

**ĐỒ ÁN MÔN HỌC LẬP TRÌNH TRÊN
MÔI TRƯỜNG WINDOW**

ỨNG DỤNG CHAT APP

Sinh Viên:

- | | |
|-----------------------------|-------------------------|
| 1. Hoàng Ân | Mssv: 2280600149 |
| 2. Phan Ngọc Cường | Mssv: 2280603457 |
| 3. Ngô Hữu Đức | Mssv: 2280600149 |
| 4. Nguyễn Quốc Trung | Mssv: 2280603457 |

GVHD: ThS. Nguyễn Huy Cường

Thành phố Hồ Chí Minh, năm 2025

**ĐỒ ÁN MÔN HỌC LẬP TRÌNH TRÊN
MÔI TRƯỜNG WINDOW**

ỨNG DỤNG CHAT APP

Sinh Viên:

- | | |
|-----------------------------|-------------------------|
| 1. Hoàng Ân | Mssv: 2280600149 |
| 2. Phan Ngọc Cường | Mssv: 2280600358 |
| 3. Ngô Hữu Đức | Mssv: 2280600725 |
| 4. Nguyễn Quốc Trung | Mssv: 2280603457 |

GVHD: ThS. Nguyễn Huy Cường

Thành phố Hồ Chí Minh, năm 2025

LỜI CAM ĐOAN

Chúng tôi xin khẳng định rằng toàn bộ nội dung trong bài báo cáo này là do chính chúng tôi thực hiện trong thời gian qua. Bài báo cáo này có tham khảo, hỗ trợ từ các tài liệu và giáo trình liên quan, với nguồn gốc được trích dẫn và chú thích đầy đủ. Chúng tôi cam kết không có bất kỳ sự sao chép nào khác. Tất cả các kết quả và số liệu trong đề tài đều là trung thực và khách quan. Chúng tôi hoàn toàn chịu trách nhiệm về lời cam kết này.

LỜI CẢM ƠN

Do lần đầu tiên viết báo cáo với khối lượng kiến thức tổng hợp từ nhiều nguồn nên còn những mảng kiến thức chưa nắm vững cho dù đã rất cố gắng tham khảo các tài liệu và vận dụng kiến thức từ các môn học song bài tiểu luận của chúng em vẫn không thể tránh được những sai sót. Nhóm em rất mong được sự hướng dẫn và chỉ bảo thêm của các thầy cô để có thể củng cố và hiểu sâu hơn, nắm vững hơn về những kiến thức đã học được.

Khi thực hiện bài báo cáo này, chúng em đã tham khảo nhiều nguồn tài liệu trên mạng, tất cả những nguồn tài liệu, hình ảnh minh họa tham khảo đều sẽ được ghi chú rõ ở phần TÀI LIỆU THAM KHẢO. Cam kết mọi nội dung trong bài báo cáo này đều do chúng em thu thập và tổng hợp lại. Nếu có bất kì tranh chấp nào về đạo văn, chúng em xin đứng ra đối chấp và chịu hậu quả nếu có sai phạm.

Cuối cùng em xin chân thành cảm ơn thầy NGUYỄN HUY CUÔNG đã trực tiếp hướng dẫn, gợi ý cho chúng em hoàn thành tốt bài báo cáo này.

Một lần nữa chúng em xin chân thành cảm ơn!

Nhóm AeSoftware

MỤC LỤC


LỜI CAM ĐOAN.....	i
LỜI CẢM ƠN.....	ii
MỤC LỤC	iii
DANH MỤC HÌNH ẢNH	iv
DANH MỤC BẢNG SQL	v
CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI.....	1
1.1 Giới thiệu đề tài.....	1
1.2 Nhiệm vụ đồ án	1
1.2.1 Tính cấp thiết và lý do hình thành đề tài	1
1.2.2 Mục tiêu nghiên cứu	2
1.2.3 Đối tượng và phạm vi giới hạn	2
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	3
2.1 Khái niệm và phương pháp	3
2.1.1 Trình bày về ứng dụng Chat App sử dụng cơ sở dữ liệu SQL.....	3
2.1.2 Trình bày về hệ thống Chat sử dụng trong lập trình Window Form.....	6
CHƯƠNG 3: Kết quả thực nghiệm.....	12
CHƯƠNG 4: KẾT LUẬN	28
TÀI LIỆU THAM KHẢO	29

DANH MỤC HÌNH ẢNH


Hình 2.1 Sơ đồ Workflow	10
Hình 3.1 Giao diện Server	12
Hình 3.2 Giao diện đăng ký	13
Hình 3.3 Chọn ảnh đại diện	14
Hình 3.4 Chọn ảnh đại diện thành công	14
Hình 3.5 Đăng ký thành công và cung cấp QR code.....	15
Hình 3.6 Quét QR code để lấy mã TOTP	16
Hình 3.7 Cung cấp TOTP sau khi quét QR	17
Hình 3.8 Gửi thông tin đăng nhập tới Email	18
Hình 3.9 Thông tin gửi đến Email	18
Hình 3.10 Giao diện đăng nhập	19
Hình 3.11 Đăng nhập thành công	20
Hình 3.12 Giao diện chính.....	20
Hình 3.13 Tạo nhóm	21
Hình 3.14 Tạo nhóm thành công	22
Hình 3.15 Các group của người dùng đã tạo hoặc tham gia.....	22
Hình 3.16 Tham gia vào group	23
Hình 3.17 Gửi kết bạn.....	23
Hình 3.18 Chờ chấp nhận lời mời kết bạn.....	24
Hình 3.19 Danh sách bạn bè và nhóm	25
Hình 3.20 Tương tác giữa 2 người dùng.....	26
Hình 3.21 Gửi hình ảnh cũng như các tệp	26
Hình 3.22 Giao diện Export.....	27

DANH MỤC BẢNG SQL


User

	Column Name	Data Type	Allow Nulls
	UserID	int	<input type="checkbox"/>
	Username	nvarchar(100)	<input type="checkbox"/>
	Email	nvarchar(255)	<input type="checkbox"/>
	Password	nvarchar(MAX)	<input type="checkbox"/>
	SecretKey	nvarchar(MAX)	<input type="checkbox"/>
	ProfilePicture	nvarchar(MAX)	<input checked="" type="checkbox"/>
	Status	nvarchar(50)	<input checked="" type="checkbox"/>
	CreatedAt	datetime	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Group

	Column Name	Data Type	Allow Nulls
	GroupID	int	<input type="checkbox"/>
	GroupName	nvarchar(200)	<input type="checkbox"/>
	GroupDescription	nvarchar(MAX)	<input checked="" type="checkbox"/>
	CreatedBy	int	<input type="checkbox"/>
	CreatedAt	datetime	<input checked="" type="checkbox"/>
	GroupImage	nvarchar(MAX)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Group members

	Column Name	Data Type	Allow Nulls
	MemberID	int	<input type="checkbox"/>
	GroupID	int	<input type="checkbox"/>
	UserID	int	<input type="checkbox"/>
	Role	nvarchar(50)	<input checked="" type="checkbox"/>
	JoinedAt	datetime	<input checked="" type="checkbox"/>
	LastSeen	datetime	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Group messages

