

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN, ĐHQG-HCM**  
**KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG**



**BÁO CÁO ĐỒ ÁN MÔN HỌC**  
**LẬP TRÌNH MẠNG CĂN BẢN**  
**ĐỀ TÀI: Ứng dụng nhắn tin (Chat App)**

**Giảng viên hướng dẫn: Trần Hồng Nghi**

**Thực hiện bởi nhóm 10, bao gồm:**

<b>Họ tên thành viên</b>	<b>MSSV</b>	<b>Vai trò</b>
1. Trương Việt Hoàng	24520575	Trưởng nhóm
2. Lê Minh Hoàng	24520544	Thành viên
3. Trần Phước Hoàng	24520572	Thành viên
4. Huỳnh Vũ khánh Hưng	24520592	Thành viên

**Thời gian thực hiện: 15/09/2026 - 30/12/2026**

## LỜI CẢM ƠN

Nhóm xin gửi lời cảm ơn chân thành đến giảng viên hướng dẫn đã tận tình hỗ trợ, góp ý và đồng hành trong suốt quá trình thực hiện đồ án. Nhờ đó, nhóm có cơ hội vận dụng kiến thức lập trình mạng và phát triển phần mềm để xây dựng một ứng dụng nhắn tin thời gian thực hoàn chỉnh.

Trong quá trình thực hiện, dù đã cố gắng nhưng đồ án vẫn có thể tồn tại một số thiếu sót do giới hạn về thời gian và kinh nghiệm. Nhóm rất mong nhận được nhận xét và góp ý để tiếp tục hoàn thiện sản phẩm.

Nhóm xin chân thành cảm ơn!

## NHẬN XÉT CỦA GIẢNG VIÊN

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

# NỘI DUNG

<b>LỜI CẢM ƠN</b>	1
<b>NHẬN XÉT CỦA GIẢNG VIÊN</b>	2
<b>NỘI DUNG</b>	3
<b>CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI</b>	5
<b>1.1 Giới thiệu đề tài</b>	5
<b>1.2 Mục tiêu đề tài</b>	5
<b>1.3 Công nghệ sử dụng</b>	5
1.3.1 Ngôn Ngữ Lập Trình C#	5
1.3.2 Windows Forms	5
1.3.3 Firebase	5
1.3.4 Catbox	5
1.3.5 Thư Viện Giao Diện Guna UI2	5
1.3.6 Quản Mã Nguồn Bằng Git/GitHub	6
<b>CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG</b>	6
<b>2.1 Tổng quan kiến trúc hệ thống</b>	6
<b>2.2 Sơ đồ use-case</b>	7
2.2.1 Chức năng Đăng nhập – Đăng ký – Đăng xuất – Quên mật khẩu	7
2.2.2 Chức năng Cài đặt – Chế độ ngày đêm	8
2.2.3 Chức năng Bạn bè – Kết bạn	9
2.2.4 Chức năng Nhắn tin	10
2.2.5 Chức năng Nhóm	11
<b>2.3 Thiết kế cơ sở dữ liệu</b>	12
2.3.1 Authentication (Firebase)	12
2.3.2 Realtime Database (Firebase)	12
2.3.3 Storage (Catbox & lưu trữ cục bộ)	22
<b>CHƯƠNG 3: THIẾT KẾ GIAO DIỆN VÀ CODE CHỨC NĂNG</b>	23
<b>3.1 Giao diện đăng nhập</b>	23
<b>3.2 Giao diện đăng ký</b>	24
<b>3.3 Giao diện quên mật khẩu</b>	25
<b>3.4 Giao diện trang chủ</b>	26
<b>3.5 Giao Diện Cài Đặt</b>	28
<b>3.6 Giao diện Nhắn Tin</b>	29

<b>3.7 Giao diện tạo nhóm, tìm bạn và quản lý lời mời.....</b>	<b>33</b>
3.7.1 Giao diện tạo nhóm.....	33
3.7.2 Giao diện tìm kiếm bạn bè .....	33
3.7.3 Giao diện lời mời kết bạn.....	34
3.7.4 Giao diện thông tin nhóm.....	34
3.7.5 Giao diện thông tin bạn bè .....	35
<b>CHƯƠNG 4: KẾT QUẢ VÀ HƯỚNG PHÁT TRIỂN.....</b>	<b>35</b>
<b>4.1 Kết quả đạt được.....</b>	<b>35</b>
<b>4.2 Hạn chế .....</b>	<b>35</b>
<b>4.3 Hướng phát triển trong tương lai.....</b>	<b>35</b>

# CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI

## 1.1 Giới thiệu đề tài

Trong bối cảnh nhu cầu trao đổi thông tin nhanh và thuận tiện ngày càng tăng, các ứng dụng nhắn tin (chat) thời gian thực đóng vai trò quan trọng trong học tập và làm việc nhóm. Đề tài ChatApp tập trung xây dựng một ứng dụng nhắn tin trên Windows (WinForms) với khả năng đồng bộ dữ liệu theo thời gian thực, hỗ trợ nhắn tin cá nhân và nhóm, quản lý bạn bè và chia sẻ tệp.

## 1.2 Mục tiêu đề tài

Mục tiêu của đề tài bao gồm:

- Xây dựng ứng dụng WinForms có giao diện trực quan, thao tác dễ sử dụng.
- Thiết kế hệ thống nhắn tin thời gian thực dựa trên Firebase Realtime Database.
- Áp dụng các kiến thức lập trình mạng, xử lý bất đồng bộ và quản lý dữ liệu phía client.
- Hỗ trợ các tính năng cơ bản của một Chat App: đăng ký/đăng nhập, quản lý tài khoản, nhắn tin, tạo nhóm, gửi tệp và trạng thái trực tuyến.

## 1.3 Công nghệ sử dụng

### 1.3.1 Ngôn Ngữ Lập Trình C#

Ứng dụng được phát triển bằng C# trên nền tảng .NET, tận dụng mô hình lập trình hướng đối tượng, thư viện phong phú và khả năng xử lý bất đồng bộ (async/await) để tương tác với dịch vụ đám mây.

### 1.3.2 Windows Forms

Windows Forms cung cấp nền tảng xây dựng giao diện ứng dụng desktop trên Windows. Nhóm sử dụng các thành phần như Form, Panel/FlowLayoutPanel, TextBox, Button và các custom control để dựng bố cục danh sách hội thoại và khung chat.

### 1.3.3 Firebase

Firebase được dùng làm backend-as-a-service:

- Firebase Authentication: (tuỳ chọn) hỗ trợ xác thực email/password.
- Firebase Realtime Database: lưu trữ dữ liệu người dùng, bạn bè, nhóm và tin nhắn theo dạng JSON, đồng bộ thời gian thực.

### 1.3.4 Catbox

Catbox dùng để lưu trữ tệp đính kèm (ảnh/tài liệu) bằng cách upload lên dịch vụ file hosting và nhận URL chia sẻ; URL này được lưu trong tin nhắn để các client khác tải về nhanh chóng.

### 1.3.5 Thư Viện Giao Diện Guna UI2

Nhóm sử dụng thư viện UI (tuỳ theo dự án) để tạo giao diện hiện đại: button bo góc, shadow panel, avatar, progress bar, giúp cải thiện trải nghiệm người dùng và giảm thời gian thiết kế giao diện.

### 1.3.6 Quản Mã Nguồn Bằng Git/GitHub

Dự án được quản lý bằng Git và GitHub: phân nhánh theo tính năng, tạo pull request để review, theo dõi issues và lịch sử commit nhằm đảm bảo phối hợp nhóm hiệu quả.

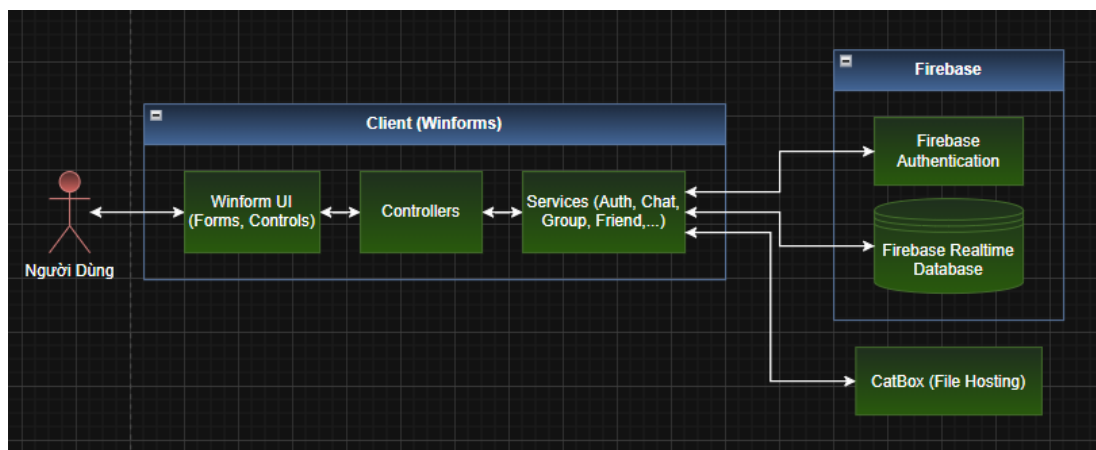
## CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### 2.1 Tổng quan kiến trúc hệ thống

Ứng dụng ChatApp được xây dựng theo mô hình Client – Cloud. Toàn bộ giao diện và xử lý phía người dùng chạy trên WinForms; dữ liệu tài khoản, danh sách bạn bè/nhóm và tin nhắn được lưu trên Firebase Realtime Database và đồng bộ theo thời gian thực thông qua cơ chế lắng nghe (listener/stream). Với tệp đính kèm (ảnh, tài liệu), ứng dụng tải lên dịch vụ Catbox để nhận URL, sau đó lưu URL và metadata (tên file, kích thước, thời gian gửi) trong Firebase để các máy khác tải về.

Các thành phần chính:

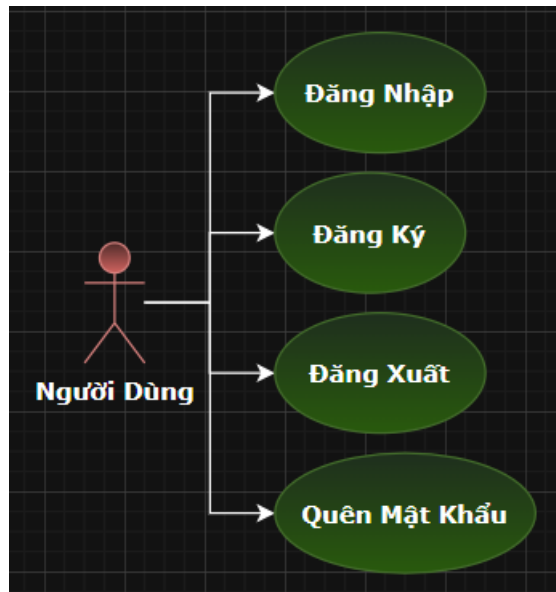
- **Client (WinForms):** hiển thị UI, kiểm tra dữ liệu nhập, gọi các service để đăng nhập/đăng ký, gửi – nhận tin nhắn, tải lên/tải xuống tệp.
- **Firebase Realtime Database:** lưu Users, Friends, Groups, Messages, Status, Themes; phát sự kiện realtime khi dữ liệu thay đổi.
- **Firebase Authentication:** hỗ trợ xác thực email/password (hoặc nhóm tự quản lý tài khoản trong RTDB).
- **Catbox (File Hosting):** lưu ảnh/tài liệu; trả về đường dẫn URL để chia sẻ nhanh.
- **Email/OTP (nếu có):** gửi mã xác nhận khi đăng ký hoặc quên mật khẩu.



