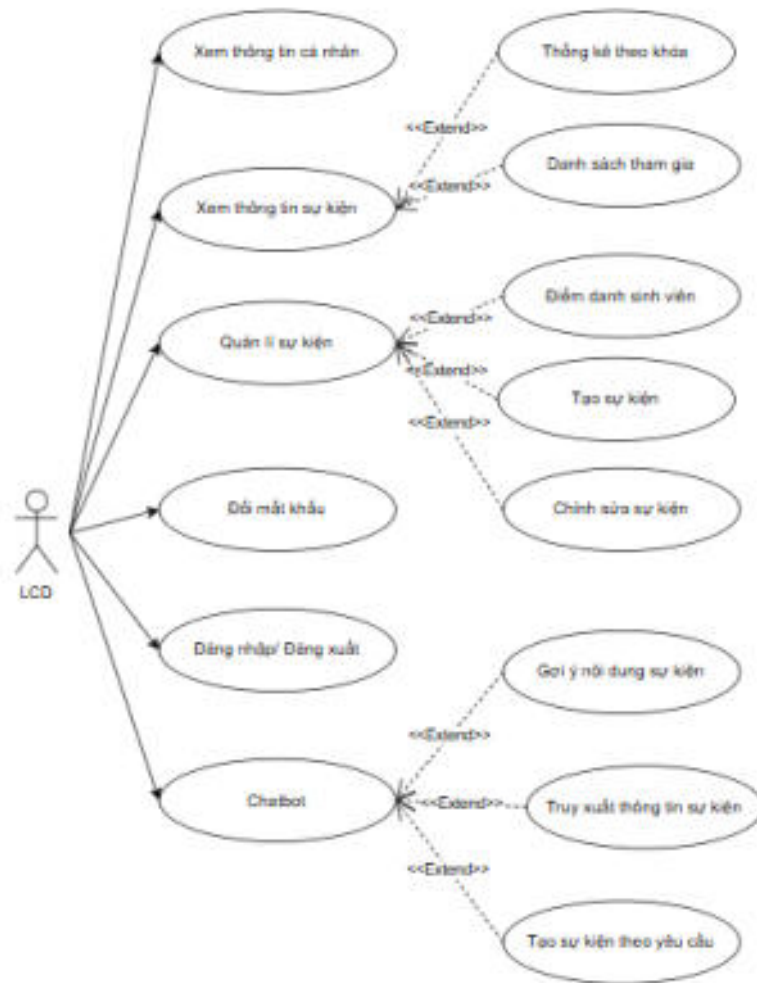
### **2.4.1 Biểu đồ ca sử dụng tổng quát cho sinh viên**

Sơ đồ sau đây cung cấp một bức tranh tổng thể về các nhóm chức năng, các Tác nhân của “*Hệ thống quản lý điểm phục vụ công đồng và sinh viên 5 tốt của Đại học Bách Khoa Đà Nẵng*”. Các chức năng của hệ thống được chia thành các ca như sau:



*Hình 11 Biểu đồ ca sử dụng tổng quát cho sinh viên*

### 2.4.2 Biểu đồ ca sử dụng tổng quát cho liên chi đoàn



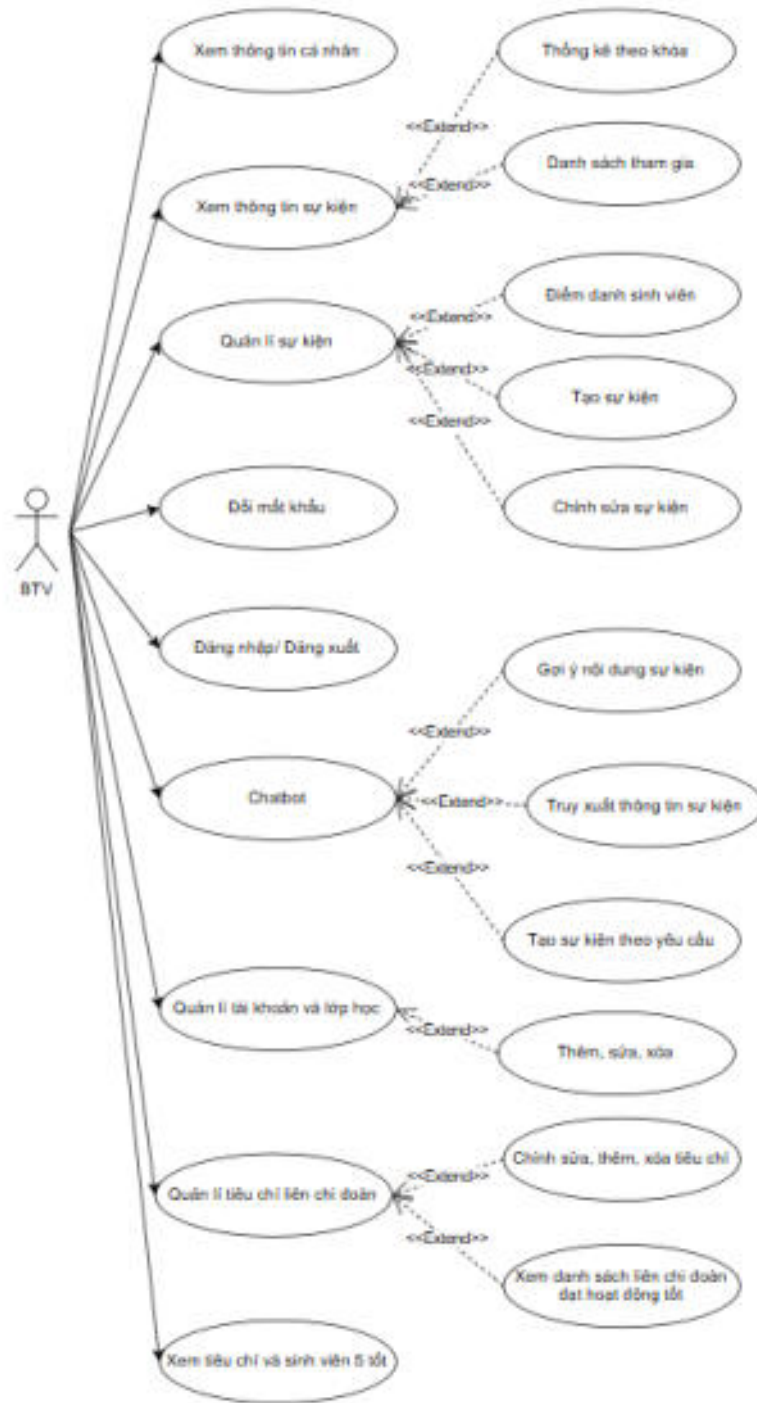
*Hình 12 Biểu đồ ca sử dụng tổng quát cho giảng viên*

### 2.4.3 Biểu đồ ca sử dụng tổng quát cho hội sinh viên



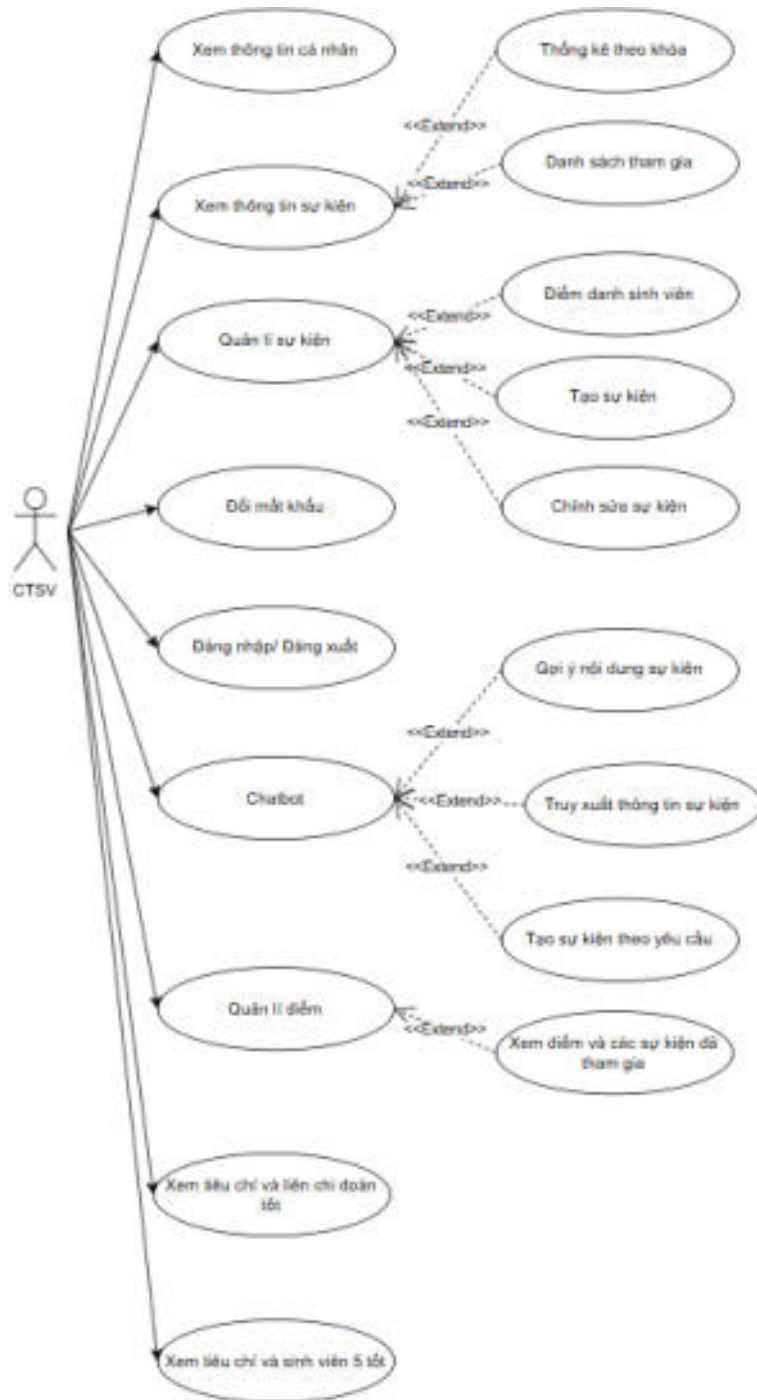
*Hình 13 Biểu đồ ca sử dụng tổng quát cho hội sinh viên*

### 2.4.4 Biểu đồ ca sử dụng tổng quát cho ban thời vụ đoàn trường



Hình 14 Biểu đồ ca sử dụng tổng quát cho ban thời vụ

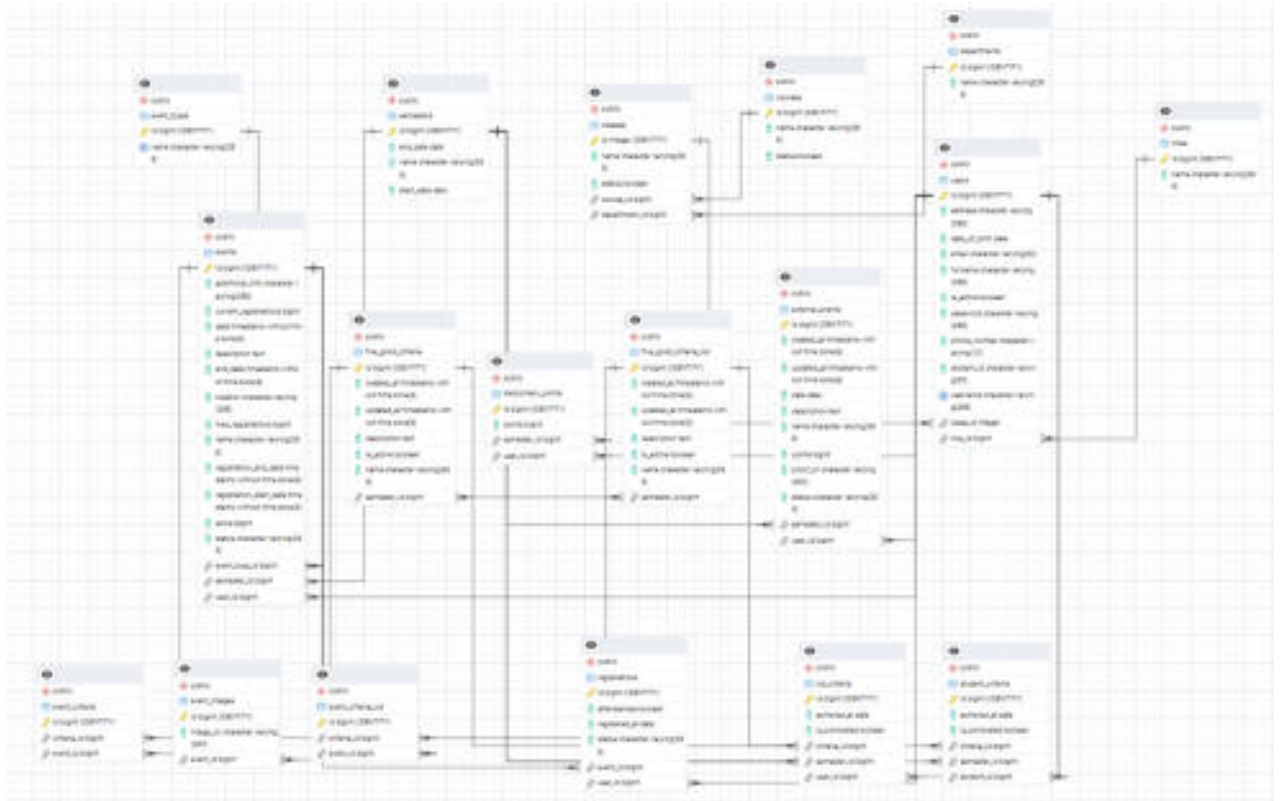
### 2.4.5 Biểu đồ ca sử dụng tổng quát cho cộng tác sinh viên



*Hình 15 Biểu đồ ca sử dụng tổng quát cho cộng tác sinh viên*

## 2.5 Lược đồ cơ sở dữ liệu

Nhóm sử dụng PostgreSQL làm cơ sở dữ liệu để lưu trữ các thông tin cần thiết nhằm phục vụ các tác vụ truy xuất, tìm kiếm dữ liệu. Hình bên dưới là thiết kế cơ sở dữ liệu của nhóm.