	Column Name	Data Type	Allow Nulls
🔑	MessageID	int	<input type="checkbox"/>
	GroupID	int	<input type="checkbox"/>
	SenderID	int	<input type="checkbox"/>
	[Content]	nvarchar(MAX)	<input checked="" type="checkbox"/>
	MessageType	nvarchar(50)	<input checked="" type="checkbox"/>
	Timestamp	datetime	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Group notifications

	Column Name	Data Type	Allow Nulls
🔑	NotificationID	int	<input type="checkbox"/>
	GroupID	int	<input type="checkbox"/>
	UserID	int	<input type="checkbox"/>
	[Content]	nvarchar(MAX)	<input type="checkbox"/>
	IsRead	bit	<input checked="" type="checkbox"/>
	Timestamp	datetime	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Friendships

	Column Name	Data Type	Allow Nulls
🔑	FriendshipID	int	<input type="checkbox"/>
	RequesterID	int	<input type="checkbox"/>
	AddressID	int	<input type="checkbox"/>
	Status	nvarchar(50)	<input checked="" type="checkbox"/>
	CreatedAt	datetime	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

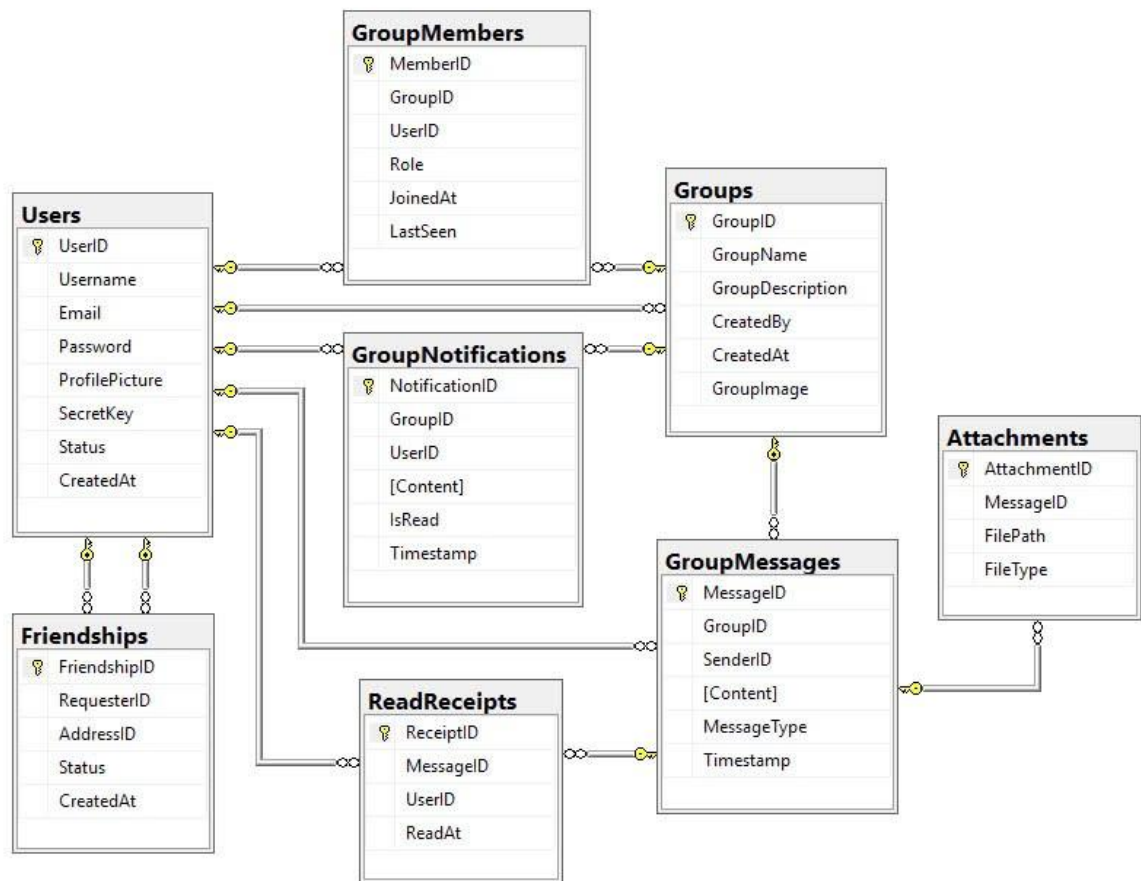
Attachments

	Column Name	Data Type	Allow Nulls
🔑	AttachmentID	int	<input type="checkbox"/>
	MessageID	int	<input type="checkbox"/>
	FilePath	nvarchar(500)	<input type="checkbox"/>
	FileType	nvarchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Read receipts

	Column Name	Data Type	Allow Nulls
🔑	ReceiptID	int	<input type="checkbox"/>
	MessageID	int	<input type="checkbox"/>
	UserID	int	<input type="checkbox"/>
	ReadAt	datetime	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

Sơ đồ mô tả các thực thể trong cơ sở dữ liệu



CHƯƠNG 1: TỔNG QUAN ĐỀ TÀI

1.1 Giới thiệu đề tài

Trong thời đại số, khi mạng xã hội ngày càng trở thành một phần không thể thiếu trong cuộc sống, nhu cầu kết nối và tương tác của con người cũng tăng lên đáng kể. Các ứng dụng như Facebook, Zalo đã trở thành những "sân chơi" quen thuộc, đáp ứng nhu cầu giao tiếp, chia sẻ thông tin của hàng triệu người dùng. Tuy nhiên, với sự phát triển không ngừng của công nghệ, người dùng ngày càng đòi hỏi những ứng dụng chat thông minh hơn, đáp ứng tốt hơn các nhu cầu cá nhân hóa. Trước đây, việc liên lạc chủ yếu dựa vào tin nhắn SMS và cuộc gọi điện thoại, thường kèm theo nhiều hạn chế về chi phí và tiện ích. Nhận thức được điều đó, các ứng dụng chat đã ra đời, mang đến một cách thức giao tiếp hoàn toàn mới, nhanh chóng và tiện lợi hơn. Với tính năng nhắn tin, gọi điện miễn phí, chia sẻ hình ảnh, video, cùng vô vàn tiện ích khác, các ứng dụng chat đã trở thành công cụ không thể thiếu trong cuộc sống hiện đại. Việc xây dựng một ứng dụng chat riêng biệt trong bối cảnh hiện nay không chỉ là một xu hướng mà còn là một nhu cầu tất yếu. Một ứng dụng chat được phát triển bài bản, với những tính năng độc đáo và trải nghiệm người dùng tuyệt vời sẽ là một lợi thế cạnh tranh lớn, thu hút được đông đảo người dùng.