Hình 2.1.1 Tổng quan kiến trúc hệ thống

## 2.2 Sơ đồ use-case

### 2.2.1 Chức năng Đăng nhập – Đăng ký – Đăng xuất – Quên mật khẩu



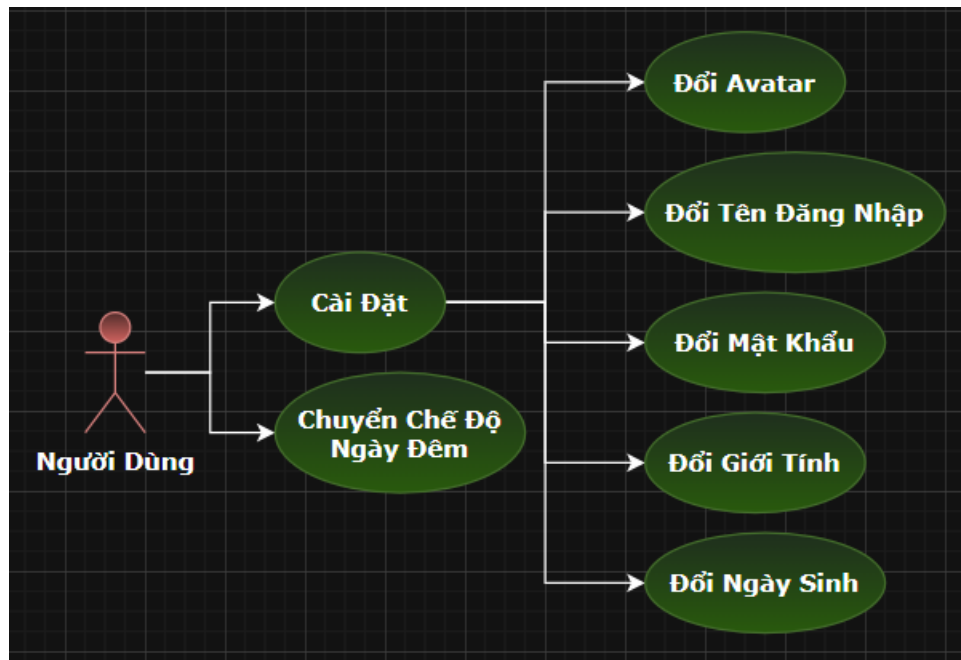
Hình 2.2.1 Sơ đồ Use case chức năng Đăng nhập – Đăng ký – Đăng xuất – Quên Mật Khẩu

#### Mô tả:

- **Đăng nhập:** Người dùng nhập tài khoản và mật khẩu để truy cập vào hệ thống. Nếu đúng thì vào màn hình chính, nếu sai thì báo lỗi.
- **Đăng ký:** Người dùng tạo tài khoản mới bằng cách nhập thông tin cần thiết. Hệ thống kiểm tra hợp lệ và lưu vào cơ sở dữ liệu.
- **Đăng xuất:** Thoát phiên đăng nhập hiện tại, đưa người dùng về màn hình đăng nhập và xoá/trả token (nếu có).
- **Quên mật khẩu:** Người dùng nhập email/tài khoản để nhận hướng dẫn đặt lại mật khẩu. Hệ thống xác thực và cho phép tạo mật khẩu mới.



## 2.2.2 Chức năng Cài đặt – Chế độ ngày đêm



Hình 2.2.2 Sơ đồ Use case chức năng Cài đặt – Chế độ ngày đêm

### Mô tả:

- **Cài đặt:** Người dùng vào mục cài đặt để chỉnh sửa thông tin cá nhân và tùy chọn tài khoản.
- **Chuyển chế độ ngày/đêm:** Bật/tắt giao diện sáng–tối và lưu lại lựa chọn.
- **Đổi avatar:** Chọn ảnh đại diện mới, hệ thống cập nhật và hiển thị lại.
- **Đổi tên đăng nhập:** Nhập tên mới, hệ thống kiểm tra hợp lệ và cập nhật.
- **Đổi mật khẩu:** Nhập mật khẩu hiện tại và mật khẩu mới, hệ thống xác thực rồi đổi.
- **Đổi giới tính:** Nhập giới tính, hệ thống lưu cập nhật.
- **Đổi ngày sinh:** Nhập ngày sinh, hệ thống lưu cập nhật.

### 2.2.3 Chức năng Bạn bè – Kết bạn

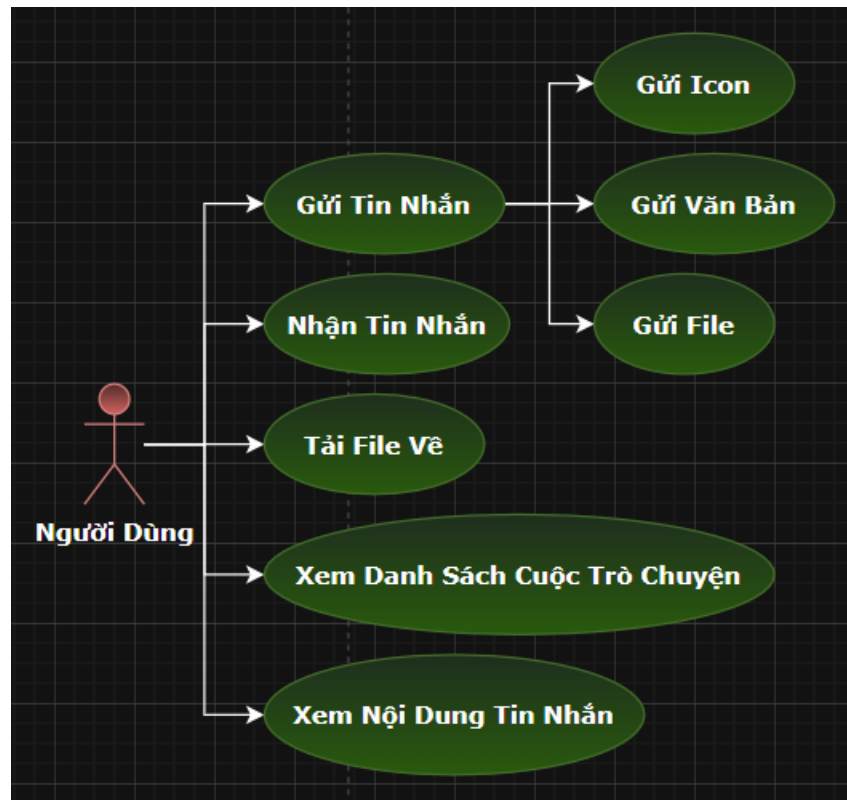


Hình 2.2.3 Sơ đồ Use case chức năng Bạn Bè – Kết bạn

#### Mô tả:

- **Gửi yêu cầu kết bạn:** Tìm người dùng và gửi lời mời kết bạn.
- **Nhận yêu cầu kết bạn:** Xem danh sách lời mời kết bạn nhận được.
- **Hủy yêu cầu kết bạn:** Hủy lời mời đã gửi khi còn trạng thái chờ.
- **Phản hồi yêu cầu kết bạn:** Xử lý lời mời với 2 lựa chọn:
  - **Chấp nhận:** Thêm nhau vào danh sách bạn bè và có thể nhắn tin.
  - **Từ chối:** Từ chối lời mời, không tạo quan hệ bạn bè.
- **Xem thông tin bạn bè:** Xem tên, giới tính, ngày sinh của bạn bè.
- **Xem danh sách bạn bè:** Xem toàn bộ bạn bè hiện có và chọn để trò chuyện.
- **Hủy kết bạn:** Xóa bạn khỏi danh sách bạn bè (hai bên không còn là bạn).

## 2.2.4 Chức năng Nhắn tin

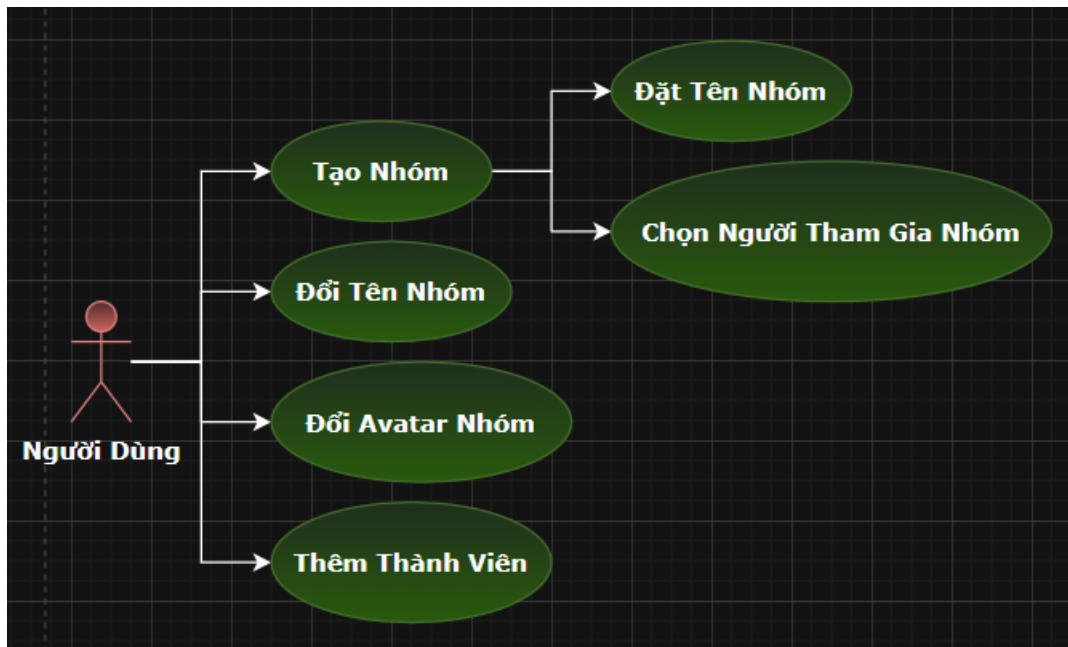


Hình 2.2.4 Sơ đồ Use case chức năng Nhắn tin

### Mô tả:

- **Xem danh sách cuộc trò chuyện:** Người dùng xem các cuộc chat (bạn bè/nhóm) để chọn mở.
- **Xem nội dung tin nhắn:** Khi chọn một cuộc chat, ứng dụng hiển thị tin nhắn (văn bản/ảnh/file).
- **Gửi tin nhắn:** Người dùng gửi nội dung trong cuộc chat, gồm:
  - **Gửi văn bản:** Nhập nội dung và bấm gửi, tin nhắn được lưu và hiển thị cho người nhận.
  - **Gửi icon (emoji):** Chọn emoji/icon và gửi như một tin nhắn.
  - **Gửi file:** Chọn file/ảnh, ứng dụng tải lên nơi lưu trữ và gửi kèm đường dẫn/file trong tin nhắn.
- **Nhận tin nhắn:** Ứng dụng nhận tin mới và hiển thị realtime trong khung chat.
- **Tải file về:** Với tin nhắn file, người dùng có thể tải file xuống máy.

## 2.2.5 Chức năng Nhóm



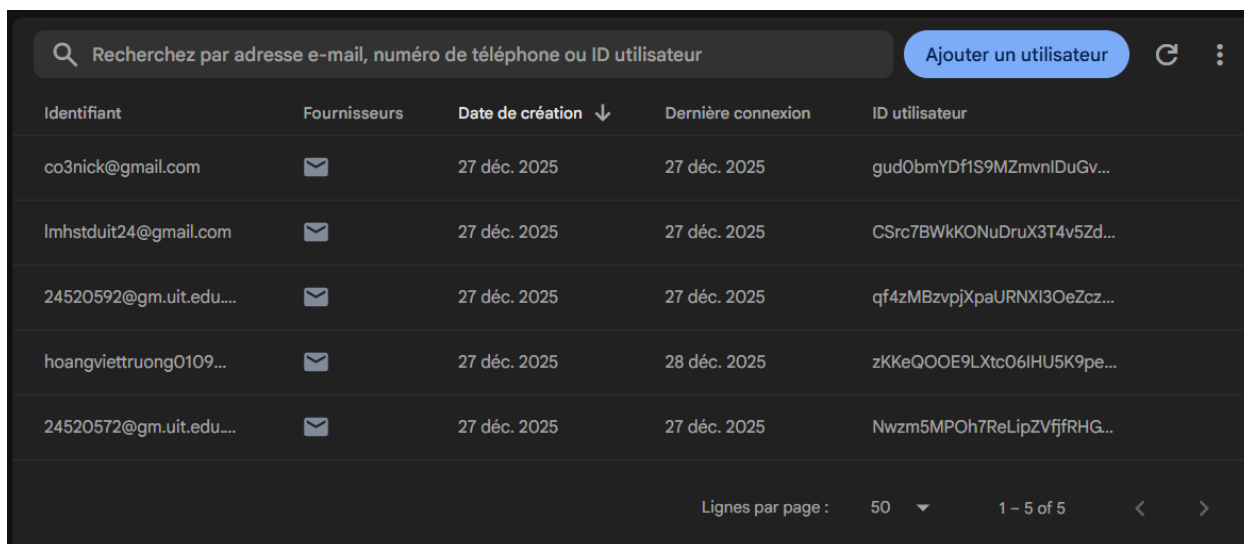
Hình 2.2.5 Sơ đồ Use case chức năng Nhóm

### Mô tả:

- **Tạo nhóm:** Người dùng tạo một nhóm chat mới để nhắn tin nhiều người.
  - **Đặt tên nhóm:** Khi tạo nhóm, người dùng nhập tên nhóm; hệ thống kiểm tra hợp lệ và lưu tên nhóm.
  - **Chọn người tham gia nhóm:** Người dùng chọn các bạn bè/thành viên cần thêm vào nhóm; hệ thống thêm thành viên và tạo nhóm thành công.
- **Đổi tên nhóm:** Cập nhật tên nhóm để hiển thị cho tất cả thành viên.
- **Đổi avatar nhóm:** Chọn và cập nhật ảnh đại diện của nhóm.
- **Thêm thành viên:** Mời/thêm người dùng khác vào nhóm để cùng trò chuyện.

## 2.3 Thiết kế cơ sở dữ liệu.2.3 Thiết kế cơ sở dữ liệu

### 2.3.1 Authentication (Firebase)



The screenshot shows the Firebase Authentication console interface. At the top, there is a search bar with the text "Recherchez par adresse e-mail, numéro de téléphone ou ID utilisateur" and a button "Ajouter un utilisateur". Below the search bar is a table with the following columns: "Identifiant", "Fournisseurs", "Date de création", "Dernière connexion", and "ID utilisateur". The table contains six rows of user data. At the bottom right, there is a pagination control showing "Lignes par page : 50" and "1 – 5 of 5".

