

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**  
**KHOA CÔNG NGHỆ THÔNG TIN**

-----o0o-----



**ĐỒ ÁN TỐT NGHIỆP**  
**XÂY DỰNG WEBSITE HỖ TRỢ QUẢN LÝ**  
**LỚP HỌC TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**Giảng viên hướng dẫn:** TS. Nguyễn Quốc Tuấn

**Sinh viên thực hiện:** Phùng Ngọc Anh

**Mã sinh viên:** 201210020

**Lớp:** Công nghệ thông tin 3

**Khoá:** 61

Hà Nội, năm 2024

## LỜI CẢM ƠN

Em xin gửi lời cảm ơn chân thành tới **thầy giáo, Tiến Sĩ Nguyễn Quốc Tuấn**, người đã tận tình giúp đỡ, quan tâm, dẫn dắt em trong suốt quá trình thực hiện đề tài. Những lời đóng góp của thầy đã giúp em có thể hoàn thiện đề tài một cách tốt nhất có thể.

Đồng thời, em cũng xin được gửi lời cảm ơn đến tất cả các **thầy cô** công tác tại **khoa Công nghệ thông tin - Trường Đại học Giao thông vận tải** đã dìu dắt, chỉ bảo cho em trong suốt quá trình học tập tại trường.

Cuối cùng, em xin gửi lời cảm ơn đến gia đình và tất cả người thân, bạn bè, những người đã luôn động viên, ủng hộ, tạo điều kiện cho em trong thời gian hoàn thành đồ án.

Mặc dù em đã cố gắng rất nhiều nhưng cũng không tránh khỏi thiếu sót và hạn chế trong quá trình thực hiện. Em rất mong nhận được sự thông cảm và chỉ bảo cũng như những đóng góp chân thành của các thầy cô và bạn bè để bài đồ án tốt nghiệp được tốt hơn.

Em xin chân thành cảm ơn!

*Hà Nội, ngày 06 tháng 06 năm 2024*

**Sinh viên thực hiện**

Phùng Ngọc Anh

# MỤC LỤC

<b>LỜI CẢM ƠN .....</b>	<b>1</b>
<b>DANH MỤC BẢNG BIỂU .....</b>	<b>4</b>
<b>DANH MỤC HÌNH ẢNH .....</b>	<b>5</b>
<b>MỞ ĐẦU.....</b>	<b>7</b>
<b>CHƯƠNG 1: TỔNG QUAN CÔNG NGHỆ .....</b>	<b>8</b>
1.1    Tổng quan công nghệ.....	8
1.1.1    Vue.js.....	8
1.1.2    NET .....	10
1.1.3    MySQL.....	13
1.1.4    MongoDB.....	15
1.2    Tổng quan đề tài.....	17
1.2.1    Bài toán .....	17
1.2.2    Khảo sát.....	17
1.2.3    Mục tiêu đề tài.....	18
<b>CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG.....</b>	<b>19</b>
2.1    Mô tả tổng quan hệ thống .....	19
2.1.1    Các tác nhân của hệ thống.....	19
2.1.2    Các yêu cầu của hệ thống.....	19
2.1.3    Các chức năng của hệ thống.....	19
2.2    Đặc tả các yêu cầu chức năng .....	21
2.2.1    Biểu đồ use-case tổng quát.....	21
2.2.2    Biểu đồ use-case chi tiết.....	23
2.3    Phân tích thiết kế cơ sở dữ liệu .....	29
2.3.1    Phân tích cơ sở dữ liệu .....	29
2.3.2    Thiết kế cơ sở dữ liệu.....	30
2.3.3    Mô hình quan hệ.....	37
<b>CHƯƠNG 3: XÂY DỰNG CHƯƠNG TRÌNH .....</b>	<b>38</b>
3.1    Cài đặt chương trình.....	38
3.1.1    Các thư viện hỗ trợ.....	38
3.1.2    Cài đặt chương trình.....	39
3.2    Kết quả thu được .....	43

3.2.1	Chức năng tài khoản.....	43
3.2.2	Chức năng nhóm .....	46
3.2.3	Chức năng bảng tin .....	47
3.2.4	Chức năng thời gian biểu .....	48
3.2.5	Chức năng nhắn tin .....	51
3.2.6	Chức năng quản lý.....	52
<b>KẾT LUẬN .....</b>		<b>55</b>
<b>DANH MỤC TÀI LIỆU THAM KHẢO .....</b>		<b>56</b>

## DANH MỤC BẢNG BIỂU

Bảng 1.1. So sánh một số tiêu chí giữa React, Angular và Vue.....	9
Bảng 1.2. So sánh một số tiêu chí giữa Spring, ASP.NET Core, Django, Express và Laravel .....	12
Bảng 1.3. So sánh một số tiêu chí giữa SQL Server, MySQL và PostgreSQL.....	14
Bảng 2.1. Mô tả use-case tổng quát.....	21
Bảng 2.2. Bảng mô tả use-case chức năng quản lý tài khoản .....	23
Bảng 2.3. Mô tả use-case chức năng tin tức .....	24
Bảng 2.4. Mô tả use-case chức năng kiểm duyệt tin tức .....	24
Bảng 2.5. Mô tả use-case chức năng thời gian biểu .....	25
Bảng 2.6. Mô tả use-case chức năng quản lý nhóm .....	27
Bảng 2.7. Mô tả use-case chức năng nhắn tin .....	28
Bảng 2.8. Mô tả use-case chức năng nhóm .....	28
Bảng 2.9. Danh sách các trường trong bảng Bulletins .....	30
Bảng 2.10. Danh sách các trường trong bảng Comments .....	30
Bảng 2.11. Danh sách các trường trong bảng Events.....	31
Bảng 2.12. Danh sách các trường trong bảng Groups.....	32
Bảng 2.13. Danh sách các trường trong bảng Notifications.....	32
Bảng 2.14. Danh sách các trường trong bảng Roles .....	33
Bảng 2.15. Danh sách các trường trong bảng Timetables.....	33
Bảng 2.16. Danh sách các trường trong bảng User .....	34
Bảng 2.17. Danh sách các trường trong tập hợp Groups .....	35
Bảng 2.18. Danh sách các trường trong tập hợp Chat.....	35
Bảng 2.19. Danh sách các trường trong tập hợp Messages.....	36
Bảng 2.20. Danh sách các trường trong tập hợp Users .....	36

## DANH MỤC HÌNH ẢNH

Hình 1.1. Logo của Vue.js.....	8
Hình 1.2. Lịch sử phát triển và các phiên bản của .NET.....	10
Hình 1.3. Một số framework của .NET trong các lĩnh vực khác nhau .....	11
Hình 1.4. Logo hệ quản trị cơ sở dữ liệu MySQL.....	13
Hình 1.5. Logo hệ quản trị cơ sở dữ liệu MongoDB.....	16
Hình 2.1. Biểu đồ use-case tổng quát .....	21
Hình 2.2. Biểu đồ use-case chức năng quản lý tài khoản.....	23
Hình 2.3. Biểu đồ use-case chức năng tin tức .....	24
Hình 2.4. Biểu đồ use-case chức năng kiểm duyệt tin tức .....	24
Hình 2.5. Biểu đồ use-case chức năng thời gian biểu .....	25
Hình 2.6. Biểu đồ use-case chức năng quản lý nhóm .....	26
Hình 2.7. Biểu đồ use-case chức năng nhắn tin .....	27
Hình 2.8. Biểu đồ use-case chức năng nhóm .....	28
Hình 2.9. Mô hình quan hệ các bảng trong hệ quản trị cơ sở dữ liệu MySQL .....	37
Hình 2.10. Mô hình quan hệ các tập hợp trong hệ quản trị cơ sở dữ liệu MongoDB ...	37
Hình 3.1. Hình ảnh các api của chức năng tài khoản .....	40
Hình 3.2. Hình ảnh các api của chức năng nhóm.....	40
Hình 3.3. Hình ảnh các api của chức năng bảng tin .....	41
Hình 3.4. Hình ảnh các api của chức năng thời gian biểu.....	41
Hình 3.5. Hình ảnh phương thức gửi phản hồi cho client từ phía server.....	42
Hình 3.6. Hình ảnh phương thức đọc file excel người dùng.....	43
Hình 3.7. Giao diện trang đăng nhập.....	44
Hình 3.8. Dialog quên mật khẩu.....	44
Hình 3.9. Hình ảnh mail do hệ thống gửi khi quên mật khẩu .....	45
Hình 3.10. Giao diện trang người dùng.....	45
Hình 3.11. Giao diện trang chọn nhóm .....	46
Hình 3.12. Dialog thêm nhóm mới.....	46
Hình 3.13. Giao diện trang bảng tin .....	47
Hình 3.14. Giao diện trang phê duyệt tin .....	47
Hình 3.15. Giao diện chi tiết tin tức .....	48
Hình 3.16. Giao diện trang thời gian biểu .....	48
Hình 3.17. Popup xác minh tài khoản của Google Calendar .....	49
Hình 3.18. Dialog thêm sự kiện.....	49
Hình 3.19. Dialog import thời khoá biểu.....	50
Hình 3.20. Dialog bật/tắt thời gian nhắc nhở .....	50
Hình 3.21. Giao diện trang nhắn tin .....	51

Hình 3.22. Dialog thêm thành viên vào cuộc trò chuyện .....	51
Hình 3.23. Giao diện trang quản lý ( <i>trong vai trò quản lý</i> ) .....	52
Hình 3.24. Giao diện trang quản lý ( <i>trong vai trò thành viên</i> ).....	52
Hình 3.25. Template import danh sách thành viên.....	53
Hình 3.26. Dialog mời thành viên .....	54

## MỞ ĐẦU

Hằng ngày, có rất nhiều yếu tố ảnh hưởng đến công việc học tập của sinh viên. Từ lỡ mất thông báo của trường lớp, cho đến việc quên lịch, thậm chí sắp xếp sai lịch học,... dẫn đến nhiều hậu quả như: đi học muộn, nghỉ học hôm có tiết, đi học hôm được thông báo nghỉ,... thậm chí nghiêm trọng hơn cũng có: không mang đầy đủ tài liệu vào phòng thi dù cán bộ lớp đã thông báo (do bỏ sót tin nhắn), nghỉ học quá nhiều dẫn đến bị cấm thi,...

Để giải quyết những vấn đề trên, đề tài nghiên cứu và phát triển "Xây dựng website hỗ trợ quản lý lớp học cho sinh viên Trường Đại học Giao thông Vận tải" đã được đặt ra với mục tiêu: cung cấp một công cụ hiệu quả giúp cán bộ lớp gửi thông báo đến sinh viên một cách nhanh chóng; hỗ trợ sinh viên trực quan hóa thời khóa biểu với độ chính xác cao và đáng tin cậy. Hệ thống này không chỉ giúp các cán bộ lớp thuận tiện hơn trong quá trình quản lý lớp học mà còn nâng cao trải nghiệm học tập của sinh viên, đảm bảo rằng họ luôn nắm bắt được các thông tin và lịch trình một cách đầy đủ và kịp thời.

Em mong việc xây dựng website hỗ trợ quản lý lớp học cho sinh viên Trường Đại học Giao thông Vận tải có thể giúp đỡ các sinh viên trong trường thuận tiện hơn trong việc tiếp cận thông tin và quản lý thời gian. Em nhận thấy đề tài có tính phát triển cao, nếu sản phẩm thành công thì có thể đem lại nhiều giá trị cho sinh viên, giúp các bạn thuận tiện hơn trong công việc học tập tại trường.



# CHƯƠNG 1: TỔNG QUAN CÔNG NGHỆ

## 1.1 Tổng quan công nghệ

### 1.1.1 Vue.js

#### 1.1.1.1 Giới thiệu về Vue.js

Thị trường framework JavaScript đã trải qua nhiều giai đoạn biến động. Vào năm 2010, Google cho ra mắt AngularJS, sự xuất hiện của nó nhanh chóng trở thành lựa chọn phổ biến cho các dự án lớn và phức tạp. Tuy nhiên đến năm 2013, React được phát hành bởi Facebook, với cấu trúc tập trung vào UI components tái sử dụng và quản lý trạng thái hiệu quả, nhanh chóng vượt qua AngularJS nhờ sự đơn giản và hiệu suất cao. Ngay sau đó một năm, Vue.js ra đời, kết hợp ưu điểm của AngularJS và React, mang lại trải nghiệm lập trình dễ dàng và hiệu quả đã thu hút nhiều lập trình viên trải nghiệm và sử dụng. Trong lúc này, AngularJS lại gặp khó khăn khi cộng đồng chuyển sang các framework khác do những bất tiện trong quá trình sử dụng. Cho đến năm 2016, Google đã cho ra mắt Angular 2 được viết lại hoàn toàn với cấu trúc và tư duy mới. Còn AngularJS, trước những nhược điểm khó có thể khắc phục và thị trường giảm sút rõ rệt, đã chính thức dừng hỗ trợ từ tháng 12/2021. Tới thời điểm hiện tại, thị trường framework Javascript chủ yếu vẫn là sân chơi của ba “ông lớn” React, Angular và Vue.



**Hình 1.1.** Logo của Vue.js

Vue.js là một framework JavaScript được thiết kế để dễ dàng tích hợp với các dự án hiện có. Nó kết hợp ưu điểm của cả Angular và React, mang lại sự linh hoạt trong phát triển ứng dụng và quản lý trạng thái hiệu quả. Với cú pháp dễ hiểu và tài liệu chi tiết, Vue.js đã nhanh chóng thu hút được sự quan tâm của cộng đồng lập trình viên và trở thành một trong những lựa chọn hàng đầu trong việc phát triển các ứng dụng web hiện đại.

### 1.1.1.2 Ưu điểm của Vue.js

- **Quá trình phát triển đơn giản:** Vue.js có cú pháp đơn giản và trực quan, giúp việc học và viết mã trở nên dễ dàng hơn. Kiến trúc dựa trên component của nó hỗ trợ khả năng tái sử dụng lại các đoạn mã, giúp tăng tốc thời gian phát triển dự án.
- **Hiệu suất cao:** VueJS tận dụng DOM ảo và các thuật toán trong việc render, mang lại hiệu suất ấn tượng. Chỉ những thành phần cần thiết mới được cập nhật lại, giảm thiểu việc render các components không có sự thay đổi, từ đó nâng cao tốc độ tổng thể.
- **Tính linh hoạt:** Vue.js là một framework có khả năng thích ứng cao, có thể dễ dàng tích hợp vào các dự án hiện có.
- **Tài liệu học tập đầy đủ:** Vue.js có tài liệu toàn diện và được viết theo tổ chức tốt, giúp các lập trình viên dễ dàng bắt đầu. Tài liệu Vue chính thức bao gồm tất cả các khía cạnh của việc phát triển Vue, từ những khái niệm cơ bản đến nâng cao.

### 1.1.1.3 So sánh giữa Vue.js và các framework khác

Dưới đây là bảng so sánh giữa Vue.js và một số framework khác phổ biến trên thị trường dựa theo một số tiêu chí nhất định.

**Bảng 1.1.** So sánh một số tiêu chí giữa React, Angular và Vue

	<b>React</b>	<b>Angular</b>	<b>Vue</b>
<b>Cộng đồng hỗ trợ</b>	Lớn	Lớn	Trung Bình
<b>Độ tiếp cận của tài liệu hướng dẫn</b>	Dễ	Trung bình	Dễ
<b>Độ khó triển khai</b>	Trung Bình	Khó	Dễ
<b>Thư viện hỗ trợ</b>	Nhiều	Nhiều	Trung Bình
<b>Hiệu suất thực thi chương trình</b>	Nhanh	Vừa	Nhanh
<b>Chi phí triển khai</b>	Vừa	Rẻ	Rẻ
<b>Hỗ trợ chính thức xây dựng ứng dụng trên thiết bị di động</b>	Tốt	Trung Bình	Kém
<b>Mã nguồn mở</b>	Có	Có	Có

Lưu ý, bảng được so sánh dựa theo các tiêu chí và thang điểm (*theo thứ tự từ lớn về bé*) như sau:

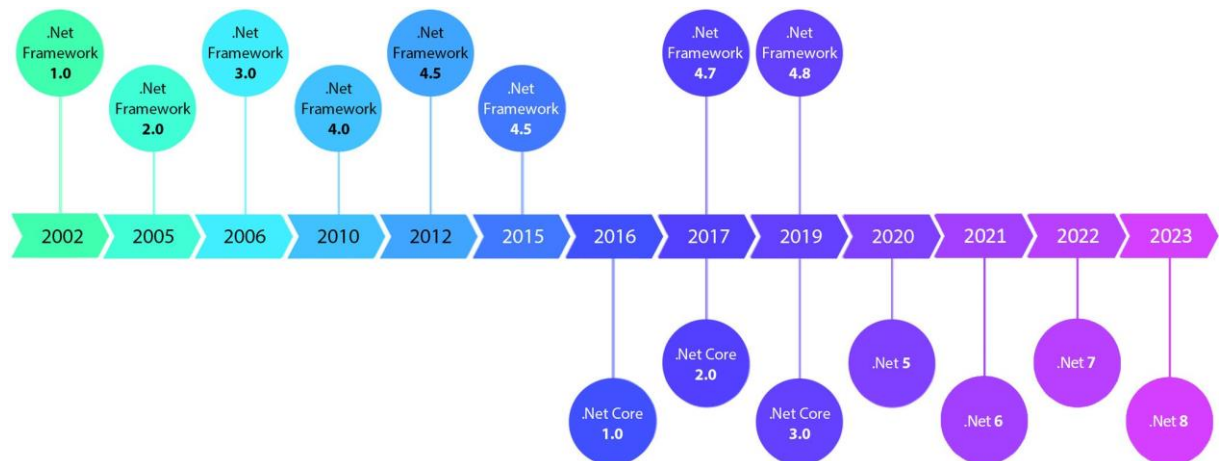
- Cộng đồng hỗ trợ: Lớn, Trung Bình, Nhỏ
- Độ tiếp cận của tài liệu hướng dẫn: Dễ, Trung Bình, Khó
- Độ khó triển khai: Dễ, Trung Bình, Khó
- Thư viện hỗ trợ: Nhiều, Trung Bình, Ít
- Hiệu suất thực thi chương trình: Nhanh, Vừa, Chậm
- Chi phí triển khai: Rẻ, Vừa, Đắt
- Hỗ trợ chính thức xây dựng ứng dụng trên thiết bị di động: Tốt, Trung Bình, Kém
- Mã nguồn mở: Có, Không

## 1.1.2 NET

### 1.1.2.1 Giới thiệu về .NET

.NET là nền tảng mã nguồn mở miễn phí trên các hệ điều hành Windows, Linux, và macOS. Đây là phiên bản đa nền tảng thừa kế từ .NET Framework. Dự án này chủ yếu được phát triển bởi các nhân viên Microsoft thuộc tổ chức .NET Foundation và được phát hành theo Giấy phép MIT. (*lưu ý là .NET, .NET Core và .NET Framework là ba khái niệm khác nhau mà mọi người rất hay nhầm lẫn*)

- **2002:** Microsoft ra mắt .NET Framework, cung cấp môi trường thực thi ứng dụng (CLR) và bộ thư viện lớn (FCL) để phát triển ứng dụng Windows, web và dịch vụ.
- **2016:** Microsoft công bố .NET Core, một phiên bản nhẹ hơn của .NET Framework, mã nguồn mở và đa nền tảng, chạy trên Windows, Linux và macOS.
- **2020:** Microsoft giới thiệu .NET 5, sau đổi tên thành .NET, tích hợp các tính năng của .NET Core cùng nhiều cải tiến khác, mang lại khả năng tương thích ngược với .NET Framework và .NET Core, hỗ trợ nhiều nền tảng và môi trường phát triển.