*Hình 16 Lược đồ cơ sở dữ liệu*

### **Tổng quát các bảng trong lược đồ cơ sở dữ liệu**

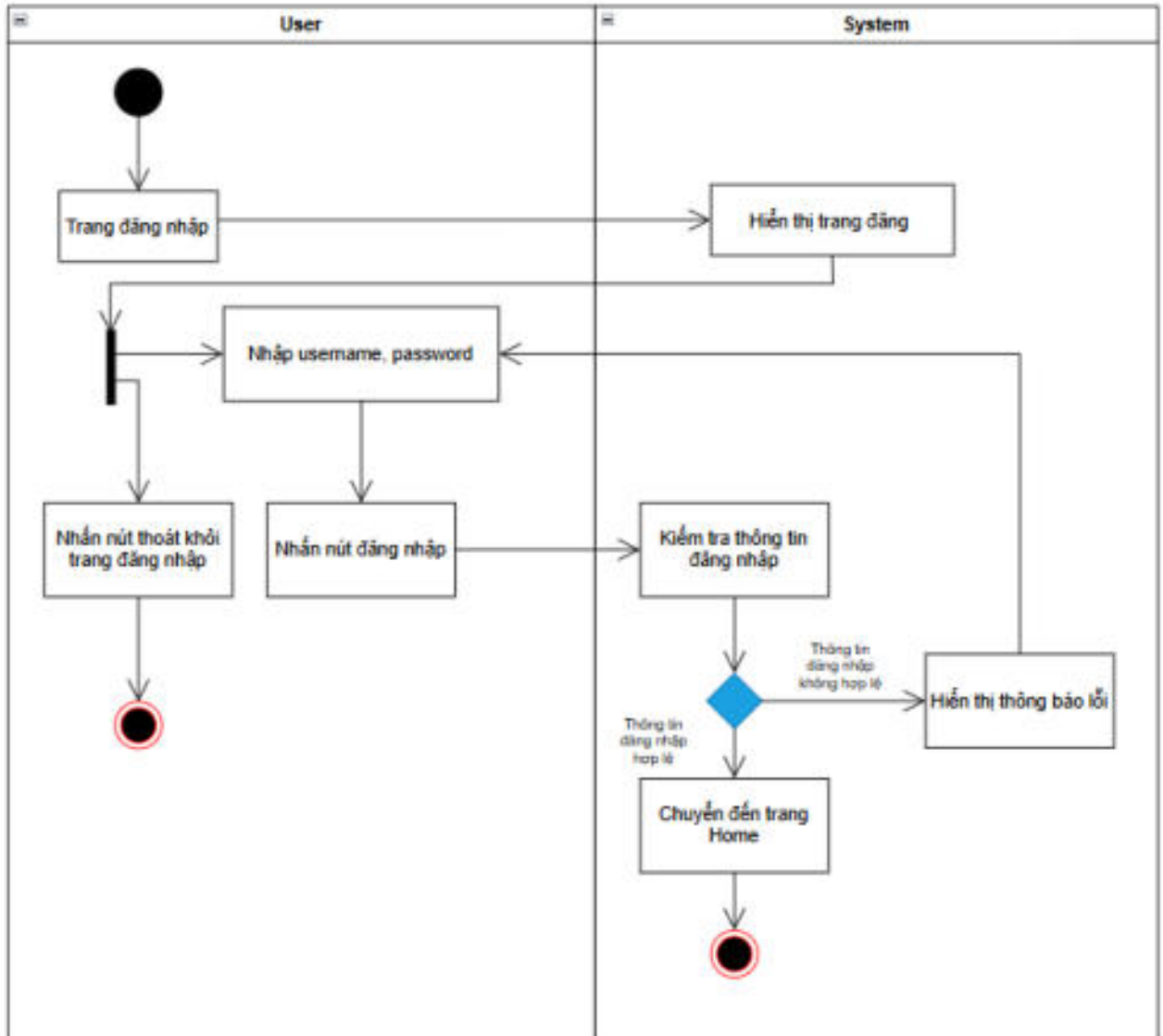
*Bảng 7 Tổng quát các bảng trong lược đồ cơ sở dữ liệu*

STT	Tên bảng	Mô tả
1	Users	Thông tin tài khoản của người dùng
2	Roles	Quyền người dùng
3	Events	Lưu trữ thông tin sự kiện
4	Event_types	Lưu trữ thông tin loại sự kiện
5	Event_images	Lưu trữ ảnh sự kiện
6	Event_criteria	Lưu trữ các tiêu chí sinh viên 5 tốt của sự kiện
7	Event_criteria_lcd	Lưu trữ các tiêu chí hoạt động tốt liên chi đoàn của sự kiện

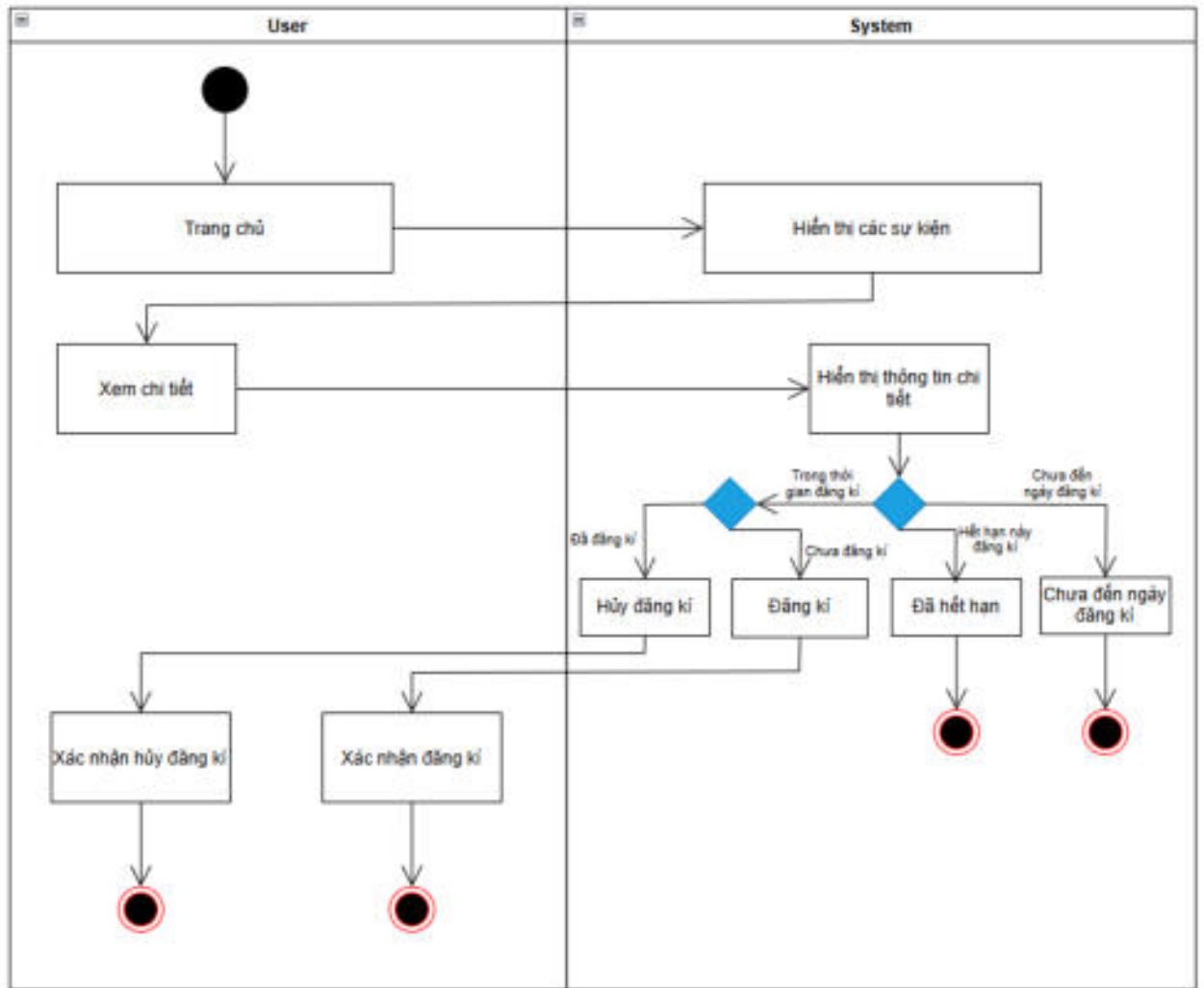
*Tên đề tài: Hệ thống quản lý điểm phục vụ công đồng và sinh viên 5 tốt của Đại Học*

8	Courses	Lưu trữ thông tin các khóa học
9	Classes	Lưu trữ thông tin các lớp học
10	Department	Lưu trữ thông tin khoa
11	Disciplinary_points	Lưu trữ điểm sinh viên
12	Semester	Lưu trữ thông tin năm học
13	External_events	Lưu trữ thông tin về sự kiện bên ngoài trường học
14	Five_good_criteria	Lưu trữ các tiêu chí sinh viên 5 tốt
15	Five_good_criteria_lcd	Lưu trữ các tiêu chí hoạt động tốt của liên chi đoàn
16	Lcd_criteria	Lưu trữ các tiêu chí liên chi đoàn đạt được
17	Student_criteria	Lưu trữ các tiêu chí sinh viên đạt được
18	Registrations	Lưu trữ thông tin các sinh viên đăng ký sự kiện