Identifiant	Fournisseurs	Date de création	Dernière connexion	ID utilisateur
co3nick@gmail.com		27 déc. 2025	27 déc. 2025	gud0bmYDf1S9MZmvnlDuGv...
lmhstduit24@gmail.com		27 déc. 2025	27 déc. 2025	CSrc7BWkKONuDruX3T4v5Zd...
24520592@gm.uit.edu....		27 déc. 2025	27 déc. 2025	qf4zMBzvpjXpaURNXI3OeZcz...
hoangviettruong0109...		27 déc. 2025	28 déc. 2025	zKKeQOOE9LXtc06IHU5K9pe...
24520572@gm.uit.edu....		27 déc. 2025	27 déc. 2025	Nwzm5MPOh7ReLipZVfjRHG...

Bảng 2.3.1 Tài khoản trên Authentication Firebase

#### Giải thích:

Hỗ trợ xác thực người dùng như: đăng ký, đăng nhập, mã hoá mật khẩu, tạo tài khoản và lưu trữ thông tin xác thực an toàn. Trong đồ án, nhóm có thể sử dụng Firebase Authentication để quản lý phiên đăng nhập, hoặc quản lý tài khoản trong Realtime Database tùy theo cấu hình.

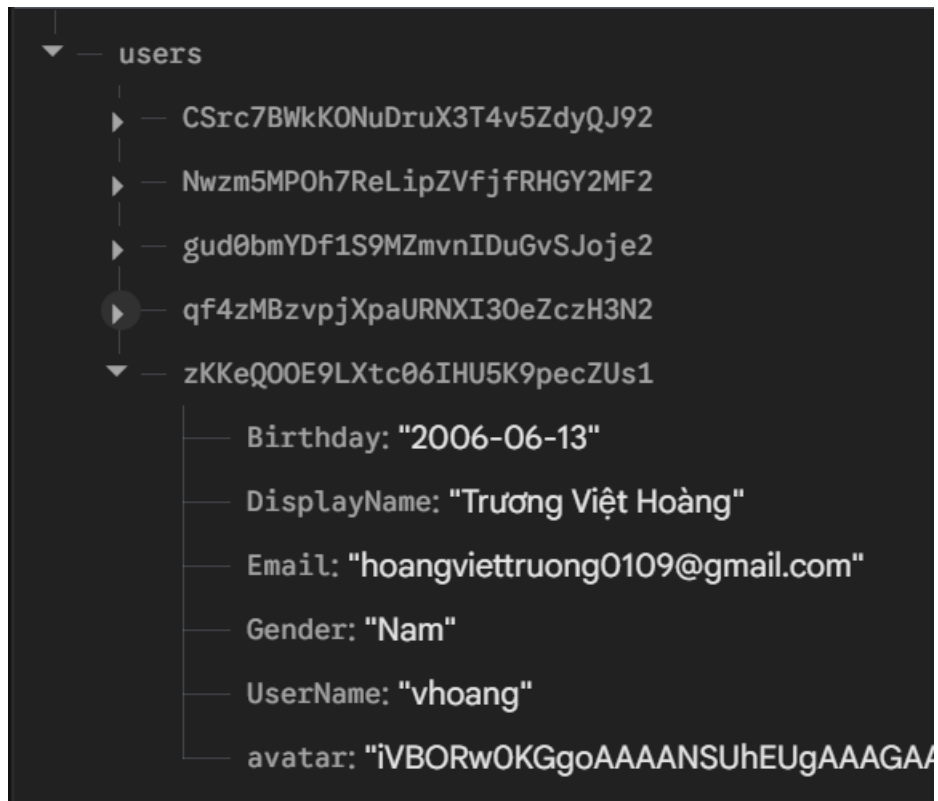
Ngoài ra, hệ thống có cơ chế xác minh email/OTP (nếu bật) nhằm hạn chế đăng ký ảo và hỗ trợ quên mật khẩu. Mã OTP được gửi qua email và chỉ có hiệu lực trong một khoảng thời gian ngắn.

### 2.3.2 Realtime Database (Firebase)

#### Users

STT	Tên Trường	Kiểu Dữ Liệu	Ý Nghĩa
1	Uid (key)		Khóa định danh người dùng (id node trong users/{uid})
2	UserName	string	Tên tài khoản (username) dùng để đăng nhập/tìm kiếm
3	DisplayName	string	Tên hiển thị trong ứng dụng
4	Email	string	Email đăng ký / đăng nhập
5	Gender	string	Giới tính (ví dụ: “Nam”, “Nữ”,...)
6	Birthday	string	Ngày sinh của người dùng
7	avatar	string	Ảnh đại diện dạng Base64 (chuỗi)

Bảng 2.3.2.1 Bảng tổ chức dữ liệu Users



Bảng 2.3.2.2 Tổ chức dữ liệu Users trên Firebase

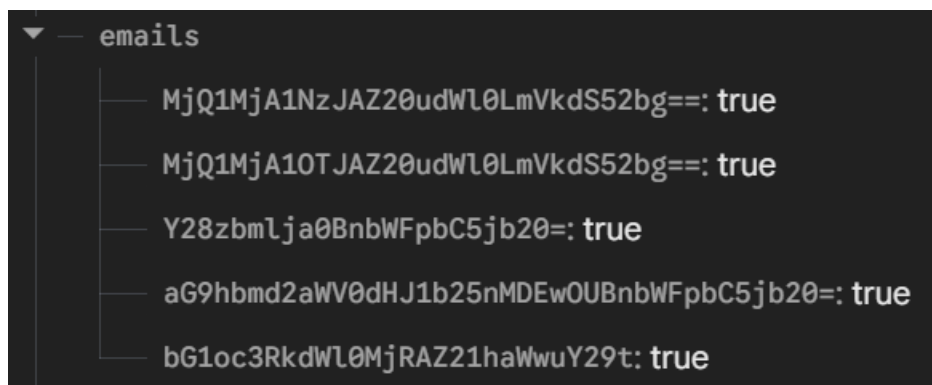
### Giải thích:

Trong nhánh **Users**, dữ liệu được chia theo từng **UID** của người dùng. Mỗi UID lưu các thông tin cơ bản như **UserName**, **DisplayName**, **Email**, **Gender**, **Birthday** và (nếu có) **avatar**, dùng để hiển thị hồ sơ và định danh người dùng trong ứng dụng.

### Emails

STT	Tên Trường	Kiểu Dữ Liệu	Ý Nghĩa
1	Email (đã mã hoá)	bool	Các email đã được đăng ký

Bảng 2.3.2.3 Bảng tổ chức dữ liệu Emails



Bảng 2.3.2.4 Tổ chức dữ liệu Emails trên Firebase

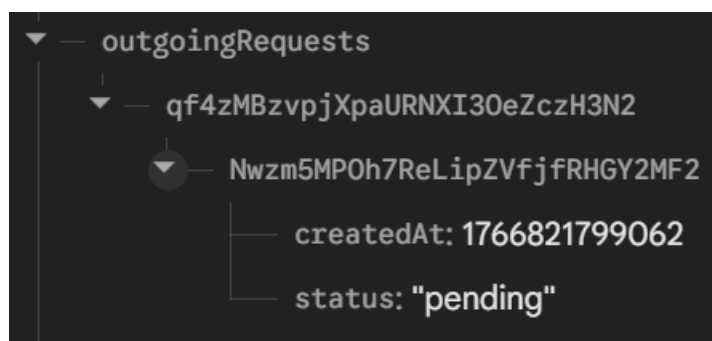
### Giải thích:

Trong nhánh emails, dữ liệu được lưu theo dạng bảng tra cứu email. Mỗi “khóa” là chuỗi email đã được mã hoá/encode để dùng làm key an toàn trong Firebase, và giá trị true dùng để đánh dấu email đó đã tồn tại trong hệ thống. Nhánh này thường được dùng để kiểm tra trùng email khi đăng ký, hỗ trợ tìm tài khoản/khôi phục mật khẩu, và tăng tốc tra cứu thay vì phải quét toàn bộ users.

### outgoingRequests

STT	Tên Trường	Kiểu Dữ Liệu	Ý Nghĩa
1	UID (người gửi)		Mã định danh người gửi (node cha)
2	UID (người nhận)		Mã định danh người nhận (node con)
3	createdAt	int	Thời điểm gửi
4	status	string	Trạng thái

Bảng 2.3.2.5 Bảng tổ chức dữ liệu outgoingRequests



Bảng 2.3.2.6 Tổ chức dữ liệu outgoingRequests trên Firebase

### Giải thích:

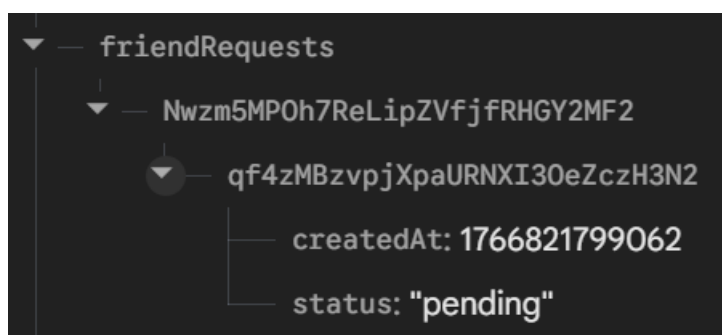
Trong nhánh outgoingRequests, dữ liệu được chia theo từng UID của người gửi. Bên trong mỗi UID sẽ có các node con là UID của người nhận, lưu thông tin yêu cầu đã gửi như createdAt (thời

điểm gửi) và status (trạng thái, ví dụ pending). Nhánh này dùng để theo dõi các lời mời đã gửi, hiển thị “đang chờ”, và hỗ trợ hủy yêu cầu hoặc cập nhật khi bên kia chấp nhận/từ chối.

### *friendRequests*

STT	Tên Trường	Kiểu Dữ Liệu	Ý Nghĩa
1	UID (người nhận)		Mã định danh người nhận (node cha)
2	UID (người gửi)		Mã định danh người gửi (node con)
3	createdAt	int	Thời điểm gửi
4	status	string	Trạng thái

Bảng 2.3.2.7 Bảng tổ chức dữ liệu friendRequests



Bảng 2.3.2.8 Tổ chức dữ liệu friendRequests trên Firebase

### **Giải thích:**

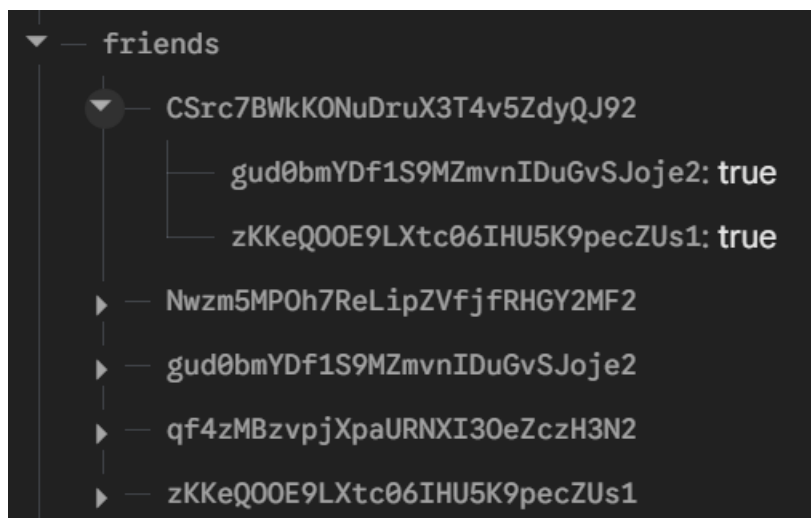
Trong nhánh friendRequests, dữ liệu được chia theo từng UID của người nhận. Bên trong mỗi UID sẽ có các node con là UID của người gửi, lưu thông tin của lời mời kết bạn như createdAt (thời điểm gửi) và status (trạng thái, ví dụ pending). Nhánh này dùng để hiển thị danh sách lời mời, và xử lý chấp nhận/từ chối yêu cầu kết bạn.