**Hình 1.2.** Lịch sử phát triển và các phiên bản của .NET

.NET có một hệ sinh thái rất phong phú, với đầy đủ các framework cho nhiều loại ứng dụng khác nhau, như web (ASP.NET), desktop (Windows Forms, Windows Presentation Foundation), cho đến mobile (MAUI), game (Unity)... Nó cung cấp hệ thống thư viện class khổng lồ chung mà tất cả các framework trên nó đều có thể sử dụng.



**Hình 1.3.** Một số framework của .NET trong các lĩnh vực khác nhau

### 1.1.2.2 Ưu điểm của .NET

- **Đa nền tảng:** Với .NET, ứng dụng có thể chạy trên bất kỳ nền tảng nào. Từ các ứng dụng di động chạy trên iOS, Android và Windows cho đến các ứng dụng máy chủ doanh nghiệp chạy trên Windows Server và Linux hay các dịch vụ vi mô, vĩ mô chạy trên đám mây, .NET đều có thể cung cấp giải pháp phù hợp.
- **Được các lập trình viên yêu thích:** .NET là một nền tảng phát triển nguồn mở, hiện đại, sáng tạo và được các nhà phát triển yêu thích, xếp hạng nhất trong Khảo sát về framework mà lập trình viên yêu thích nhất trên Stack Overflow (*Stack Overflow Developer Survey*) ba năm liên tiếp (2019, 2020 và 2021).
- **Hiệu suất:** .NET rất nhanh. Điều đó có nghĩa là các ứng dụng phản hồi nhanh hơn và yêu cầu ít hơn trong việc tính toán.
- **Bảo mật tốt:** Nền tảng .NET được Microsoft hỗ trợ chính thức và được hàng nghìn công ty cũng như hàng triệu lập trình viên tin tưởng. Microsoft rất coi trọng vấn đề bảo mật và phát hành các bản cập nhật nhanh chóng khi phát hiện ra các mối đe dọa.
- **Hệ sinh thái lớn:** Với hơn 5.000.000 lập trình viên .NET trên toàn thế giới, .NET sở hữu hệ sinh thái rộng với sự kết hợp giữa các thư viện từ trình quản lý gói NuGet cho đến Visual Studio Marketplace. Cộng đồng .NET rất mạnh và đông đảo, sẵn sàng hỗ trợ cho mọi người dùng với các vấn đề của họ.
- **Mã nguồn mở:** .NET Foundation là một tổ chức phi lợi nhuận độc lập hỗ trợ hệ sinh thái mã nguồn mở của .NET. .NET có hơn 100.000 đóng góp từ các lập trình viên từ hơn 3.700 công ty bên ngoài Microsoft.

### 1.1.2.3 So sánh giữa .NET và các đối thủ cạnh tranh

Dưới đây là bảng so sánh giữa .NET và một số framework khác phổ biến trên thị trường dựa theo một số tiêu chí nhất định.

**Bảng 1.2.** So sánh một số tiêu chí giữa Spring, ASP.NET Core, Django, Express và Laravel

	<b>Spring</b>	<b>ASP.NET Core</b>	<b>Django</b>	<b>Express</b>	<b>Laravel</b>
<b>Cộng đồng hỗ trợ</b>	Lớn	Lớn	Lớn	Trung Bình	Lớn
<b>Tài liệu hướng dẫn dễ tiếp cận</b>	Dễ	Dễ	Dễ	Trung Bình	Dễ
<b>Vấn đề bảo mật</b>	Tốt	Tốt	Tốt	Trung Bình	Tốt
<b>Hệ sinh thái</b>	Đa dạng	Đa dạng	Đa dạng	Trung Bình	Đa dạng
<b>Hiệu suất thực thi chương trình</b>	Nhanh	Nhanh	Vừa	Nhanh	Vừa
<b>Chi phí triển khai lên internet</b>	Vừa	Đắt	Rẻ	Rẻ	Rẻ
<b>Hiệu suất thực thi chương trình</b>	Tốt	Tốt	Trung Bình	Tốt	Trung Bình
<b>Đặc biệt mạnh với hệ điều hành</b>	Không	Window	Không	Không	Không
<b>Mã nguồn mở</b>	Có	Có	Có	Có	Có
<b>Đa nền tảng</b>	Có	Có	Có	Có	Có

Lưu ý, bảng được so sánh dựa theo các tiêu chí và thang điểm (*theo thứ tự từ lớn về bé*) như sau:

- Cộng đồng hỗ trợ: Lớn, Trung Bình, Nhỏ
- Tài liệu hướng dẫn dễ tiếp cận: Dễ, Trung Bình, Khó
- Vấn đề bảo mật: Tốt, Trung Bình, Yếu
- Hệ sinh thái: Đa dạng, Trung Bình, Ít
- Hiệu suất thực thi chương trình: Nhanh, Vừa, Chậm
- Chi phí triển khai: Rẻ, Vừa, Đắt
- Hiệu suất thực thi chương trình: Tốt, Trung Bình, Kém
- Đặc biệt mạnh với hệ điều hành: Linux, Window, Khác, Không (*ngang nhau với mọi hệ điều hành*)
- Mã nguồn mở: Có, Không
- Đa nền tảng: Có, Không

### 1.1.3 MySQL

#### 1.1.3.1 Giới thiệu về MySQL

SQL (Structured Query Language) là một ngôn ngữ lập trình được thiết kế để quản lý và truy xuất dữ liệu từ các hệ thống quản lý cơ sở dữ liệu quan hệ (Relational Database Management System - RDBMS). Với các câu lệnh mạnh mẽ và linh hoạt, SQL đã trở thành chuẩn công nghiệp cho việc làm việc với cơ sở dữ liệu. Phần lớn các hệ thống cơ sở dữ liệu quan hệ phổ biến như PostgreSQL, Microsoft SQL Server, Oracle Database,... đều sử dụng SQL làm ngôn ngữ truy vấn chính, trong đó có MySQL.



**Hình 1.4.** Logo hệ quản trị cơ sở dữ liệu MySQL

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở sử dụng SQL làm ngôn ngữ truy vấn chính. Được phát triển bởi MySQL AB, sau này thuộc sở hữu của Oracle Corporation, MySQL nổi tiếng với hiệu suất cao, đáng tin cậy và dễ sử dụng. MySQL hỗ trợ một loạt các ứng dụng từ các dự án nhỏ cho đến các hệ thống lớn và phức

tap. Nó thường được sử dụng trong các ứng dụng web và đã trở thành một phần không thể thiếu của bộ LAMP (Linux, Apache, MySQL, PHP/Python/Perl). Nhờ tính linh hoạt và khả năng tối ưu hóa truy vấn mạnh mẽ, MySQL là một trong những lựa chọn phổ biến cho việc quản lý cơ sở dữ liệu trên toàn thế giới.

### 1.1.3.2 Ưu điểm của MySQL

MySQL là một hệ thống quản lý cơ sở dữ liệu mã nguồn mở, chúng quản lý dữ liệu thông qua các cơ sở dữ liệu. Mỗi cơ sở dữ liệu có thể có nhiều bảng quan hệ chứa dữ liệu. Nó được phát triển bởi Oracle dựa trên ngôn ngữ truy vấn dữ liệu có cấu trúc (SQL).

Là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng, MySQL có độ phủ sóng rất cao trên thị trường với những ưu điểm không thể phủ nhận. Nó là hệ quản trị cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh, nên rất thích hợp cho các ứng dụng nhu cầu có truy cập cơ sở dữ liệu trên internet.

Ngoài ra, MySQL cung cấp bản download miễn phí trên trang chủ của nó, với nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows, Linux, Mac OS X, Unix, FreeBSD, NetBSD, Novell NetWare, SGI Irix, Solaris, SunOS,...

Cũng bởi vì là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở, nó sở hữu một cộng đồng lớn và tích cực bao gồm các nhà phát triển và người dùng, những người đóng góp cho sự phát triển và cung cấp hỗ trợ cho những người khác sử dụng.

Hầu như tất cả các nhà phát triển web ưa chuộng MySQL bởi tính chất đơn giản, dễ học, dễ thiết lập và bảo trì. Với các tính năng như phân bổ bộ nhớ dựa trên luồng, tham gia vòng lặp lồng nhau được tối ưu hóa, MySQL có khả năng hoạt động xuất sắc, và các khung công cụ lưu trữ đảm bảo hiệu suất hoàn hảo.

### 1.1.3.3 So sánh giữa MySQL và các cơ sở dữ liệu tương đồng

Dưới đây là bảng so sánh giữa MySQL và một số database khác phổ biến trên thị trường dựa theo một số tiêu chí nhất định.

**Bảng 1.3.** So sánh một số tiêu chí giữa SQL Server, MySQL và PostgreSQL

	SQL Server	MySQL	PostgreSQL
<b>Chi phí triển khai</b>	Đắt	Rẻ	Vừa
<b>Cộng đồng hỗ trợ</b>	Lớn	Lớn	Lớn

<b>Tốc độ truy vấn và thực hiện thao tác</b>	Nhanh	Nhanh	Nhanh
<b>Vấn đề bảo mật</b>	Tốt	Tốt	Tốt
<b>Tương thích với các hệ điều hành khác nhau</b>	Có	Có	Có

Lưu ý, bảng được so sánh dựa theo các tiêu chí và thang điểm (*theo thứ tự từ lớn về bé*) như sau:

- Chi phí triển khai: Rẻ, Vừa, Đắt
- Cộng đồng hỗ trợ: Lớn, Trung Bình, Nhỏ
- Tốc độ truy vấn và thực hiện thao tác: Nhanh, Trung Bình, Chậm
- Vấn đề bảo mật: Tốt, Trung Bình, Kém
- Tương thích với các hệ điều hành khác nhau: Có, Không

#### 1.1.4 MongoDB

##### 1.1.4.1 Giới thiệu về MongoDB

NoSQL là một nhóm các hệ quản trị cơ sở dữ liệu được thiết kế để xử lý dữ liệu không có cấu trúc hoặc bán cấu trúc, không yêu cầu tuân theo các mô hình bảng cố định như SQL. NoSQL phù hợp với các ứng dụng hiện đại yêu cầu tốc độ truy xuất cao, khả năng mở rộng linh hoạt và khả năng lưu trữ dữ liệu đa dạng. Các loại NoSQL chính bao gồm:

- **Cơ sở dữ liệu Key-Value:** Lưu trữ dữ liệu dưới dạng cặp khóa-giá trị, ví dụ như Redis và DynamoDB.
- **Cơ sở dữ liệu Document:** Lưu trữ dữ liệu dưới dạng tài liệu JSON hoặc XML, ví dụ như MongoDB và CouchDB.
- **Cơ sở dữ liệu Column-Family:** Tổ chức dữ liệu thành các cột, giúp tối ưu hóa truy vấn cho các tập dữ liệu lớn, ví dụ như Cassandra và HBase.
- **Cơ sở dữ liệu Graph:** Chuyên lưu trữ và xử lý các mối quan hệ phức tạp giữa các dữ liệu, ví dụ như Neo4j và ArangoDB.





**Hình 1.5.** Logo hệ quản trị cơ sở dữ liệu MongoDB

Một trong những hệ quản trị cơ sở dữ liệu NoSQL phổ biến nhất hiện nay là MongoDB, được thiết kế để lưu trữ và truy vấn dữ liệu dưới dạng tài liệu. MongoDB lưu trữ dữ liệu dưới dạng BSON (Binary JSON), cho phép linh hoạt trong việc lưu trữ dữ liệu có cấu trúc không cố định. Điều này giúp MongoDB dễ dàng mở rộng và quản lý các loại dữ liệu phức tạp.

#### 1.1.4.2 Ưu điểm của MongoDB

- **Tính linh hoạt cao:** MongoDB sử dụng định dạng JSON-like (BSON) để lưu trữ dữ liệu, cho phép bạn dễ dàng lưu trữ các cấu trúc dữ liệu phức tạp và thay đổi cấu trúc dữ liệu mà không cần phải chỉnh sửa schema.

- **Khả năng mở rộng:** MongoDB hỗ trợ sharding, giúp dễ dàng mở rộng ngang bằng cách phân chia dữ liệu trên nhiều máy chủ. Điều này rất hữu ích cho các ứng dụng yêu cầu xử lý lượng dữ liệu lớn.

- **Hiệu suất cao:** Với các tính năng như indexing và in-memory storage, MongoDB cung cấp khả năng truy vấn nhanh chóng và hiệu quả, phù hợp cho các ứng dụng yêu cầu tốc độ truy xuất cao.

- **Dễ dàng tích hợp:** MongoDB có khả năng tích hợp tốt với nhiều ngôn ngữ lập trình và framework phổ biến, giúp lập trình viên dễ dàng phát triển ứng dụng.

- **Cộng đồng và tài liệu phong phú:** MongoDB có một cộng đồng lớn và rất nhiều tài liệu, công cụ hỗ trợ, giúp bạn dễ dàng tìm kiếm sự trợ giúp và giải pháp cho các vấn đề gặp phải.

- **Tính năng phong phú:** MongoDB cung cấp nhiều tính năng hữu ích như replication (sao lưu và phục hồi), aggregation framework (xử lý và phân tích dữ liệu), và Atlas (dịch vụ cơ sở dữ liệu đám mây quản lý toàn diện).

- **Quản lý dễ dàng:** Các công cụ như MongoDB Atlas giúp dễ dàng triển khai, quản lý và giám sát các cơ sở dữ liệu MongoDB trên đám mây mà không cần lo lắng về hạ tầng cơ sở.

### 1.1.4.3 So sánh giữa MongoDB và các cơ sở dữ liệu tương đồng

Tùy vào yêu cầu cụ thể của ứng dụng mà mỗi cơ sở dữ liệu sẽ có những ưu điểm và nhược điểm riêng:

- **MongoDB:** Phù hợp cho các ứng dụng yêu cầu lưu trữ dữ liệu linh hoạt và truy vấn phong phú. Hỗ trợ mở rộng và quản lý dễ dàng.
- **Cassandra:** Lý tưởng cho các hệ thống phân tán với khối lượng ghi lớn, yêu cầu khả năng mở rộng cao và độ tin cậy cao.
- **Redis:** Tốt cho caching và message brokering nhờ tốc độ truy xuất cao, nhưng không phù hợp cho lưu trữ dữ liệu dài hạn lớn.
- **CouchDB:** Thích hợp cho các ứng dụng cần đồng bộ hóa dữ liệu dễ dàng và độ tin cậy trong các môi trường không ổn định.
- **Neo4j:** Tối ưu cho các ứng dụng yêu cầu xử lý và phân tích mối quan hệ phức tạp giữa các dữ liệu.

## 1.2 Tổng quan đề tài

### 1.2.1 Bài toán

Trong một lớp của trường Đại học Giao thông Vận tải, cán bộ lớp thường là người thông báo các thông tin từ thầy cô/đoàn/trường... đến các sinh viên khác. Tuy nhiên trong quá trình hoạt động và làm việc, các cán bộ lớp nhận thấy một số điều sau:

Vì nhiều lý do, đôi khi một số sinh viên có thể không nhận được thông báo và bỏ lỡ thông tin, ví dụ như không vào nhóm lớp, không đi học, tắt thông báo tin nhắn nhóm...

Mỗi khi đến ngày đi học hoặc buổi thi, đôi khi sinh viên đến muộn vì một số lý do nào đó như quên mất lịch, nhớ nhầm ngày, ngủ quên...

Mỗi khi đăng ký tín chỉ của một kỳ xong, sinh viên thường sẽ nhận được một tệp excel thời khóa biểu, tuy nhiên file này có ux/ui không quá tiện cho việc tra cứu. Thường các sinh viên sẽ phải tự tạo một thời khoá biểu dựa trên file đó.

Để giải quyết những vấn đề trên, một hệ thống hỗ trợ quản lý lớp học có thể được xây dựng. Hệ thống sẽ cho phép cán bộ lớp gửi thông báo đến các sinh viên khác trong lớp, hỗ trợ việc trực quan hóa thời khoá biểu dựa trên tệp excel được xuất ra từ trang web quản lý đào tạo của Trường Đại học Giao thông Vận tải ([qltd.utc.edu.vn](http://qltd.utc.edu.vn)) và đồng bộ hóa với tài khoản Google Calendar.

### 1.2.2 Khảo sát

Hiện tại chưa có ứng dụng nào hỗ trợ cho các công việc trên. Tuy nhiên, nếu tách nhỏ bài toán, thì chúng ta có một số ứng dụng đáp ứng được khía cạnh nào đó của nhu cầu:

- Import thông tin qua file: Chức năng này thường được xây dựng trong hệ thống nội bộ, vì để import thì file sẽ cần tuân thủ theo mẫu template nhất định.

⇒ Nhận xét: Hầu như các trường Đại học và Cao đẳng đều có triển khai chức năng cho phép import một lượng thông tin gì đó thông qua file excel này để giảm thiểu thời gian nhập liệu.

- Gửi thông báo khi có tin mới đến tất cả người dùng trong nhóm: Google Classroom, Facebook group,...
- Cho phép các người dùng trong nhóm trao đổi thông tin qua việc nhắn tin: Facebook, Zalo,...

Tuy nhiên, cũng có một số chức năng mà rất khó có thể tìm thấy được sự hỗ trợ từ các ứng dụng trên thị trường hiện tại như:

- Trực quan hoá thời khoá biểu dựa trên file nhà trường cung cấp (qltd.utc.edu.vn)
  - ⇒ Nhận xét: Do template của thời khóa biểu/tài liệu khác nhau nên thường chỉ triển khai nội bộ.
- Cho phép người dùng đăng ký tin nhắn nhắc nhở lịch học gửi tới điện thoại trong thời gian tự chọn căn cứ theo thời khoá biểu người dùng đã nhập.
  - ⇒ Nhận xét: Có thể sử dụng Google Calendar, tuy nhiên người dùng sẽ phải nhập sự kiện thủ công, rất tốn thời gian.

### 1.2.3 Mục tiêu đề tài

Đề tài được thực hiện nhằm tạo ra ứng dụng hỗ trợ sinh viên trong việc gửi/nhận thông báo và quản lý thời gian biểu. Thông qua ứng dụng, các cán bộ lớp có thể dễ dàng hơn trong việc gửi thông tin tới các sinh viên mà không lo bỏ sót.

Đề tài cũng giúp em giúp nâng cao kiến thức và tiếp cận việc xây dựng một ứng dụng thực tế, nâng cao khả năng nghiên cứu và làm việc độc lập.

## CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### 2.1 Mô tả tổng quan hệ thống

#### 2.1.1 Các tác nhân của hệ thống

Hệ thống được xây dựng để tách thành các nhóm nhỏ và người dùng tự quản lý. Các nhóm là độc lập và không có liên hệ gì với nhau.

Hệ thống sẽ có hai tác nhân chính:

- Thành viên (sinh viên trong lớp)
- Quản lý (thầy cô, cán bộ lớp,...)

Một người dùng có thể tham gia nhiều nhóm với các vai trò khác nhau. Một tài khoản có thể là người dùng của nhóm này, nhưng là quản lý của một nhóm khác.

#### 2.1.2 Các yêu cầu của hệ thống

##### - Đối với thành viên:

- + Tài khoản: người dùng có thể đăng nhập, đăng kí, đổi mật khẩu, lấy lại mật khẩu nếu quên, thay đổi thông tin cá nhân, xóa tài khoản
- + Tin tức: tin của người dùng sẽ được quản trị phê duyệt trước khi gửi thông báo tới tất cả mọi người trong nhóm
- + Thời khóa biểu: người dùng có thể import thời khóa biểu của nhà trường vào hệ thống để trực quan hóa, thêm sự kiện, đồng bộ với Google Calendar, bật/tắt thời gian nhắc nhở của các sự kiện trong Google Calendar
- + Trò chuyện: nhắn tin với người trong nhóm
- + Nhóm: người dùng có thể tạo nhóm mới với vai trò quản trị viên, mời người vào nhóm thông qua mã mời, rời khỏi nhóm

##### - Đối với quản lý: ngoại trừ các chức năng cơ bản của người dùng, quản lý sẽ có thêm một số chức năng:

- + Tin tức: quản lý có thể phê duyệt bài đăng
- + Quản lý: quản lý có thể import người dùng dựa trên file excel theo template mẫu, thay đổi vai trò của người dùng, xóa người dùng ra khỏi nhóm, tạo người dùng mới cho nhóm.

#### 2.1.3 Các chức năng của hệ thống

Thành viên có thể:

- Đăng nhập
- Đăng ký
- Hỗ trợ quên mật khẩu
- Thay đổi mật khẩu

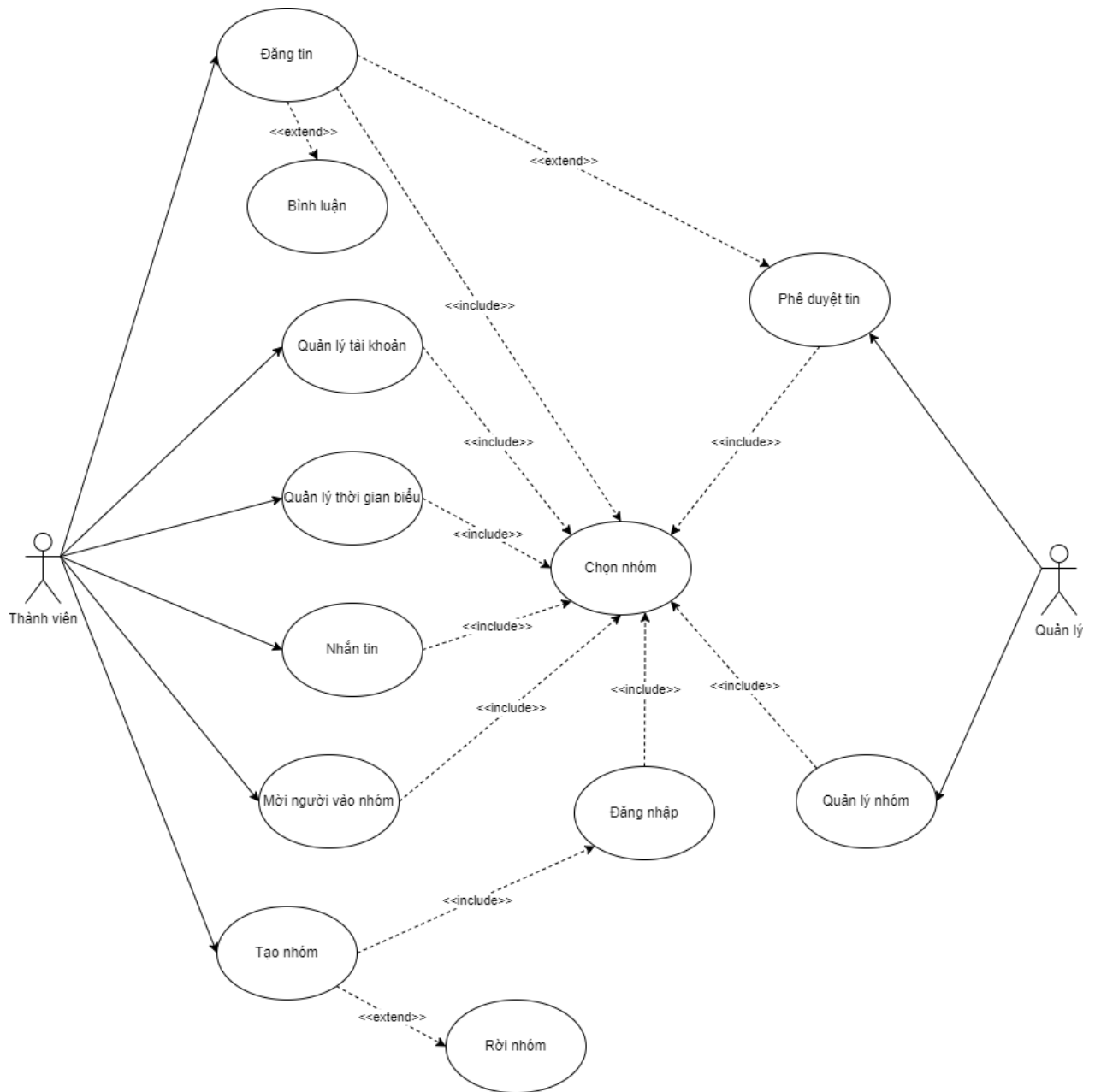
- Sửa đổi thông tin cá nhân
- Xoá tài khoản
- Mời người vào nhóm
- Rời khỏi nhóm
- Tạo nhóm mới
- Đăng tin mới
- Viết bình luận dưới tin
- Xem thông báo
- Nhận thông báo khi người khác có hoạt động tương tác với mình
- Trực quan hoá thời khóa biểu
- Đồng bộ thời gian biểu với Google Calendar
- Thêm, sửa, xoá sự kiện
- Bật/tắt thời gian nhắc nhở
- Nhắn tin với người trong nhóm
- Thêm người vào nhóm chat

Quản lý, kế thừa các chức năng cơ bản của thành viên, và có quyền sử dụng thêm một số chức năng khác:

- Import danh sách thành viên
- Đăng ký tài khoản cho người dùng khác
- Kiểm duyệt tin được đăng lên tường chung
- Xoá người dùng khỏi nhóm
- Thay đổi vai trò của người dùng
- Thay đổi các cài đặt của nhóm
- Xoá nhóm

## 2.2 Đặc tả các yêu cầu chức năng

### 2.2.1 Biểu đồ use-case tổng quát



**Hình 2.1.** Biểu đồ use-case tổng quát

**Bảng 2.1.** Mô tả use-case tổng quát

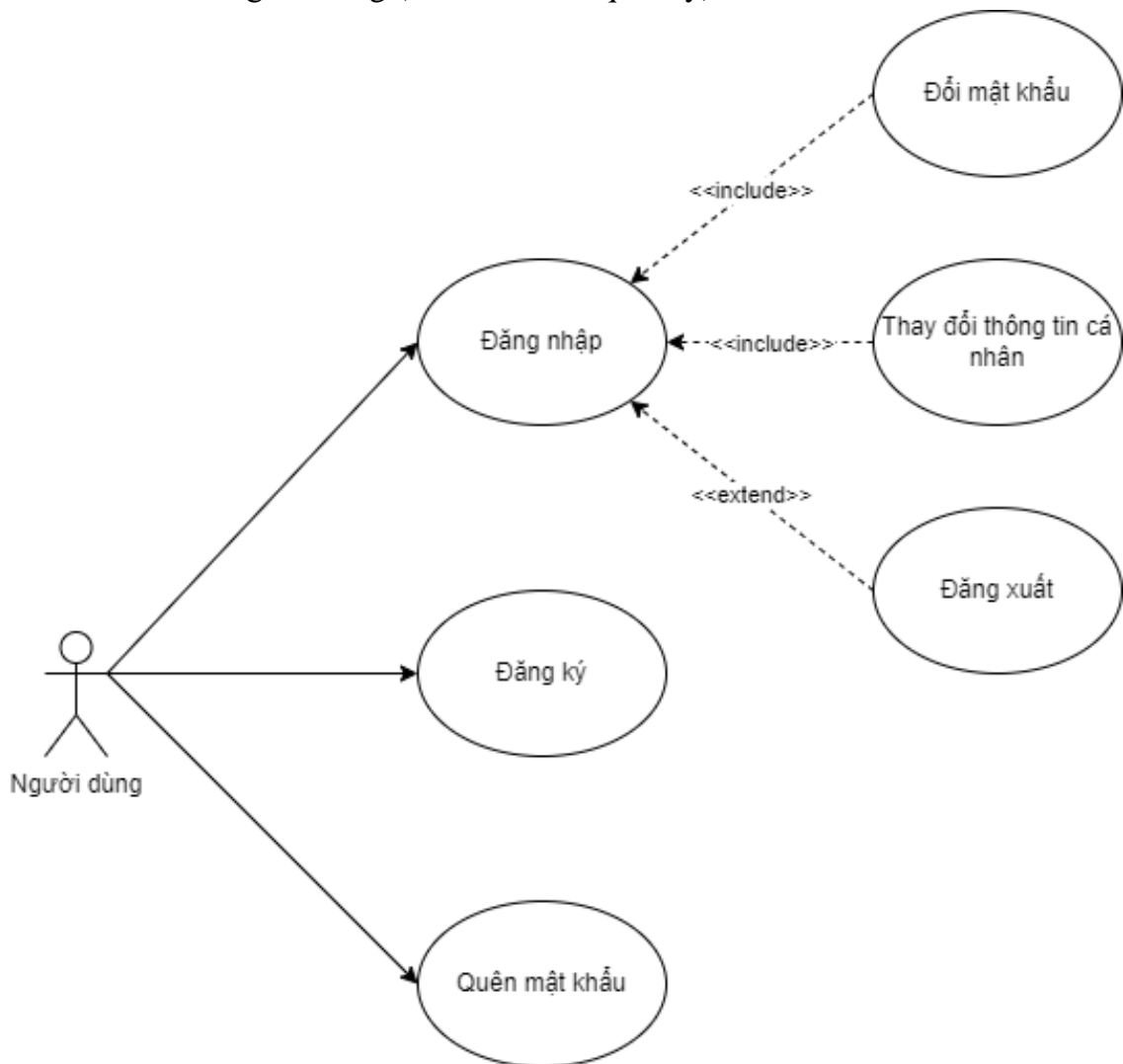
STT	Tên use-case	Mô tả
1	Đăng nhập	Người dùng truy cập vào ứng dụng
2	Chọn nhóm	Người dùng chọn nhóm để truy cập vào nhóm

3	Quản lý tài khoản	Người dùng quản lý tài khoản của họ (quên mật khẩu, sửa đổi thông tin cá nhân như tên hiển thị, ảnh đại diện, đổi mật khẩu)
4	Quản lý thời gian biểu	Người dùng quản lý thời gian biểu của họ (nhập file thời khoá biểu, đồng bộ hoá với Google Calendar, thêm/sửa/xoá sự kiện)
5	Nhắn tin	Người dùng tạo cuộc trò chuyện mới, nhắn tin với các thành viên khác hoặc mở rộng cuộc trò chuyện của họ
6	Đăng tin	Người dùng đăng tin tức mới tới thành viên trong nhóm
7	Bình luận	Người dùng bình luận về tin tức
8	Tạo nhóm	Người dùng tạo một nhóm mới với vai trò quản lý
9	Rời nhóm	Người dùng rời khỏi nhóm
10	Mời người vào nhóm	Người dùng gửi lời mời đến tài khoản khác
11	Phê duyệt tin	Người quản lý kiểm duyệt tin tức thành viên đăng lên
12	Quản lý nhóm	Người quản lý có quyền chỉnh sửa các cài đặt của nhóm (như cho phép thành viên mời người vào nhóm, tự rời nhóm, chỉnh sửa tên nhóm,...), thêm danh sách thành viên vào nhóm, xoá thành viên khỏi nhóm

### 2.2.2 Biểu đồ use-case chi tiết

- **Chức năng:** Quản lý tài khoản

**Tác nhân:** Người dùng (thành viên và quản lý)



**Hình 2.2.** Biểu đồ use-case chức năng quản lý tài khoản

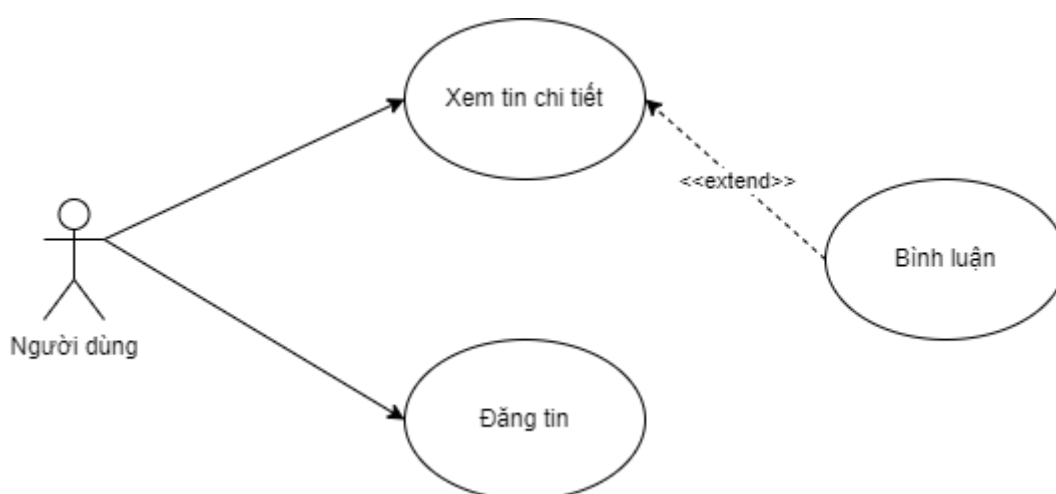
**Bảng 2.2.** Bảng mô tả use-case chức năng quản lý tài khoản

STT	Tên use-case	Mô tả
1	Đăng nhập	Người dùng đăng nhập vào ứng dụng
2	Đăng ký	Người dùng đăng ký tài khoản
3	Quên mật khẩu	Người dùng thực hiện quy trình xác minh để đăng nhập lại vào hệ thống khi quên mật khẩu
4	Thay đổi thông tin cá nhân	Người dùng thay đổi thông tin cá nhân như ảnh đại diện, tên hiển thị, email, số điện thoại,...
5	Đổi mật khẩu	Người dùng thay đổi mật khẩu
6	Đăng xuất	Người dùng đăng xuất khỏi hệ thống



- **Chức năng:** Tin tức

**Tác nhân:** Người dùng (thành viên và quản lý)



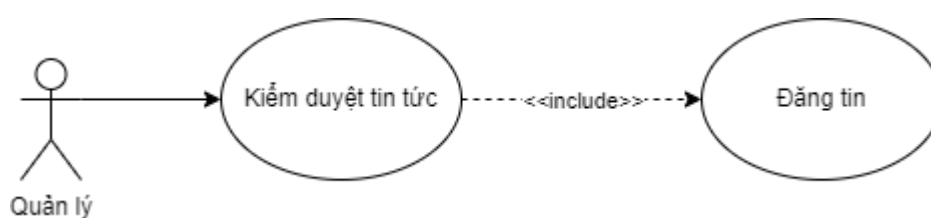
**Hình 2.3.** Biểu đồ use-case chức năng tin tức

**Bảng 2.3.** Mô tả use-case chức năng tin tức

STT	Tên use-case	Mô tả
1	Xem tin chi tiết	Người dùng xem tin chi tiết
2	Đăng tin	Người dùng đăng tin
3	Bình luận	Người dùng bình luận tin

- **Chức năng:** Kiểm duyệt tin tức

**Tác nhân:** Quản lý



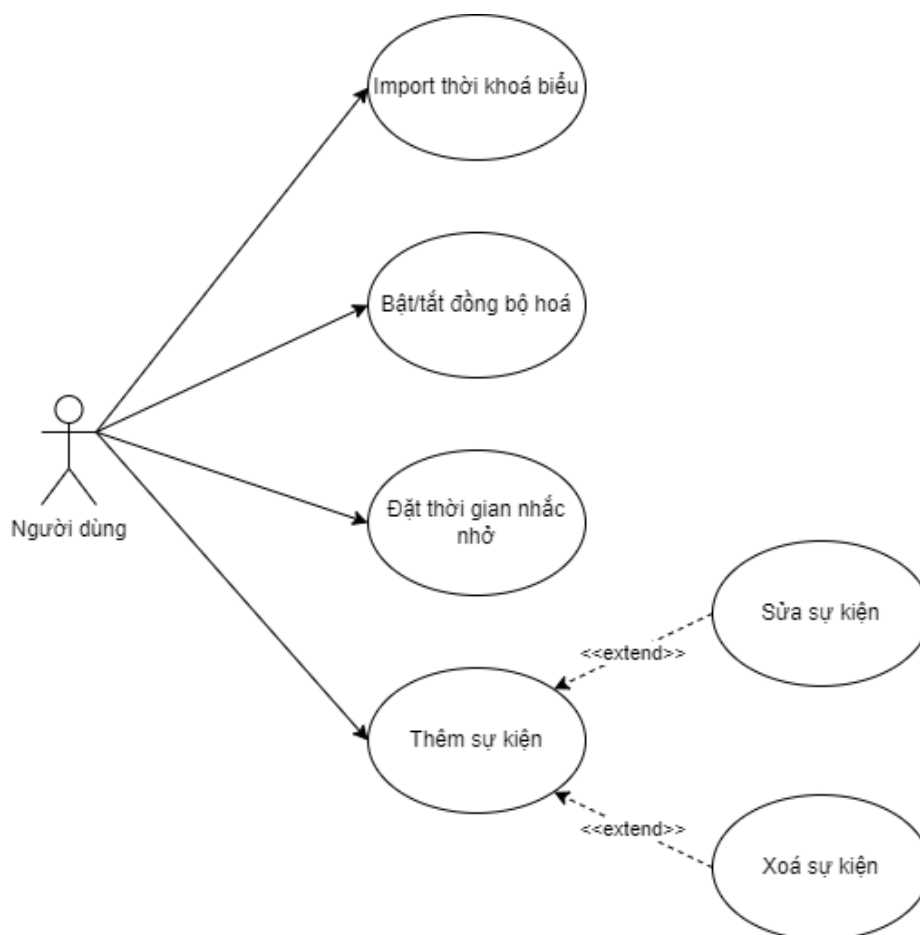
**Hình 2.4.** Biểu đồ use-case chức năng kiểm duyệt tin tức