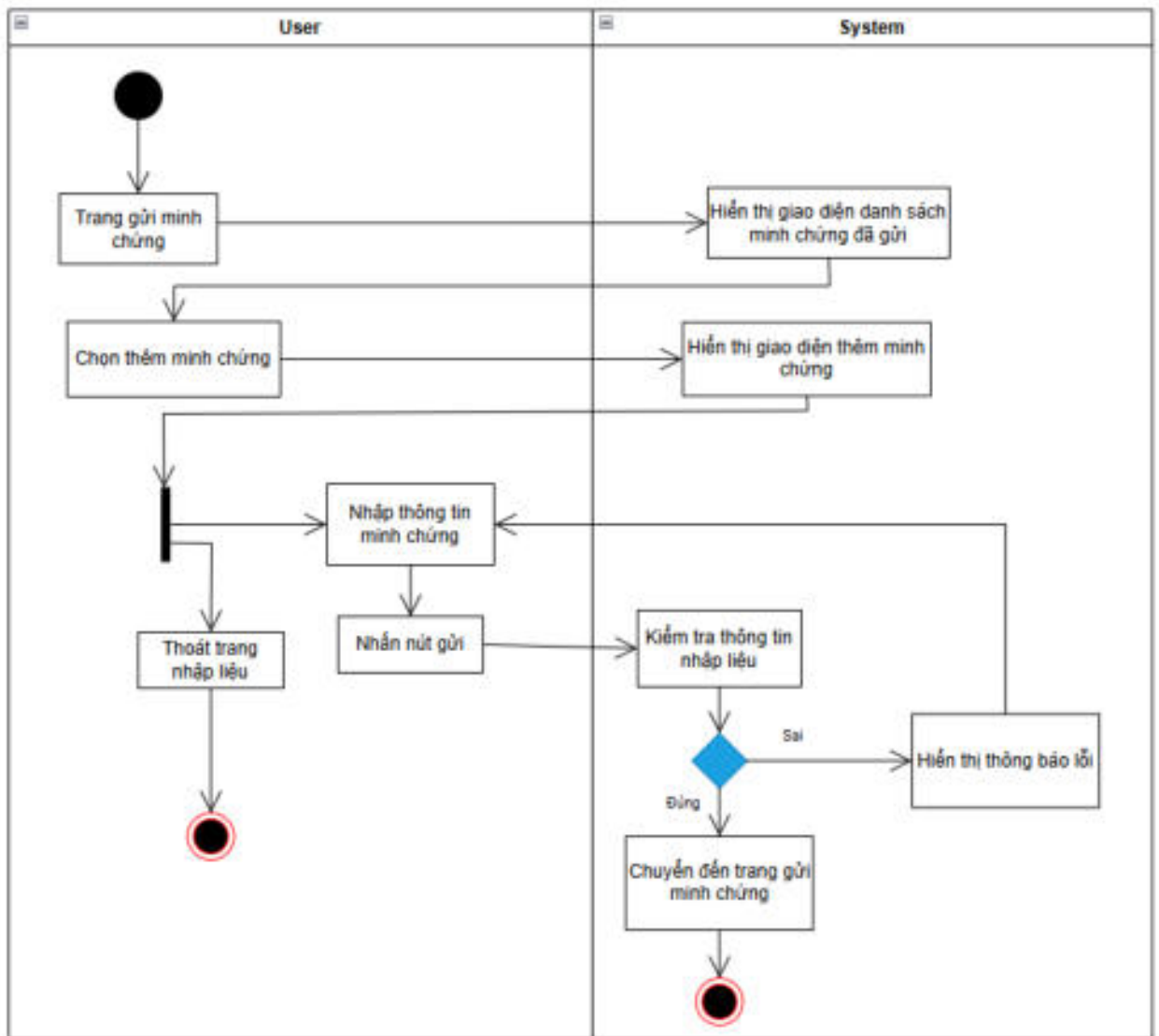
## 2.6 Sơ đồ hoạt động



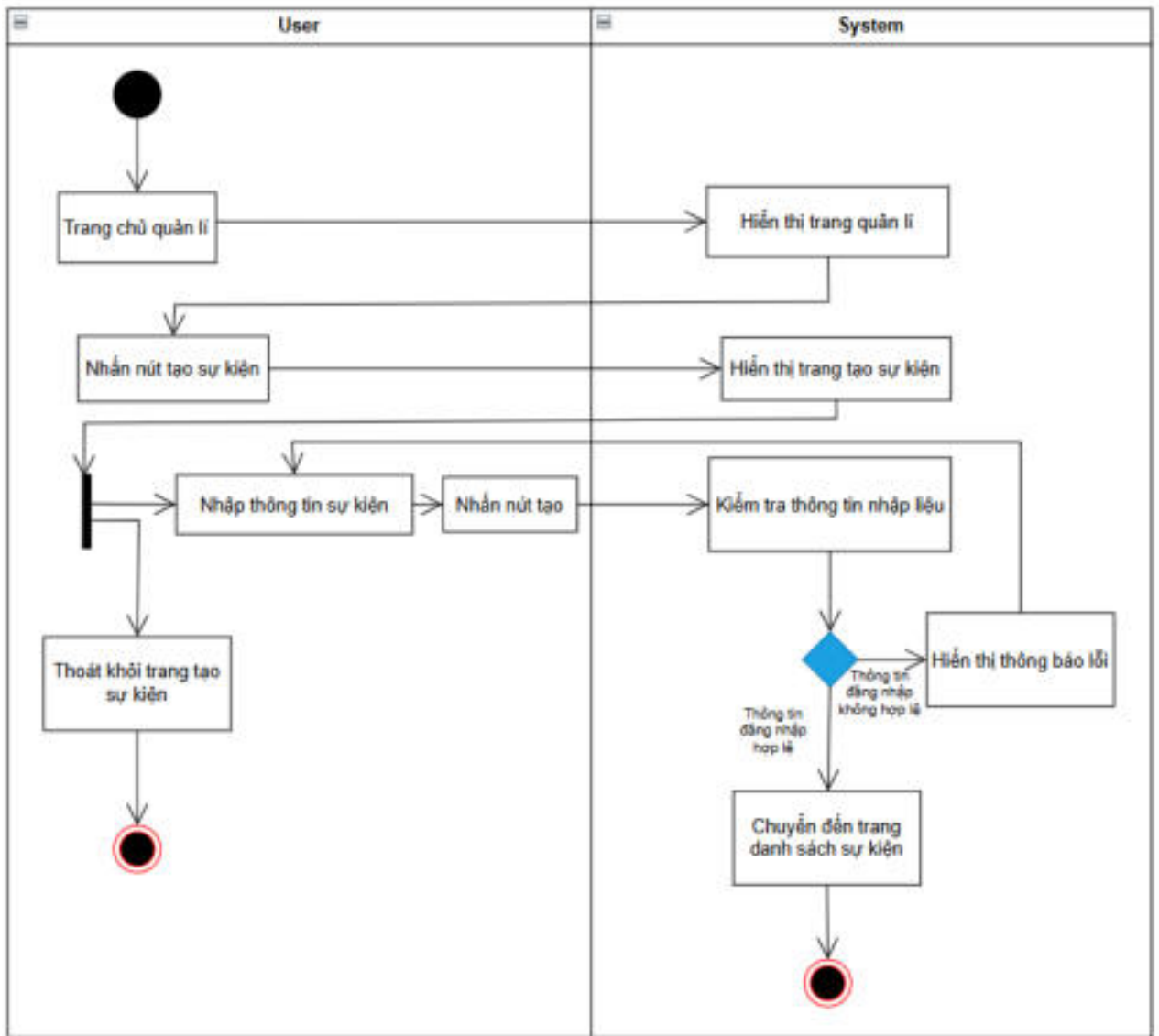
Hình 17 Sơ đồ Activity đăng nhập



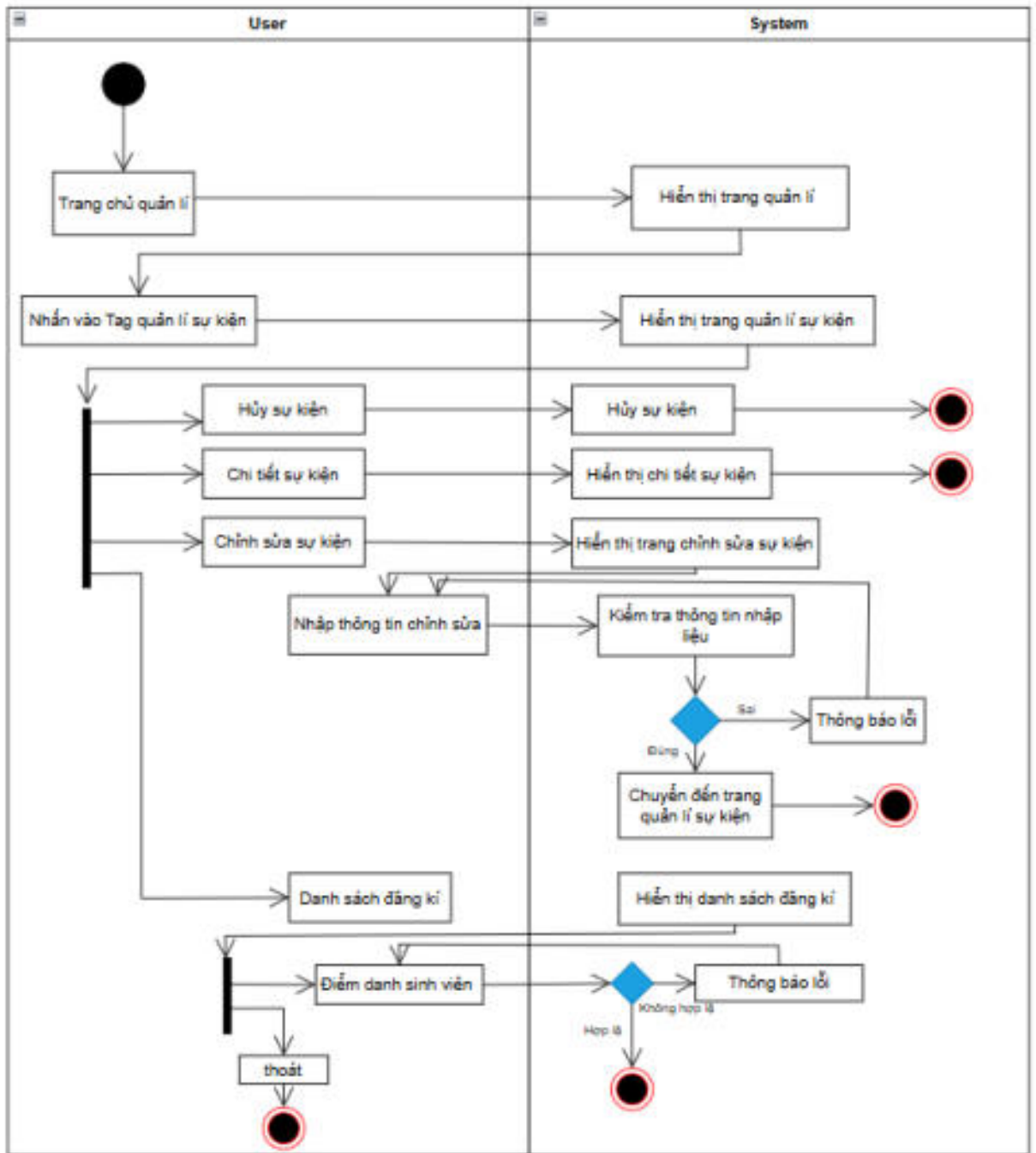
Hình 18 Sơ đồ hoạt động - Đăng ký sự kiện



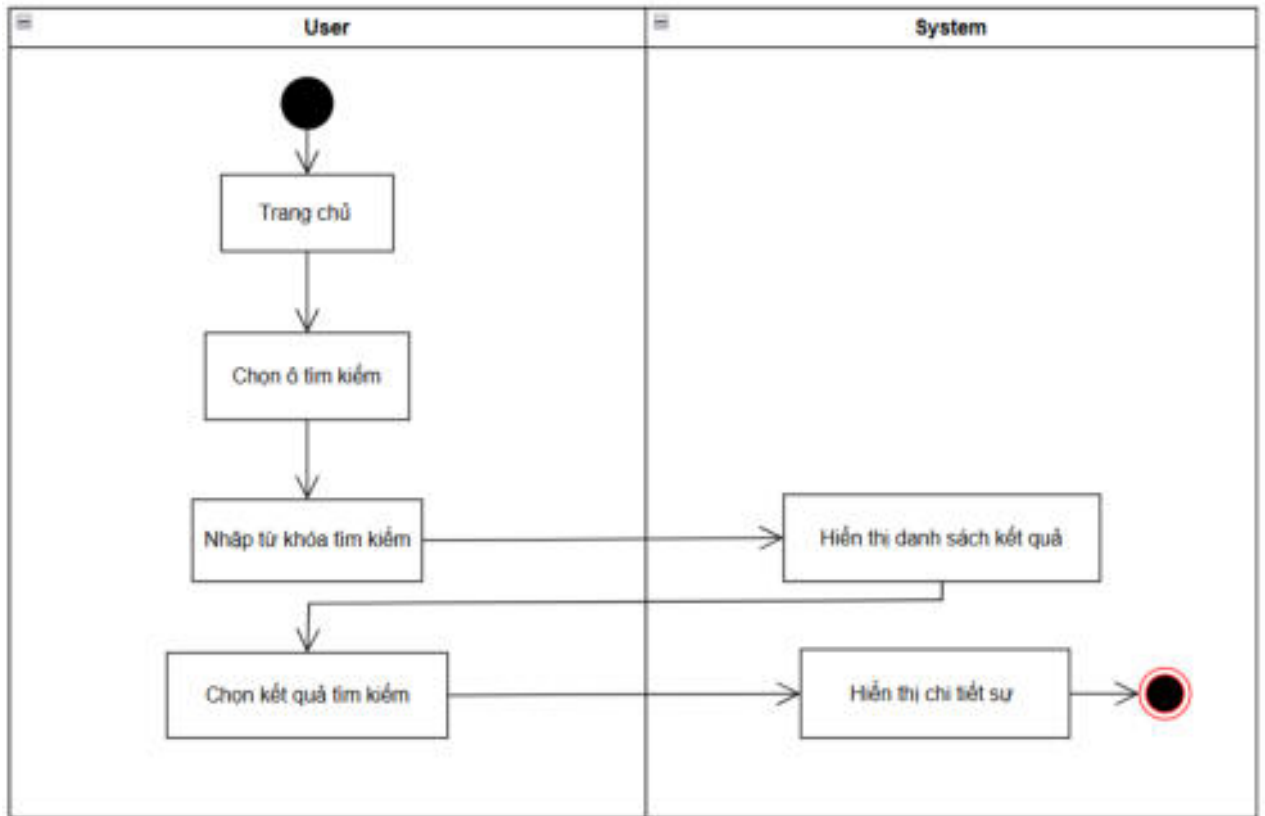
Hình 19 Sơ đồ hoạt động - Gửi minh chứng