### *friends*

STT	Tên Trường	Kiểu Dữ Liệu	Ý Nghĩa
1	UID (key)		Mã định danh người dùng
2	UID (Bạn bè)	bool	Danh sách UID bạn bè

Bảng 2.3.2.9 Bảng tổ chức dữ liệu friends





Bảng 2.3.2.10 Tổ chức dữ liệu friends trên Firebase

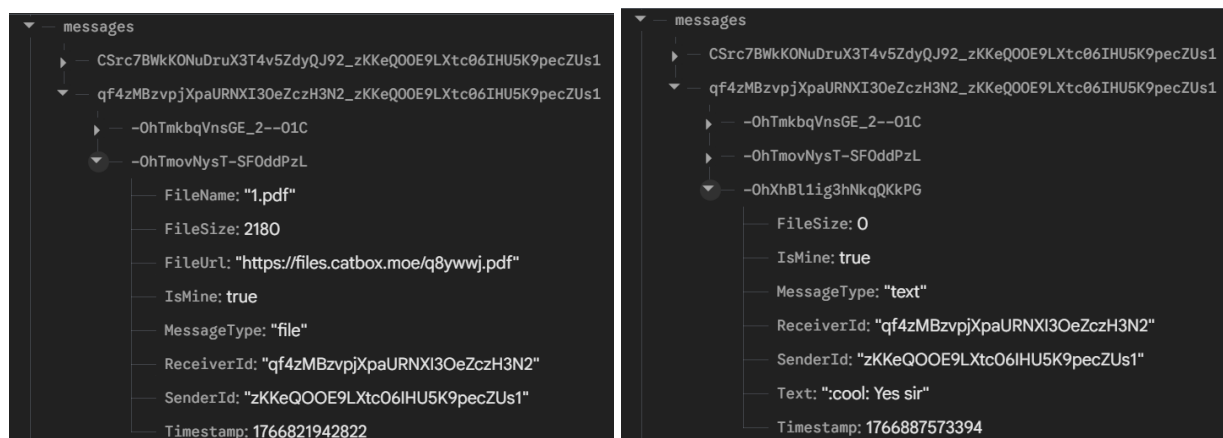
### Giải thích:

Trong nhánh friends, dữ liệu được chia theo từng UID của người dùng. Bên trong mỗi UID sẽ là danh sách UID bạn bè, với giá trị true dùng để đánh dấu đã là bạn bè. Cấu trúc này giúp hệ thống truy xuất nhanh danh sách bạn bè, kiểm tra quan hệ bạn bè (có/không), và hỗ trợ huỷ kết bạn bằng cách xoá UID tương ứng ở hai phía.

### messages

STT	Tên Trường	Kiểu Dữ Liệu	Ý Nghĩa
1	ConversationID		Mã cuộc trò chuyện
2	MessageID		Mã tin nhắn
3	MessageType	String	Loại tin nhắn (Text,File)
4	SenderId	String	Mã định danh người gửi
5	ReceiverId	String	Mã định danh người nhận
6	Timestamp	Int	Thời gian gửi
7	IsMine	bool	Đánh dấu tin nhắn của chính người gửi
8	Text	String	Tin nhắn dạng văn bản và emoji đã được chuyển thành văn bản.
9	FileSize	Int	Kích thước File
10	FileName	String	Tên File
11	FileUrl	string	Url dẫn tới file lưu ở Catbox

Bảng 2.3.2.11 Bảng tổ chức dữ liệu messages



Bảng 2.3.2.12 Tổ chức dữ liệu messages trên Firebase

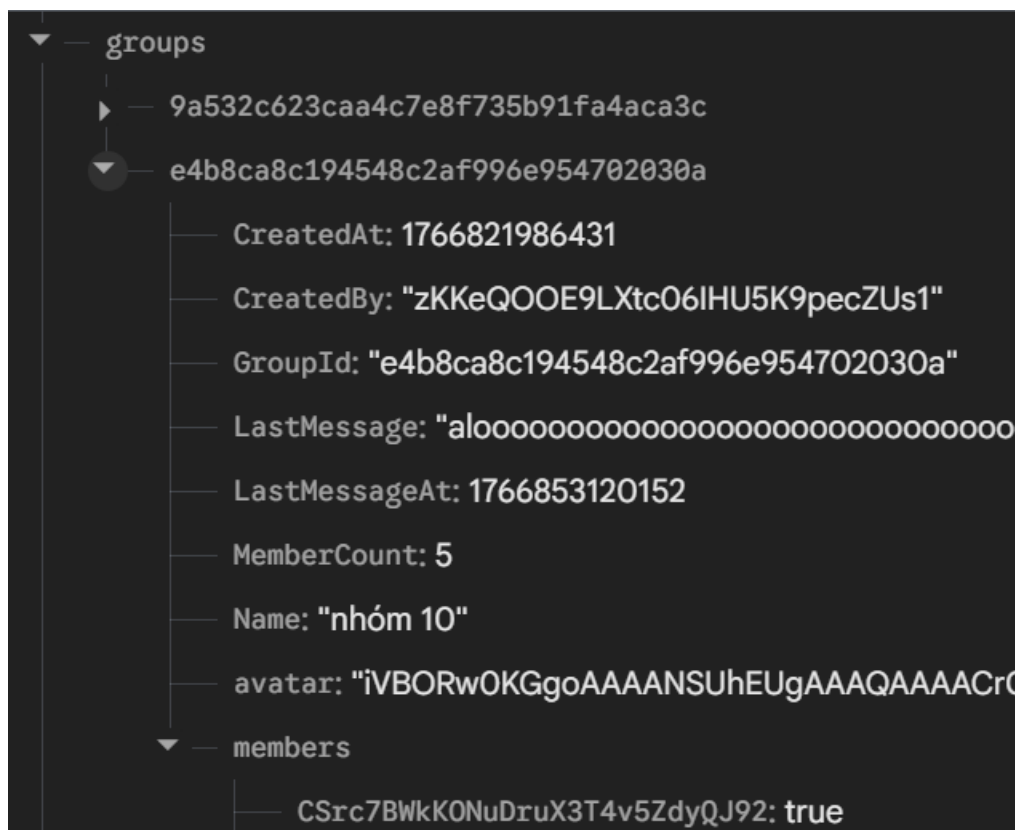
### Giải thích:

Trong nhánh messages, dữ liệu được chia theo từng conversationId (thường là ghép từ uidA\_uidB). Bên trong mỗi cuộc trò chuyện sẽ có nhiều node con là messageId (Firebase auto key), lưu nội dung từng tin nhắn và metadata. Mỗi tin nhắn có MessageType để phân loại (ví dụ text, file), kèm các thông tin như SenderId, ReceiverId, Timestamp, IsMine. Với tin nhắn văn bản sẽ có trường Text; với tin nhắn file sẽ có FileName, FileSize, FileUrl (URL lưu trên Catbox). Cấu trúc này giúp hệ thống lấy lịch sử chat theo từng cuộc trò chuyện, đồng thời hỗ trợ realtime khi có tin mới.

### groups

STT	Tên Trường	Kiểu Dữ Liệu	Ý Nghĩa
1	GroupId		Mã nhóm
2	CreatedAt	Int	Thời gian tạo nhóm
3	CreatedBy	String	Mã người tạo
4	GroupID	String	Mã nhóm
5	LastMessage	String	Tin nhắn gần nhất
6	LastMessageAt	Int	Thời gín gửi của tin nhắn gần nhất
7	MemberCount	Int	Số lượng thành viên
8	Name	String	Tên nhóm
9	Avatar	String	Avatar nhóm
10	Member		Lưu các thành viên
11	UID member	bool	Mã của các thành viên

Bảng 2.3.2.13 Bảng tổ chức dữ liệu groups



Bảng 2.3.2.14 Tổ chức dữ liệu groups trên Firebase

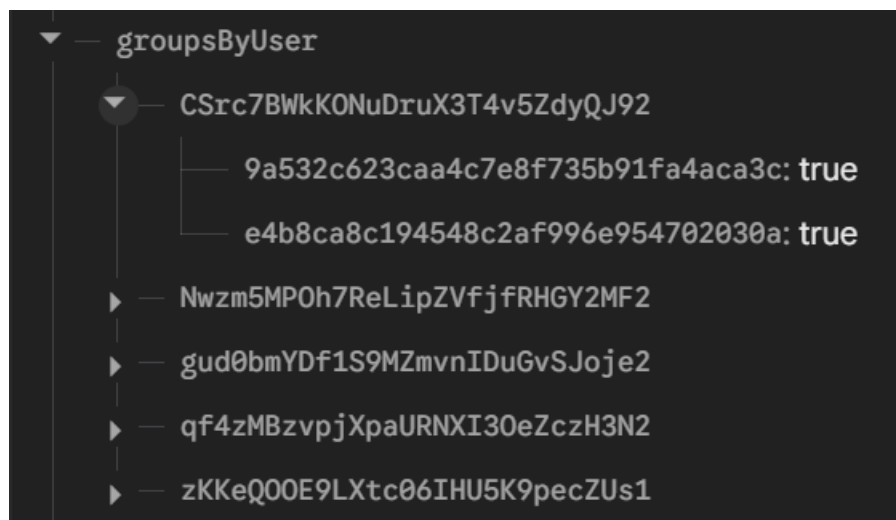
### Giải thích:

Trong nhánh groups, dữ liệu được chia theo từng GroupId (mỗi node con là một nhóm chat). Mỗi nhóm lưu các thông tin như Name (tên nhóm), CreatedAt, CreatedBy (UID người tạo), MemberCount, LastMessage và LastMessageAt để hiển thị nhanh ở danh sách nhóm. Ngoài ra nhóm có nhánh members chứa danh sách UID thành viên với giá trị true để đánh dấu thuộc nhóm, và có thể có avatar (Base64/URL) làm ảnh đại diện nhóm. Cấu trúc này giúp hệ thống quản lý nhóm, tải danh sách nhóm nhanh và kiểm tra quyền thành viên trước khi cho nhắn tin.

### *groupByUser*

STT	Tên Trường	Kiểu Dữ Liệu	Ý Nghĩa
1	UID		Mã định danh người dùng
2	groupId	bool	Id các nhóm mà người dùng tham gia

Bảng 2.3.2.15 Bảng tổ chức dữ liệu groupByUsers



Bảng 2.3.2.16 Tổ chức dữ liệu groupByUsers trên Firebase

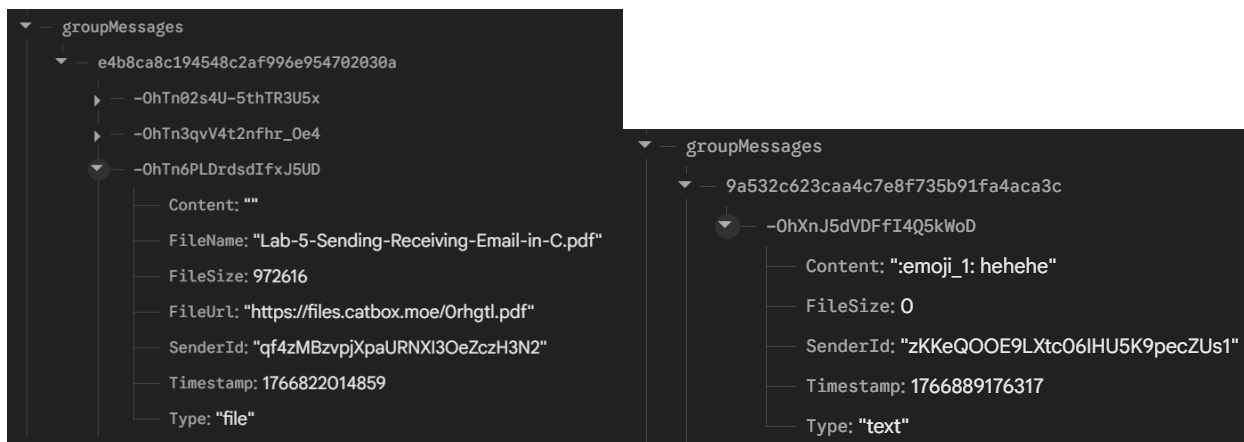
### Giải thích:

Trong nhánh groupsByUser, dữ liệu được chia theo từng UID của người dùng. Bên trong mỗi UID là danh sách các GroupId mà người đó tham gia, với giá trị true để đánh dấu. Nhánh này dùng để truy xuất nhanh danh sách nhóm của một người dùng mà không cần quét toàn bộ groups, giúp tải giao diện danh sách nhóm nhanh hơn và thuận tiện khi kiểm tra người dùng đang thuộc những nhóm nào.

### groupMessages

STT	Tên Trường	Kiểu Dữ Liệu	Ý Nghĩa
1	GroupId		Mã nhóm
2	MessageId		Mã tin nhắn
3	Type	String	Loại tin nhắn (Text, File)
4	Content	String	Nội Dung
5	SenderId	String	Mã người gửi
6	Timestamp	Int	Thời gian gửi
7	FileSize	Int	Kích cỡ File
8	FileName	String	Tên File
9	FileUrl	string	Url dẫn tới file lưu ở Catbox

Bảng 2.3.2.17 Bảng tổ chức dữ liệu groupMessages



Bảng 2.3.2.18 Tổ chức dữ liệu groupMessages trên Firebase

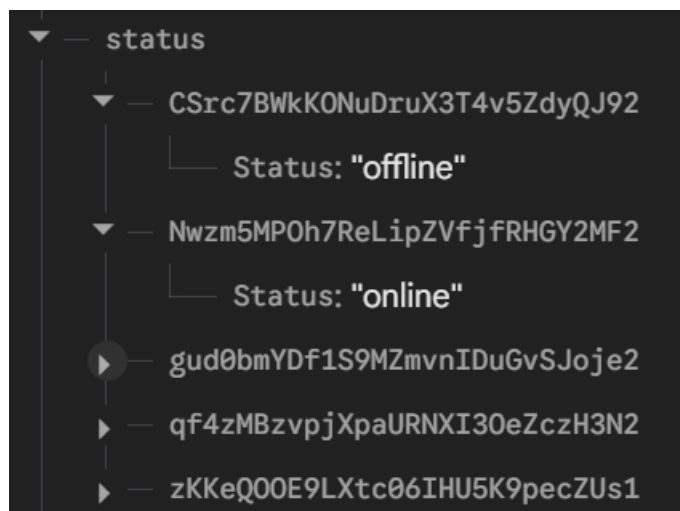
### Giải thích:

Trong nhánh groupMessages, dữ liệu được chia theo từng GroupId. Bên trong mỗi nhóm sẽ có nhiều node con là messageId (Firebase auto key), lưu nội dung tin nhắn của nhóm. Mỗi tin nhắn có Type để phân loại (ví dụ text, file), kèm SenderId và Timestamp. Với tin nhắn văn bản sẽ dùng Content; với tin nhắn file sẽ có thêm FileName, FileSize, FileUrl (URL lưu trên Catbox). Cấu trúc này giúp hệ thống lấy lịch sử chat theo nhóm và hỗ trợ nhận tin realtime cho tất cả thành viên trong nhóm.