**Bảng 2.4.** Mô tả use-case chức năng kiểm duyệt tin tức

STT	Tên use-case	Mô tả
1	Đăng tin	Người dùng đăng tin tức
2	Kiểm duyệt tin tức	Quản lý chấp nhận tin tức do người dùng đăng hoặc từ chối nó

- **Chức năng:** Quản lý thời gian biểu

**Tác nhân:** Người dùng (thành viên và quản lý)



**Hình 2.5.** Biểu đồ use-case chức năng thời gian biểu

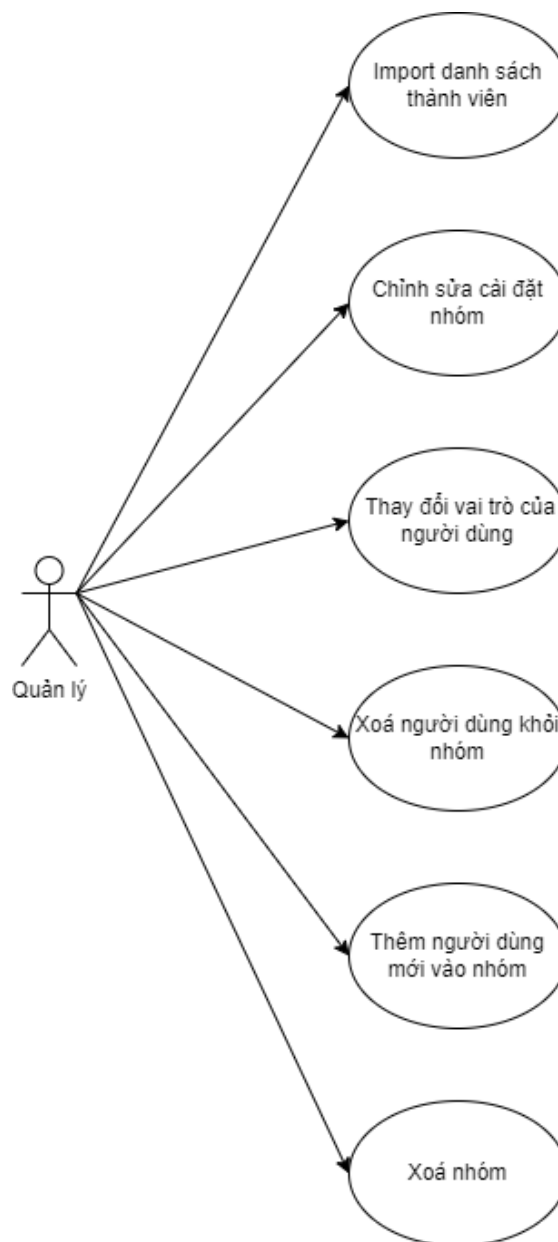
**Bảng 2.5.** Mô tả use-case chức năng thời gian biểu

STT	Tên use-case	Mô tả
1	Import thời khoá biểu	Người dùng import file thời khoá biểu từ trang Quản lý đào tạo của trường Đại học Giao thông vận tải để hệ thống trực quan hoá
2	Bật/tắt đồng bộ hoá	Người dùng bật/tắt đồng bộ hoá với Google Calendar.
3	Đặt thời gian nhắc nhở	Người dùng đặt thời gian nhắc nhở cho các sự kiện trong Google Calendar. Điều này chỉ có hiệu lực cho các sự kiện được thêm vào từ thời điểm bật đồng bộ hoá.
4	Thêm sự kiện	Người dùng thêm sự kiện vào hệ thống. Nếu bật đồng bộ hoá sẽ lưu vào Google Calendar.

5	Sửa sự kiện	Người dùng sửa sự kiện. Chỉ áp dụng được cho các sự kiện nội bộ (không phải sự kiện của Google Calendar)
6	Xoá sự kiện	Người dùng xoá sự kiện. Chỉ áp dụng được cho các sự kiện nội bộ (không phải sự kiện của Google Calendar)

- **Chức năng:** Quản lý nhóm

**Tác nhân:** Quản lý



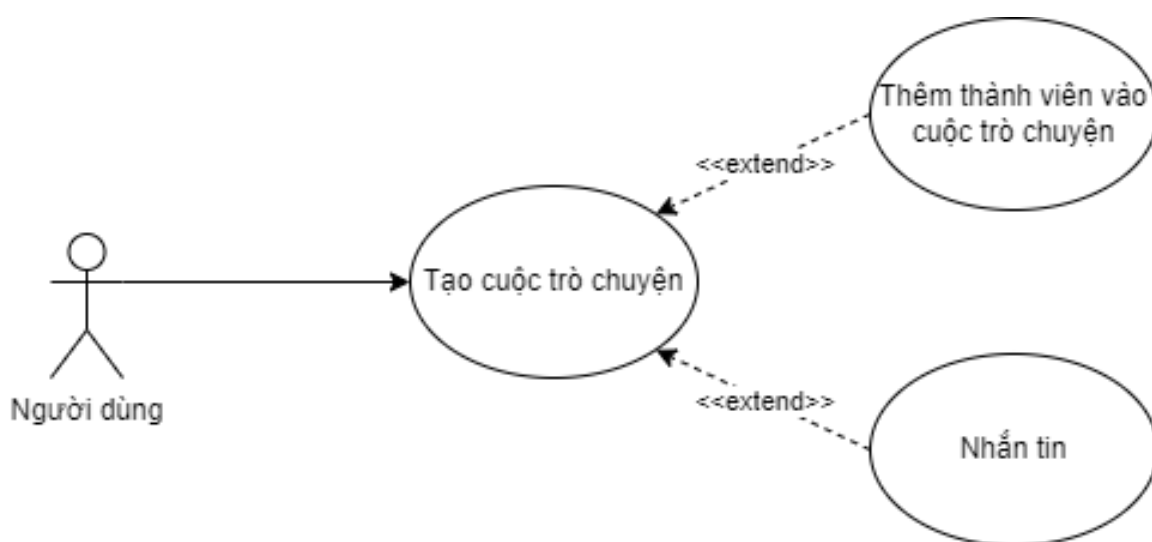
**Hình 2.6.** Biểu đồ use-case chức năng quản lý nhóm

**Bảng 2.6.** Mô tả use-case chức năng quản lý nhóm

STT	Tên use-case	Mô tả
1	Import danh sách thành viên	Quản lý import file danh sách các thành viên vào nhóm theo mẫu. Nếu thành viên chưa tồn tại, tạo tài khoản mới. Nếu thành viên tồn tại nhưng chưa vào nhóm, gửi thông báo và lời mời đến email. Nếu thành viên đã tồn tại, bỏ qua.
2	Chỉnh sửa cài đặt nhóm	Quản lý chỉnh sửa các cài đặt của nhóm như đổi tên, bật/tắt việc thành viên mời người vào nhóm,...
3	Thay đổi vai trò của người dùng	Quản lý thay đổi vai trò của người dùng trong nhóm
4	Xoá người dùng khỏi nhóm	Quản lý xoá người dùng ra khỏi nhóm
5	Thêm tài khoản mới vào nhóm	Quản lý tạo một tài khoản mới và thêm vào nhóm với vai trò mặc định là Thành viên. Mật khẩu sẽ trùng với tên tài khoản.
6	Xoá nhóm	Quản lý xoá nhóm

- **Chức năng:** Trò chuyện

**Tác nhân:** Người dùng (thành viên và quản lý)

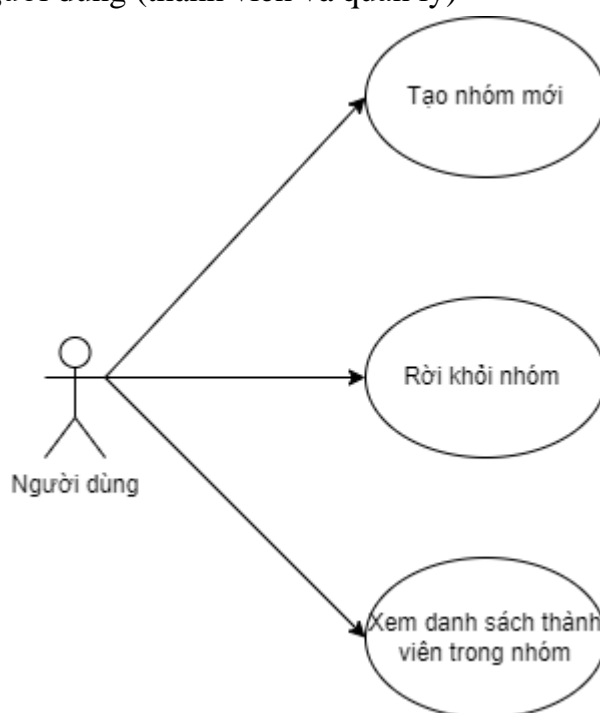
**Hình 2.7.** Biểu đồ use-case chức năng nhắn tin

**Bảng 2.7.** Mô tả use-case chức năng nhắn tin

STT	Tên use-case	Mô tả
1	Tạo cuộc trò chuyện	Người dùng tạo cuộc trò chuyện với thành viên trong nhóm
2	Nhắn tin	Người dùng nhắn tin trong cuộc trò chuyện
3	Thêm thành viên vào cuộc trò chuyện	Người dùng thêm thành viên vào cuộc trò chuyện

- **Chức năng:** Nhóm

**Tác nhân:** Người dùng (thành viên và quản lý)

**Hình 2.8.** Biểu đồ use-case chức năng nhóm**Bảng 2.8.** Mô tả use-case chức năng nhóm

STT	Tên use-case	Mô tả
1	Tạo nhóm mới	Người dùng tạo nhóm mới
2	Rời khỏi nhóm	Người dùng rời khỏi nhóm (nếu quản lý bật chế độ cho phép tự rời nhóm)
3	Xem danh sách thành viên trong nhóm	Người dùng xem danh sách thành viên trong nhóm

## 2.3 Phân tích thiết kế cơ sở dữ liệu

### 2.3.1 Phân tích cơ sở dữ liệu

Theo mô tả và đặc tả các yêu cầu của bài toán được nêu ở các mục trước đây, em có thể xác định được các bảng sử dụng trong hệ quản trị cơ sở dữ liệu **MySQL**:

- Bulletins: bao gồm các thông tin về tin tức được người dùng đăng tải.
- Comments: bao gồm các bình luận của người dùng dưới các tin.
- Events: bao gồm các sự kiện trong thời gian biểu của người dùng.
- Groups: bao gồm các nhóm.
- Notifications: bao gồm các thông báo.
- Roles: bao gồm các vai trò.
- Timetables: bao gồm thông tin chung về thời gian biểu.
- Users: bao gồm các thông tin về người dùng và tài khoản.

Ngoài ra còn một số bảng phụ khác để hỗ trợ liên kết giữa các bảng.

Tuy nhiên, khi phân tích về các chức năng, em nhận thấy chức năng nhắn tin có một số yêu cầu đặc thù như:

- Linh hoạt về mô hình dữ liệu: tin nhắn có nhiều kiểu như text, image, link,...
- Hiệu suất cao với các thao tác đọc ghi nhanh.
- Khả năng lưu trữ và truy xuất dữ liệu lớn.

Các cơ sở dữ liệu quan hệ như MySQL vẫn có thể dùng để lưu trữ dữ liệu của chức năng này, tuy nhiên em nhận thấy rằng việc sử dụng các cơ sở dữ liệu phi quan hệ sẽ giúp việc phát triển và mở rộng hệ thống thuận tiện và dễ dàng hơn khi số lượng bản ghi trong cơ sở dữ liệu ngày một tăng.

Vì vậy, em quyết định sử dụng cơ sở dữ liệu phi quan hệ MongoDB cho chức năng này.

Danh sách các tập hợp trong cơ sở dữ liệu **MongoDB**:

- Groups: bao gồm thông tin về nhóm và tham chiếu tới tập hợp Users cho thông tin những người ở trong nhóm, có mã định danh là trace id của Groups trong MySQL.
- Chats: bao gồm thông tin về cuộc hội thoại, tham chiếu tới Users cho thông tin những người tham gia, Groups cho thông tin ở trong nhóm nào và Messages cho thông tin những tin nhắn trong cuộc hội thoại.
- Messages: bao gồm thông tin về tin nhắn, tham chiếu tới Users cho thông tin người tạo.
- User: bao gồm thông tin về người dùng, có mã định danh là trace id của Users trong MySQL

### 2.3.2 Thiết kế cơ sở dữ liệu

MySQL sẽ được dùng để lưu trữ dữ liệu của tất cả các chức năng trong hệ thống, ngoại trừ chức năng nhắn tin:

#### - Bulletins:

Bảng Bulletins dùng để lưu các tin được đăng trong hệ thống.

**Bảng 2.9.** Danh sách các trường trong bảng Bulletins

STT	Thuộc tính	Kiểu dữ liệu	Mô tả	Ghi chú
1	id	varchar	mã định danh	
2	content	longtext	nội dung	chứa html dưới dạng text
3	createdDate	datetime	ngày đăng	
4	createdBy	varchar	người đăng	
5	lastModifiedDate	datetime	ngày sửa cuối cùng	
6	lastModifiedBy	varchar	người sửa cuối cùng	
7	groupId	varchar	mã nhóm	
8	approved	tinyint	được duyệt	kiểu boolean

#### - Comments:

Bảng Comments lưu thông tin các bình luận của tin khác nhau.

**Bảng 2.10.** Danh sách các trường trong bảng Comments

STT	Thuộc tính	Kiểu dữ liệu	Mô tả	Ghi chú
1	id	varchar	mã định danh	
2	postId	varchar	mã tin	
3	content	text	nội dung	
4	createdDate	datetime	ngày đăng	
5	createdBy	varchar	người đăng	

**- Events:**

Bảng Events lưu thông tin các sự kiện trong thời gian biểu. Có hai loại sự kiện là sự kiện một lần và sự kiện lặp lại.

**Bảng 2.11.** Danh sách các trường trong bảng Events

STT	Thuộc tính	Kiểu dữ liệu	Mô tả	Ghi chú
1	id	int	mã định danh	
2	timetableId	varchar	mã thời gian biểu	
3	day	int	thứ trong tuần	
4	from	datetime	từ ngày	sử dụng cho sự kiện không lặp lại
5	to	datetime	đến ngày	sử dụng cho sự kiện không lặp lại
6	isLoopPerDay	tinyint	là sự kiện lặp lại theo thời gian hay không	kiểu boolean
7	periodStart	time	thời gian bắt đầu cho sự kiện lặp	sử dụng cho sự kiện lặp
8	periodEnd	time	thời gian kết thúc cho sự kiện lặp	sử dụng cho sự kiện lặp
9	title	text	tiêu đề	
10	description	text	mô tả	
11	location	text	địa chỉ	
12	subject	text	môn học	dùng cho thời khoá biểu
13	subjectCode	text	mã môn học	dùng cho thời khoá biểu
14	subjectClass	text	tên lớp học	dùng cho thời khoá biểu
15	credit	int	credit	dùng cho thời khoá biểu
16	category	text	thư mục	(1) Timetable (2) User (3) Google (4) Unclassify



**- Groups:**

Bảng Groups lưu thông tin về các nhóm của hệ thống.

**Bảng 2.12.** Danh sách các trường trong bảng Groups

STT	Thuộc tính	Kiểu dữ liệu	Mô tả	Ghi chú
1	id	varchar	mã định danh	
2	name	varchar	tên	
3	allowInvite	tinyint	cho phép mời người vào nhóm	kiểu boolean
4	allowOut	tinyint	cho phép thành viên tự rời nhóm	kiểu boolean

**- Notifications:**

Bảng Notifications lưu thông tin về các thông báo gửi đến người dùng khi có những người dùng khác thực hiện tương tác với họ.

**Bảng 2.13.** Danh sách các trường trong bảng Notifications

STT	Thuộc tính	Kiểu dữ liệu	Mô tả	Ghi chú
1	id	varchar	mã định danh	
2	content	varchar	nội dung	
3	action	varchar	loại thao tác	(1) NewPost, (2) PendingPost, (3) AcceptPost, (4) RejectPost, (5) ReplyPost, (6) Mention, (7) InviteToGroup, (8) KickFromGroup, (9) ChangeRole
4	range	varchar	phạm vi thông báo	(1) User (2) Group (3) All
5	groupId	varchar	mã nhóm	sử dụng trong các thông báo trong phạm vi nhóm, có thể null

6	userId	varchar	mã người dùng	sử dụng cho các thông báo với người dùng, có thể null
7	createdDate	datetime	ngày đăng	
8	createdBy	varchar	người đăng	
9	objectId	varchar	mã đính kèm	lưu mã của đối tượng liên quan đến thông báo, có thể null
10	isSeen	tinyint	đã đọc hay chưa	

**- Roles:**

Bảng Roles lưu thông tin về các vai trò trong hệ thống.

**Bảng 2.14.** Danh sách các trường trong bảng Roles

STT	Thuộc tính	Kiểu dữ liệu	Mô tả	Ghi chú
1	id	varchar	mã định danh	
2	name	varchar	tên	
3	normalizeName	varchar	chuẩn hoá tên	
4	concurencyStamp	varchar	concurencyStamp	dùng để xác định phiên bản hiện tại của dữ liệu

**- Timetables:**

Bảng Timetables lưu các thông tin chung về thời gian biểu.

**Bảng 2.15.** Danh sách các trường trong bảng Timetables

STT	Thuộc tính	Kiểu dữ liệu	Mô tả	Ghi chú
1	id	varchar	mã định danh	
2	groupId	varchar	mã nhóm	
3	url	varchar	đường dẫn file import	có thể null
4	createdDate	datetime	concurencyStamp	dùng để xác định phiên bản hiện tại của dữ liệu
5	createdBy	varchar	người tạo	
6	isSynchronize	tinyint	trong trạng thái đồng bộ hoá	kiểu boolean
7	remindTime	int	nhắc nhở trước thời gian	tính theo phút

				-1 là tắt nhắc nhở
--	--	--	--	--------------------

**- Users:**

Bảng Users lưu các thông tin về tài khoản và người dùng.

**Bảng 2.16.** Danh sách các trường trong bảng User

STT	Thuộc tính	Kiểu dữ liệu	Mô tả	Ghi chú
1	id	varchar	mã định danh	
2	displayName	varchar	tên hiển thị	
3	userName	varchar	tên đăng nhập	
4	nomalizedUserName	varchar	tên đăng nhập chuẩn hoá	dùng để xác định phiên bản hiện tại của dữ liệu
5	email	varchar	email	
6	normalizedEmail	varchar	email chuẩn hoá	
7	emailConfirmed	tinyint	email đã xác nhận hay chưa	kiểu boolean
8	passwordHash	longtext	chuỗi mật khẩu mã hoá	
9	avatar	text	ảnh đại diện	lưu đường dẫn, có thể null
10	securityStamp	longtext	securityStamp	
11	cocurrencyStamp	longtext	cocurrencyStamp	
12	phoneNumber	longtext	số điện thoại	có thể null
13	phoneNumberConfirmed	tinyint	số điện thoại đã xác nhận hay chưa	kiểu boolean
14	twoFactorEnabled	tinyint	bảo vệ 2 lớp bật/tắt	kiểu boolean
15	lockoutEnd	datetime	thời gian mở khoá tài khoản	
16	lockoutEnabled	tinyint	khóa tài khoản bật/tắt	kiểu boolean
17	accessFailedCount	int	số lần đăng nhập thất bại	

MongoDB được dùng để lưu trữ dữ liệu của chức năng nhắn tin:

**- Groups:**

Bảng Groups lưu một số thông tin cần thiết về nhóm.