1.2 Nhiệm vụ đề án

1.2.1 Tính cấp thiết và lý do hình thành đề tài

Trong kỷ nguyên số, ứng dụng chat đã trở thành một phần không thể thiếu trong cuộc sống hàng ngày của chúng ta. Sự cạnh tranh trên thị trường ứng dụng chat ngày càng khốc liệt, đòi hỏi các sản phẩm phải không ngừng đổi mới và nâng cao trải nghiệm người dùng. Việc phát triển một ứng dụng chat thành công không chỉ đơn thuần là tạo ra một nền tảng giao tiếp mà còn là một chiến lược kinh doanh thông minh. Để thành công, các ứng dụng chat cần phải đáp ứng được những nhu cầu đa dạng và ngày càng cao của người dùng, từ việc giao tiếp cá nhân đến làm việc nhóm, từ giải trí đến thương mại điện tử. Bên cạnh đó, việc tạo ra một cộng đồng người dùng gắn bó, trung thành cũng là yếu tố quan trọng quyết định sự thành bại của một ứng dụng chat. Việc phát triển ứng

dụng chat hiện nay không chỉ là một xu hướng mà còn là một cơ hội kinh doanh lớn. Để thành công, các ứng dụng chat cần phải liên tục đổi mới, đáp ứng nhu cầu của người dùng và tạo ra sự khác biệt so với các đối thủ cạnh tranh.

1.2.2 Mục tiêu nghiên cứu

Sử dụng C# Winforms để tạo một ứng dụng có các chức năng như:

- Đăng ký, đăng nhập.
- Bảo mật tài khoản 2 lớp với TOTP.
- Gửi thông tin tài khoản về mail khi đăng ký.
- Cho phép người dùng tương tác với nhau.
- Tạo group chat với 2 người trở lên.
- Các tính năng cơ bản của việc nhắn tin, gửi ảnh.

1.2.3 Đối tượng và phạm vi giới hạn

Việc thiết kế ứng dụng này nhằm mục đích phục vụ cho việc liên lạc, tương tác giữa các người dùng với nhau, bảo mật thông tin người dùng và mang lại những trải nghiệm cơ bản của một ứng dụng Chat.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1 Khái niệm và phương pháp

2.1.1 Trình bày về ứng dụng Chat App sử dụng cơ sở dữ liệu SQL Server

Giới thiệu

- Mục tiêu của hệ thống: Tạo ra một hệ thống cho phép các người dùng tương tác, nhắn tin, liên lạc qua lại, bảo mật tài khoản, thông tin người dùng bằng 2FA.
- Phạm vi hệ thống: Nhắn tin, liên lạc, tương tác với nhau thông qua Server.

Các khái niệm cơ bản:

- Quản lý tài khoản: hệ thống sẽ lưu trữ tài khoản đăng ký của người dùng và đồng thời tích hợp tính năng gửi tài khoản về email cho người dùng khi đăng ký.
- Bảo mật tài khoản bằng 2FA: Người dùng chỉ đăng nhập được khi nhập thêm mã TOTP mà hệ thống gửi về Authenticator của người dùng.
- Quản lý danh sách bạn bè: Lưu trữ thông tin các bạn bè mà người dùng đã gửi yêu cầu kết bạn và đã chấp nhận.
- Quản lý nhóm: Lưu trữ thông tin nhóm bao gồm Admin (người lập ra nhóm) và các thành viên đã tham gia vào nhóm.
- Quản lý tin nhắn: hệ thống lưu trữ các đoạn hội thoại giữa các người dùng với nhau hoặc lưu trong đoạn hội thoại trong các nhóm chat nhiều người dùng.

Công nghệ sử dụng:

- Cơ sở dữ liệu: SQL Sever hoặc MySQL cho việc lưu trữ và xử lý dữ liệu.
- Ngôn ngữ truy vấn: SQL (Structured Query Language) để truy vấn và thao tác dữ liệu.
- Công cụ phát triển: Sử dụng công cụ như SQL Management Studio.

- Giao diện người dùng: Tạo giao diện thân thiện với người sử dụng thông qua ứng dụng Web hoặc Desktop.

Phương pháp giải quyết vấn đề:

- Phân tích yêu cầu: Xác định các yêu cầu chức năng (quản lý người dùng, nhóm, tin nhắn, bạn bè, ...) và phi chức năng (bảo mật, hiệu suất).
- Thiết kế hệ thống:
 - Kiến trúc hệ thống: Chia thành các module với các chức năng rõ ràng.
 - Mô hình hóa dữ liệu: Sử dụng ERD (Entity – Relationship Diagram) để mô tả các thực thể và mối quan hệ.

Các ràng buộc:

- **Ràng buộc:** Bảo vệ thông tin người dùng bằng cách mã hóa trường Password và có thể sử dụng SecretKey để tăng cường bảo mật.
- **Ràng buộc logic:** Một người dùng có thể là thành viên của nhiều nhóm, nhưng một người chỉ có thể xuất hiện một lần trong một nhóm.
- RequesterID và AddressID phải đảm bảo duy nhất để tránh các yêu cầu kết bạn trùng lặp trong bảng **Friendships**.

Các giải pháp và tối ưu hóa:

- Duy trì tính toàn vẹn dữ liệu:
 - Sử dụng **khóa ngoại** để đảm bảo dữ liệu liên kết giữa các bảng luôn nhất quán.
 - Đảm bảo sử dụng các **trigger** hoặc **stored procedure** để tự động xử lý khi xóa/sửa dữ liệu trong bảng liên quan.
- Quản lý quyền và bảo mật:
 - Mã hóa thông tin nhạy cảm như Password.
 - Kiểm tra quyền của người dùng khi thực hiện các thao tác như gửi tin nhắn, thêm thành viên vào nhóm.
- Xử lý dữ liệu lớn:

- Với bảng **GroupMessages**, có thể phân vùng dữ liệu (partitioning) dựa trên GroupID hoặc Timestamp để giảm tải khi truy vấn.
- Sử dụng cơ chế lưu trữ tệp tin ngoài (như Amazon S3) cho các tệp lớn trong bảng **Attachments**.
- Tăng trải nghiệm người dùng:
 - Thêm các cột như LastSeen trong bảng **GroupMembers** để hiển thị trạng thái hoạt động của người dùng trong nhóm.
 - Sử dụng trường IsRead trong bảng **GroupNotifications** để đánh dấu thông báo đã đọc/chưa đọc.
- Ràng buộc logic nâng cao:
 - Trong bảng **Friendships**, đảm bảo rằng quan hệ RequesterID và AddressID không thể đảo ngược (nghĩa là nếu A gửi yêu cầu đến B, không thể tạo bản ghi ngược lại).
 - Trong bảng **GroupMembers**, đảm bảo mỗi nhóm luôn có ít nhất một Role = 'admin'

Mô hình: Mô hình được thiết kế để hỗ trợ các tính năng chính của một ứng dụng chat bao gồm quản lý người dùng, nhóm, tin nhắn, file đính kèm, bạn bè, thông báo nhóm và trạng thái đọc tin nhắn.

Lý giải xây dựng mô hình: Dựa trên đặc điểm của một ứng dụng chat đa năng (nhắn tin, nhóm, thông báo, file đính kèm...), mô hình này được xây dựng nhằm đảm bảo tính **tối ưu, bảo mật, và linh hoạt** trong việc quản lý dữ liệu.