### Status

STT	Tên Trường	Kiểu Dữ Liệu	Ý Nghĩa
1	UID		Mã định danh người dùng
2	status	string	Trạng thái người dùng (online, offline)

Bảng 2.3.2.19 Bảng tổ chức dữ liệu Status



Bảng 2.3.2.20 Tổ chức dữ liệu status trên Firebase

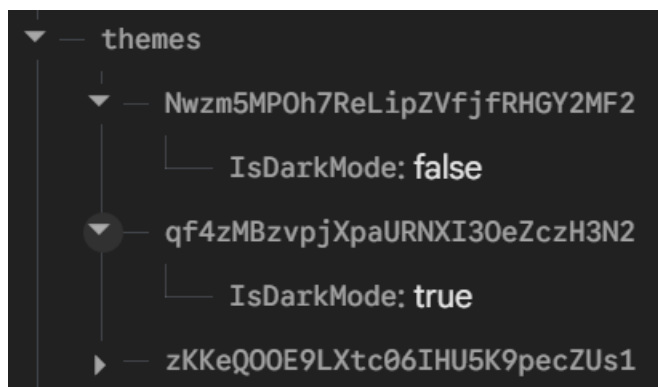
### Giải thích:

Trong nhánh status, dữ liệu được chia theo từng UID của người dùng. Mỗi UID lưu trường Status (ví dụ online, offline) để thể hiện trạng thái hoạt động của người dùng. Nhánh này giúp ứng dụng hiển thị online/offline trong danh sách bạn bè, trong khung chat, và cập nhật realtime khi người dùng đổi trạng thái.

### Themes

STT	Tên Trường	Kiểu Dữ Liệu	Ý Nghĩa
1	UID		Mã định danh người dùng
2	IsDarkMode	bool	Chế độ ngày đêm của người dùng

Bảng 2.3.2.21 Bảng tổ chức dữ liệu Themes



Bảng 2.3.2.22 Tổ chức dữ liệu Themes trên Firebase

**Giải thích:**

Trong nhánh themes, dữ liệu được chia theo từng UID của người dùng. Mỗi UID lưu tùy chọn giao diện như IsDarkMode (true/false) để xác định chế độ tối/sáng mà người dùng đã chọn. Nhánh này giúp ứng dụng ghi nhớ cài đặt theme, và khi đăng nhập lại sẽ tự áp dụng đúng chế độ ngày/đêm cho từng tài khoản.

### 2.3.3 Storage (Catbox & lưu trữ cục bộ)

**Tài liệu nhóm (tệp đính kèm)**

**Giải thích:** Các tệp đính kèm (ảnh, tài liệu) được lưu ở dịch vụ lưu trữ ngoài (Catbox) để giảm tải cho Realtime Database và giúp tải nhanh. Firebase chỉ lưu URL và metadata của tệp.

- Upload tệp lên Catbox → nhận URL chia sẻ (public link).
- Ghi URL + FileName + FileSize + Timestamp vào messages/groupMessages; khi người dùng bấm tải, client tải file từ URL và có thể cache cục bộ.

## CHƯƠNG 3: THIẾT KẾ GIAO DIỆN VÀ CODE CHỨC NĂNG

### 3.1 Giao diện đăng nhập.



The screenshot shows a web browser window with the title 'ĐĂNG NHẬP'. The main content is a login form with a blue background. The form has a white rounded rectangle in the center with the title 'ĐĂNG NHẬP HỆ THỐNG'. Below the title, there are two input fields: the first is for 'Email Hoặc Tên Đăng Nhập' with a user icon, and the second is for 'Mật Khẩu' with a lock icon and a toggle eye icon. Below the password field is a link 'Quên mật khẩu'. At the bottom of the form are two buttons: 'Đăng Nhập' (blue) and 'Đăng Ký' (black).

Bảng 3.1.1 Giao diện đăng nhập

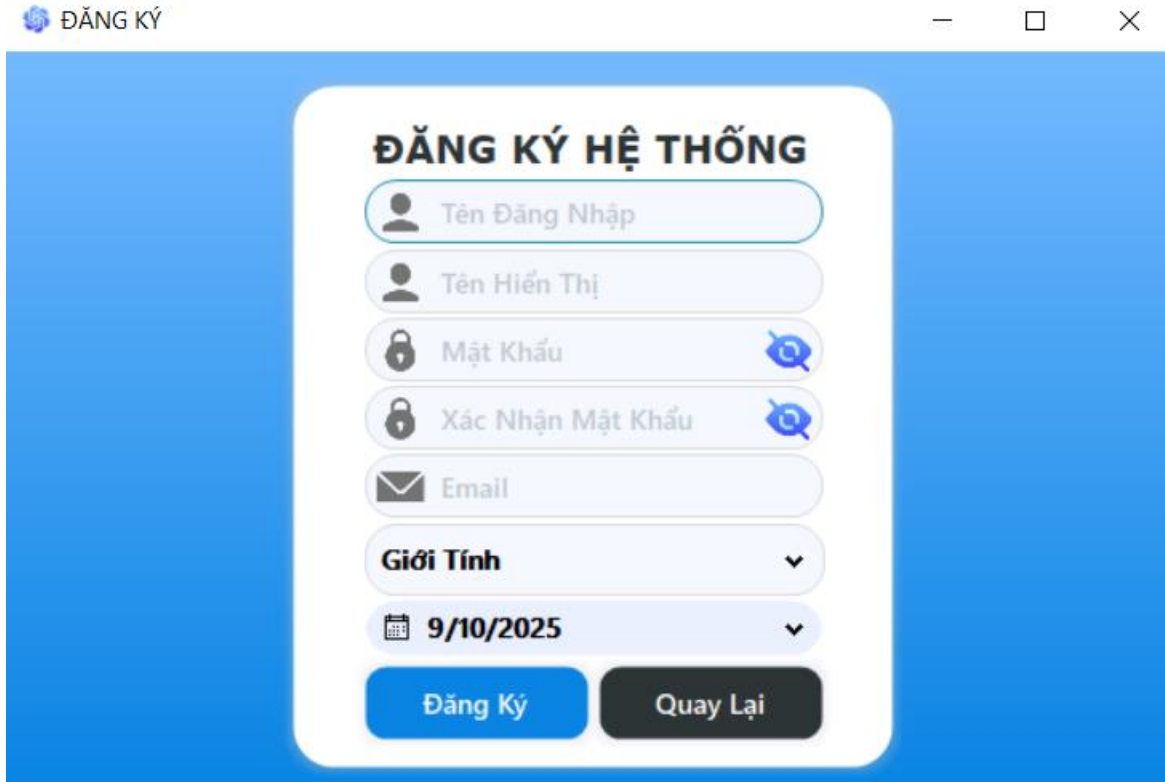
Cần điền đầy đủ thông tin trước khi đăng nhập nếu không sẽ không thể đăng nhập.

Báo lỗi nếu không nhập đầy đủ thông tin hoặc một vài lỗi xảy ra khi đăng nhập.

Có Link quên mật khẩu để chuyển qua Form quên mật khẩu và Nút đăng ký để chuyển qua Form đăng ký.



### 3.2 Giao diện đăng ký.

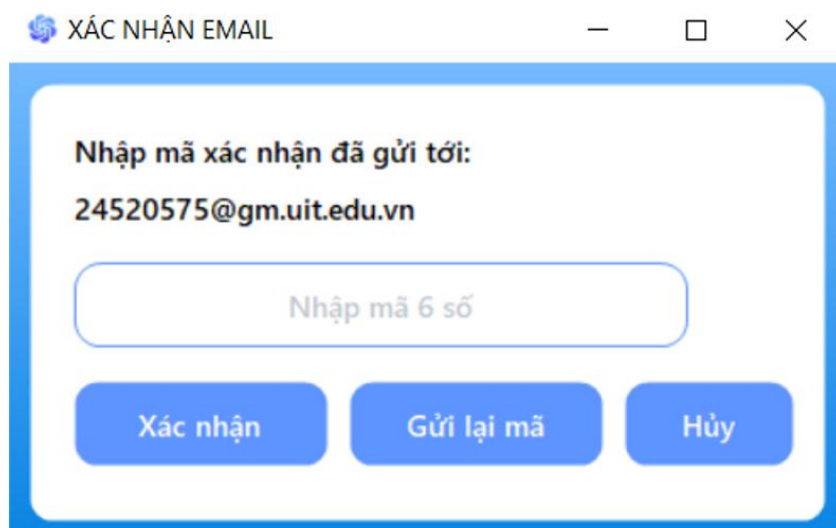


The screenshot shows a registration window titled "ĐĂNG KÝ" (Registration). The form is titled "ĐĂNG KÝ HỆ THỐNG" (System Registration) and contains the following fields and controls:

- Tên Đăng Nhập (Login Name)
- Tên Hiển Thị (Display Name)
- Mật Khẩu (Password) with a toggle icon
- Xác Nhận Mật Khẩu (Confirm Password) with a toggle icon
- Email
- Giới Tính (Gender) dropdown menu
- 9/10/2025 date selector
- Đăng Ký (Register) button
- Quay Lại (Back) button

Bảng 3.2.1. Giao diện đăng ký

Người dùng sẽ phải điền đầy đủ các thông tin thì mới đăng ký được.

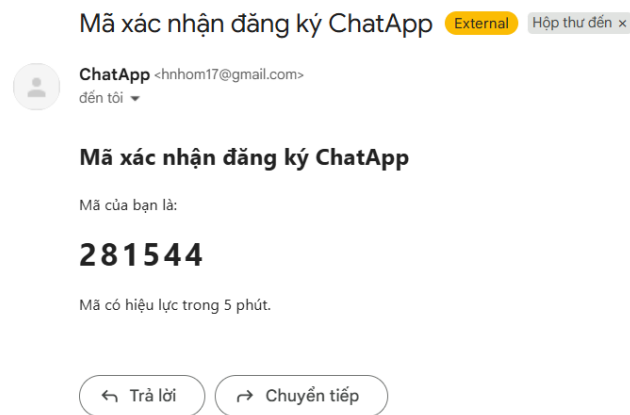


The screenshot shows an email confirmation window titled "XÁC NHẬN EMAIL" (Email Confirmation). The form contains the following elements:

- Nhập mã xác nhận đã gửi tới: 24520575@gm.uit.edu.vn
- Nhập mã 6 số (Enter 6-digit code) input field
- Xác nhận (Confirm) button
- Gửi lại mã (Resend code) button
- Hủy (Cancel) button

Bảng 3.2.2 Giao diện Xác nhận email

## Báo cáo cuối kỳ: Ứng dụng nhắn tin (Chat App)



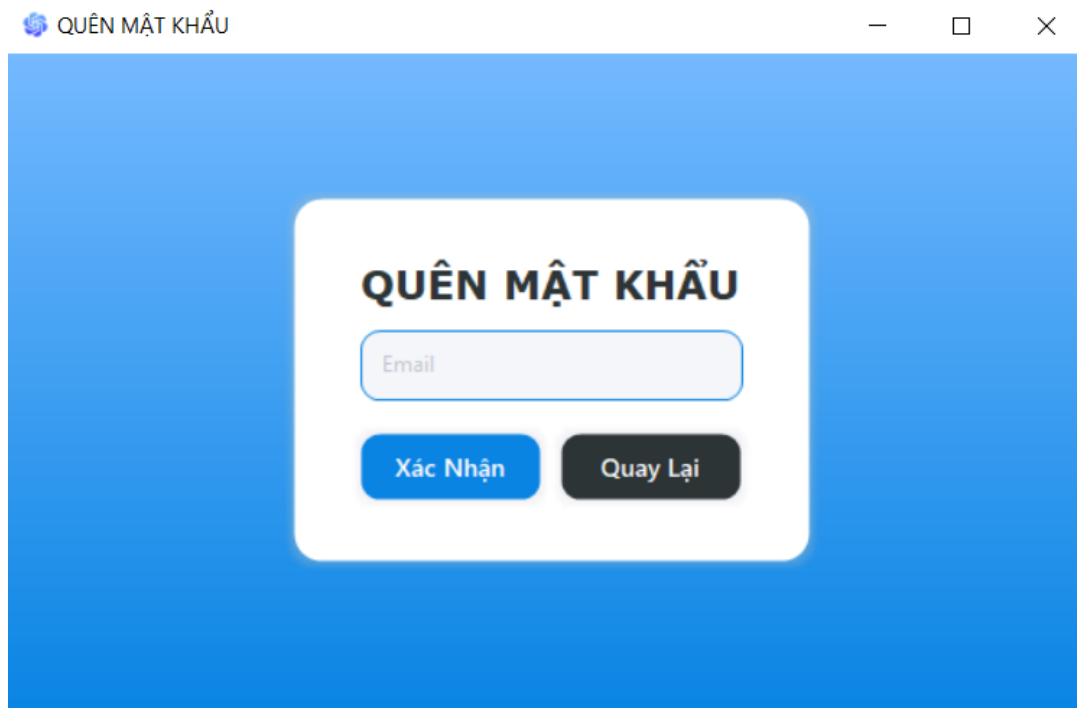
**Bảng 3.2.3 Gửi mã xác nhận tài khoản có tồn tại về mail**

Khi điền email và đầy đủ thông tin xong, bấm Nút đăng ký sẽ hiện Form Xác nhận mật khẩu, đồng thời gửi mã xác nhận về email người dùng.