**Bảng 2.17.** Danh sách các trường trong tập hợp Groups

STT	Thuộc tính	Kiểu dữ liệu	Mô tả	Ghi chú
1	id	binary	mã định danh	
2	name	string	tên nhóm	
3	users	array	danh sách người thuộc nhóm	chứa object là User có thể null

**- Chats:**

Bảng Chats lưu thông tin về cuộc hội thoại, chứa bình luận và những người dùng tham gia.

**Bảng 2.18.** Danh sách các trường trong tập hợp Chat

STT	Thuộc tính	Kiểu dữ liệu	Mô tả	Ghi chú
1	id	binary	mã định danh	
2	name	string	tên cuộc hội thoại	
3	avatar	string	ảnh đại diện	lưu đường dẫn, có thể null
4	createdDate	date	ngày tạo	
5	createdBy	object	người tạo	object là User
6	users	array	danh sách người tham gia	chứa object là User
7	messages	array	danh sách tin nhắn	chứa object là Message
8	groups	object	nhóm của cuộc hội thoại	object là Group

**- Messages:**

Bảng Messages lưu thông tin về các tin nhắn trong hội thoại. Các kiểu tin nhắn được chấp nhận hiện tại là chuỗi, đường dẫn và ảnh.

**Bảng 2.19.** Danh sách các trường trong tập hợp Messages

STT	Thuộc tính	Kiểu dữ liệu	Mô tả	Ghi chú
1	id	binary	mã định danh	
2	text	string	nội dung	
3	isImage	boolean	có phải ảnh	
4	imageUrl	string	đường dẫn ảnh	có thể null
5	createdAt	date	ngày tạo	
6	createdBy	object	người tạo	object là User
7	updatedAt	date	thời gian cập nhật lần cuối	có thể null
8	isDeleted	boolean	đã xóa	

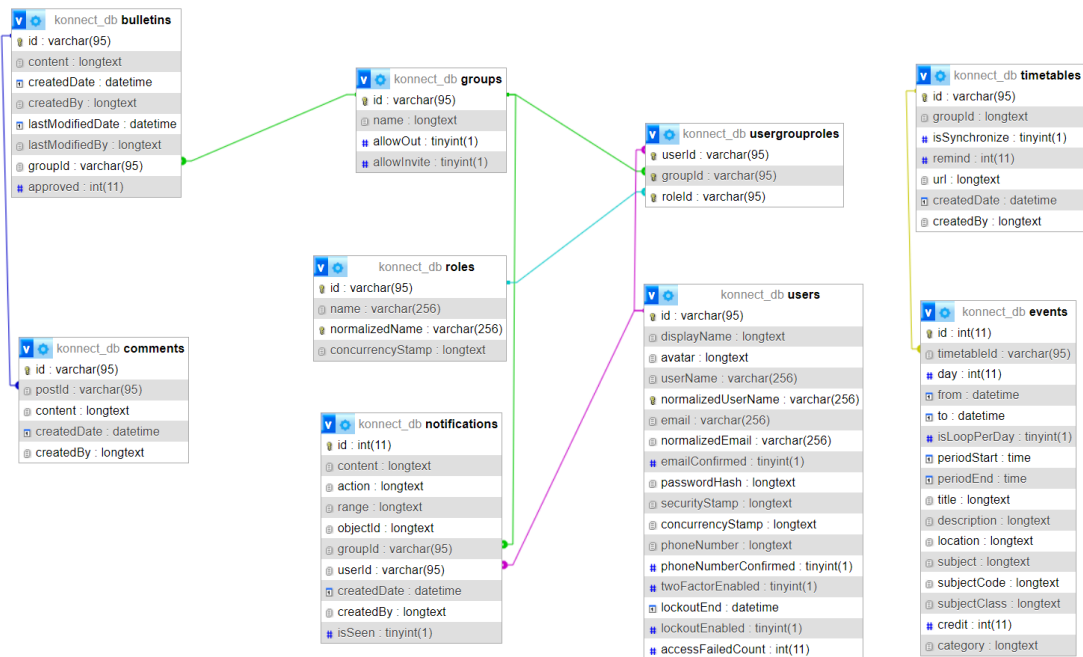
**- Users:**

Bảng Users lưu một số thông tin cần thiết của người dùng trong hội thoại.

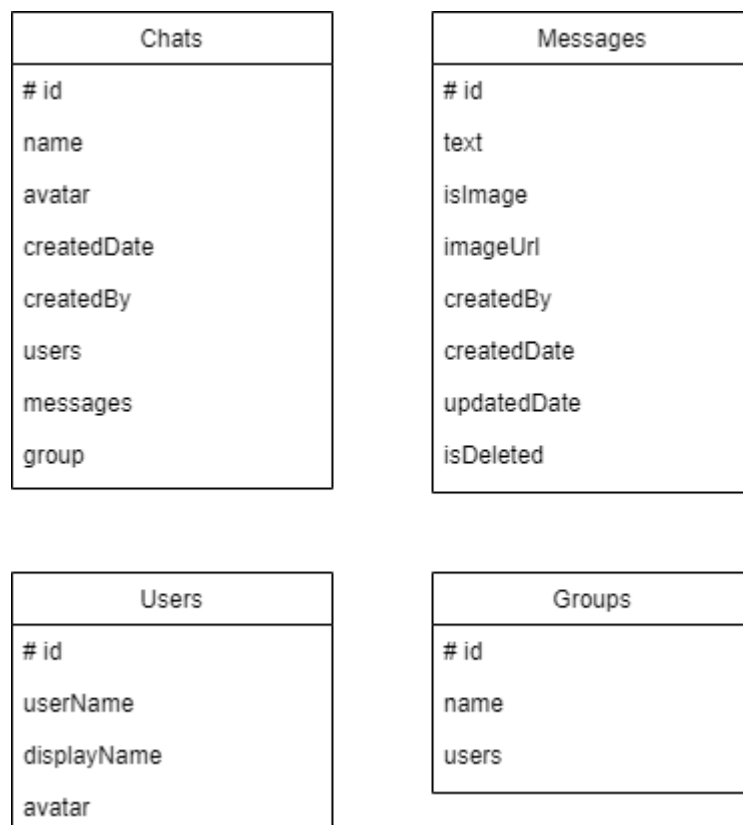
**Bảng 2.20.** Danh sách các trường trong tập hợp Users

STT	Thuộc tính	Kiểu dữ liệu	Mô tả	Ghi chú
1	id	binary	mã định danh	
2	userName	string	tên đăng nhập	
3	displayName	string	tên hiển thị	
4	avatar	string	ảnh đại diện	đường dẫn ảnh

### 2.3.3 Mô hình quan hệ



**Hình 2.9.** Mô hình quan hệ các bảng trong hệ quản trị cơ sở dữ liệu MySQL



**Hình 2.10.** Mô hình quan hệ các tập hợp trong hệ quản trị cơ sở dữ liệu MongoDB

## CHƯƠNG 3: XÂY DỰNG CHƯƠNG TRÌNH

### 3.1 Cài đặt chương trình

#### 3.1.1 Các thư viện hỗ trợ

Để xây dựng ứng dụng, em sử dụng các công nghệ và thư viện hỗ trợ:

##### 3.1.1.1 Back-end

Sử dụng framework ASP.NET Core API với phiên bản .NET 6.0 cùng các thư viện:

- NPOI: đọc và ghi các tệp Microsoft Office, bao gồm Excel và Word.
- Pomelo.EntityFrameworkCore.MySql: cung cấp EntityFrameworkCore cho MySQL, cho phép kết nối và tương tác với cơ sở dữ liệu MySQL.
- AutoMapper: công cụ ánh xạ đối tượng, giúp chuyển đổi dữ liệu giữa các đối tượng mà không cần viết mã thủ công.
- MailKit: thư viện gửi và nhận email trong .NET, hỗ trợ các giao thức SMTP, POP3 và IMAP.
- MimeKit: thư viện xử lý MIME, hỗ trợ tạo và phân tích cú pháp các thông điệp email.
- Newtonsoft.Json: thư viện phổ biến để xử lý JSON trong .NET, hỗ trợ chuyển đổi giữa các đối tượng .NET và chuỗi JSON.
- EntityFrameworkCore: ORM của Microsoft cho .NET, giúp tương tác với cơ sở dữ liệu một cách dễ dàng và mạnh mẽ.
- SignalR: thư viện cho phép xây dựng các ứng dụng thời gian thực, hỗ trợ giao tiếp hai chiều giữa máy chủ và máy khách.
- MongoDB.Driver: trình điều khiển chính thức cho MongoDB trong .NET, cho phép tương tác với cơ sở dữ liệu NoSQL MongoDB.
- MediatR: thư viện hỗ trợ việc triển khai mẫu thiết kế Mediator, giúp quản lý luồng dữ liệu và giao tiếp giữa các thành phần trong ứng dụng.
- Microsoft.AspNetCore.Identity: hệ thống quản lý nhận dạng và xác thực người dùng trong các ứng dụng ASP.NET Core.
- Microsoft.AspNetCore.Authentication.JwtBearer: hỗ trợ xác thực bằng JSON Web Tokens (JWT) trong các ứng dụng ASP.NET Core, giúp bảo mật các API và dịch vụ web.

##### 3.1.1.2 Front-end

Sử dụng Vue3 cùng các package hỗ trợ:

- Ck5Editor: trình soạn thảo văn bản WYSIWYG mạnh mẽ cho web, hỗ trợ chỉnh sửa và định dạng văn bản trực quan.

- Bootstrap: thư viện CSS phổ biến giúp phát triển giao diện web nhanh chóng và dễ dàng với các thành phần UI sẵn có và responsive.
- ElementUI+: thư viện UI cho Vue.js, cung cấp các thành phần giao diện người dùng đẹp mắt và dễ sử dụng, hỗ trợ xây dựng ứng dụng web hiệu quả.
- Mitt: thư viện sự kiện nhẹ cho JavaScript, giúp quản lý và xử lý sự kiện dễ dàng trong ứng dụng.
- SignalR: thư viện cho phép xây dựng các ứng dụng thời gian thực, hỗ trợ giao tiếp hai chiều giữa máy chủ và máy khách.
- Axios: thư viện HTTP client cho JavaScript, giúp thực hiện các yêu cầu HTTP dễ dàng và hỗ trợ các promise.
- Vue Cal: thư viện lịch cho Vue.js, cung cấp các thành phần lịch tùy biến và tương tác, giúp hiển thị và quản lý các sự kiện.
- VuePluginLoadScript: plugin cho Vue.js, giúp tải động các tập lệnh (script) JavaScript vào ứng dụng một cách dễ dàng.

### 3.1.1.3 Cơ sở dữ liệu

- MySQL: hệ quản trị cơ sở dữ liệu quan hệ phổ biến, lưu trữ hầu hết các thông tin của hệ thống.
- MongoDB: hệ quản trị cơ sở dữ liệu NoSQL hướng tài liệu, dùng để lưu trữ và quản lý dữ liệu cho chức năng tin nhắn.

## 3.1.2 Cài đặt chương trình

### 3.1.2.1 Chức năng tài khoản

Người dùng cần đăng ký để có tài khoản truy cập vào hệ thống. Có hai trường hợp:

- + Người dùng có mã mời: khi đăng ký thành công, người dùng sẽ được thêm vào nhóm tương ứng với mã mời dưới vai trò thành viên bình thường.

- + Người dùng không có mã mời: một nhóm mới sẽ được tạo ra cho người dùng với vai trò Quản lý.

Sau khi đăng ký xong, người dùng sẽ nhận được thông báo và chuyển hướng về trang đăng nhập. Người dùng có thể dùng tài khoản đã đăng ký để đăng nhập bình thường.

Khi người dùng đăng nhập, hệ thống sẽ kiểm tra tên đăng nhập và mật khẩu của người dùng có chính xác không. Nếu có thì một đoạn mã JWT sẽ được sinh ra, dùng cho việc xác minh khi client và server giao tiếp. Khi phía client gửi thông tin thì đoạn mã này sẽ được đính kèm trong header.



Access			^
POST	/login		✓ 🔒
POST	/login/{groupId}		✓ 🔒
POST	/register		✓ 🔒
POST	/password/forgot		✓ 🔒
POST	/join/{token}		✓ 🔒

**Hình 3.1.** Hình ảnh các api của chức năng tài khoản

Đôi khi, người dùng có thể quên mất mật khẩu của tài khoản. Khi đó, hệ thống sẽ hỗ trợ bằng cách hỏi thông tin email của tài khoản. Phía server thực hiện việc sinh mật khẩu mới, đặt lại mật khẩu cho tài khoản này và gửi một bức thư đến địa chỉ mail chứa mật khẩu mới.

Người dùng truy cập vào hòm thư cá nhân để có thể lấy mật khẩu mới. Nếu muốn đổi lại mật khẩu, sau khi người dùng đăng nhập xong có thể truy cập vào trang cá nhân để đổi mật khẩu.

### 3.1.2.2 Chức năng nhóm

Sau khi đăng nhập, người dùng đến với giao diện chọn nhóm. Hệ thống sẽ đọc thông tin các nhóm từ cơ sở dữ liệu và hiển thị ra màn hình.

Người dùng có thể tham gia nhóm thông qua nhập chính xác id của nhóm. Hệ thống sẽ đối chiếu và nếu chính xác thì người dùng sẽ được thêm vào nhóm với vai trò Thành viên.

Group			^
GET	/group		✓ 🔒
POST	/group		✓ 🔒
POST	/group/join/{id}		✓ 🔒
DELETE	/group/out/{id}		✓ 🔒
POST	/group/invite		✓ 🔒

**Hình 3.2.** Hình ảnh các api của chức năng nhóm

### 3.1.2.3 Chức năng bảng tin

Tin tức sẽ có 3 trạng thái là Reject, Pending và Approved. Khi người dùng truy cập vào bảng tin, phía client gọi đến server để lấy được các tin đang ở trạng thái Approved và hiển thị ra màn hình.

Nếu người dùng đăng một tin, khi client gửi request thì phía server sẽ kiểm tra vai trò của người dùng trong nhóm hiện tại. Với vai trò người dùng, tin sẽ được đặt trạng thái Pending và gửi thông báo/email đến cho quản lý. Với vai trò quản lý, tin sẽ được đặt trạng thái Approved và gửi thông báo/email đến tất cả người dùng trong nhóm, còn

không thì tin sẽ được đặt trạng thái Reject và gửi thông báo kèm theo lời nhắn của quản lý phê duyệt đến người dùng.

Khi người dùng bình luận, phía client sẽ gửi request đến cho server rồi hiển thị bình luận. Phía server sẽ lưu bình luận vào cơ sở dữ liệu rồi tạo một thông báo cho chủ bài đăng.

Bulletin			^
POST	/bulletin		✓ 🔒
GET	/bulletin		✓ 🔒
POST	/bulletin/{postId}/state		✓ 🔒
GET	/bulletin/{id}		✓ 🔒
POST	/bulletin/{id}/comment		✓ 🔒

**Hình 3.3.** Hình ảnh các api của chức năng bảng tin

### 3.1.2.4 Chức năng thời gian biểu

Phía client dùng thư viện Vue Cal để hiển thị thời khoá biểu cho người dùng.

Để đọc được file import thời khoá biểu, hệ thống thực hiện các bước sau:

- Phía server sẽ nhận được file người dùng gửi lên và kiểm tra nó có phải file excel không. Nếu không, gửi thông báo cho người dùng.
- Hệ thống sẽ sử dụng NPOI library để đọc file excel thời khoá biểu và chuyển dữ liệu trong đó thành một DataTable. Sau đó, hệ thống sẽ xử lý từng cell trong DataTable thành dữ liệu phù hợp với thực thể và lưu vào cơ sở dữ liệu.
- Transaction được bật trong quá trình xử lý và lưu thực thể vào cơ sở dữ liệu. Nếu có lỗi xảy ra thì các bản ghi sẽ được rollback và gửi thông báo về cho người dùng.
- Sau khi đọc file thành công, server sẽ gửi thông báo về cho client. Client gọi đến api lấy thời gian biểu và xử lý nó thành dạng danh sách sự kiện mà Vue Cal có thể nhận được.

Timetable			^
GET	/timetable		✓ 🔒
DELETE	/timetable		✓ 🔒
POST	/timetable/synchronize		✓ 🔒
POST	/timetable/remind/{time}		✓ 🔒
POST	/timetable/event		✓ 🔒

**Hình 3.4.** Hình ảnh các api của chức năng thời gian biểu

Trong trường hợp người dùng muốn kết hợp sự kiện trong tài khoản Google Calendar của họ với thời khoá biểu hiện tại, phía client sẽ cần kiểm tra đã có quyền truy cập từ Google chưa. Nếu chưa có, một request sẽ được gửi đến Google để người dùng xác minh.

Sau khi người dùng xác minh xong, phía client sẽ gọi đến api của Google để lấy toàn bộ danh sách các sự kiện. Sau đó, nó sẽ xử lý các sự kiện thành object mà Vue Cal có thể nhận được rồi đẩy vào danh sách sự kiện của Vue Cal.

### 3.1.2.5 Chức năng nhắn tin

Chức năng này sử dụng SignalR để thực hiện việc nhận tin nhắn mà không cần tải lại trang. Phía server cung cấp một hub và client đăng ký với hub đó. Mỗi client có một connection đại diện. Khi một người dùng nhắn tin, client sẽ gọi đến phương thức tương ứng trong hub và hub sẽ gửi phản hồi dựa theo các connection.

Sau khi nhận được phản hồi từ hub, client sẽ hiển thị tin nhắn.

```
public async Task SendMessage(string chatId, string obj)
{
    var message = JsonConvert.DeserializeObject<AddMessageRequest>(obj);
    var chat = await _unitOfWork.Chats.GetAsync(chatId);
    var mess = new Message()
    {
        Text = message.Text,
        IsImage = message.IsImage,
        ImageUrl = message.ImageUrl,
        CreatedDate = DateTime.Now,
        CreatedBy = await _unitOfWork.Users.GetAsync(message.CreatedBy),
        IsDeleted = false
    };
    _ = _unitOfWork.Messages.CreateAsync(mess);
    if (chat.Messages == null)
    {
        chat.Messages = new List<Message>();
    }
    chat.Messages.Add(mess);
    _ = _unitOfWork.Chats.UpdateAsync(chatId, chat);
    await Clients.Group(chatId).SendAsync(ChatAction.ReceiveMessage,
        JsonSerializer.SerializeObjectCamelCase(mess));
}
```

**Hình 3.5.** Hình ảnh phương thức gửi phản hồi cho client từ phía server

### 3.1.2.6 Chức năng quản lý

Quản lý có thể nhanh chóng tạo tài khoản cho các thành viên trong lớp thông qua việc import file excel theo mẫu đã cung cấp. Ban đầu, phía server sẽ nhận được file người dùng gửi lên và kiểm tra nó có phải file excel không. Nếu không, gửi thông báo cho người dùng.