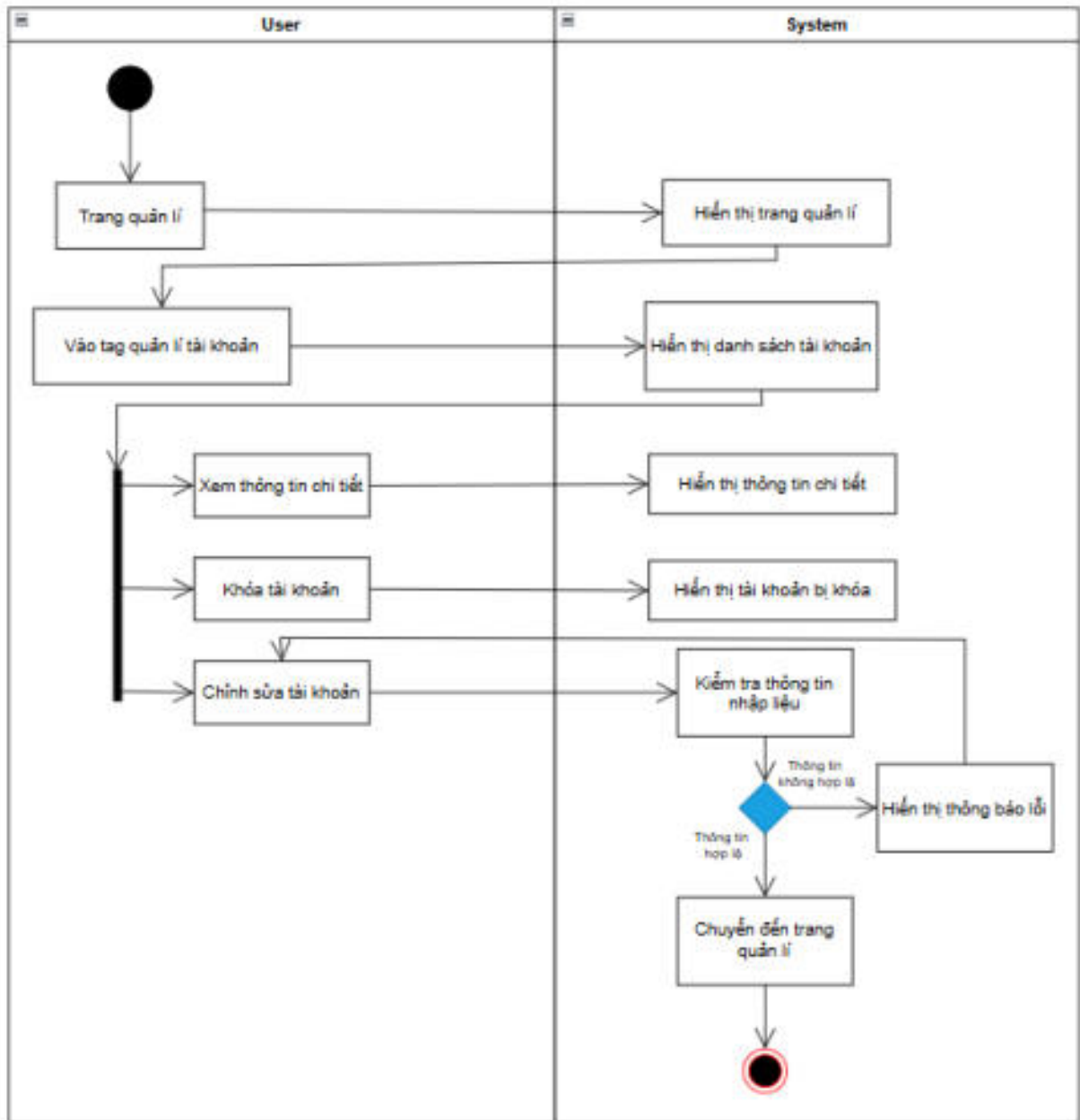
Hình 20 Sơ đồ hoạt động - tạo sự kiện



Hình 21 Sơ đồ hoạt động - quản lý sự kiện

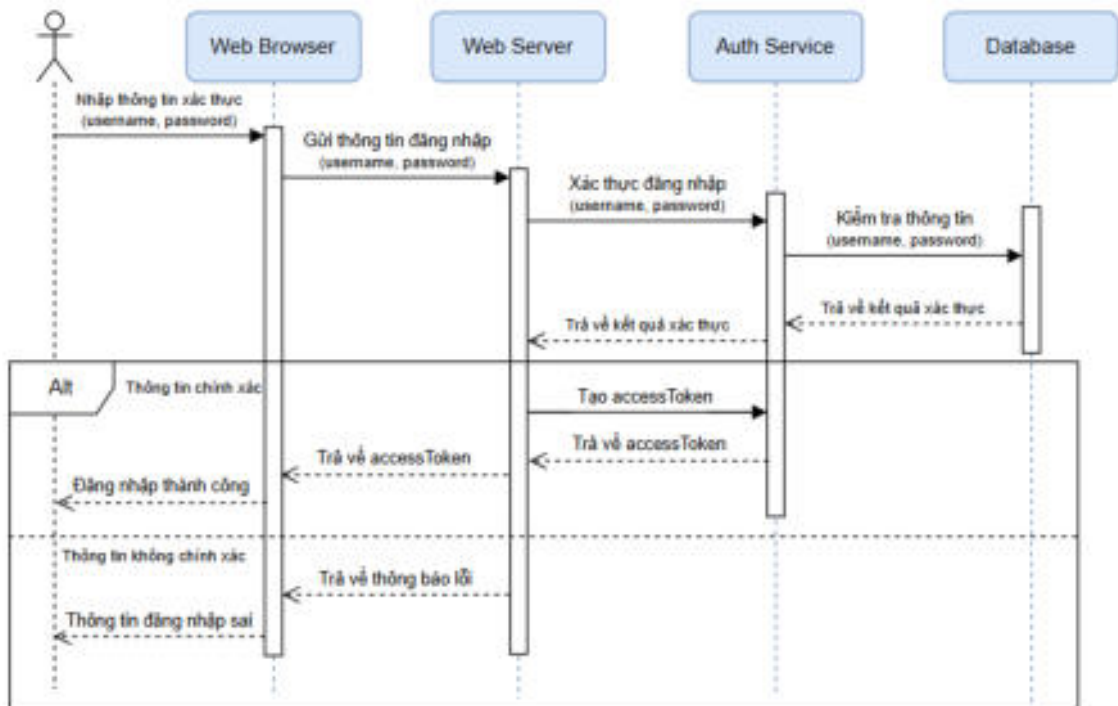


*Hình 22 Sơ đồ hoạt động - tìm kiếm sự kiện*

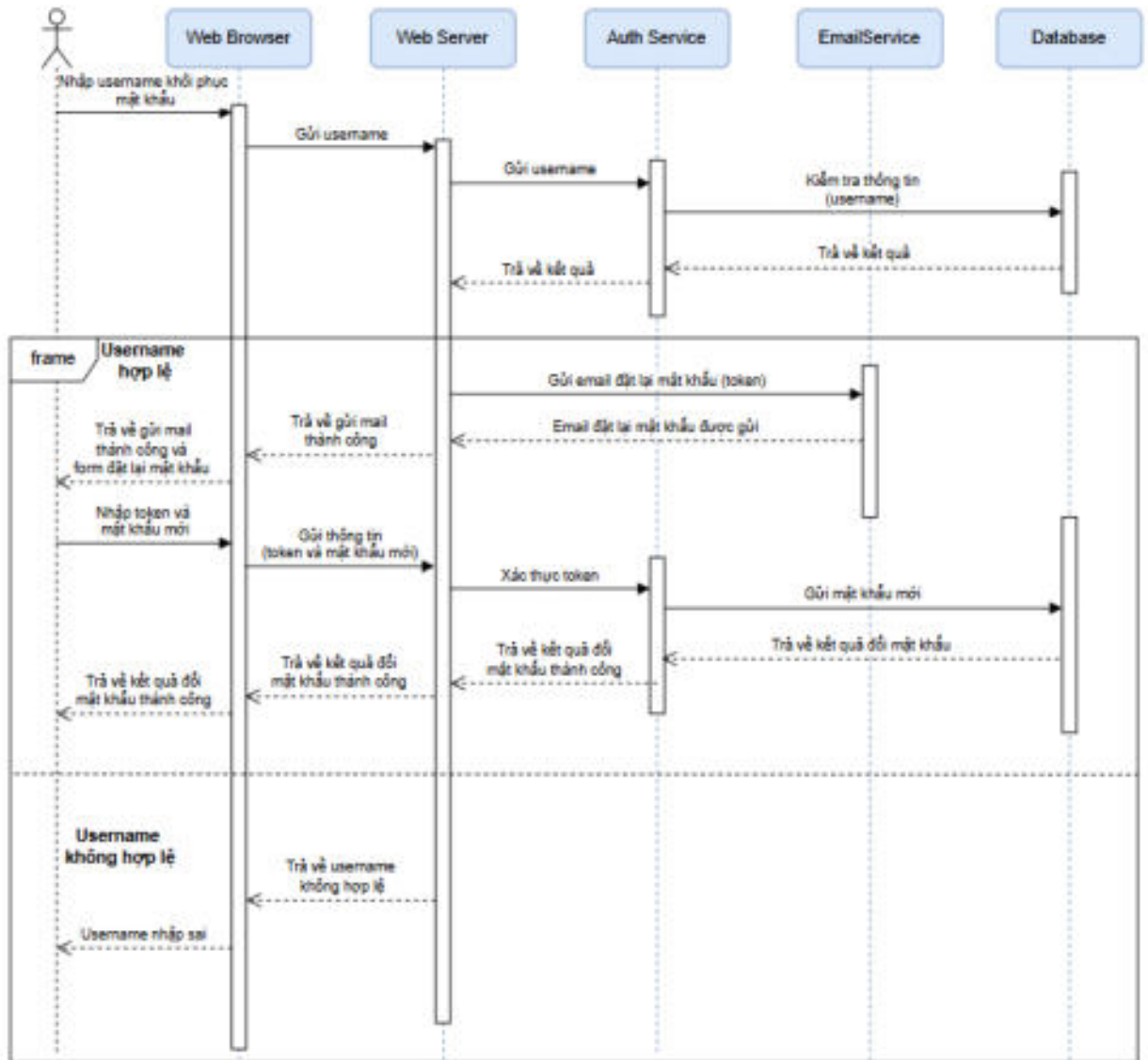


Hình 23 Sơ đồ hoạt động - quản lý tài khoản

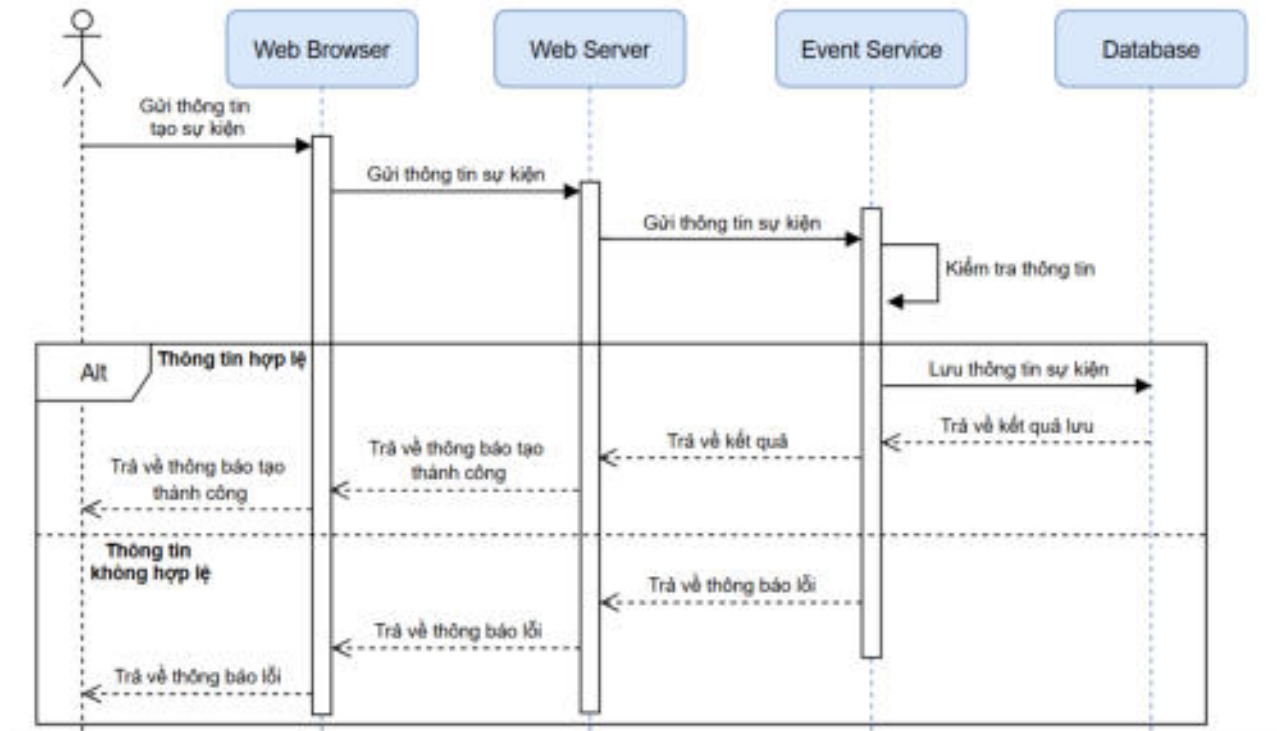
## 2.7 Sơ đồ tuần tự



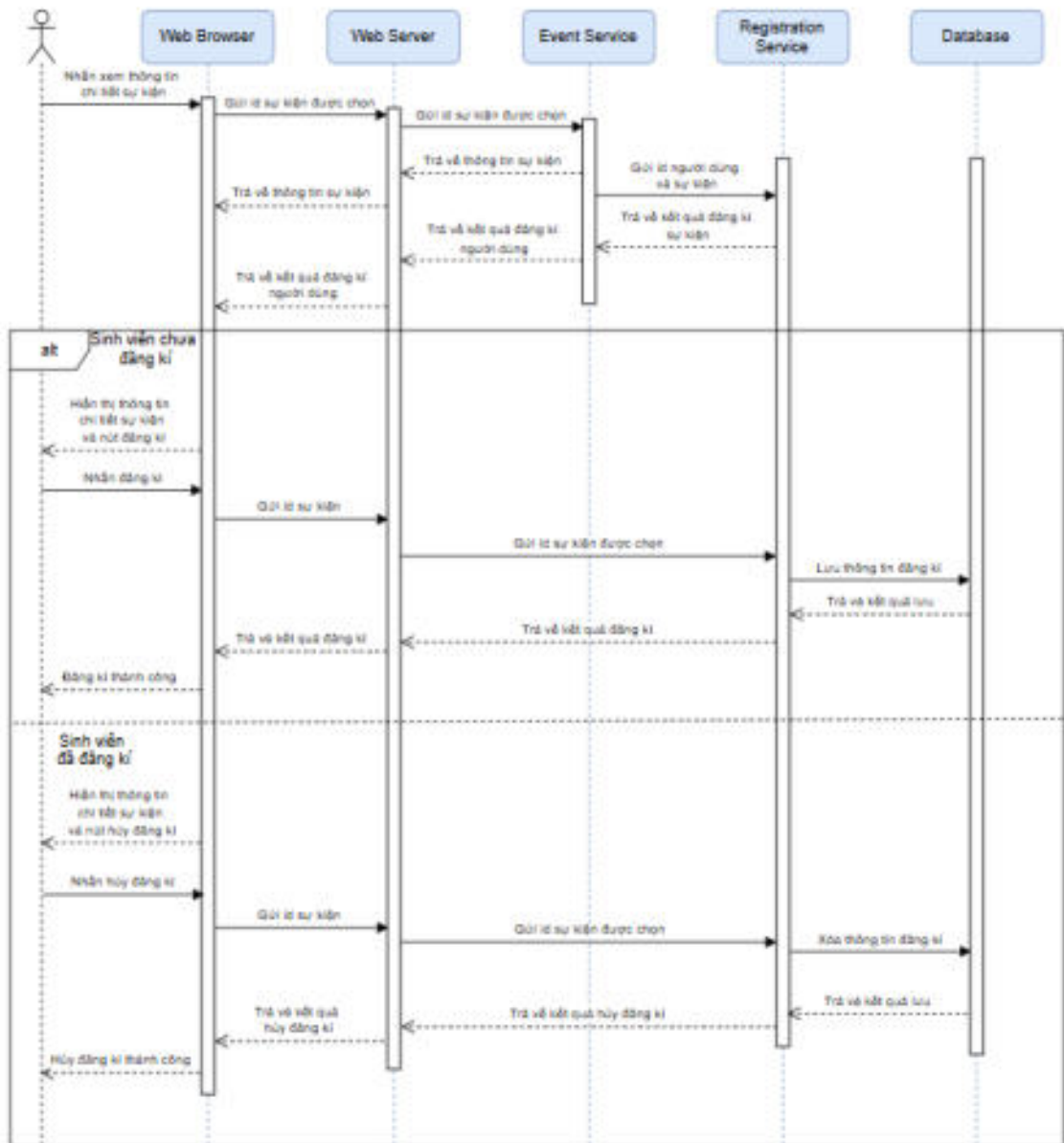
*Hình 24 Sơ đồ tuần tự đăng nhập*



Hình 25 Sơ đồ tuần tự đặt lại mật khẩu



Hình 26 Sơ đồ tuần tự tạo sự kiện



Hình 27 Sơ đồ tuần tự đăng ký sự kiện

## 2.8 Kết luận chương 2

Chương 2 đi sâu vào việc phân tích và thiết kế hệ thống dựa trên nền tảng lý thuyết đã được trình bày trong chương trước. Các yêu cầu chức năng và phi chức năng của hệ thống đã được xác định rõ ràng. Các sơ đồ usecase, lược đồ cơ sở dữ liệu, sơ đồ hoạt động, và sơ đồ tuần tự cung cấp một cái nhìn trực quan và chi tiết về cách hệ thống sẽ hoạt động. Quá trình này đảm bảo rằng mọi khía cạnh của hệ thống đều được xem xét kỹ lưỡng, tạo điều kiện thuận lợi cho giai đoạn triển khai sau này.

## **CHƯƠNG 3: TRIỂN KHAI HỆ THỐNG**

### **3.1 Xây dựng chatbot hỏi đáp nội quy nhà trường**

Trong bối cảnh số hóa giáo dục ngày càng mạnh mẽ, việc cung cấp thông tin kịp thời và chính xác về các quy định, chính sách và nội quy trong môi trường đại học đóng vai trò quan trọng trong việc hỗ trợ sinh viên tuân thủ đúng quy định và hòa nhập tốt với đời sống học đường. Để đáp ứng nhu cầu này, chatbot tư vấn nội quy đã nổi lên như một giải pháp hiệu quả. Với sự phát triển vượt bậc của công nghệ và các mô hình ngôn ngữ lớn (Large Language Models - LLM), chúng ta có thể xây dựng các chatbot thông minh, có khả năng hiểu và giải đáp thắc mắc của sinh viên một cách tự nhiên, rõ ràng và chính xác. Đặc biệt trong dự án lần này, em xây dựng một chatbot chuyên biệt dành riêng cho tư vấn nội quy của từng trường đại học, cho phép chatbot trả lời dựa trên các quy định cụ thể do nhà trường ban hành. Qua đó, sinh viên có thể dễ dàng tiếp cận thông tin, nhanh chóng giải quyết thắc mắc, và chủ động hơn trong việc tuân thủ các quy định học vụ, đạo đức, và sinh hoạt trong môi trường đại học.

#### **3.1.1 Lý do lựa chọn n8n và mô hình ngôn ngữ lớn (LLM) để xây dựng chatbot**

Một trong những lý do chính khiến em lựa chọn sử dụng n8n kết hợp với các mô hình ngôn ngữ lớn (Large Language Models - LLM) thay vì các mô hình truyền thống là nhờ khả năng tích hợp linh hoạt, tự động hóa quy trình và xử lý thông tin một cách hiệu quả.

Khác với các mô hình truyền thống – thường yêu cầu huấn luyện lại từ đầu mỗi khi cập nhật dữ liệu mới – n8n cho phép chúng tôi dễ dàng kết nối chatbot với nhiều nguồn dữ liệu như Google Sheets, Notion, cơ sở dữ liệu SQL, API nội bộ của trường, v.v... Điều này giúp hệ thống có thể cập nhật thông tin gần như thời gian thực mà không cần can thiệp sâu vào mô hình.

Khi kết hợp với các mô hình ngôn ngữ lớn như GPT-3.5 hoặc GPT-4, hệ thống chatbot có thể hiểu và trả lời các câu hỏi một cách tự nhiên, chính xác và có ngữ cảnh. LLM có khả năng xử lý ngôn ngữ tự nhiên rất tốt mà không cần phải huấn luyện lại mô hình mỗi khi có thay đổi về nội dung.

Nhờ vào n8n, em có thể xây dựng các quy trình tự động như: chuyên đổi văn bản thành định dạng phù hợp cho chatbot, gửi phản hồi đến người dùng qua giao diện trò chuyện, và thậm chí ghi lại lịch sử tương tác để cải thiện hiệu suất sau này.

Giải pháp này không những giảm thiểu chi phí vận hành, mà còn giúp mở rộng quy mô hệ thống chatbot một cách linh hoạt, đảm bảo rằng sinh viên luôn nhận được thông tin mới nhất, chính xác và phù hợp với nhu cầu thực tế.

### **3.1.2 Các bước xây dựng chatbot hỏi đáp**

#### **Bước 1: Xác định Mục tiêu và Yêu cầu của Chatbot**

- **Mục tiêu:** Xây dựng một chatbot hỏi đáp thông minh tư vấn nội quy, giúp sinh viên nhận được câu trả lời nhanh chóng và chính xác cho các câu hỏi về nội quy nhà trường.
- **Yêu cầu chức năng:** Chatbot phải có khả năng xử lý ngôn ngữ tự nhiên, hiểu ngữ cảnh, và cung cấp các câu trả lời phù hợp.
- **Yêu cầu phi chức năng:** Độ chính xác cao, thời gian phản hồi ngắn, khả năng mở rộng, tính bảo mật cao, và dễ dàng sử dụng.

#### **Bước 2: Thu thập**

- **Thu thập dữ liệu:** Giảng viên sẽ upload tài liệu học tập, câu hỏi thường gặp, đánh giá khóa học, và các thông tin liên quan khác lên google drive.

#### **Bước 3: Lựa chọn mô hình ngôn ngữ**

- **Lựa chọn mô hình:** Sử dụng GPT-4o-mini cho khả năng xử lý ngôn ngữ tự nhiên mạnh mẽ.

#### **Bước 4: Xây dựng Hệ thống Lưu trữ Dữ liệu với Pinecone**

- **Lựa chọn Vector Database:** Sử dụng Pinecone để lưu trữ và quản lý các vector biểu diễn dữ liệu khóa học.
- **Chuyển đổi dữ liệu thành vector:**
  - Embeddings các dữ liệu của khóa học thành các vector sau khi đã chia nhỏ.
  - Lưu trữ các vector này trong Pinecone để dễ dàng tìm kiếm và truy vấn.
- **Tích hợp Pinecone vào hệ thống:** Kết nối Pinecone với mô hình text-embedding-3-small để thực hiện các truy vấn vector và tìm kiếm thông tin liên quan trong cơ sở dữ liệu.

#### **Bước 5: Xây dựng prompt**

Mục tiêu của prompt là cung cấp ngữ cảnh đầy đủ để GPT-4o-mini có thể hiểu và trả lời đúng câu hỏi liên quan đến nội quy nhà trường

### 3.1.3 Kết quả

Sinh viên có thể hỏi các quy định của nhà trường bằng chatbot mà không cần phải tra thông tin tài liệu cụ thể.