Nếu có các lỗi như xác nhận mật khẩu không khớp, nhập sai mã xác nhận, không gửi được mã,... thì sẽ có thông báo lỗi.

Nút quay lại để quay lại Form đăng nhập.

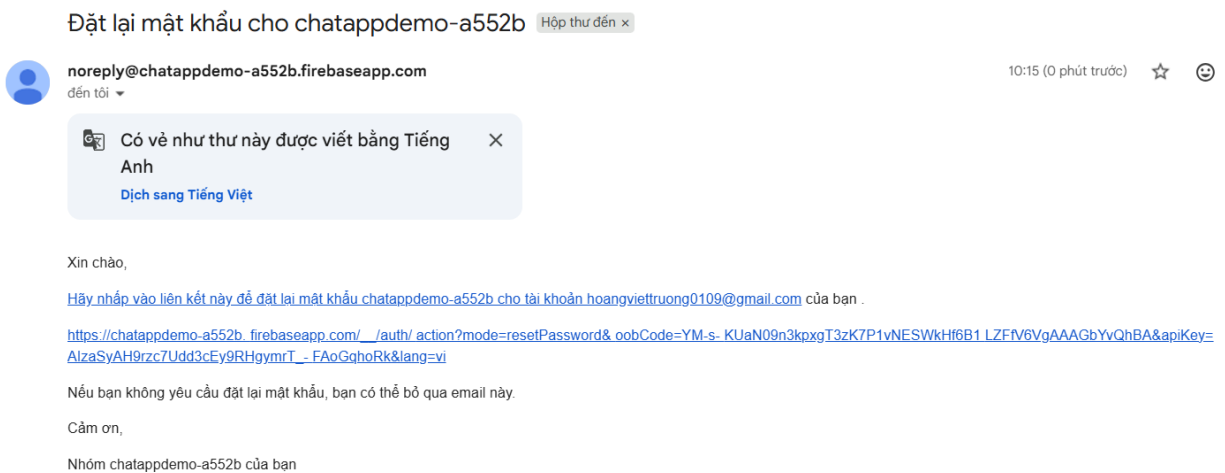
### 3.3 Giao diện quên mật khẩu.



**Bảng 3.3.1 Giao diện Quên mật khẩu**

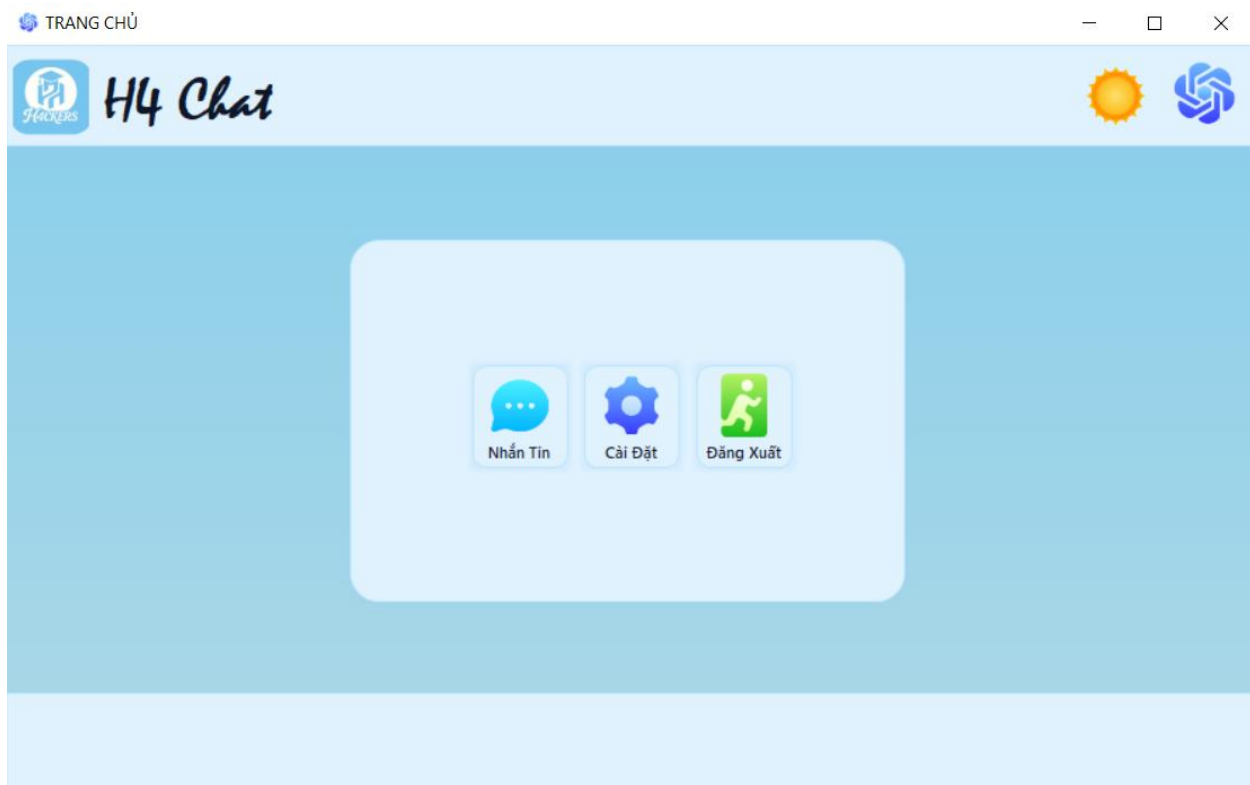
Báo cáo cuối kỳ: Ứng dụng nhắn tin (Chat App)

Điền email đã đăng ký tài khoản sau đó bấm Nút xác nhận, hệ thống sẽ gửi email để bạn đổi mật khẩu (Kiểm tra tin nhắn spam, tin nhắn rác,...).

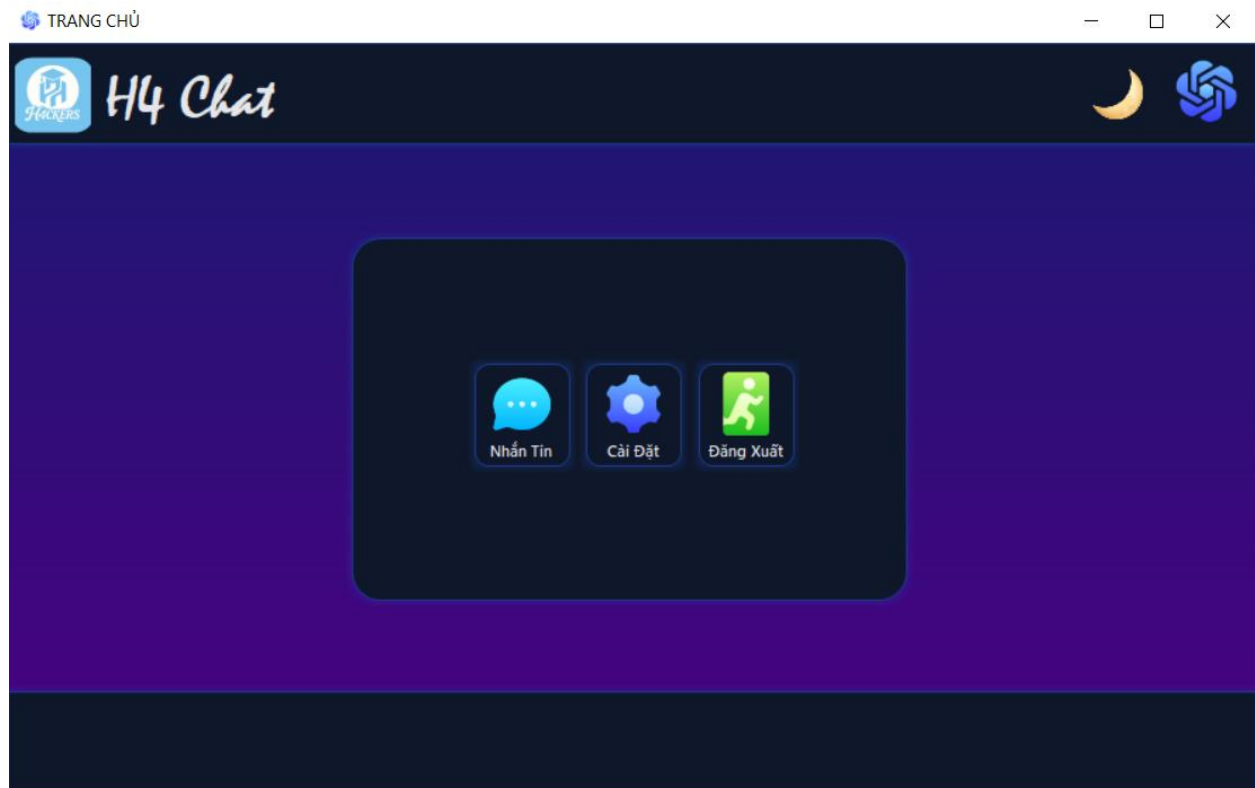


Bảng 3.3.2 Gửi link đổi mật khẩu về mail

### 3.4 Giao diện trang chủ.



Bảng 3.4.1 Giao diện trang chủ (Chế độ ban ngày)



**Bảng 3.4.2** Giao diện trang chủ (Chế độ ban đêm)

Khi đăng nhập thành công thì hệ thống sẽ chuyển đến Form trang chủ.

Có các nút điều hướng như:

**Nút nhắn tin:** chuyển đến Form nhắn tin.

**Nút cài đặt:** chuyển đến Form cài đặt.


**Nút đăng xuất:** để đăng xuất.

Ngoài ra còn có biểu tượng mặt trời / mặt trăng để chuyển qua lại chế độ giữa ngày và đêm.

### 3.5 Giao Diện Cài Đặt

Cài đặt tài khoản

**CÀI ĐẶT TÀI KHOẢN**



[Đổi avatar](#)

Tên Đăng Nhập :  [Đổi tên đăng nhập](#)

Mật Khẩu :  [Đổi mật khẩu](#)

Tên Hiển Thị :  [Đổi tên hiển thị](#)

Giới Tính :  [Đổi giới tính](#)

Ngày Sinh :  [Đổi ngày sinh](#)


Email :

[Đóng](#)

Bảng 3.5.1 Giao diện Cài đặt (Chế độ ban ngày)

Cài đặt tài khoản

**CÀI ĐẶT TÀI KHOẢN**



[Đổi avatar](#)

Tên Đăng Nhập :  [Đổi tên đăng nhập](#)

Mật Khẩu :  [Đổi mật khẩu](#)

Tên Hiển Thị :  [Đổi tên hiển thị](#)

Giới Tính :  [Đổi giới tính](#)

Ngày Sinh :  [Đổi ngày sinh](#)

Email :

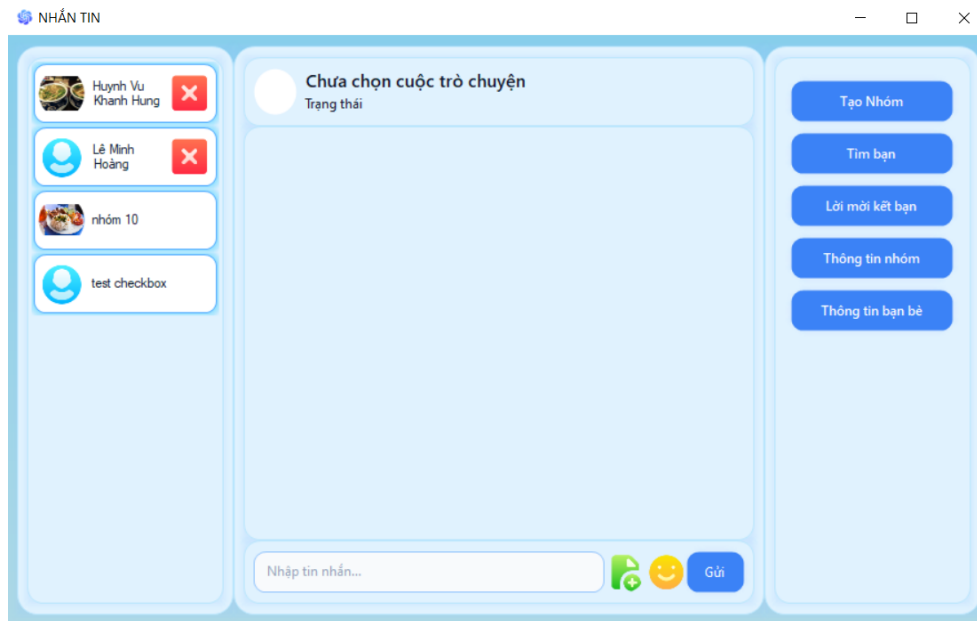
[Đóng](#)