1 reference

```

public static System.Data.DataTable ConvertUserTemplateToDataTable(ISheet sheet)
{
    var dataTable = new DataTable(sheet.SheetName);

    // write the header row
    var headerRow = sheet.GetRow(0);
    foreach (var headerCell in headerRow)
    {
        dataTable.Columns.Add(headerCell.ToString());
    }

    // write the record
    for (int i = 1; i < sheet.PhysicalNumberOfRows; i++)
    {
        var sheetRow = sheet.GetRow(i);
        if (sheetRow == null) continue;
        var dtRow = dataTable.NewRow();
        dtRow.ItemArray = dataTable.Columns
            .Cast<DataColumn>()
            .Select(c => sheetRow.GetCell(c.Ordinal,
                MissingCellPolicy.CREATE_NULL_AS_BLANK)?.ToString() ?? String.Empty)
            ?.ToArray();
        dataTable.Rows.Add(dtRow);
    }
    return dataTable;
}

```

**Hình 3.6.** Hình ảnh phương thức đọc file excel người dùng

Sau đó, hệ thống sẽ sử dụng thư viện NPOI để đọc file excel người dùng và chuyển dữ liệu trong đó thành một DataTable. Có 3 trường hợp:

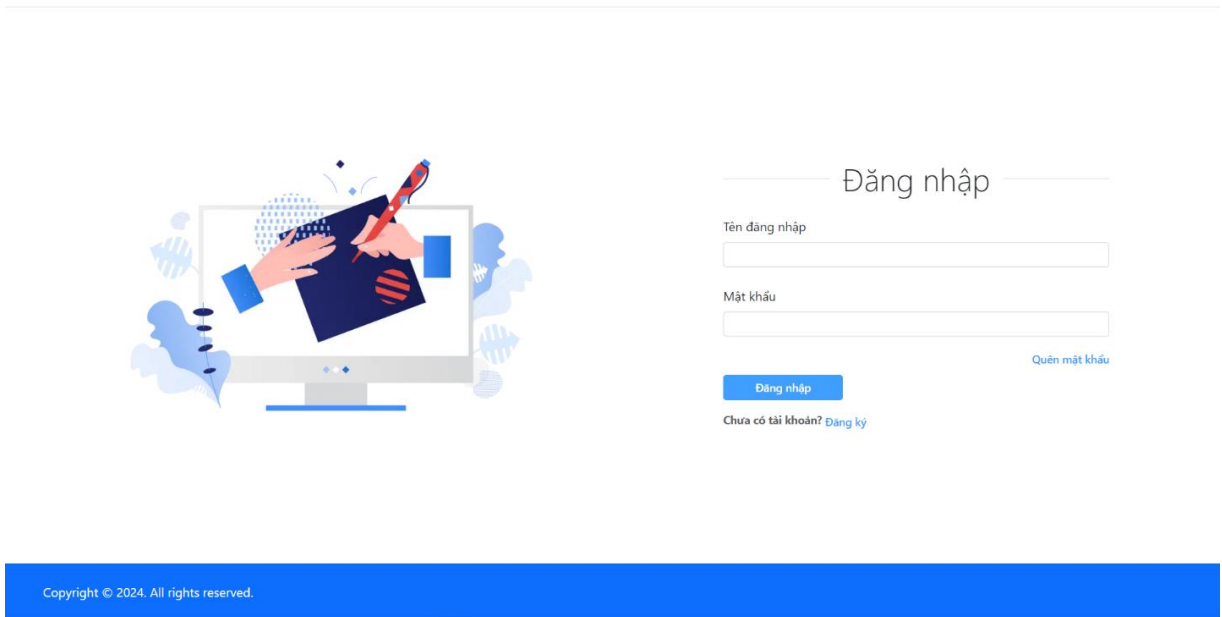
- + Người dùng chưa có tài khoản: hệ thống tạo tài khoản sẵn cho họ dựa theo mã sinh viên (mật khẩu giống tên tài khoản).
- + Người dùng có tài khoản và ở trong nhóm: hệ thống xử lý bản ghi kế tiếp.
- + Người dùng có tài khoản nhưng chưa ở trong nhóm: hệ thống gửi thông báo và email mời người dùng vào nhóm.

Trong quá trình xử lý file thì transaction luôn được bật. Nếu có lỗi xảy ra thì các bản ghi sẽ được rollback và gửi thông báo về cho người dùng.

## 3.2 Kết quả thu được

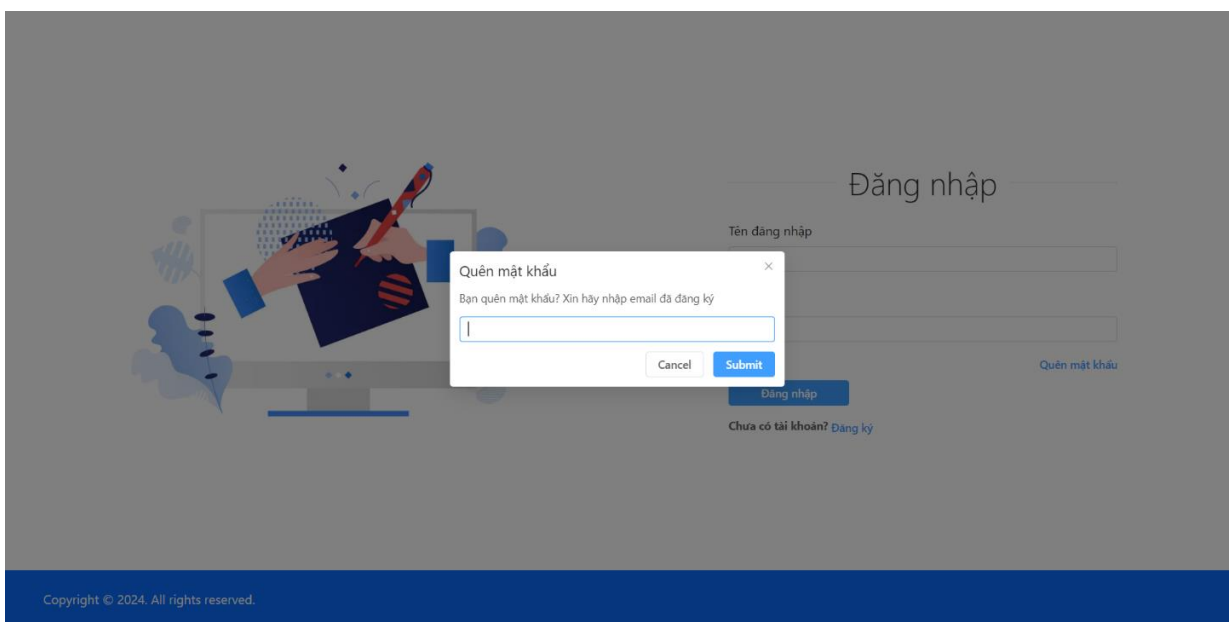
### 3.2.1 Chức năng tài khoản

Để truy cập được vào hệ thống, người dùng cần thực hiện đăng nhập.



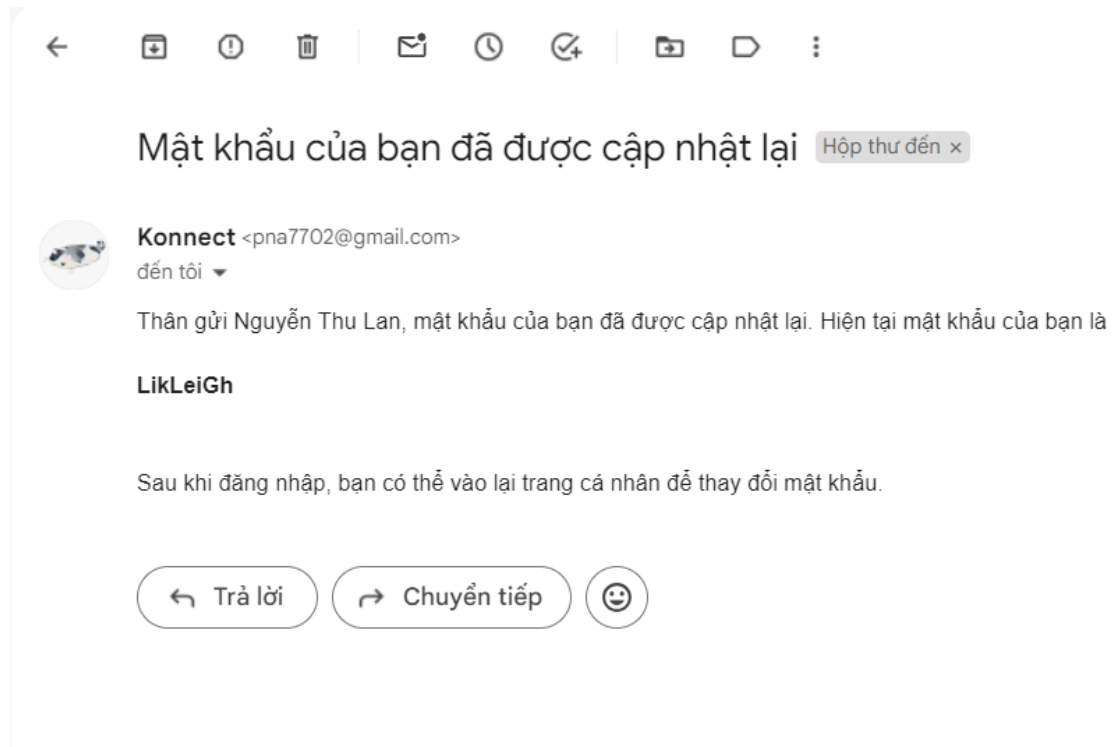
**Hình 3.7.** Giao diện trang đăng nhập

Người dùng có thể lấy lại mật khẩu nếu quên mất bằng cách nhập email đăng ký.



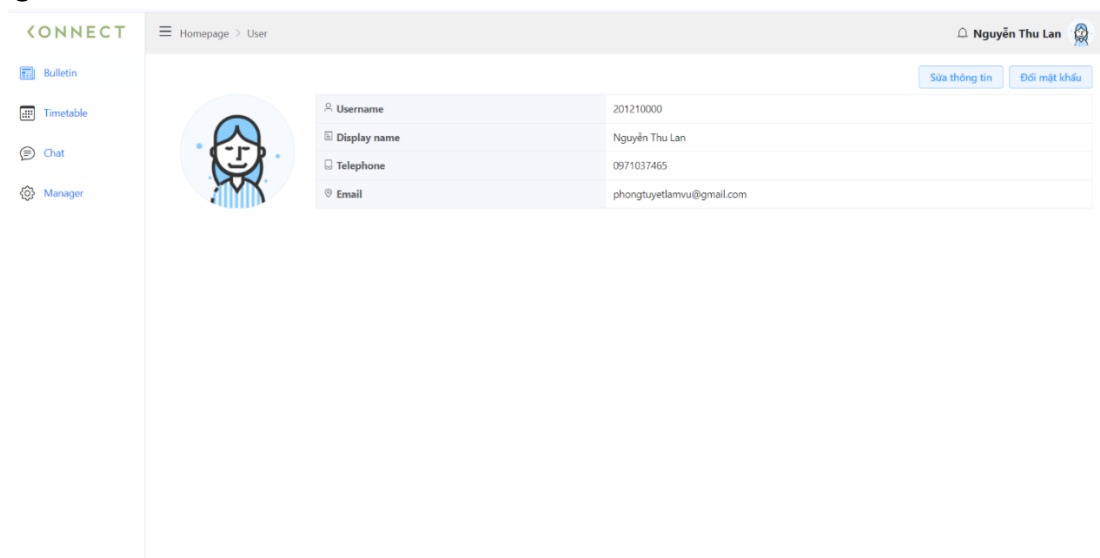
**Hình 3.8.** Dialog quên mật khẩu

Khi đó một mail sẽ được gửi đến hòm thư của tài khoản đó.



**Hình 3.9.** Hình ảnh mail do hệ thống gửi khi quên mật khẩu

Sau khi truy cập vào hệ thống, người dùng có thể thay đổi ảnh đại diện mặc định của hệ thống thông qua việc ấn vào ảnh đại diện. Người dùng cũng có thể sửa những thông tin cơ bản hoặc đổi mật khẩu.



**Hình 3.10.** Giao diện trang người dùng

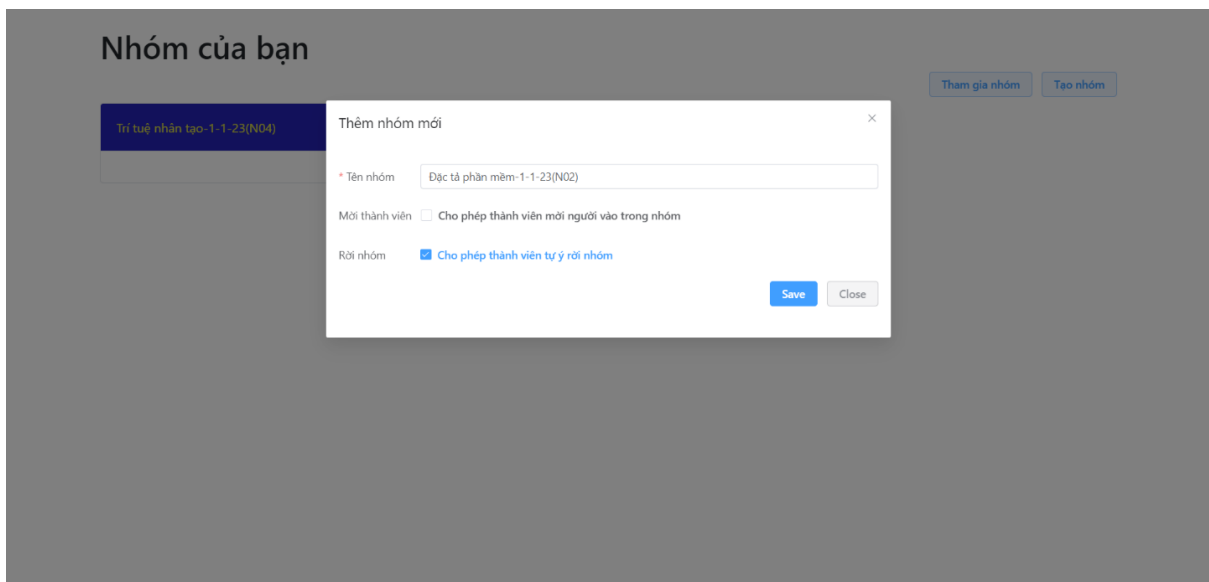
### 3.2.2 Chức năng nhóm

Sau khi đăng nhập xong, người dùng cần chọn nhóm để truy cập.



**Hình 3.11.** Giao diện trang chọn nhóm

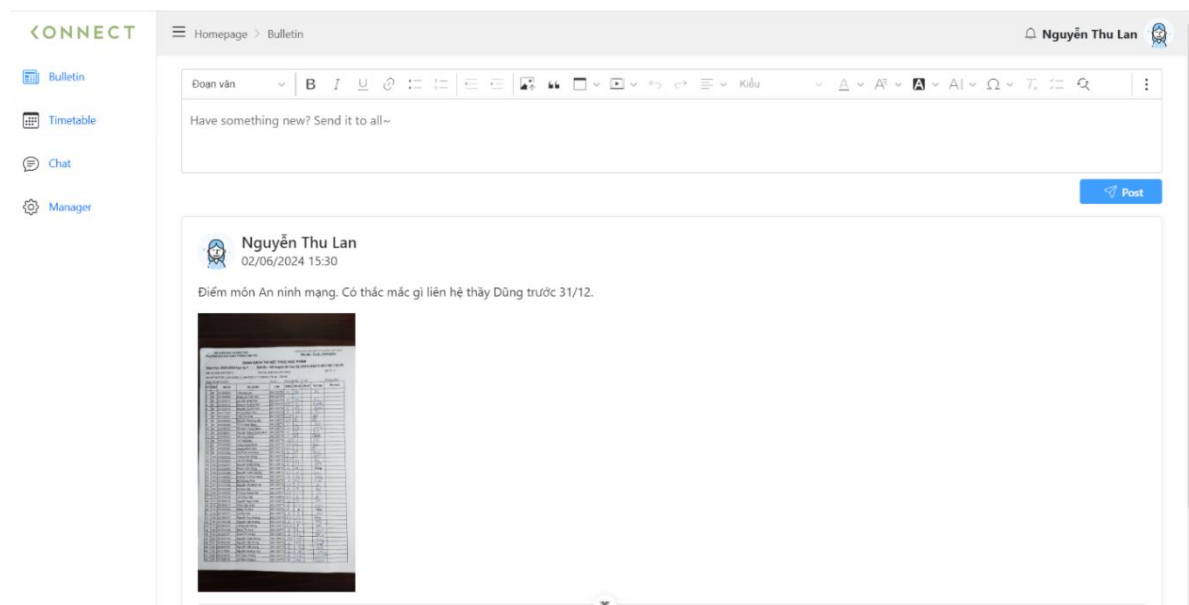
Người dùng có thể tham gia một nhóm thông qua mã mời hoặc tạo một nhóm mới với vai trò Quản lý.



**Hình 3.12.** Dialog thêm nhóm mới

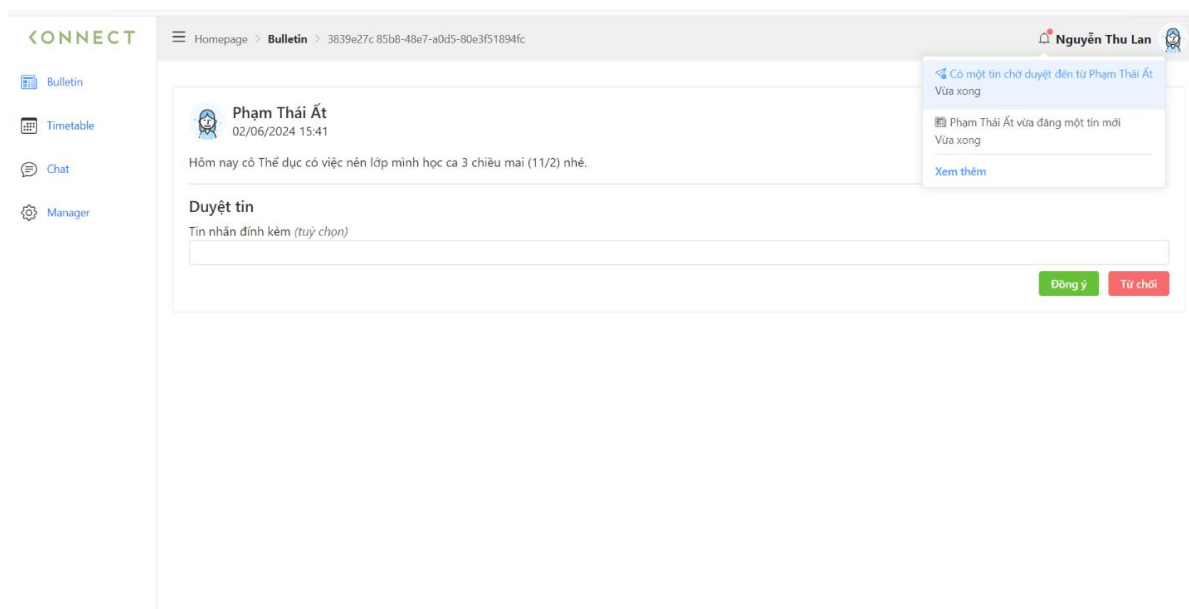
### 3.2.3 Chức năng bảng tin

Các tin tức đã được phê duyệt của nhóm sẽ hiển thị trước mặt người dùng khi người dùng truy cập vào nhóm.