Dưới đây là các lý giải cụ thể:

- Bảng Users: Là nền tảng chính quản lý thông tin người dùng. Lưu các thông tin cơ bản như tên đăng nhập, email, mật khẩu để xác thực và nhận dạng. Bao gồm các cột Status (trạng thái online/offline) và secretKey để hỗ trợ bảo mật và trạng thái người dùng trong hệ thống.
- Bảng group: Là tính năng của ứng dụng chat, nơi nhiều người có thể tương tác. Quản lý thông tin nhóm như tên nhóm, mô tả, người tạo và thời gian tạo.

- Bảng GroupMembers: Một người có thể tham gia nhiều nhóm, và một nhóm có nhiều thành viên. Lưu các thông tin về nhóm, người dùng, quyền (Role), thời gian tham gia (JoinedAt), và thời gian hoạt động cuối (LastSeen).
- Bảng GroupMessages: Tin nhắn là phần cốt lõi của ứng dụng chat. Lưu trữ các nội dung, loại tin nhắn, thời gian gửi tin nhắn và thông tin người gửi, đồng thời hỗ trợ các loại tin nhắn khác nhau như văn bản, hình ảnh,...
- Bảng Attachments: Người dùng thường chia sẻ file (ảnh, video, tài liệu) trong tin nhắn. Lưu trữ các file này thay vì lưu trực tiếp trong bảng GroupMessages, giúp giảm tải lưu trữ.
- Bảng Friendships: Quản lý các mối quan hệ bạn bè giữa người dùng với các trạng thái như “đang chờ”, “đã chấp nhận”, “bị từ chối”. Quan hệ bạn bè là một phần quan trọng trong các tính năng nhắn tin cá nhân.
- Bảng GroupNotifications: Người dùng cần nhận thông báo từ nhóm như lời mời tham gia, thông báo tin nhắn, hoặc sự kiện trong nhóm. Lưu trữ trạng thái “đã đọc” hoặc “chưa đọc” của thông báo.
- Bảng RedReceipts: Trạng thái đọc tin nhắn là một tính năng phổ biến trong các ứng dụng chat. Lưu thông tin ai đã đọc tin nhắn nào và thời gian đọc.

Kết luận: Mô hình này không chỉ đáp ứng các yêu cầu cơ bản của một ứng dụng chat mà còn có khả năng mở rộng linh hoạt, tối ưu hóa lưu trữ và truy vấn. Việc chia tách các thực thể và mối quan hệ rõ ràng giúp cải thiện hiệu năng cũng như quản lý dữ liệu một cách dễ dàng.

2.1.2 Trình bày về hệ thống Chat sử dụng trong lập trình Window Form

Hệ thống Chat được xây dựng trên nền tảng lập trình Windows Forms cung cấp một giao diện đồ họa thân thiện để người dùng gửi/nhận tin nhắn, chia sẻ file, quản lý nhóm, và thực hiện các hoạt động khác trong thời gian thực. Dưới đây là trình bày chi tiết về hệ thống này:

- Kiến trúc hệ thống: Hệ thống được chia thành hai thành phần chính

- Server
- Client: Người dùng có thể sử dụng trong một sever

Server

- Mô tả:
 - Server chịu trách nhiệm xử lý dữ liệu nhận từ Client và gửi phản hồi tương ứng.
 - Hoạt động như trung gian để quản lý các yêu cầu từ nhiều Client.
- Tính năng chính:
 - Quản lý kết nối Client.
 - Lưu trữ tin nhắn và lịch sử chat vào cơ sở dữ liệu.
 - Gửi thông báo thời gian thực đến các Client.
 - Xử lý các thao tác nhóm như thêm, xóa thành viên.
 - Đảm bảo bảo mật dữ liệu qua mã hóa.
- Công nghệ và thư viện:
 - **TcpListener**: Lắng nghe các kết nối đến từ client.
 - **Threading**: Xử lý nhiều client đồng thời thông qua các luồng riêng biệt.
 - **JSON Serialization**: Chuyển đổi dữ liệu giữa các client và server.
 - **SQL Server**: Lưu trữ dữ liệu người dùng, tin nhắn, nhóm, và thông báo.

Client

- Mô tả:
 - Đây là ứng dụng giao diện mà người dùng sử dụng để thực hiện các hoạt động chat.
 - Giao tiếp với server qua giao thức **TCP/IP**.
 - Gửi yêu cầu và nhận phản hồi từ server.
- Tính năng chính:
 - Đăng ký/Đăng nhập.
 - Nhắn tin cá nhân hoặc nhóm.

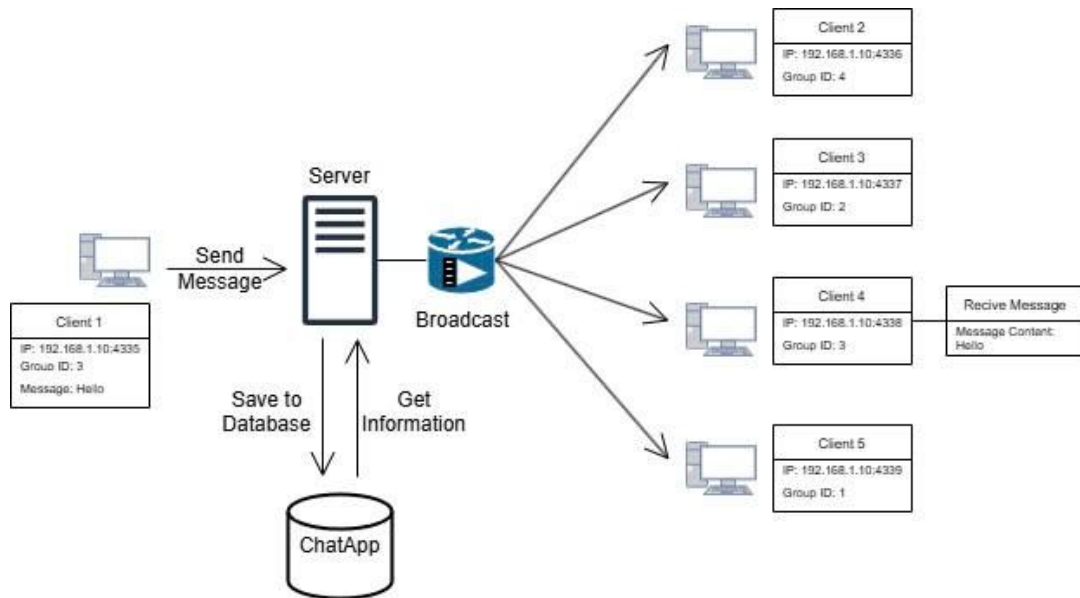
- Gửi tệp tin (file).
- Hiển thị thông báo nhóm.
- Quản lý danh sách bạn bè.
- Thêm biểu tượng cảm xúc trong tin nhắn.
- Các thành phần giao diện trong Windows Form:
 - **TextBox**: Nhập nội dung tin nhắn, tìm kiếm...
 - **RichTextBox**: Hiển thị nội dung hộp thoại (tin nhắn, hình ảnh, icon, ...).
 - **Button**: Gửi tin nhắn, tệp tin, hoặc tạo nhóm...
 - **ListBox**: Hiển thị danh sách tin nhắn hoặc thành viên nhóm...
 - **DataGridView**: Hiển thị dữ liệu phức tạp như danh sách nhóm hoặc bạn bè...
 - **PictureBox**: Hiển thị hình ảnh đại diện, các icon, hoặc file đính kèm...