Bảng 3.5.2 Giao diện Cài đặt (Chế độ ban đêm)

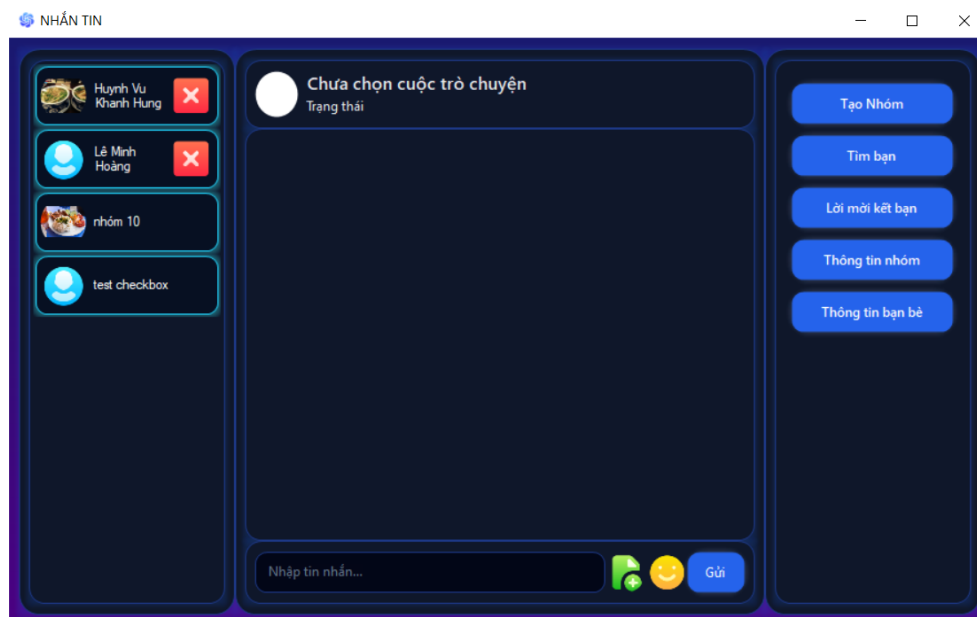
Báo cáo cuối kỳ: Ứng dụng nhắn tin (Chat App)

Form này để đổi avatar, tên đăng nhập, mật khẩu, tên hiển thị, giới tính, ngày sinh bằng cách nhập vào ô và bấm Nút đổi tương ứng; Riêng avatar khi bấm đổi sẽ được chọn ảnh trong quản lý tệp của máy.

### 3.6 Giao diện Nhắn Tin



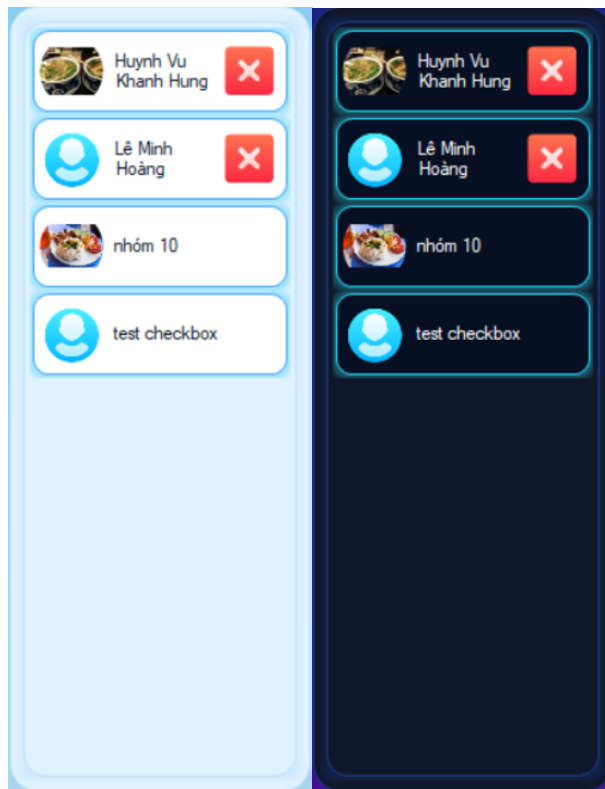
Bảng 3.6.1 Giao diện Nhắn tin (Chế độ ban ngày)



Bảng 3.6.2 Giao diện Cài đặt (Chế độ ban đêm)

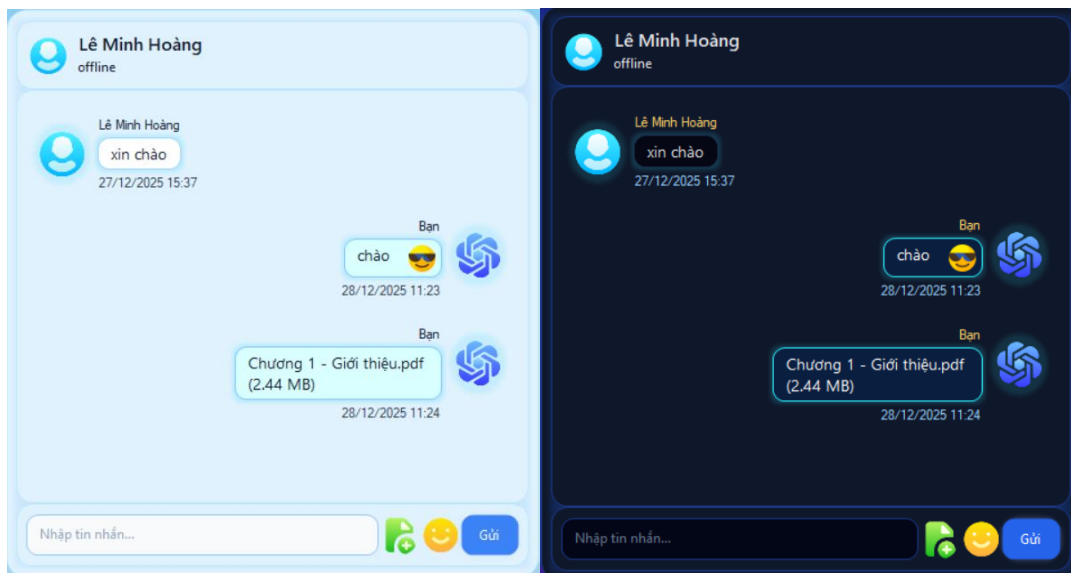
Khi mới đầu vào Form nhắn tin, chưa có tin nhắn (ở giữa) nào bởi vì ta chưa chọn Cuộc trò chuyện (bên trái) nào.

## Báo cáo cuối kỳ: Ứng dụng nhắn tin (Chat App)



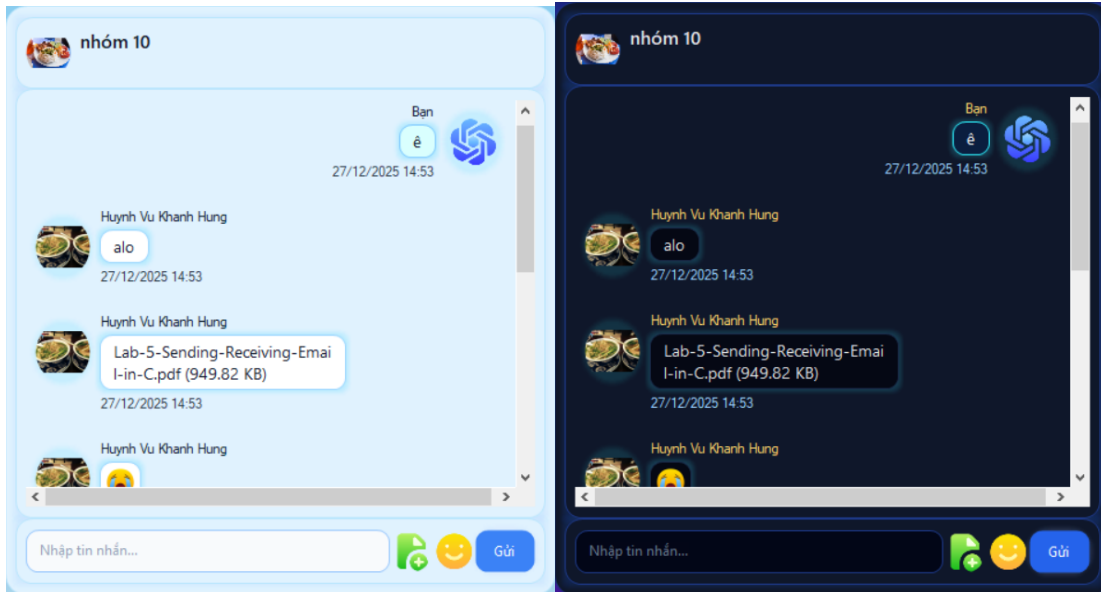
**Bảng 3.6.3 Các Conversation (panel Trái)**

Phía bên trái là danh sách các Cuộc trò chuyện nhóm hoặc bạn bè, cũng xem nó như một danh sách bạn bè cũng được. Ở các cuộc trò chuyện với bạn bè có Nút X để hủy kết bạn.



**Bảng 3.6.4 Giao diện Chat riêng (panel Giữa)**

## Báo cáo cuối kỳ: Ứng dụng nhắn tin (Chat App)



**Bảng 3.6.5 Giao diện Chat nhóm (panel Giữa)**



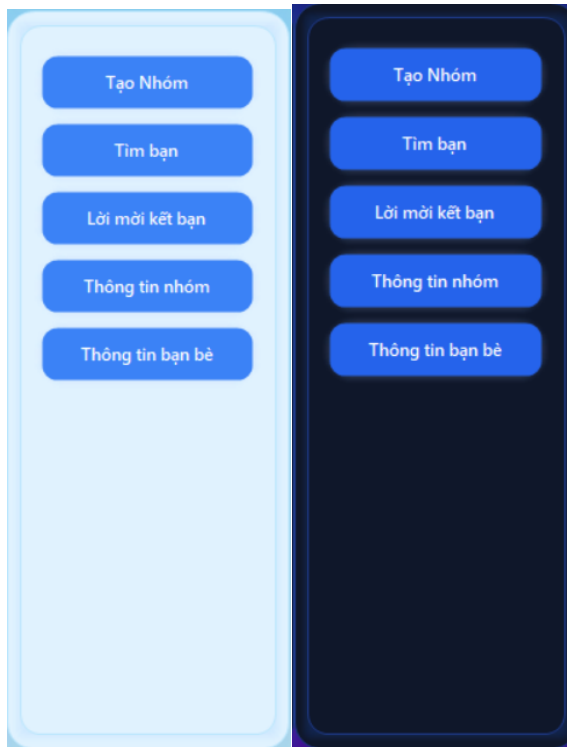
**Bảng 3.6.6 Các loại emoji**

Ở giữa là vùng để nhắn tin:

Dùng để coi tin nhắn, viết tin nhắn, gửi tin nhắn dạng văn bản, emoji và file.

Dùng để coi mình đang ở trong Cuộc trò chuyện nào và coi trạng thái (online/offline) nếu đó là nhắn tin với bạn bè.





**Bảng 3.6.7 Các nút chức năng (panel Phải)**

Phía bên phải là các Nút thực hiện vài công việc như sau:

Tạo Nhóm: để tạo nhóm.

Tìm bạn: để tìm và gửi yêu cầu kết bạn.

Lời mời kết bạn: để chấp nhận hoặc từ chối các yêu cầu kết bạn được gửi tới.

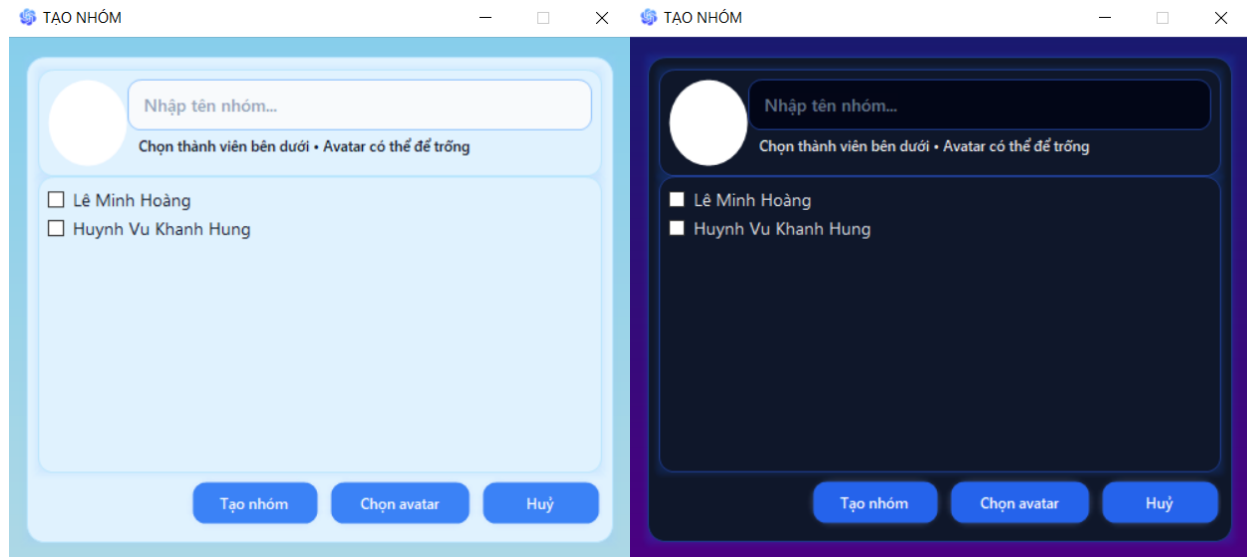
Thông tin nhóm: Chỉ hoạt động khi bạn đã chọn một nhóm (bên trái) để nhắn tin trước đó. Dùng để thêm thành viên, đổi một số thông tin nhóm.