**Hình 3.13.** Giao diện trang bảng tin

Khi một thành viên trong nhóm đăng bài, quản lý sẽ nhận được thông báo để đưa ra lựa chọn phê duyệt tin hay không cùng với phản hồi cho người đăng.



**Hình 3.14.** Giao diện trang phê duyệt tin



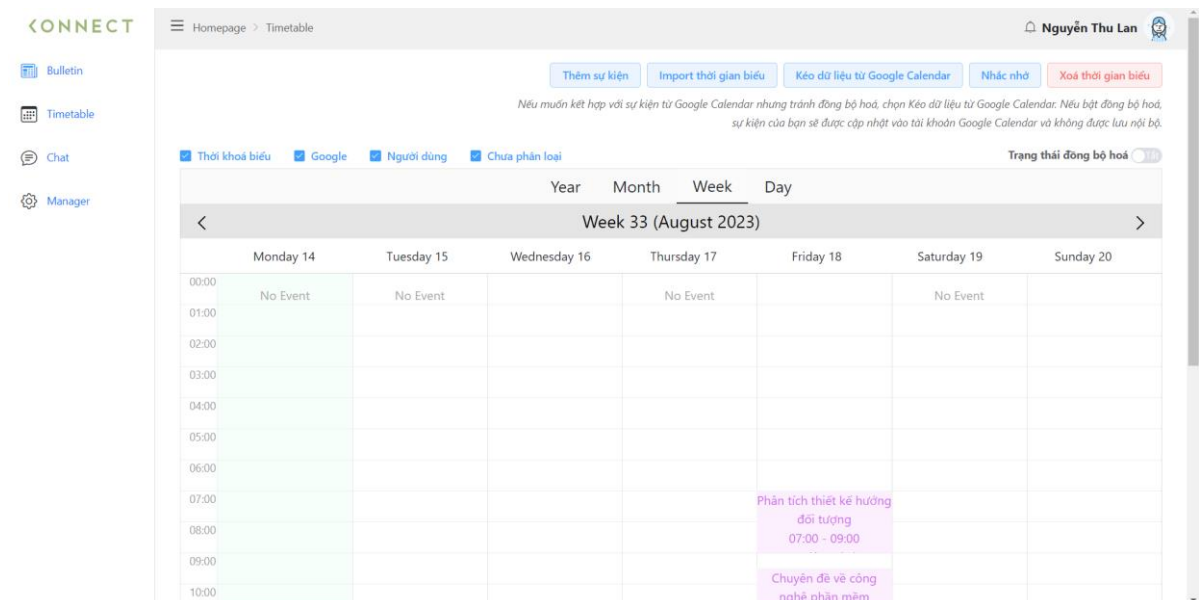
Người dùng cũng có thể bình luận khi nhấn vào tin tức.



Hình 3.15. Giao diện chi tiết tin tức

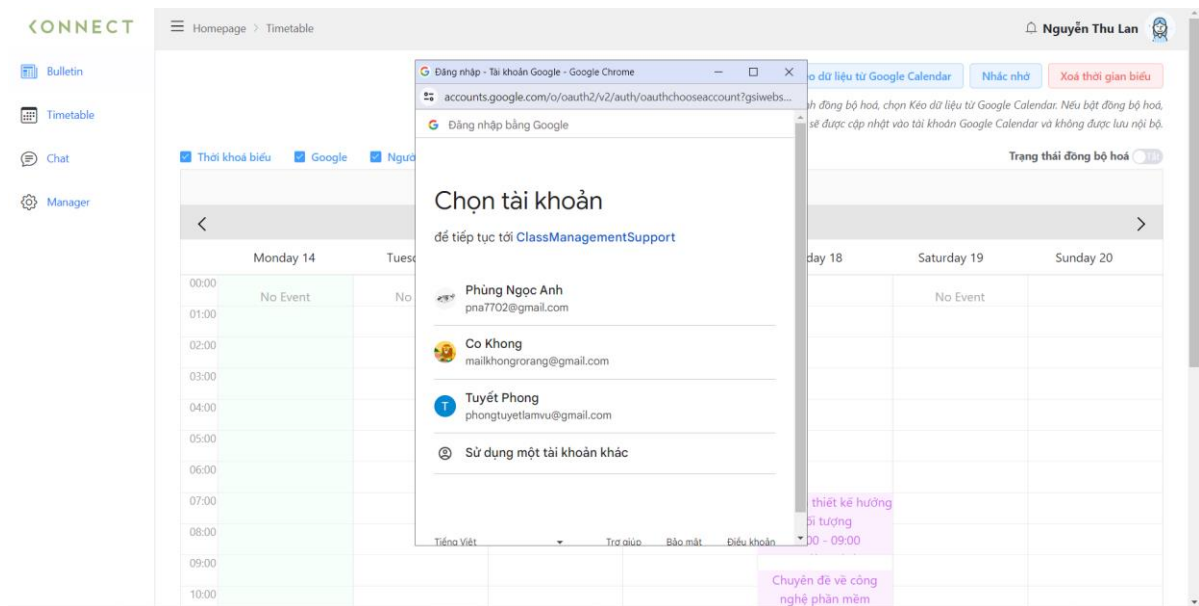
### 3.2.4 Chức năng thời gian biểu

Giao diện trang thời gian biểu:



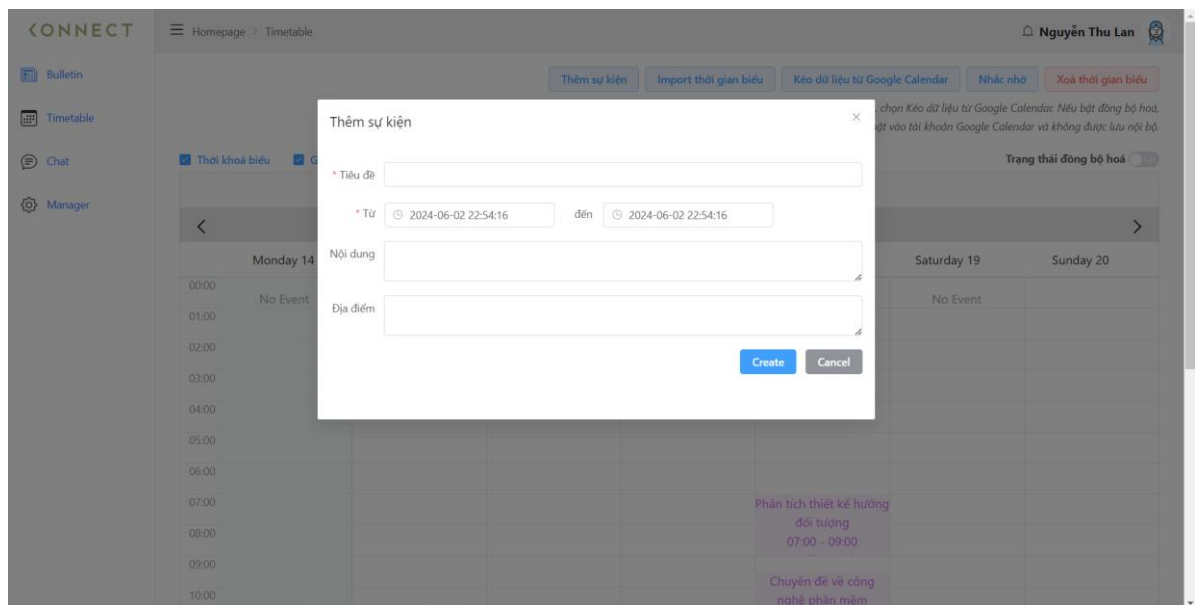
Hình 3.16. Giao diện trang thời gian biểu

Nếu bật đồng bộ hoá hoặc chọn Kéo dữ liệu từ Google Calendar về thì popup xác minh danh tính của Google sẽ bật ra.



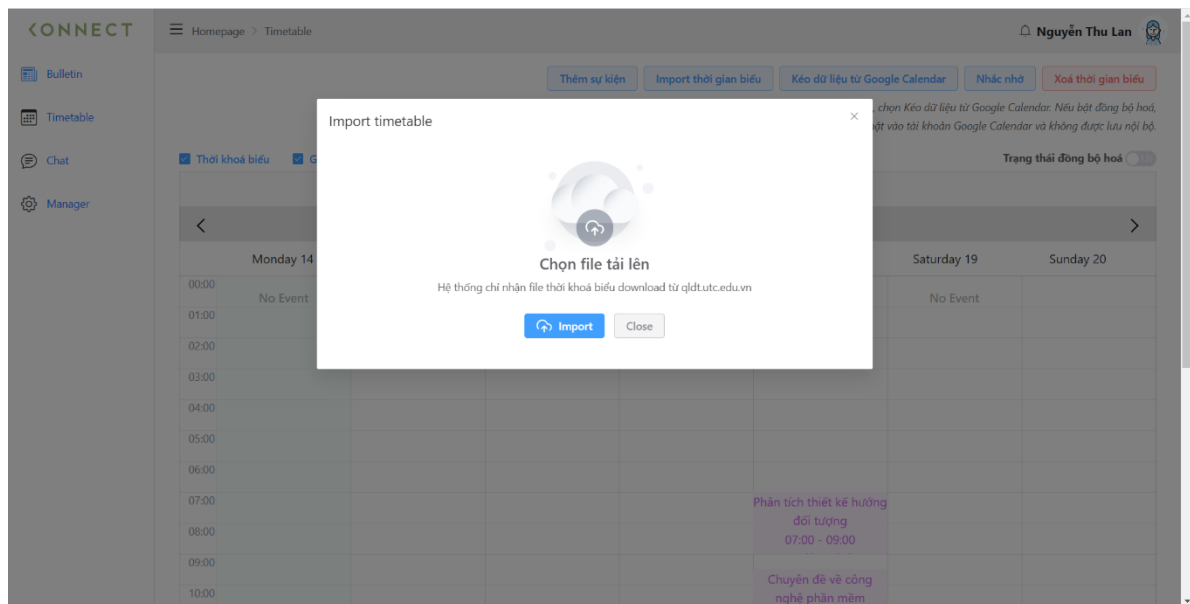
**Hình 3.17.** Popup xác minh tài khoản của Google Calendar

Người dùng có thể thêm sự kiện thông qua nút “Thêm sự kiện” hoặc sửa/xoá sự kiện bằng cách ấn vào sự kiện trên thời gian biểu.



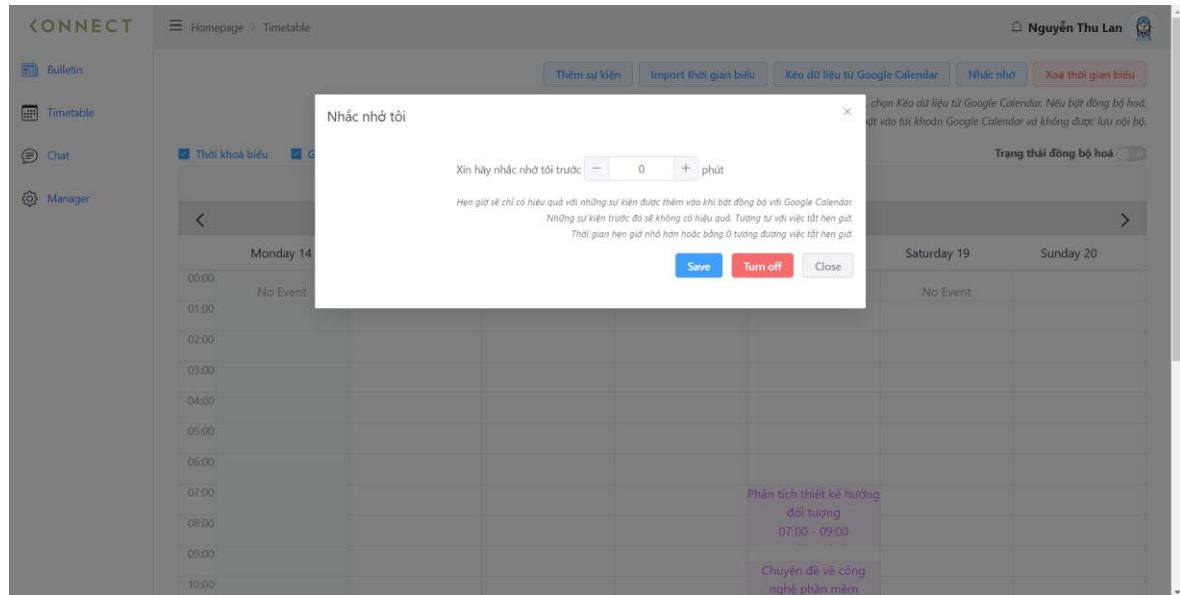
**Hình 3.18.** Dialog thêm sự kiện

Người dùng có thể tải thời khoá biểu từ trang Quản lý đào tạo của nhà trường để hệ thống tiến hành trực quan hoá.



**Hình 3.19.** Dialog import thời khoá biểu

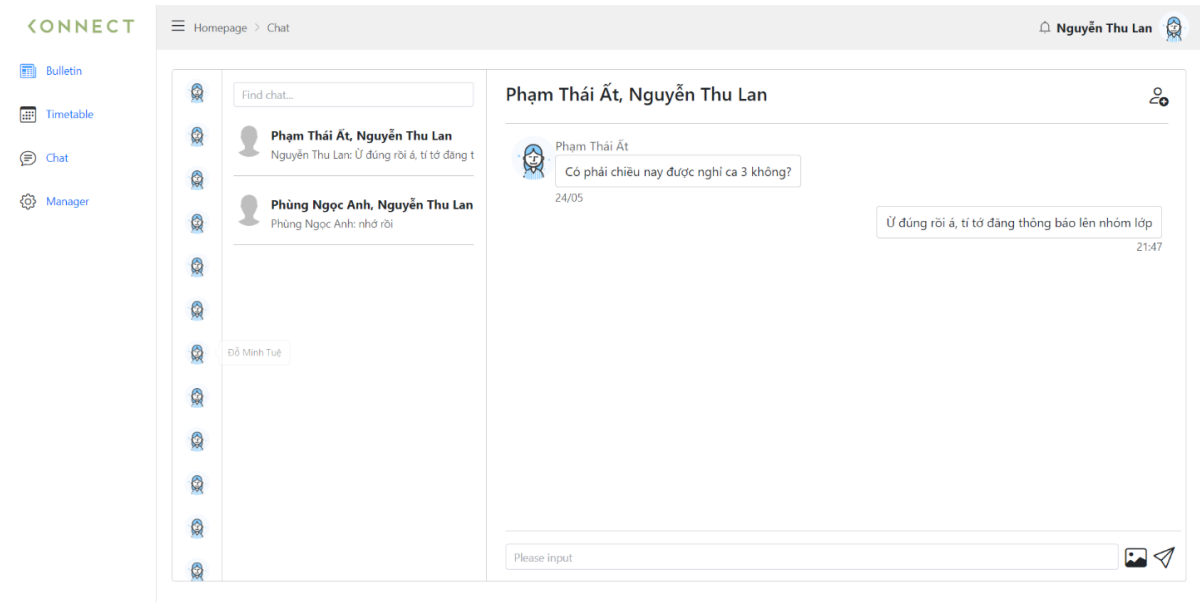
Người dùng có thể bật/tắt nhắc nhở các sự kiện thông qua Google Calendar.



**Hình 3.20.** Dialog bật/tắt thời gian nhắc nhở

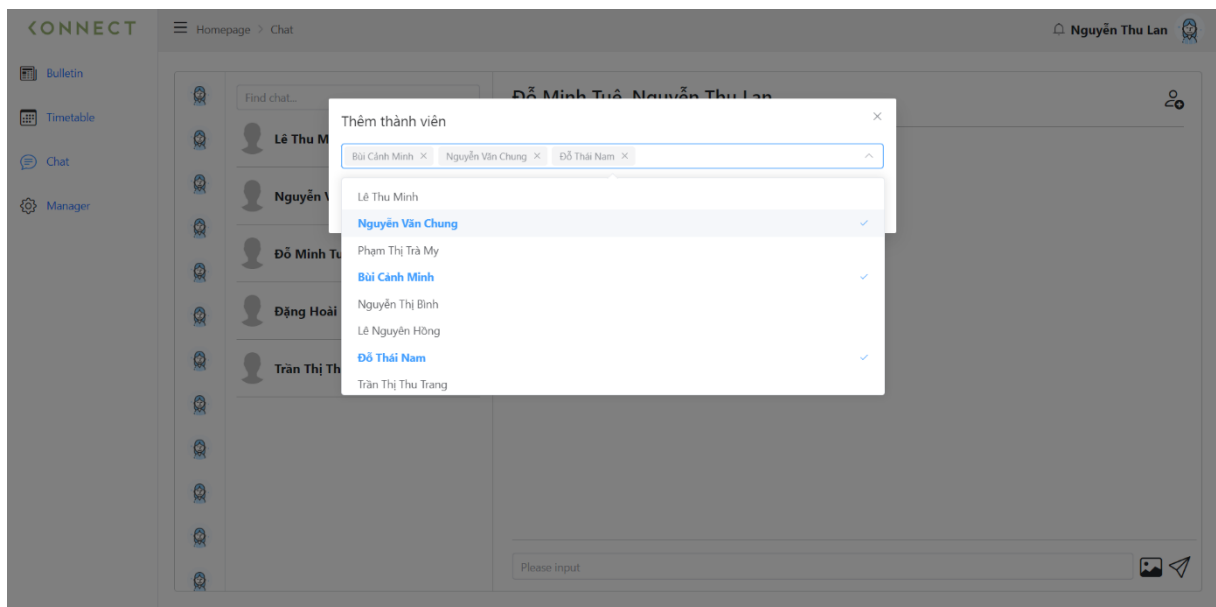
### 3.2.5 Chức năng nhắn tin

Người dùng có thể nhấn vào avatar của các thành viên trong nhóm để tạo cuộc hội thoại với họ và tiến hành trò chuyện.