Quy trình hoạt động

- Đăng ký/Đăng nhập
 - Người dùng nhập thông tin đăng nhập (username, password, OTP Token)..
 - Server kiểm tra thông tin trong cơ sở dữ liệu:
 - Nếu hợp lệ: Trả về thông báo đăng nhập thành công.
 - Nếu không hợp lệ: Trả về lỗi.
 - Client cập nhật giao diện để hiển thị trạng thái. Đồng thời, gửi dữ liệu này dưới dạng JSON qua Server để thực hiện cơ chế Broadcast - truyền thông tin cho tất cả các Clients còn lại (để biết được bạn bè của họ đang online).
- Nhắn tin
 - Người dùng nhập nội dung tin nhắn trong **TextBox** và nhấn nút gửi.
 - Tin nhắn sẽ được gửi từ client đến server, kèm theo thông tin người gửi, người nhận (hoặc nhóm).
 - Server xử lý:
 - Lưu tin nhắn vào cơ sở dữ liệu.

- Gửi lại tin nhắn đến client đích (hoặc tất cả thành viên trong nhóm).
 - Tin nhắn hiển thị trên giao diện của cả người gửi và người nhận.
- Quản lý nhóm
 - Người dùng tạo nhóm với thông tin như tên nhóm, mô tả, và danh sách thành viên.
 - Dữ liệu nhóm được gửi từ client đến server.
 - Server:
 - Lưu thông tin nhóm vào cơ sở dữ liệu.
 - Gửi thông báo đến các thành viên được thêm vào nhóm.
 - Giao diện người dùng cập nhật danh sách nhóm.
- Gửi tệp tin
 - Người dùng chọn tệp từ máy tính thông qua **OpenFileDialog**.
 - Client nén và gửi file đến server cùng với thông tin tin nhắn.
 - Server lưu trữ file trên hệ thống hoặc cơ sở dữ liệu.
 - Tệp được gửi lại client đích và hiển thị trong giao diện.
- Hiển thị thông báo
 - Khi có sự kiện mới (ví dụ: lời mời tham gia nhóm, tin nhắn chưa đọc), server gửi thông báo đến client tương ứng.
 - Thông báo hiển thị trên giao diện người dùng (ví dụ: popup hoặc biểu tượng đỏ).

Sơ đồ thể hiện



Hình 2.1 Sơ đồ Workflow

Công nghệ và kỹ thuật sử dụng

- Ngôn ngữ lập trình C#
- Nền tảng phát triển: Windows Forms.
- Cơ sở dữ liệu: SQL Server cho việc lưu trữ dữ liệu.
- TCP/IP: Đảm bảo kết nối thời gian thực giữa Client và Server.
- JSON Serialization: Đóng gói dữ liệu (tin nhắn, thông báo) để truyền qua mạng.

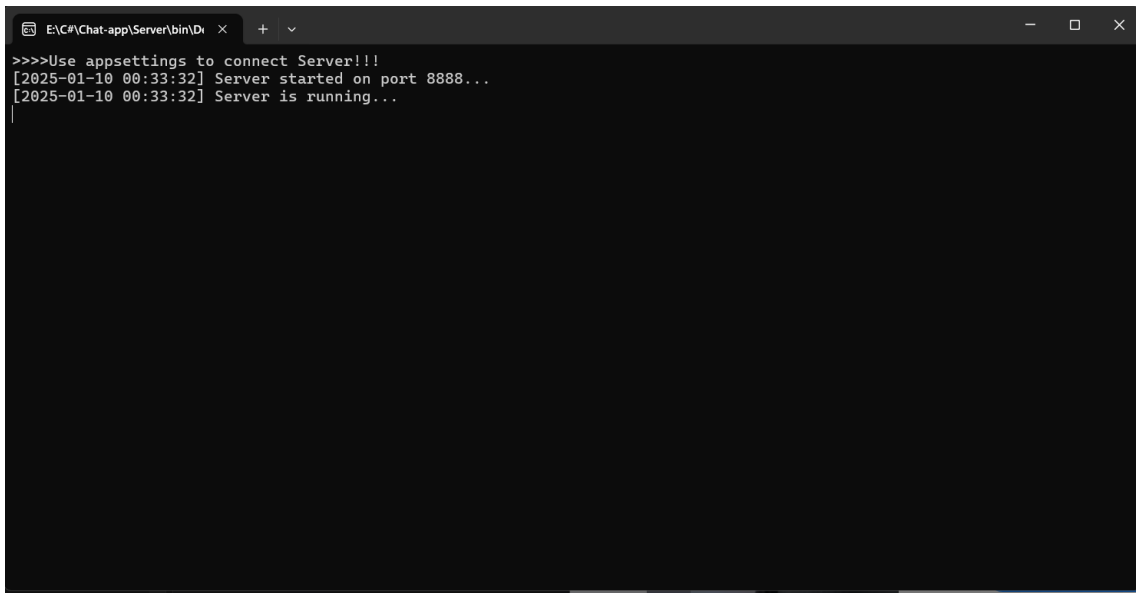
Ưu điểm và hạn chế của hệ thống

- Ưu điểm
 - **Giao diện thân thiện:** Dễ sử dụng với Windows Forms.
 - **Thời gian thực:** Nhờ sử dụng TCP/IP và đa luồng.
 - **Mở rộng linh hoạt:** Hỗ trợ quản lý nhóm, thông báo, và file đính kèm.
 - **Bảo mật:** Mã hóa mật khẩu và các giao thức bảo mật.
- Hạn chế
 - **Phụ thuộc môi trường Windows:** Không hoạt động trên các nền tảng khác.
 - **Khó mở rộng giao diện:** Windows Forms không lý tưởng cho thiết kế hiện đại.

- **Hiệu năng giới hạn:** Với số lượng người dùng lớn, cần tối ưu hóa server.

Kết luận: Hệ thống chat sử dụng Windows Forms là giải pháp phù hợp cho các ứng dụng nhỏ đến trung bình, cung cấp đầy đủ tính năng nhắn tin thời gian thực và quản lý nhóm. Với việc tối ưu hóa server và cơ sở dữ liệu, hệ thống có thể được mở rộng để hỗ trợ lượng người dùng lớn hơn.