Thông tin bạn bè: Chỉ hoạt động khi bạn đã chọn một người (bên trái) để nhắn tin trước đó. Dùng để xem một số thông tin bạn bè.

## 3.7 Giao diện tạo nhóm, tìm bạn và quản lý lời mời

Các màn hình phụ dưới đây hỗ trợ hoàn thiện trải nghiệm nhắn tin và làm việc nhóm.

### 3.7.1 Giao diện tạo nhóm

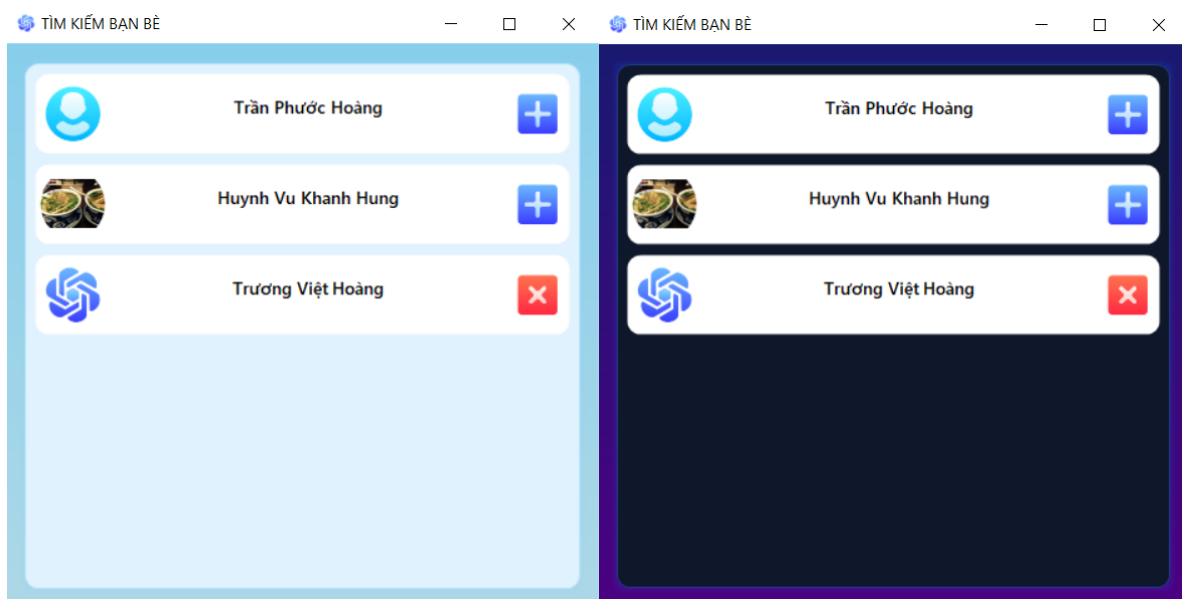


Hình 3.7.1 Giao diện tạo nhóm

Nhập tên nhóm, chọn các bạn bè tham gia nhóm, có thể chọn avatar ở Nút chọn avatar và cuối cùng nhấn Nút tạo nhóm.

Nút huỷ để đóng form.

### 3.7.2 Giao diện tìm kiếm bạn bè

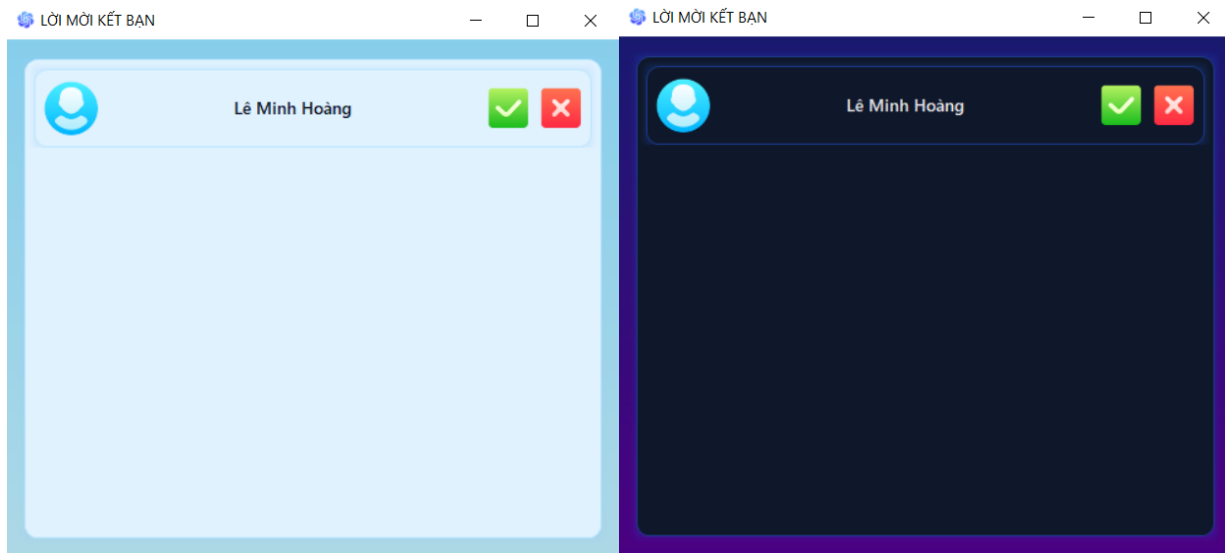


Hình 3.7.2 Giao diện tìm kiếm bạn bè

Báo cáo cuối kỳ: Ứng dụng nhắn tin (Chat App)

Form tự hiển thị những người chưa phải là bạn bè lên danh sách, để người dùng có thể gửi yêu cầu kết bạn hoặc huỷ yêu cầu kết bạn nếu trước đó đã gửi r.

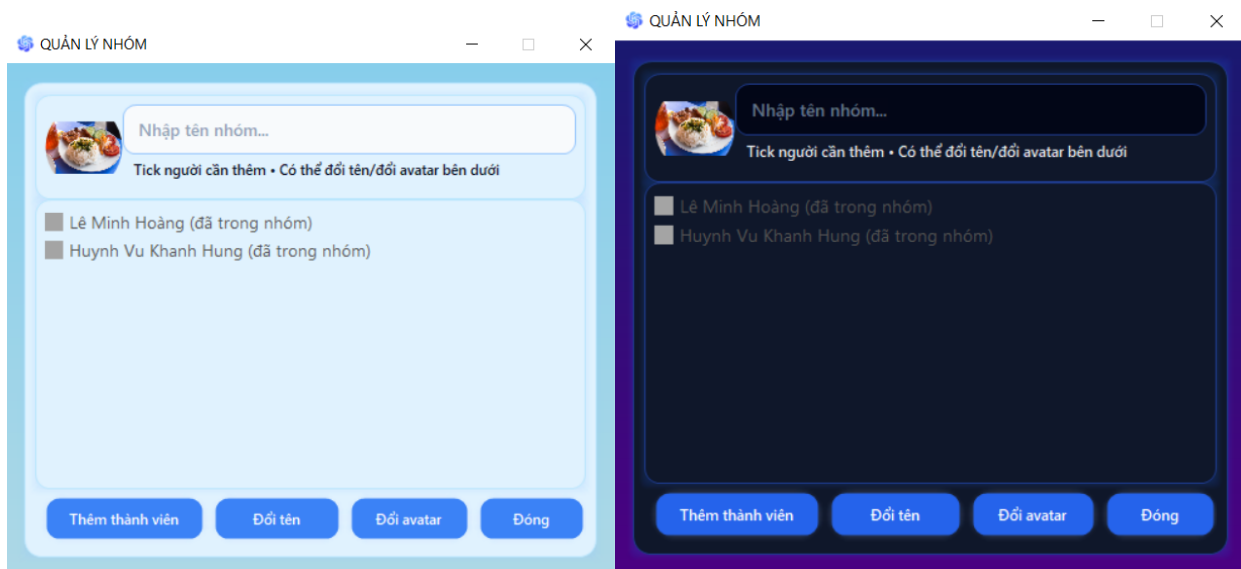
### 3.7.3 Giao diện lời mời kết bạn



Hình 3.7.3 Giao diện lời mời kết bạn

Hiển thị danh sách lời mời và cho phép chấp nhận hoặc từ chối yêu cầu kết bạn.

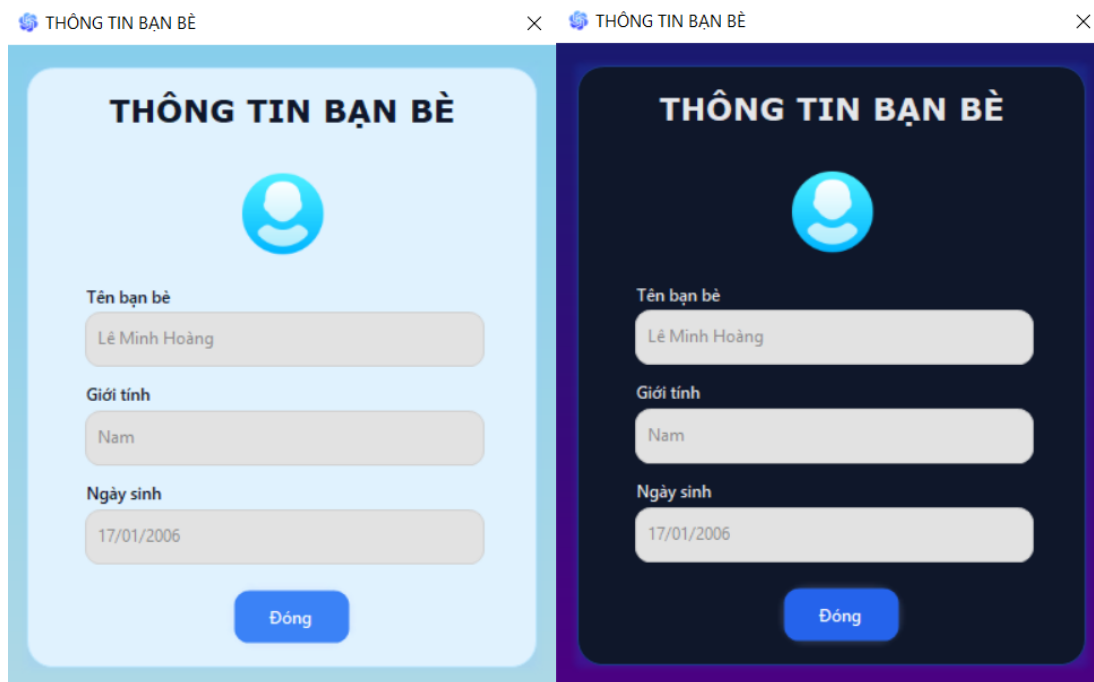
### 3.7.4 Giao diện thông tin nhóm



Hình 3.7.4 Giao diện thông tin nhóm

Xem danh sách thành viên và thao tác quản lý nhóm bao gồm: thêm thành viên, đổi tên nhóm và đổi avatar nhóm.

### 3.7.5 Giao diện thông tin bạn bè



Hình 3.7.5 Giao diện thông tin bạn bè

Hiện thị hồ sơ thông tin bạn bè bao gồm tên hiển thị, giới tính, ngày sinh và avatar.

## CHƯƠNG 4: KẾT QUẢ VÀ HƯỚNG PHÁT TRIỂN

### 4.1 Kết quả đạt được

- Vận dụng được các kiến thức đã học được trong môn Lập trình mạng căn bản để tạo ra được một sản phẩm thực tế.
- Học hỏi được nhiều kinh nghiệm trong hoạt động làm việc nhóm.
- Học được cách dùng C# và công cụ winforms, Firebase, Git, GitHub.

### 4.2 Hạn chế

- Cơ sở dữ liệu còn nhiều khuyết điểm
- Một số lỗi logic mà nhóm vẫn chưa thể khắc phục kịp thời.
- Còn khá ít tính năng.
- Tổ chức thư mục chưa được tốt.

### 4.3 Hướng phát triển trong tương lai

- Giải quyết được các lỗi logic mà người dùng có thể gặp phải trong quá trình trải nghiệm ứng dụng.

Báo cáo cuối kỳ: Ứng dụng nhắn tin (Chat App)

- Xây dựng một cơ sở dữ liệu tốt hơn
- Bổ sung thêm một số chức năng để chương trình ngày càng hoàn thiện hơn.
- Hoàn thiện tốt hơn về phân lập trình bằng ngôn ngữ C# và ngôn ngữ truy vấn dữ liệu.