*Hình 28 Kết quả chatbot trả lời câu hỏi sinh viên*

## **3.2 Xây dựng chatbot tư vấn và tự động tạo sự kiện**

Trong bối cảnh chuyển đổi số ngày càng phát triển sâu rộng trong giáo dục, việc hỗ trợ cán bộ nhà trường trong công tác tổ chức và quản lý các sự kiện, hoạt động nội bộ trở nên vô cùng cần thiết. Để đáp ứng nhu cầu này, em phát triển một chatbot agent thông minh, không chỉ có khả năng gợi ý nội dung mà còn được cấp quyền tự động tạo và quản lý sự kiện một cách chủ động và hiệu quả.

Ứng dụng các mô hình ngôn ngữ lớn (Large Language Models - LLM), chatbot agent có thể hiểu sâu sắc yêu cầu từ cán bộ nhà trường, đề xuất các nội dung phù hợp và thực hiện việc tạo sự kiện trực tiếp trong hệ thống quản lý. Với quyền truy cập và tương tác linh hoạt, chatbot có thể tạo và truy xuất thông tin chi tiết về các sự kiện theo từng đặc thù của nhà trường.

Giải pháp này không chỉ giúp nâng cao hiệu suất làm việc của cán bộ mà còn đảm bảo quá trình tổ chức sự kiện được thực hiện nhanh chóng, chính xác và minh bạch, góp phần tạo nên môi trường giáo dục hiện đại, chuyên nghiệp và hiệu quả.

### **3.2.1 Lý do lựa chọn n8n để phát triển chatbot agent tạo sự kiện**

Em lựa chọn n8n làm nền tảng phát triển chatbot agent cho dự án này nhờ vào những ưu điểm vượt trội trong khả năng tự động hóa quy trình, tích hợp linh hoạt và mở rộng dễ dàng. Cụ thể:

**Khả năng tích hợp mạnh mẽ:** n8n hỗ trợ hơn 350 dịch vụ và ứng dụng phổ biến như Google Calendar, Notion, Airtable, MySQL, và các dịch vụ API REST. Điều này cho phép chatbot agent dễ dàng tạo hoặc truy xuất sự kiện trong hệ thống quản lý của nhà trường mà không cần xây dựng tất cả từ đầu.

**Luồng xử lý linh hoạt và trực quan:** Với giao diện kéo thả trực quan, n8n giúp dễ dàng thiết kế các quy trình tự động (workflows) từ việc nhận yêu cầu của người dùng → xử lý văn bản → truy xuất dữ liệu → tạo sự kiện. Điều này phù hợp cho các tác vụ như gợi ý nội dung sự kiện, kiểm tra thời gian trống, và xác nhận thông tin trước khi tạo.

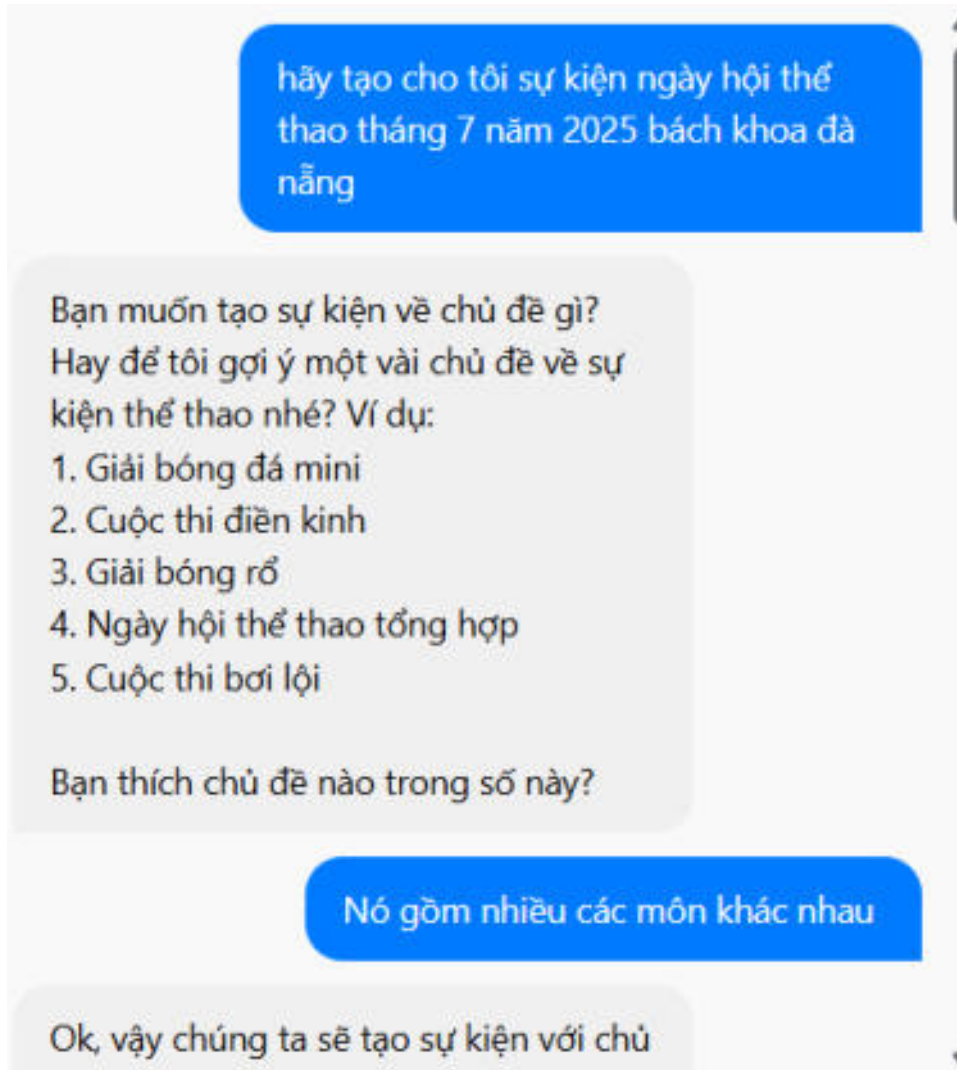
**Khả năng kết hợp với mô hình ngôn ngữ lớn (LLM):** n8n hỗ trợ tích hợp API của các mô hình LLM như GPT-4, GPT-4o, Claude, v.v., giúp chatbot agent có thể hiểu ngôn ngữ tự nhiên, sinh phản hồi thông minh, và đưa ra gợi ý nội dung phù hợp với ngữ cảnh của từng sự kiện.

**Tự động hóa không cần lập trình phức tạp:** Đối với môi trường giáo dục, nơi nguồn lực kỹ thuật có thể hạn chế, việc sử dụng n8n giúp giảm đáng kể chi phí phát triển và bảo trì, đồng thời cho phép cán bộ kỹ thuật dễ dàng chỉnh sửa hoặc thêm mới chức năng mà không cần kỹ năng lập trình chuyên sâu.

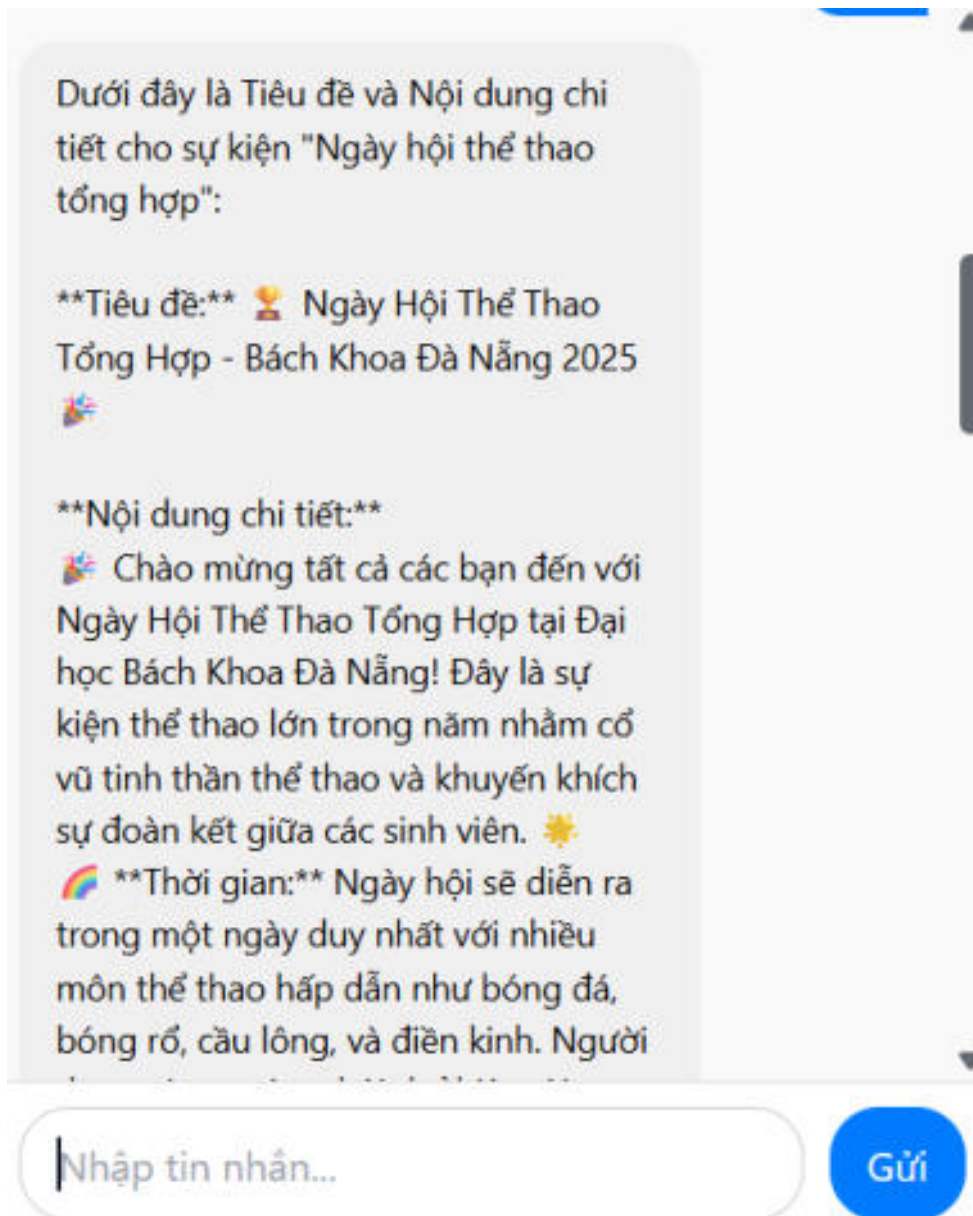
**Nguồn mở và triển khai linh hoạt:** n8n là mã nguồn mở, cho phép triển khai nội bộ trên server của nhà trường nếu cần đảm bảo bảo mật dữ liệu. Đồng thời, n8n có thể mở rộng bằng JavaScript, giúp dễ dàng cá nhân hóa logic xử lý khi cần thiết.

### **3.2.2 Kết quả**

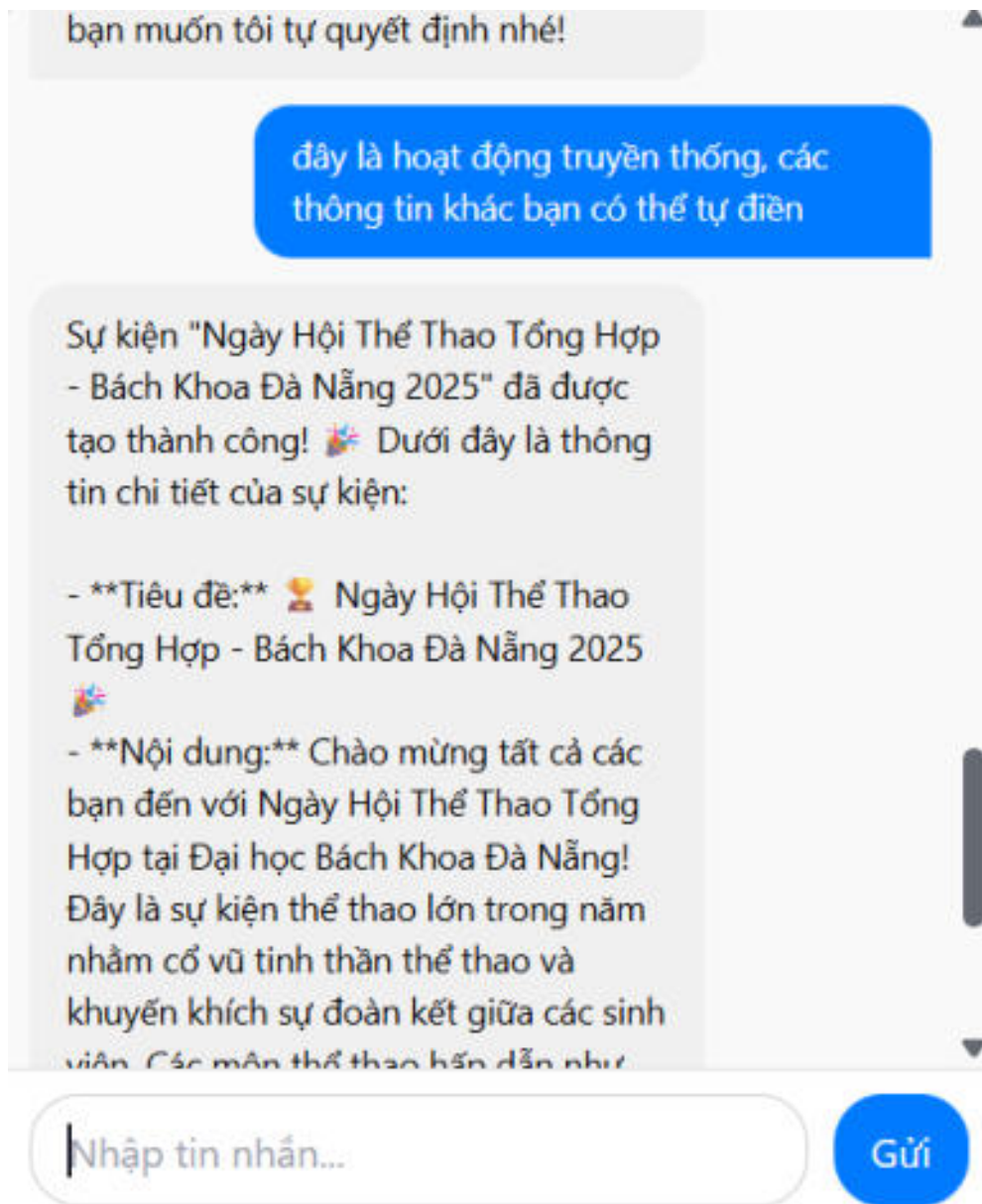
Cán bộ nhà trường có thể truy xuất thông tin của sự kiện hoặc tạo sự kiện bằng cách yêu cầu