**Hình 3.21.** Giao diện trang nhắn tin

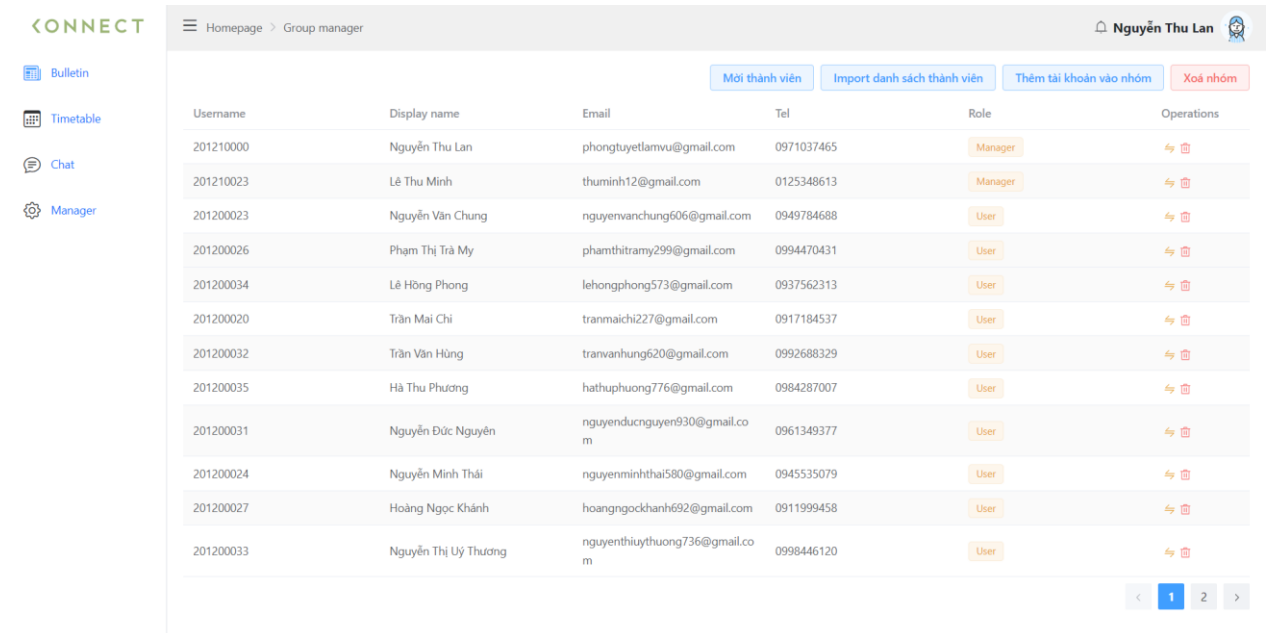
Người dùng cũng có thể thêm thành viên trong nhóm vào cuộc trò chuyện.



**Hình 3.22.** Dialog thêm thành viên vào cuộc trò chuyện

### 3.2.6 Chức năng quản lý

Chỉ có quản lý mới có thể nhìn thấy các nút. Người dùng chỉ có thể xem được danh sách thành viên và chức vụ.



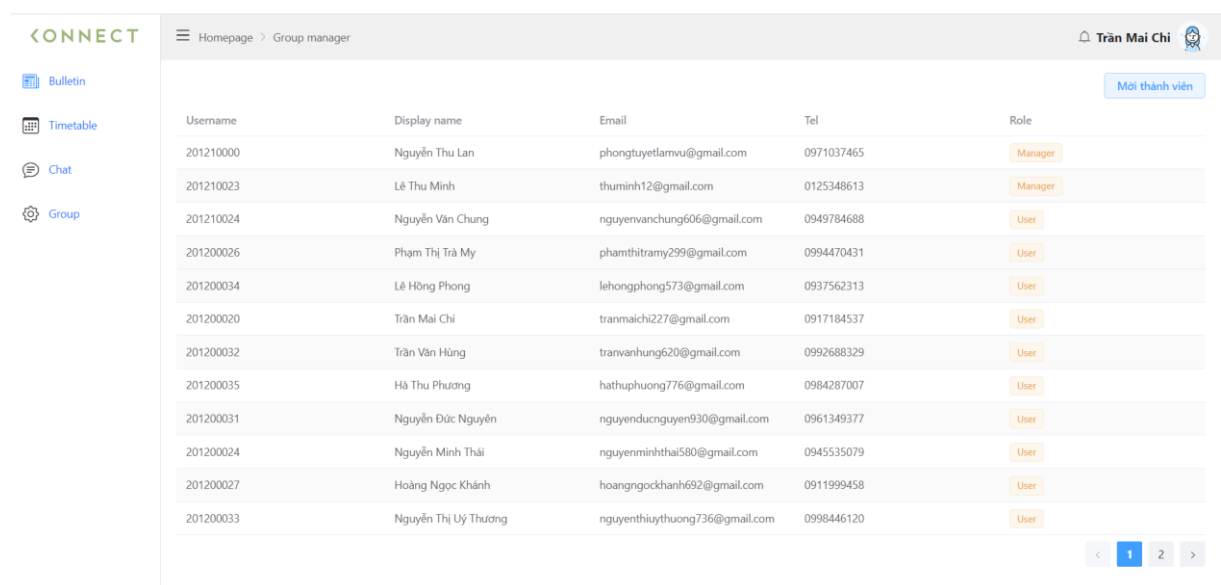
**CONNECT** | Homepage > Group manager | Nguyễn Thu Lan

Môi thành viên | Import danh sách thành viên | Thêm tài khoản vào nhóm | Xóa nhóm

Username	Display name	Email	Tel	Role	Operations
201210000	Nguyễn Thu Lan	phongtuyetlamvu@gmail.com	0971037465	Manager	
201210023	Lê Thu Minh	thuminh12@gmail.com	0125348613	Manager	
201200023	Nguyễn Văn Chung	nguyenvanchung606@gmail.com	0949784688	User	
201200026	Phạm Thị Trà My	phamthitramy299@gmail.com	0994470431	User	
201200034	Lê Hồng Phong	lehongphong573@gmail.com	0937562313	User	
201200020	Trần Mai Chi	tranmaichi227@gmail.com	0917184537	User	
201200032	Trần Văn Hùng	tranvanhung620@gmail.com	0992688329	User	
201200035	Hà Thu Phương	hathuphuong776@gmail.com	0984287007	User	
201200031	Nguyễn Đức Nguyễn	nguyenducnguyen930@gmail.com	0961349377	User	
201200024	Nguyễn Minh Thái	nguyenminhthai580@gmail.com	0945535079	User	
201200027	Hoàng Ngọc Khánh	hoangngockhanh692@gmail.com	0911999458	User	
201200033	Nguyễn Thị Uyên Thương	nguyenthuythuong736@gmail.com	0998446120	User	

< 1 2 >

**Hình 3.23.** Giao diện trang quản lý (trong vai trò quản lý)



**CONNECT** | Homepage > Group manager | Trần Mai Chi

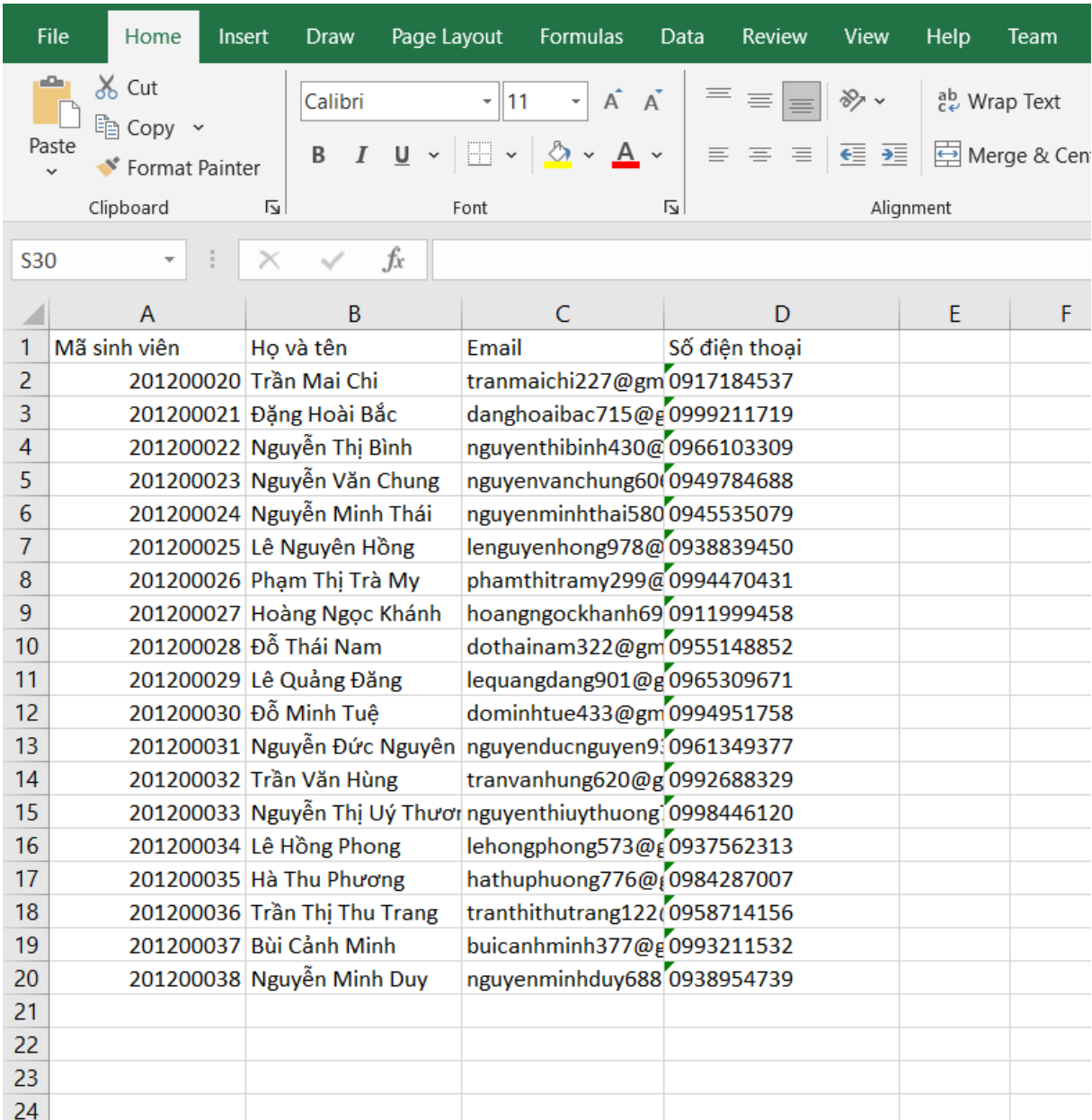
Môi thành viên

Username	Display name	Email	Tel	Role
201210000	Nguyễn Thu Lan	phongtuyetlamvu@gmail.com	0971037465	Manager
201210023	Lê Thu Minh	thuminh12@gmail.com	0125348613	Manager
201210024	Nguyễn Văn Chung	nguyenvanchung606@gmail.com	0949784688	User
201200026	Phạm Thị Trà My	phamthitramy299@gmail.com	0994470431	User
201200034	Lê Hồng Phong	lehongphong573@gmail.com	0937562313	User
201200020	Trần Mai Chi	tranmaichi227@gmail.com	0917184537	User
201200032	Trần Văn Hùng	tranvanhung620@gmail.com	0992688329	User
201200035	Hà Thu Phương	hathuphuong776@gmail.com	0984287007	User
201200031	Nguyễn Đức Nguyễn	nguyenducnguyen930@gmail.com	0961349377	User
201200024	Nguyễn Minh Thái	nguyenminhthai580@gmail.com	0945535079	User
201200027	Hoàng Ngọc Khánh	hoangngockhanh692@gmail.com	0911999458	User
201200033	Nguyễn Thị Uyên Thương	nguyenthuythuong736@gmail.com	0998446120	User

< 1 2 >

**Hình 3.24.** Giao diện trang quản lý (trong vai trò thành viên)

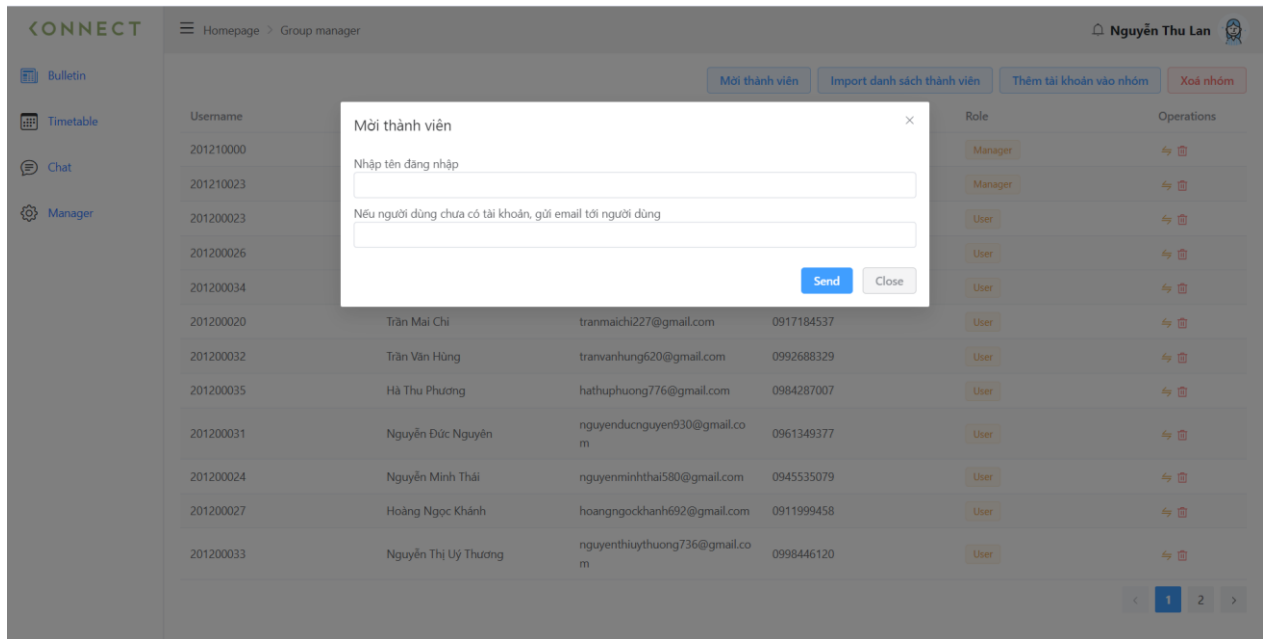
Quản lý có thể nhập danh sách các thành viên thông qua việc import file excel theo mẫu. File excel sẽ yêu cầu có các trường Mã sinh viên (tên đăng nhập và mật khẩu), Họ và tên, Email và Số điện thoại.



	A	B	C	D	E	F
1	Mã sinh viên	Họ và tên	Email	Số điện thoại		
2	201200020	Trần Mai Chi	tranmaichi227@gm	0917184537		
3	201200021	Đặng Hoài Bắc	danghoaibac715@g	0999211719		
4	201200022	Nguyễn Thị Bình	nguyenthibinh430@	0966103309		
5	201200023	Nguyễn Văn Chung	nguyenvanchung60	0949784688		
6	201200024	Nguyễn Minh Thái	nguyenminhthai580	0945535079		
7	201200025	Lê Nguyên Hồng	lenguyenhong978@	0938839450		
8	201200026	Phạm Thị Trà My	phamthitramy299@	0994470431		
9	201200027	Hoàng Ngọc Khánh	hoangngockhanh69	0911999458		
10	201200028	Đỗ Thái Nam	dothainam322@gm	0955148852		
11	201200029	Lê Quảng Đăng	lequangdang901@g	0965309671		
12	201200030	Đỗ Minh Tuệ	dominhthue433@gm	0994951758		
13	201200031	Nguyễn Đức Nguyễn	nguyenducnguyen9	0961349377		
14	201200032	Trần Văn Hùng	tranvanhung620@g	0992688329		
15	201200033	Nguyễn Thị Uyên Thương	nguyenthuiythuong	0998446120		
16	201200034	Lê Hồng Phong	lehongphong573@g	0937562313		
17	201200035	Hà Thu Phương	hathuphuong776@g	0984287007		
18	201200036	Trần Thị Thu Trang	tranthithutrang122	0958714156		
19	201200037	Bùi Cảnh Minh	buicanhminh377@g	0993211532		
20	201200038	Nguyễn Minh Duy	nguyenminhduy688	0938954739		
21						
22						
23						
24						

**Hình 3.25.** Template import danh sách thành viên

Người dùng có thể gửi lời mời bằng email (*đến những người chưa có tài khoản*) hoặc tên đăng nhập.



**Hình 3.26.** Dialog mời thành viên

## KẾT LUẬN

Sau thời gian thực hiện đề tài em đã được củng cố thêm nhiều kiến thức về công nghệ, cách hoạt động cũng như quy trình phát triển một ứng dụng. Trong quá trình xây dựng ứng dụng, em đã gặp một số khó khăn như xử lý bất đồng bộ khi kết nối với api của Google, vấn đề trong quá trình cài đặt WebSocket cho chức năng tin nhắn... Ngoài ra, đồ án của em cũng có một số thay đổi so với đề cương ban đầu như việc bỏ chức năng Lưu trữ, do việc thuê cloud sẽ rất tốn kém để duy trì.

Tuy nhiên nhờ sự góp ý của thầy và sự giúp đỡ của các bạn trong nhóm, cuối cùng em đã hoàn thành được đề tài với các chức năng cần thiết. Cụ thể như chức năng bảng tin giúp cho người dùng có thể đăng tin, bình luận và nhận được thông báo kịp thời. Chức năng tin nhắn giúp cho thành viên trong nhóm có thể trao đổi trực tiếp trên website. Chức năng thời gian biểu giúp người dùng trực quan hoá file thời khoá biểu của nhà trường, hỗ trợ kết hợp với Google Calendar cá nhân khiến việc lên lịch và quản lý thời gian trở nên đơn giản, dễ dàng hơn. Chức năng quản lý nhóm hỗ trợ đọc file danh sách học sinh tiện lợi cho các thầy cô và cán bộ lớp quản lý.

Dù vậy, phần mềm vẫn có đôi chút hạn chế như UX/UI chưa thật sự thân thiện với người dùng, giao diện có thể cải thiện cho đẹp mắt hơn. Ngoài ra, đề tài vẫn có tiềm năng phát triển và rộng mở, ví dụ như phát triển thêm phiên bản mobile app.



## DANH MỤC TÀI LIỆU THAM KHẢO

1. <https://stackoverflow.com/>
2. <https://chatgpt.com/>
3. <https://developers.google.com/calendar/api/guides/overview?hl=vi>
4. <https://learn.microsoft.com/en-us/aspnet/core/tutorials/signalr?view=aspnetcore-8.0&tabs=visual-studio>
5. <https://element-plus.org/en-US/component/overview.html>
6. <https://github.com/developit/mitt#usage>
7. <https://antoniandre.github.io/vue-cal/>
8. <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
9. <https://vuejs.org/>
10. <https://learn.microsoft.com/en-us/aspnet/core/tutorials/first-mongo-app?view=aspnetcore-8.0&tabs=visual-studio>
11. <https://mysqlconnector.net/>
12. <https://github.com/PomeloFoundation/Pomelo.EntityFrameworkCore.MySql>
13. <https://ckeditor.com/docs/index.html>
14. <https://axios-http.com/docs/intro>