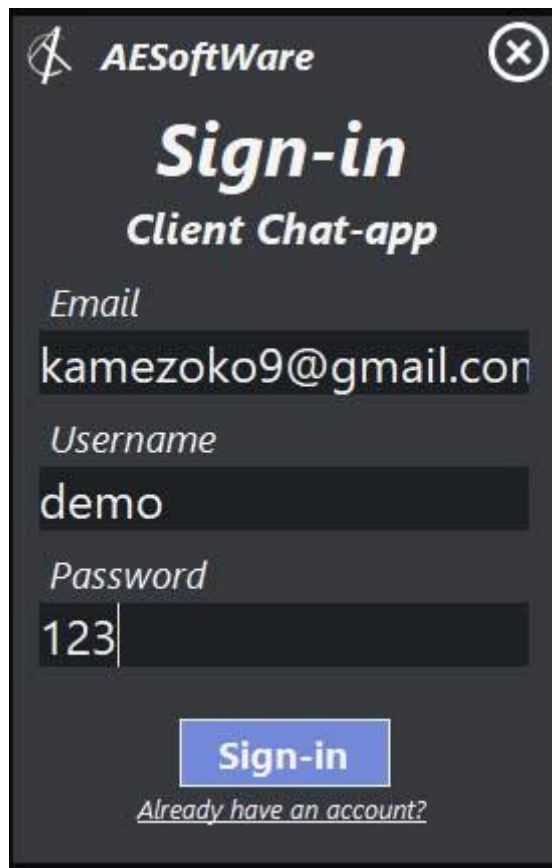
CHƯƠNG 3: Kết quả thực nghiệm

A screenshot of a terminal window with a dark background. The title bar shows the file path 'E:\C#\Chat-app\Server\bin\Di'. The terminal content displays the following text: '>>>>Use appsettings to connect Server!!!', '[2025-01-10 00:33:32] Server started on port 8888...', and '[2025-01-10 00:33:32] Server is running...'.

```
E:\C#\Chat-app\Server\bin\Di >>>>Use appsettings to connect Server!!!
[2025-01-10 00:33:32] Server started on port 8888...
[2025-01-10 00:33:32] Server is running...
```

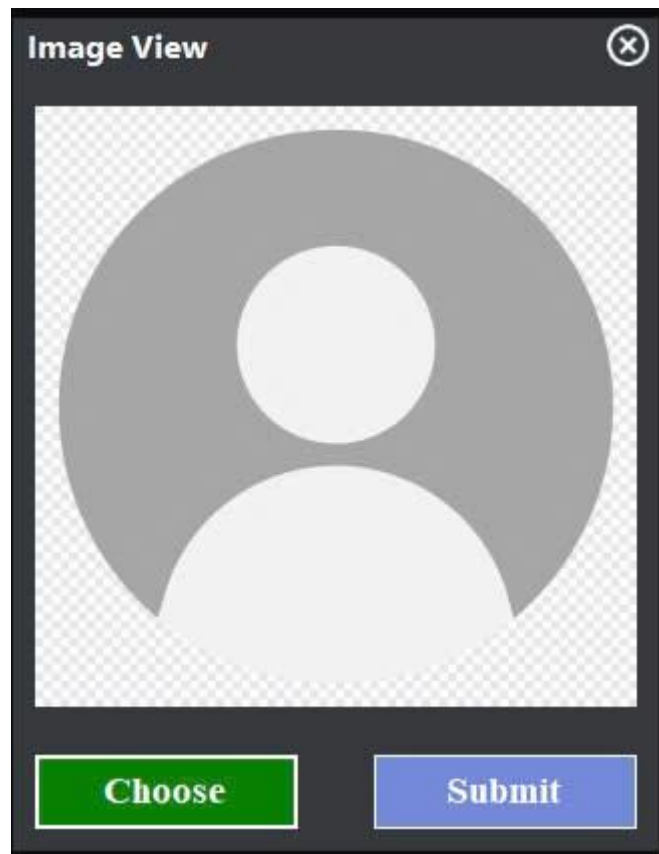
Hình 3.1 Giao diện Server

Hình 3.1: Giao diện server này là một thành phần backend trong hệ thống Chat Client-Server, hoạt động trên giao thức TCP/IP và lắng nghe các kết nối từ ứng dụng Chat Client thông qua cổng 8888. Server đảm nhiệm vai trò trung gian để xử lý các yêu cầu từ client như đăng nhập, gửi tin nhắn, quản lý nhóm, và phản hồi kết quả tương ứng. Nó điều phối tin nhắn giữa các client, đảm bảo rằng tin nhắn được chuyển tiếp chính xác đến người nhận hoặc tất cả thành viên trong nhóm. Đồng thời, server quản lý kết nối phiên (session) của các client, lưu trữ thông tin người dùng, nhóm, và lịch sử tin nhắn vào cơ sở dữ liệu, giúp đồng bộ hóa dữ liệu giữa các phiên làm việc. Ngoài ra, server cũng hiển thị trạng thái hoạt động, như thông báo "Server is running...", nhằm xác nhận hệ thống đang vận hành ổn định và hỗ trợ gỡ lỗi khi cần thiết. Đây là thành phần quan trọng để đảm bảo tính thời gian thực và hiệu quả của hệ thống chat.



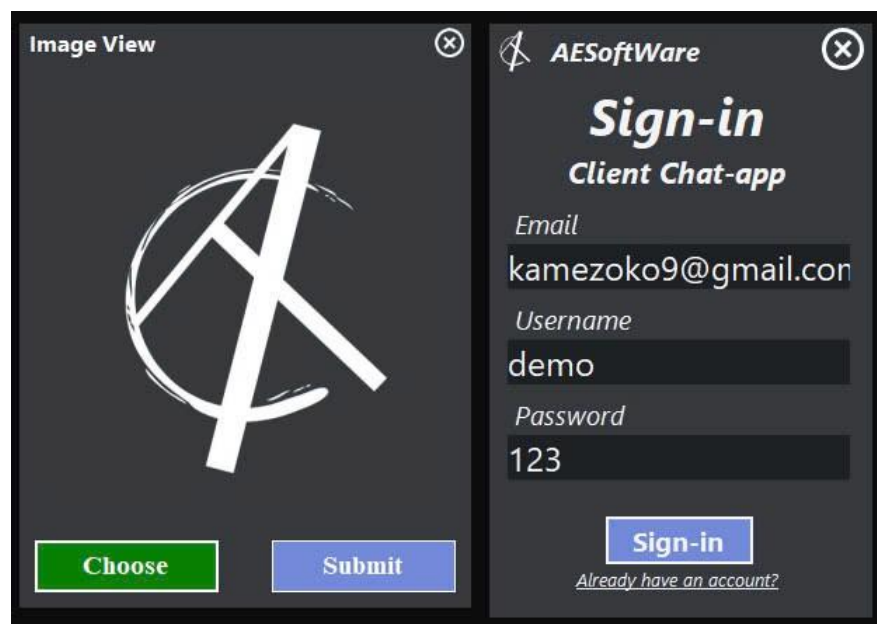
Hình 3.2 Giao diện đăng ký

Hình 3.2: Giao diện đăng ký trong hệ thống Chat Client là nơi người dùng tạo tài khoản mới để sử dụng ứng dụng. Giao diện này thường bao gồm các trường nhập liệu như tên đăng nhập, email, mật khẩu (nhập hai lần để xác nhận), và có thể bổ sung thêm ảnh đại diện hoặc thông tin bổ sung như ngày sinh, số điện thoại. Hệ thống kiểm tra và xác thực dữ liệu đầu vào, đảm bảo các trường không bị bỏ trống, email có định dạng hợp lệ, mật khẩu đáp ứng yêu cầu bảo mật, và hai trường mật khẩu phải khớp nhau. Khi người dùng nhấn nút "Đăng ký," ta thực hiện truy vấn lên Server để kiểm tra các điều kiện như tên đăng nhập đã tồn tại hay chưa, email có tồn tại hay không, sau đó phản hồi kết quả về ứng dụng. Nếu đăng ký thành công, giao diện sẽ hiển thị thông báo và chuyển hướng sang trang đăng nhập; nếu có lỗi, hệ thống sẽ báo lỗi để người dùng sửa. Đây là bước đầu tiên và quan trọng trong trải nghiệm người dùng, yêu cầu thiết kế đơn giản, trực quan và đảm bảo tính bảo mật cao.