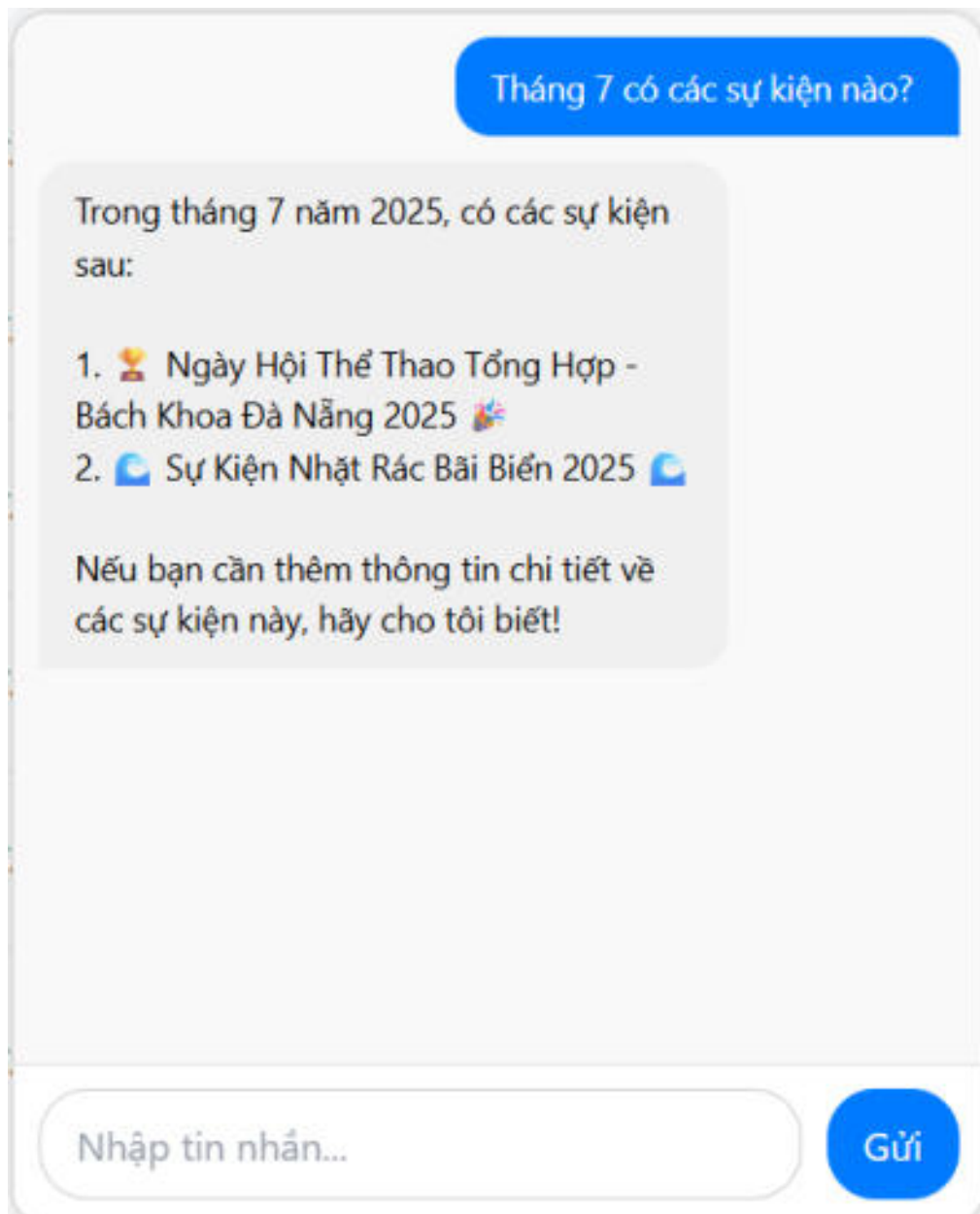
*Hình 29 Kết quả chatbot*



Hình 30 Kết quả chatbot



*Hình 31 Kết quả chatbot*

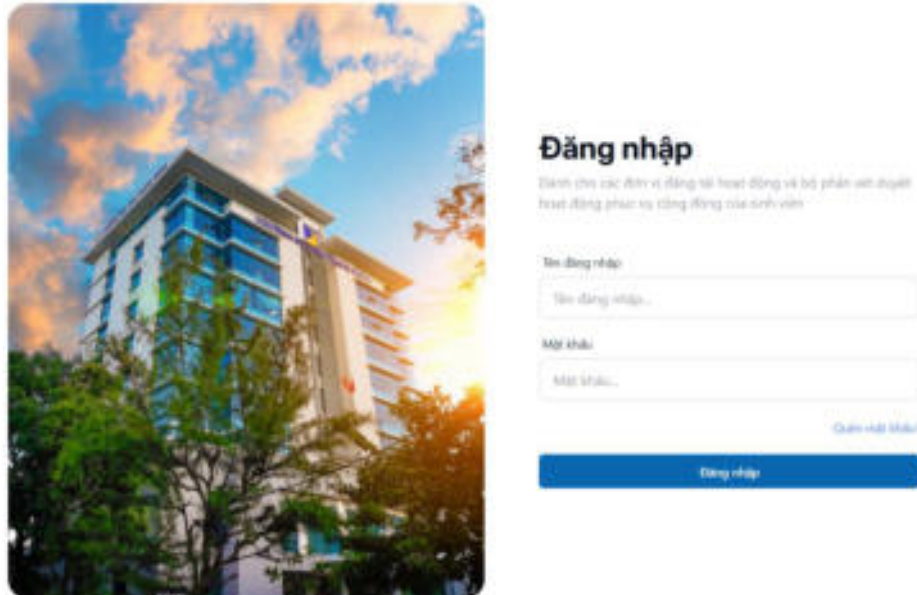


Hình 32 Kết quả chatbot

### 3.3 Một số giao diện hệ thống

#### 3.3.1 Trang đăng nhập

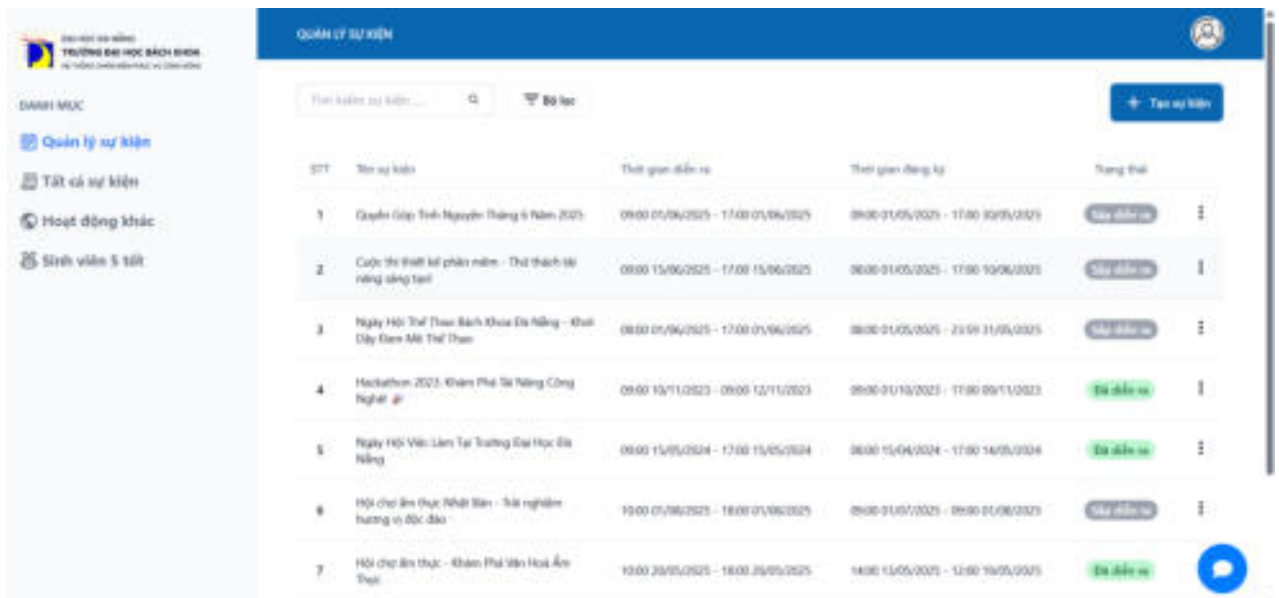
Để sử dụng hầu hết các chức năng chính trong hệ thống, người dùng cần phải đăng nhập bằng username và mật khẩu.



Hình 33 Màn hình đăng nhập

#### 3.3.2 Màn hình trang quản lý sự kiện

Trang quản lý sự kiện là nơi người dùng có thể tạo mới, xem, chỉnh sửa và quản lý các sự kiện được tổ chức trong hệ thống. Đây là trang trung tâm giúp các quản trị viên dễ dàng điều phối các hoạt động sự kiện một cách hiệu quả.



Hình 34 Màn hình trang chủ

### 3.3.3 Màn hình chi tiết sự kiện

Trang chi tiết sự kiện cung cấp đầy đủ thông tin về một sự kiện cụ thể, giúp người dùng nắm bắt toàn bộ nội dung, thời gian, địa điểm, cũng như các hoạt động liên quan đến sự kiện đó. Trang này hỗ trợ cả người quản lý và người tham gia trong việc theo dõi và tương tác với sự kiện.



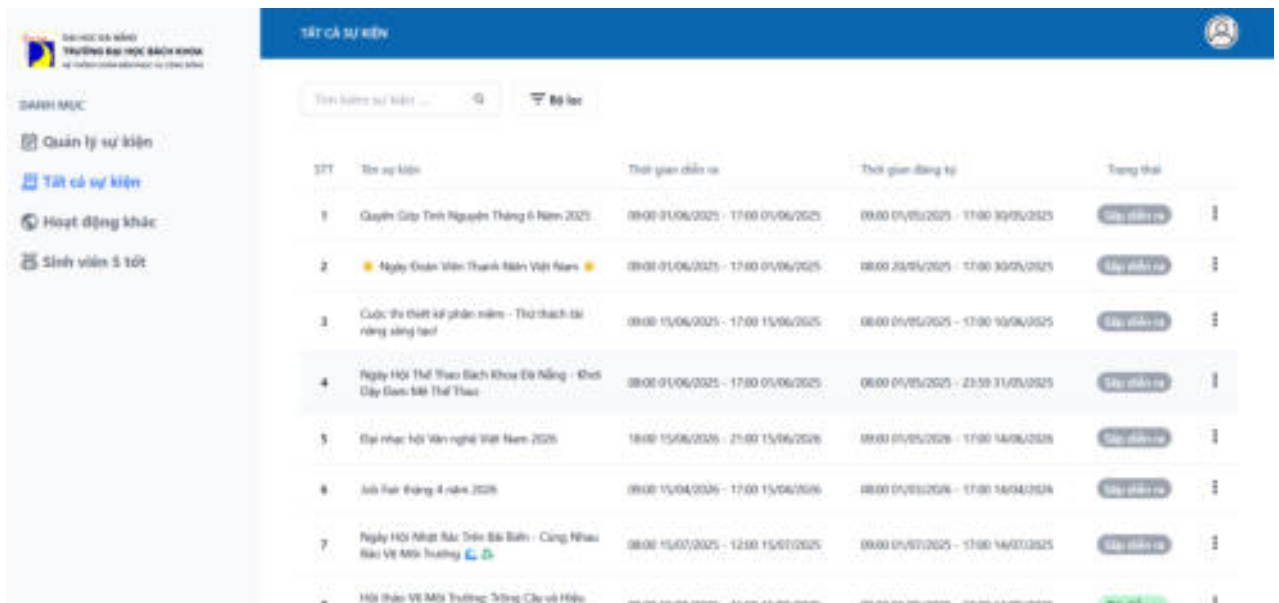
Hình 35 Màn hình chi tiết sự kiện



Hình 36 Màn hình chi tiết sự kiện

### 3.3.4 Màn hình tất cả sự kiện

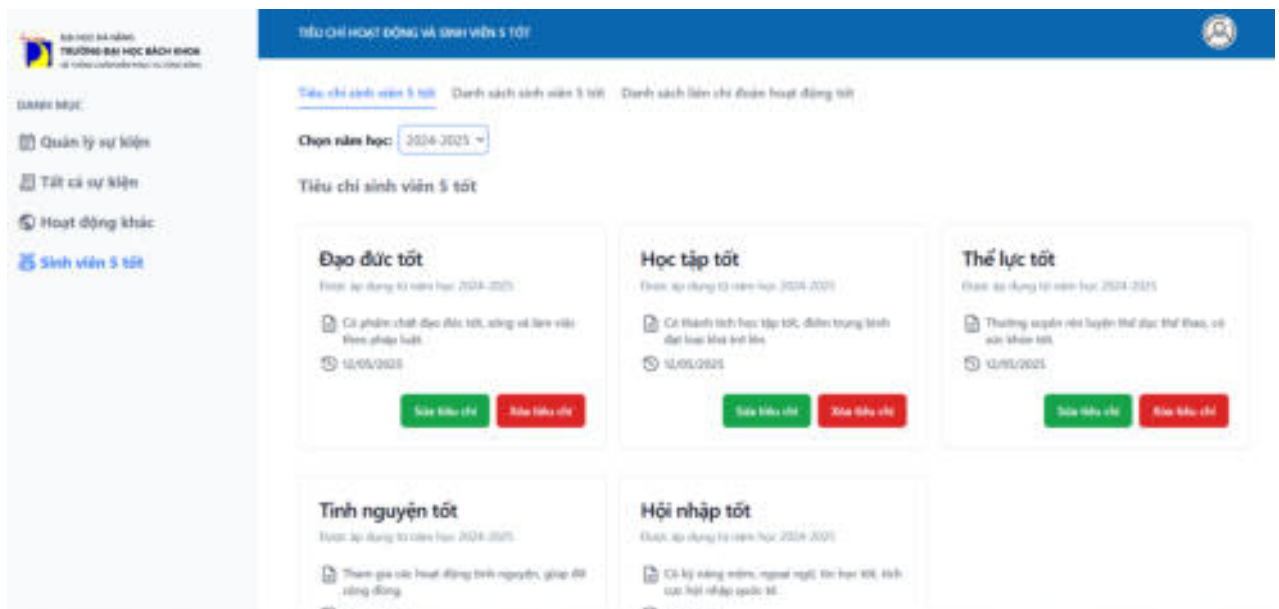
Trang tất cả sự kiện cung cấp cho người dùng cái nhìn tổng quan về toàn bộ các sự kiện trong hệ thống.



*Hình 37 Màn hình tất cả sự kiện*

### 3.3.5 Màn hình tiêu chí hoạt động và sinh viên 5 tốt

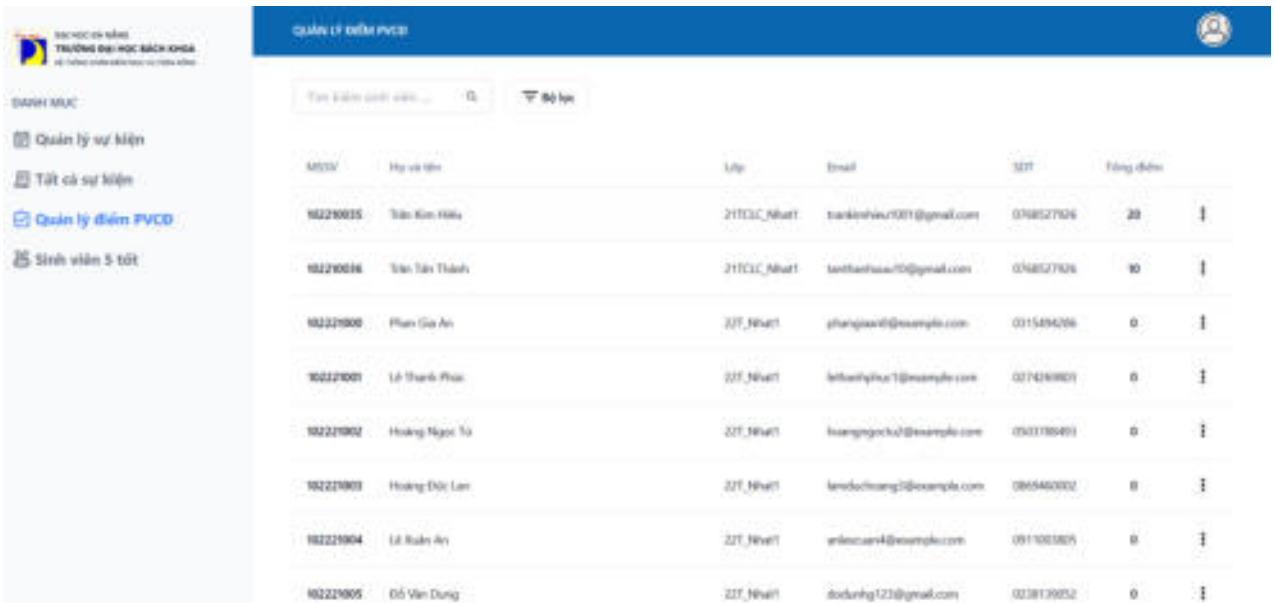
Cung cấp cho cán bộ cái nhìn rõ ràng về các tiêu chí cần đạt để trở thành “Sinh viên 5 tốt”, đồng thời giúp theo dõi các tiêu chí theo từng năm, sinh viên 5 tốt và liên chi đoàn hoạt động tốt



*Hình 38 Màn hình tiêu chí hoạt động và sinh viên 5 tốt*

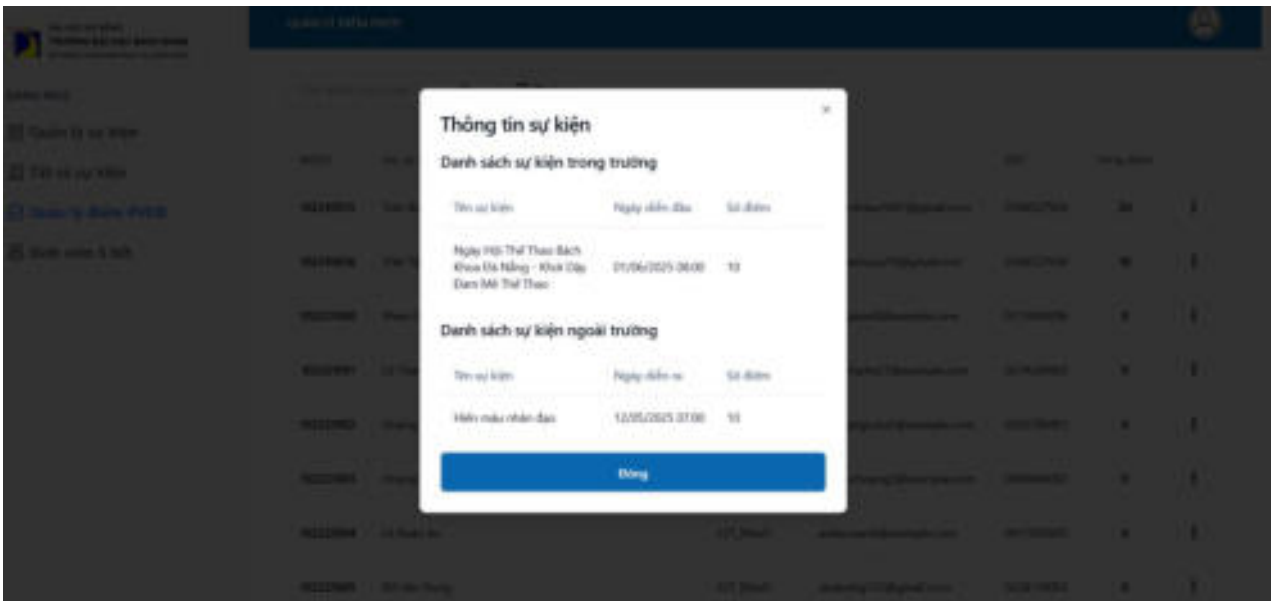
### 3.3.6 Màn hình quản lý điểm phục vụ cộng đồng

Hỗ trợ cộng tác sinh viên theo dõi điểm phục vụ cộng đồng dựa trên các hoạt động đã tham gia.



MSSV	Họ và tên	Vị trí	Email	SĐT	Điểm điểm
102210015	Tần Kim Hùng	21TC1C_NhưT	tan.kimhieu2001@gmail.com	0748527526	20
102210016	Tần Tấn Thành	21TC1C_NhưT	tan.tan.thanh70@gmail.com	0748527526	10
102210000	Phan Gia An	22T_NhưT	phan.gia.an@example.com	0315494206	0
102210001	Lê Thanh Phúc	22T_NhưT	le.thanh.phuc7@example.com	0274248801	0
102210002	Hoàng Ngọc Tú	22T_NhưT	hoangngoc.tu@example.com	0501780401	0
102210003	Hoàng Đức Lan	22T_NhưT	hduyhoang20@example.com	0864400002	0
102210004	Lê Xuân An	22T_NhưT	le.xuan.an4@example.com	0917003805	0
102210005	Đỗ Văn Dũng	22T_NhưT	do.van.dung123@gmail.com	0238139052	0

*Hình 39 Màn hình quản lý điểm phục vụ cộng đồng*



Tên sự kiện	Ngày diễn ra	Số điểm
Ngày 19/03 Tháng 3 Bạch Khoa Đà Nẵng - Khu Dãy Dặm Mũi Thi Thảo	01/06/2025 06:00	10

Tên sự kiện	Ngày diễn ra	Số điểm
Hội thảo nhân đạo	12/05/2025 07:00	10

*Hình 40 Màn hình quản lý điểm phục vụ cộng đồng*

### 3.4.6 Màn hình tạo sự kiện

Cho phép cán bộ quản lý, ban tổ chức được quyền tạo và cấu hình sự kiện mới để phục vụ cho hoạt động sinh viên, tính điểm rèn luyện, điểm phục vụ cộng đồng, hoặc xét danh hiệu "Sinh viên 5 tốt".

The screenshot shows a web interface for creating a new event. The title is 'Tạo sự kiện mới'. Under the 'Thông tin chung' (General information) section, there are several input fields: 'Tên sự kiện' (Event name), 'Địa điểm diễn ra' (Location), 'Năm học' (Academic year) with a dropdown menu, 'Thời gian bắt đầu' (Start time) and 'Thời gian kết thúc' (End time) with date pickers, and 'Thời gian bắt đầu đồng bộ' (Synchronized start time) and 'Thời gian kết thúc đồng bộ' (Synchronized end time) with date pickers. A sidebar on the left contains navigation links: 'Quản lý sự kiện', 'Tất cả sự kiện', 'Quản lý điểm PVCD', and 'Sinh viên 5 tốt'.

*Hình 41 Màn hình tạo sự kiện*

### 3.4.7 Màn hình quản lý lớp học

Cho phép cán bộ quản lý, giáo viên phụ trách, hoặc ban tổ chức sự kiện theo dõi và thao tác với danh sách lớp học.

The screenshot shows a web interface for managing the class list. The title is 'Danh sách lớp học'. There is a search bar and a 'Thêm lớp học' (Add class) button. The table below lists classes with the following data:

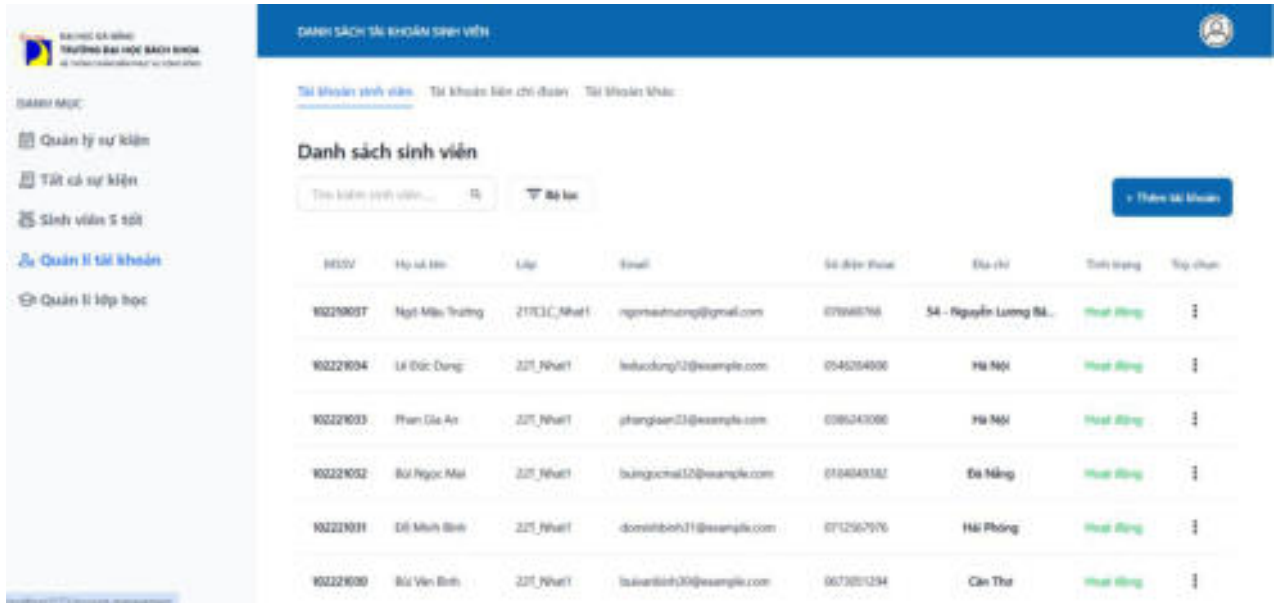
STT	Lớp	Khóa học	Class	Tác vụ
1	2102C_Nhật	2021-2025	Khoa CNTT	⋮
2	2202C	2022-2026	Khoa Cơ Khí	⋮
3	2301	2023-2027	Khoa Điện	⋮
4	2404	2024-2028	Khoa Hóa	⋮
5	2504D	2025-2029	Khoa Xây Dựng	⋮
6	2102C_DT1	2021-2025	Khoa CNTT	⋮
7	22T_Nhật	2022-2026	Khoa CNTT	⋮

The sidebar on the left contains navigation links: 'Quản lý sự kiện', 'Tất cả sự kiện', 'Sinh viên 5 tốt', 'Quản lý tài khoản', and 'Quản lý lớp học'.

*Hình 42 Màn hình quản lý lớp học*

### 3.4.8 Màn hình quản lý tài khoản

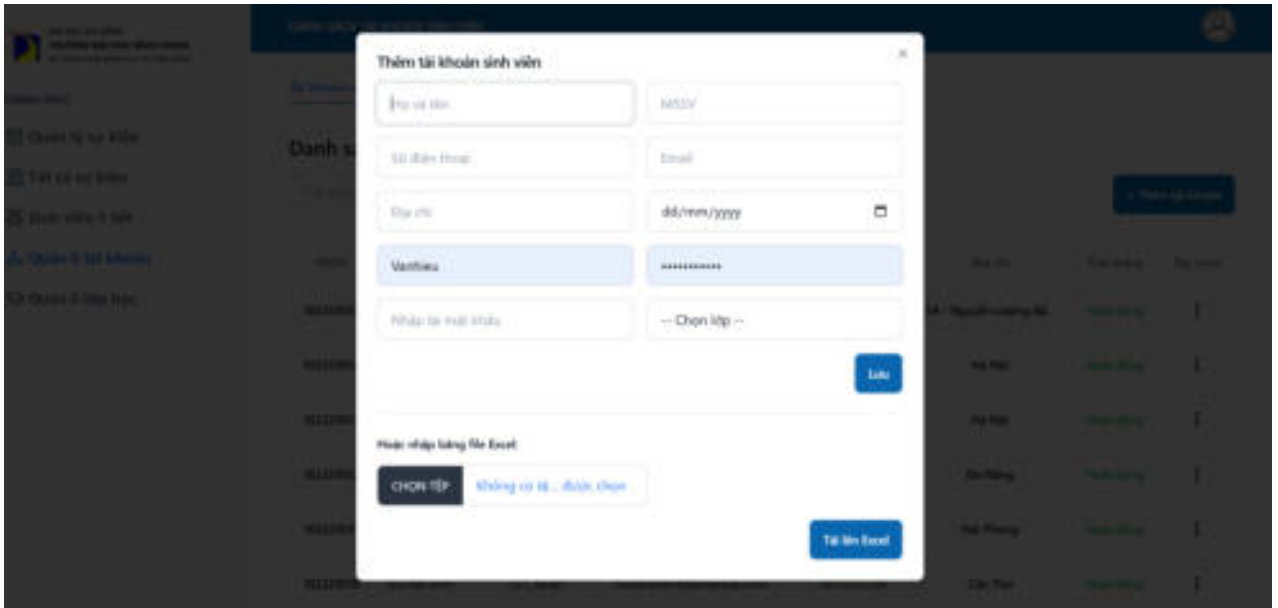
Cho phép quản trị viên hệ thống hoặc cán bộ được phân quyền thực hiện các thao tác quản lý tài khoản người dùng như sinh viên, giảng viên, ban tổ chức, và quản trị viên khác. Màn hình hỗ trợ các chức năng như tìm kiếm, tạo mới, khóa/mở tài khoản và chỉnh sửa thông tin.



*Hình 43 Màn hình quản lý tài khoản*

### 3.4.9 Màn hình thêm tài khoản

Cung cấp giao diện cho quản trị viên hoặc cán bộ có quyền tạo mới tài khoản người dùng trong hệ thống. Đảm bảo việc thêm người dùng được thực hiện đầy đủ, chính xác và an toàn.



*Hình 44 Màn hình thêm tài khoản*

### 3.4.6 Màn hình quản lý tham gia sự kiện

Cung cấp cho quản trị viên hoặc người tổ chức sự kiện khả năng theo dõi, xác nhận và quản lý danh sách người dùng (sinh viên) đã đăng ký hoặc tham gia sự kiện.



*Hình 45 Màn hình quản lý khóa học*

Mã số	Họ và tên	Lớp sinh hoạt	Khóa	E-mail
K0220005	Trần Kim Hòa	21XLC_Nhat1	Khóa CNTT	trankimhau1000@gmail.com
K0220006	Trần Tấn Thành	21XLC_Nhat1	Khóa CNTT	trankimhau1001@gmail.com
K0220000	Phan Gia An	22T_Nhat1	Khóa CNTT	phangiaan@example.com
K0220025	Bùi Thái Trung	22T_Nhat1	Khóa CNTT	buitaithung25@example.com
K0220007	Ngô Mậu Trường	21XLC_Nhat1	Khóa CNTT	ngomautruong@gmail.com

Hình 46 Màn hình quản lý khóa học

### 3.4.10 Màn hình điểm danh sinh viên

Cung cấp cho quản trị viên hoặc người tổ chức sự kiện khả năng theo dõi, xác nhận và quản lý danh sách người dùng (sinh viên) đã đăng ký hoặc tham gia sự kiện.