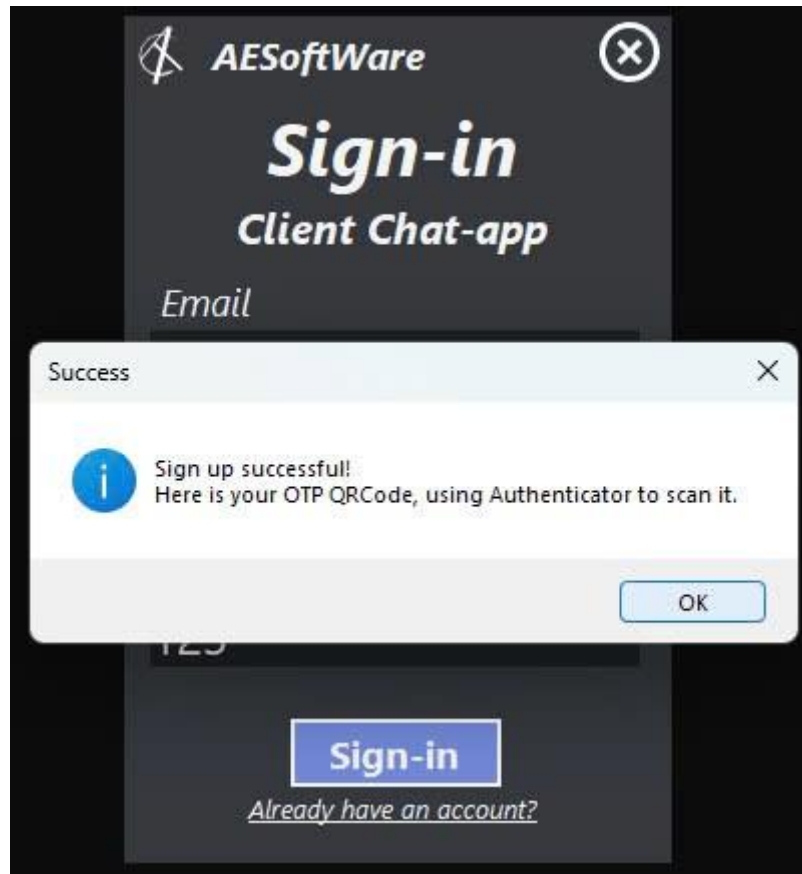
Hình 3.3 Chọn ảnh đại diện

Hình 3.3: Sau khi đăng ký tài khoản thành công, hệ thống sẽ yêu cầu chọn ảnh đại diện cho người dùng, người dùng có thể nhấn “Submit” để chọn ảnh mặc định của hệ thống.



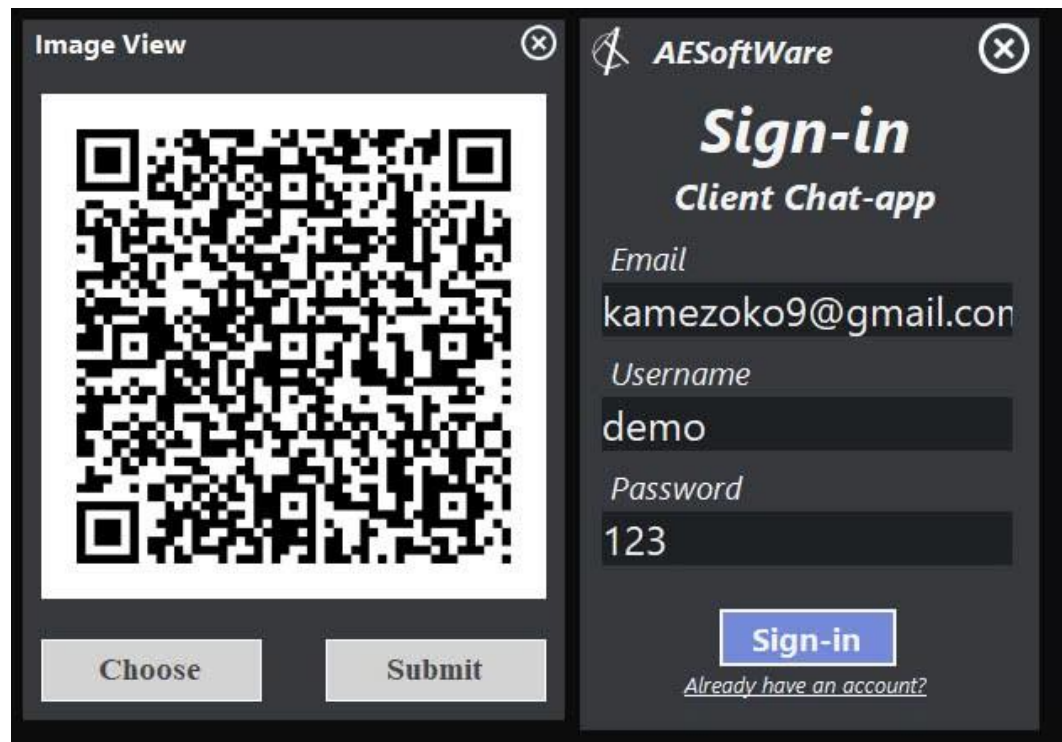
Hình 3.4 Chọn ảnh đại diện thành công

Hình 3.4: Nếu người dùng nhấn Choose hệ thống sẽ mở ra tệp hình ảnh trên máy để người dùng chọn ảnh đại diện mong muốn sau đó nhấn “Submit” server sẽ lưu vào database.