Mã số	Họ và tên	Lớp sinh hoạt	Khóa	Trạng thái	
<input type="checkbox"/>	K0220005	Trần Kim Hòa	21XLC_Nhat1	Khóa CNTT	Chưa xác nhận ***
<input type="checkbox"/>	K0220006	Trần Tấn Thành	21XLC_Nhat1	Khóa CNTT	Chưa xác nhận ***
<input type="checkbox"/>	K0220000	Phan Gia An	22T_Nhat1	Khóa CNTT	Chưa xác nhận ***
<input type="checkbox"/>	K0220025	Bùi Thái Trung	22T_Nhat1	Khóa CNTT	Chưa xác nhận ***
<input type="checkbox"/>	K0220007	Ngô Mậu Trường	21XLC_Nhat1	Khóa CNTT	Chưa xác nhận ***

Hình 47 Màn hình điểm danh sinh viên

### 3.4.11 Màn hình trang chủ bên phía sinh viên

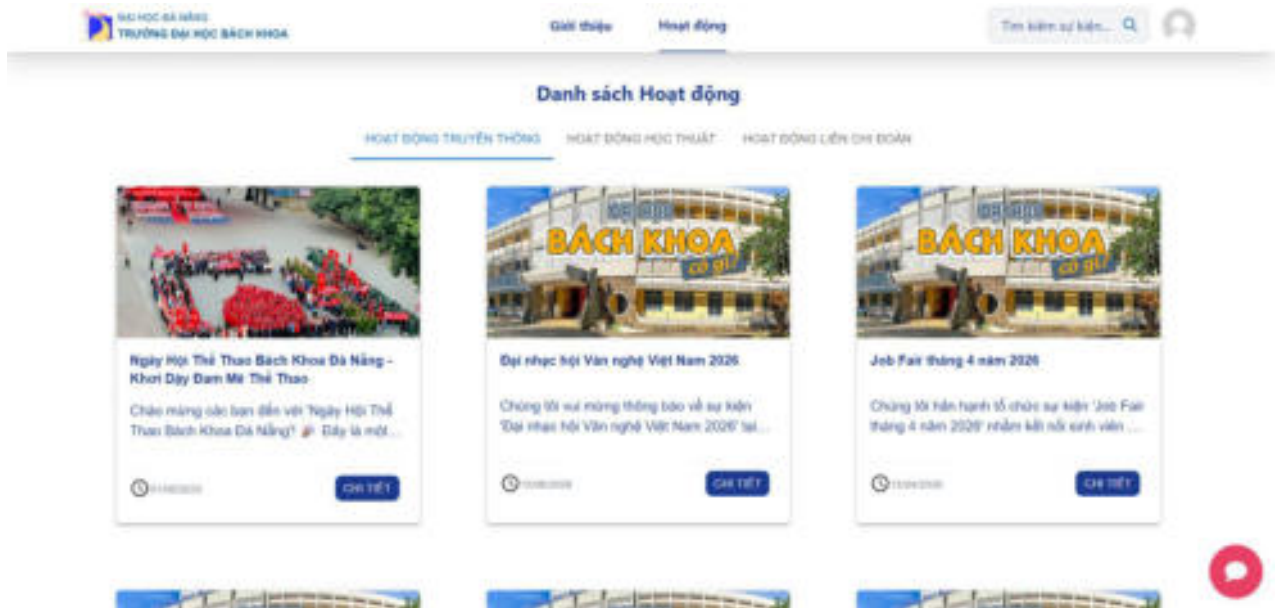


*Hình 48 Màn hình trang chủ sinh viên*



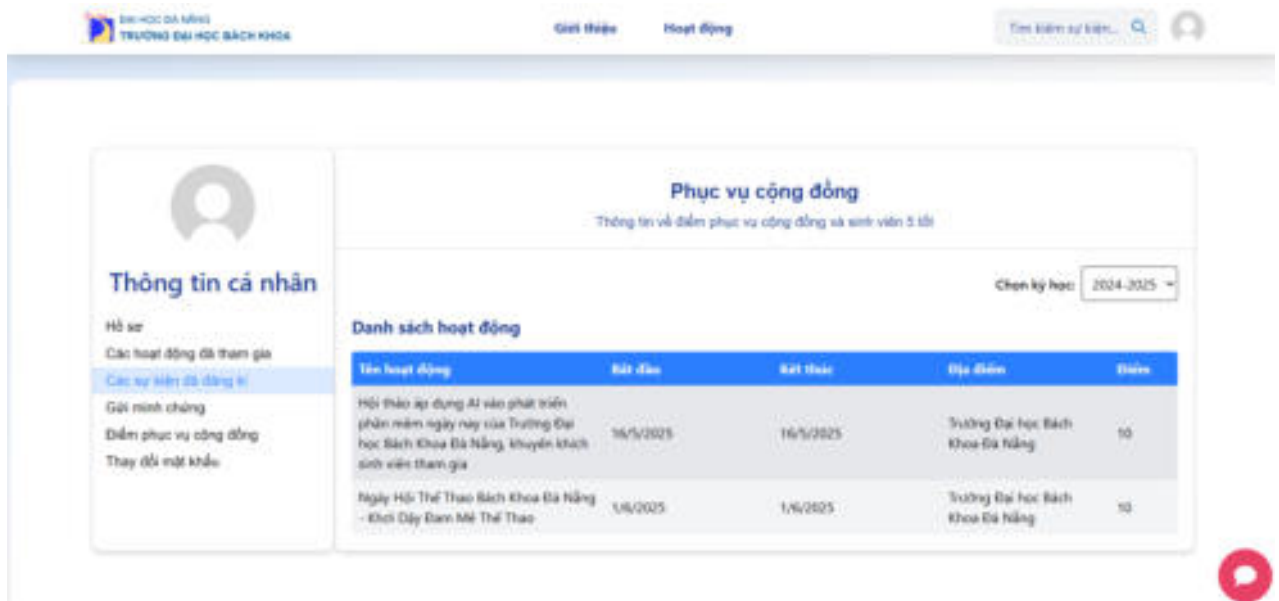
*Hình 49 Màn hình trang chủ sinh viên*

### 3.4.12 Màn hình danh sách hoạt động



Hình 50 Màn hình danh sách hoạt động

### 3.4.13 Màn hình danh sách menu phía sinh viên



Hình 51 Màn hình menu sinh viên

### **3.5 Kết luận Chương 3**

Chương 3 đã trình bày chi tiết quá trình triển khai hệ thống quản lý điểm phục vụ công đồng và sinh viên 5 tốt, dựa trên thiết kế đã xây dựng ở Chương 2. Hệ thống được phát triển bằng Spring Boot kết hợp PostgreSQL.

Ngoài các chức năng cốt lõi như ghi nhận hoạt động, xét điểm sinh viên, xác minh minh chứng, phân quyền người dùng,... hệ thống còn tích hợp hai chatbot nhằm tăng tính tương tác và hỗ trợ người dùng:

Chatbot RAG (Retrieval-Augmented Generation) được xây dựng nhằm tư vấn, giải đáp nhanh chóng các câu hỏi liên quan đến nội quy, quy định và quy trình chấm điểm rèn luyện. Chatbot này sử dụng kỹ thuật truy xuất thông tin kết hợp mô hình ngôn ngữ lớn (LLM), giúp mang lại câu trả lời chính xác, trích xuất từ tài liệu thực tế của nhà trường.

Chatbot Agent đóng vai trò như một trợ lý ảo, hỗ trợ tạo và quản lý sự kiện công đồng. Người dùng có thể tương tác bằng ngôn ngữ tự nhiên để đề xuất sự kiện, thiết lập thời gian, nội dung – giúp quy trình tổ chức sự kiện trở nên thuận tiện và thân thiện hơn với sinh viên.

Việc triển khai thành công cả hệ thống quản lý và hai chatbot trên không chỉ chứng minh tính khả thi của giải pháp, mà còn cho thấy tiềm năng tích hợp trí tuệ nhân tạo trong các hệ thống hỗ trợ sinh viên. Hệ thống góp phần nâng cao hiệu quả quản lý, tiết kiệm thời gian, giảm tải công việc thủ công, đồng thời tạo ra môi trường tương tác thông minh, minh bạch và hiện đại trong đánh giá và phát triển sinh viên toàn diện.

## KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### Kết luận

Trong báo cáo này, chúng tôi đã nghiên cứu và triển khai một hệ thống phần mềm quản lý điểm phục vụ công đồng và sinh viên 5 tốt, sử dụng các công nghệ hiện đại như Spring Boot, Docker, Cloudinary, Supabase, Render cùng với hệ thống gợi ý và chatbot được phát triển riêng. Mỗi thành phần công nghệ đều được lựa chọn cẩn thận nhằm đáp ứng hiệu quả các yêu cầu chức năng, phi chức năng và khả năng mở rộng của hệ thống.

Spring Boot đóng vai trò là nền tảng chính để xây dựng backend, với kiến trúc RESTful rõ ràng, giúp dễ dàng phát triển, bảo trì và tích hợp. Docker đảm bảo môi trường phát triển và triển khai đồng nhất, giảm thiểu lỗi do sự khác biệt môi trường. Các giải pháp lưu trữ và phân phối nội dung như Render, Cloudinary, Supabase được sử dụng để đảm bảo hiệu suất cao, độ tin cậy và khả năng mở rộng.

Đặc biệt, hệ thống đã tích hợp thành công hai chatbot thông minh được xây dựng bằng n8n:

Chatbot tư vấn nội quy giúp sinh viên dễ dàng tra cứu các quy định và quy trình liên quan đến điểm rèn luyện và danh hiệu “sinh viên 5 tốt”.

Chatbot hỗ trợ tạo sự kiện cho phép người dùng tương tác trực tiếp để tạo, cập nhật và theo dõi các hoạt động cộng đồng một cách nhanh chóng và thuận tiện.

Tổng thể, hệ thống đã đạt được các mục tiêu ban đầu về mặt chức năng, hiệu năng và tính mở rộng. Tuy nhiên, vẫn còn một số khía cạnh cần được cải tiến, chẳng hạn như giao diện người dùng, bảo mật và tối ưu hóa cho quy mô lớn. Với kiến trúc hiện đại và khả năng tích hợp cao, hệ thống sẵn sàng được áp dụng vào thực tế và mở rộng trong tương lai.

### Hướng phát triển

**Chatbot RAG tư vấn nội quy:** Tiếp tục nâng cao khả năng xử lý ngôn ngữ tự nhiên và phản hồi chính xác bằng cách mở rộng dữ liệu nội quy, đồng thời tối ưu các workflow trên n8n để tự động hóa việc cập nhật nội dung và phản hồi theo ngữ cảnh người dùng.

**Chatbot agent tạo sự kiện:** Phát triển thêm các workflow tích hợp đa kênh trên n8n, hỗ trợ tự động lên lịch, gửi thông báo nhắc nhở và đồng bộ dữ liệu với các hệ thống quản lý sự kiện khác, nhằm tăng tính linh hoạt và hiệu quả trong tổ chức sự kiện.

**Tối ưu hóa hiệu suất:** Nghiên cứu và áp dụng thêm các kỹ thuật tối ưu hóa hiệu suất khác cho ứng dụng, như việc sử dụng bộ nhớ đệm (caching) hiệu quả hơn và cân bằng tải (load balancing) để đảm bảo hệ thống hoạt động mượt mà dưới tải cao.

**Tăng cường bảo mật:** Trong tương lai, để nâng cao bảo mật hơn nữa, hệ thống sẽ mở rộng Spring Security với xác thực đa yếu tố (MFA), áp dụng mã hóa dữ liệu nhạy cảm, tăng cường phân quyền chi tiết hơn, cập nhật vá lỗi định kỳ, triển khai giám sát và ghi log truy cập nâng cao, đồng thời bổ sung các cơ chế bảo vệ API vượt trội hơn bên cạnh JWT như giới hạn tần suất truy cập (rate limiting) và kiểm soát nguồn gốc truy cập (CORS).

**Tích hợp với các dịch vụ khác:** Mở rộng hệ thống bằng cách tích hợp với các dịch vụ và API bên thứ ba để tăng cường chức năng và cung cấp trải nghiệm người dùng tốt hơn.

**Khả năng mở rộng:** Nghiên cứu và triển khai các kiến trúc microservices để tăng khả năng mở rộng và quản lý hệ thống hiệu quả hơn khi có sự tăng trưởng về số lượng người dùng và dữ liệu.

**Nâng cao trải nghiệm người dùng:** Liên tục cải tiến giao diện và trải nghiệm người dùng (UX/UI) dựa trên phản hồi và dữ liệu sử dụng của người dùng. Áp dụng các nguyên tắc thiết kế hiện đại và tối ưu hóa quy trình tương tác người dùng để nâng cao sự hài lòng và tương tác của người dùng.

**Hỗ trợ đa ngôn ngữ:** Mở rộng hỗ trợ nhiều ngôn ngữ hơn để phục vụ người dùng toàn cầu, bao gồm cả ngôn ngữ ít phổ biến và các ngôn ngữ vùng miền.

## TÀI LIỆU THAM KHẢO

- [1] Vietnix, "Mô hình Client-Server là gì?," *Vietnix*, [Online]. Available: <https://vietnix.vn/mo-hinh-client-server/#mo-hinh-client-server-la-gi>. [Accessed: Jun. 3, 2025].
- [2] 200Lab, "Git là gì?," *200Lab*, [Online]. Available: <https://200lab.io/blog/git-la-gi>. [Accessed: Jun. 3, 2025].
- [3] 200Lab, "GitHub là gì?," *200Lab*, [Online]. Available: <https://200lab.io/blog/github-la-gi>. [Accessed: Jun. 3, 2025].
- [4] Letdiv, "Git Flow là gì?," *Letdiv*, [Online]. Available: <https://letdiv.com/git-flow/>. [Accessed: Jun. 3, 2025].
- [5] Javarush, "Ưu và nhược điểm của Spring Boot và các hàm xử lý chuỗi trong Java," *Javarush*, [Online]. Available: <https://javarush.com/vi/groups/posts/vi.3380.nghe-gii-lao-75-u-im-v-nhc-im-ca-vic-s-dng-spring-boot-cc-hm-cho-chui-trong-java>. [Accessed: Jun. 3, 2025].
- [6] Tenten, "PostgreSQL là gì?," *Tenten*, [Online]. Available: <https://tenten.vn/tin-tuc/postgresql-la-gi/>. [Accessed: Jun. 3, 2025].
- [7] Viblo, "Sử dụng Cloudinary để quản lý ảnh cho ứng dụng của bạn," *Viblo*, [Online]. Available: <https://viblo.asia/p/su-dung-cloudinary-de-quan-ly-anh-cho-ung-dung-cua-ban-E7bGoxggv5e2>. [Accessed: Jun. 3, 2025].
- [8] 200Lab, "Supabase là gì?," *200Lab*, [Online]. Available: <https://200lab.io/blog/supabase-la-gi>. [Accessed: Jun. 3, 2025].
- [9] Viblo, "Docker là gì? Kiến thức cơ bản về Docker," *Viblo*, [Online]. Available: <https://viblo.asia/p/docker-la-gi-kien-thuc-co-ban-ve-docker-maGK7qeelj2>. [Accessed: Jun. 3, 2025].

- [10] Render, "Deploys," Render, [Online]. Available: <https://render.com/docs/deploys>. [Accessed: Jun. 3, 2025].
- [11] 200Lab, "n8n là gì?," 200Lab, [Online]. Available: <https://200lab.io/blog/n8n-la-gi>. [Accessed: Jun. 3, 2025].
- [12] ReactJS, "React Tutorial," React, [Online]. Available: <https://vi.legacy.reactjs.org/tutorial/tutorial.html>. [Accessed: Jun. 3, 2025].
- [13] Aptech, "Tailwind là gì?," Aptech, [Online]. Available: <https://aptech.fpt.edu.vn/tailwind-la-gi.html>. [Accessed: Jun. 3, 2025].
- [14] W3Schools, "TypeScript Introduction," W3Schools, [Online]. Available: [https://www.w3schools.com/typescript/typescript\\_intro.php](https://www.w3schools.com/typescript/typescript_intro.php). [Accessed: Jun. 3, 2025].
- [15] Viblo, "Tìm hiểu về Retrieval-Augmented Generation (RAG)," Viblo, [Online]. Available: <https://viblo.asia/p/chatgpt-series-5-tim-hieu-ve-retrieval-augmented-generation-rag-Ny0VGRd7LPA>. [Accessed: Jun. 3, 2025].
- [16] Airbyte, "Pinecone Vector Database," Airbyte, [Online]. Available: <https://airbyte.com/data-engineering-resources/pinecone-vector-database>. [Accessed: Jun. 3, 2025].

## **PHỤ LỤC 1**

## **PHỤ LỤC 2**