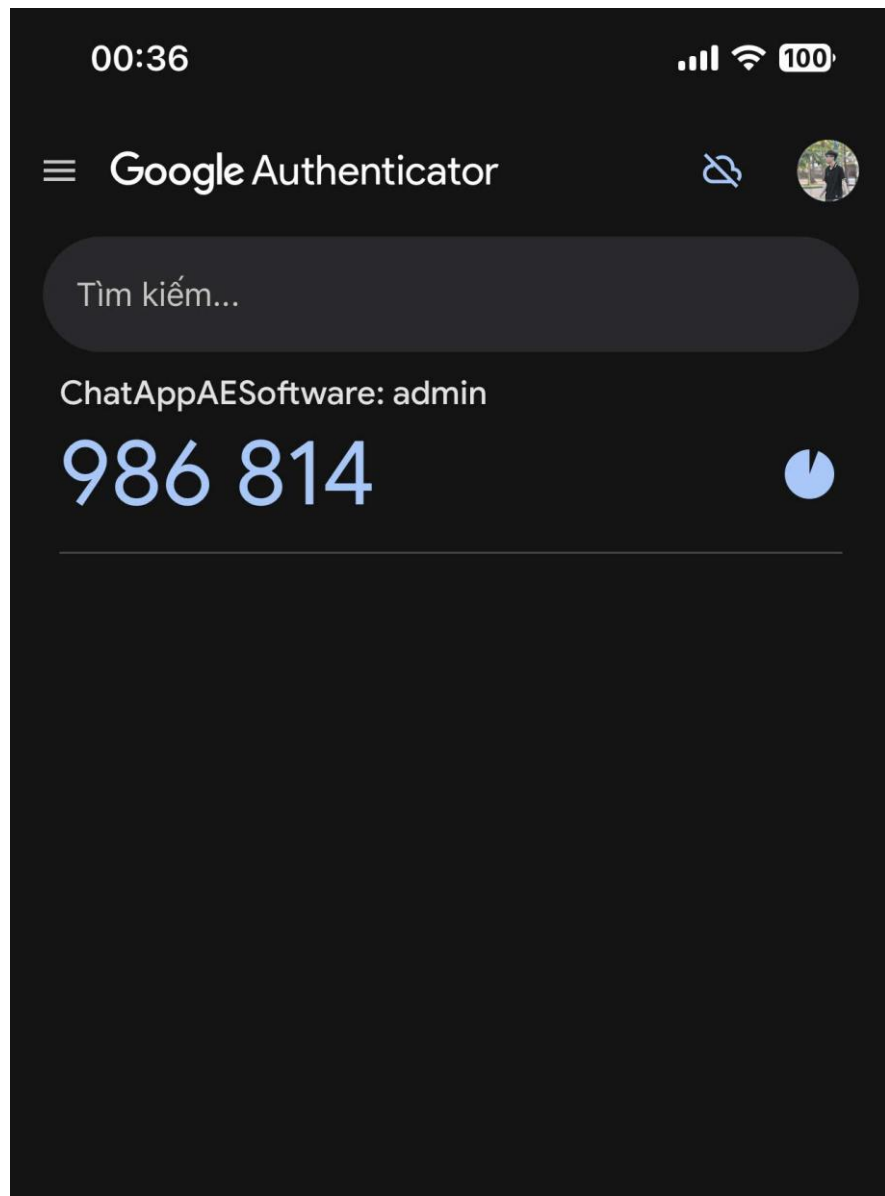
Hình 3.5 Đăng ký thành công và cung cấp QR code

Hình 3.5: Sau khi chọn ảnh đại diện xong, hệ thống sẽ thông báo người dùng đăng ký thành công và sẽ cấp mã QR Code để người dùng quét.



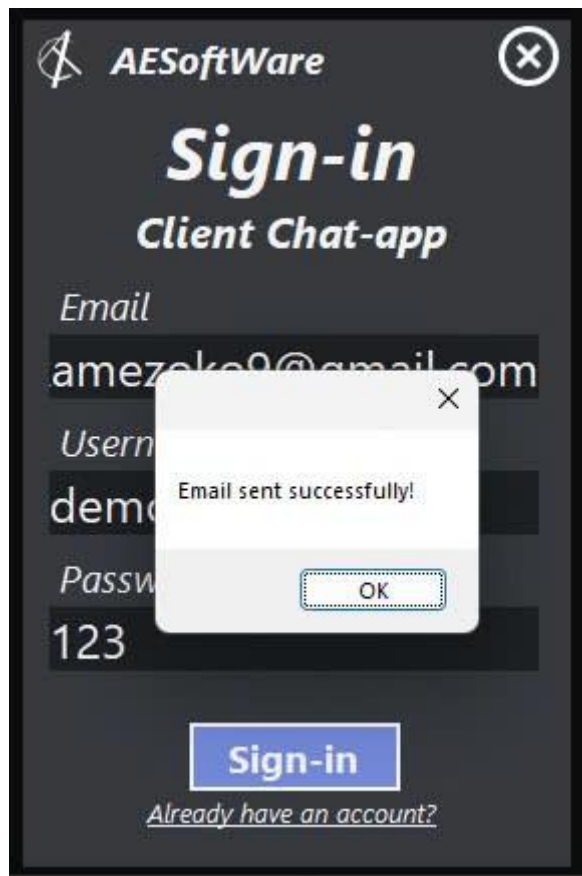
Hình 3.6 Quét QR code để lấy mã TOTP

Hình 3.6: Sau khi hiện QR code, người dùng sẽ dùng ứng dụng Authenticator để quét lấy mã TOTP mà hệ thống cấp.



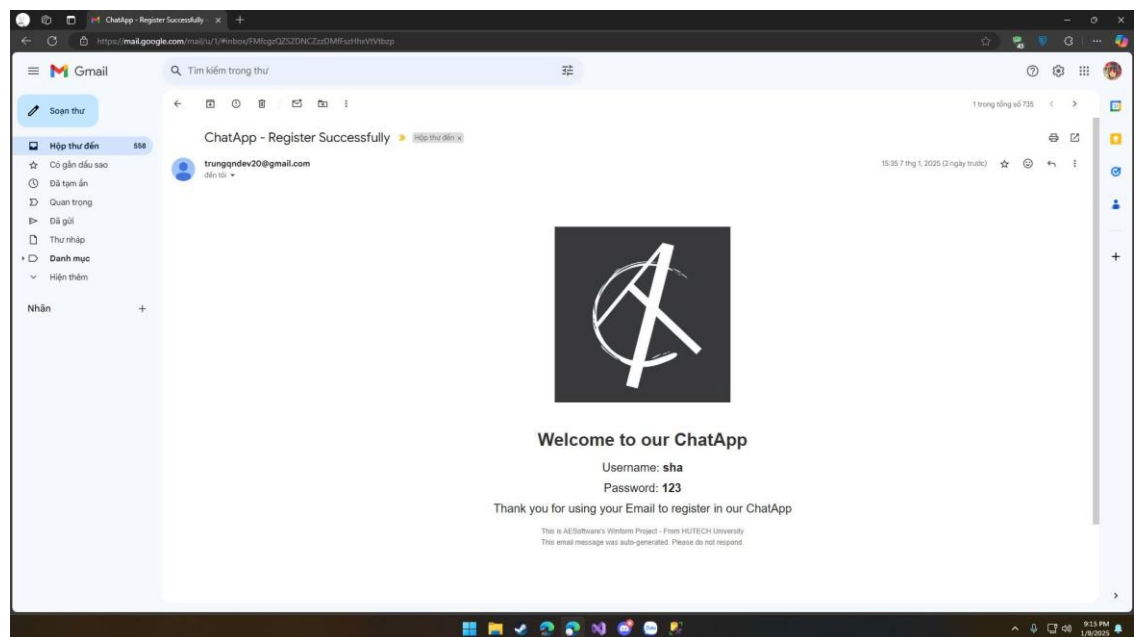
Hình 3.7 Cung cấp TOTP sau khi quét QR

Hình 3.7: Sau khi quét bằng ứng dụng Authenticator thì giao diện của ứng dụng sẽ hiện lên thông tin của mã OTP và tên người dùng. Mã OTP sẽ thay đổi sau 30 giây theo thời gian thực tế.

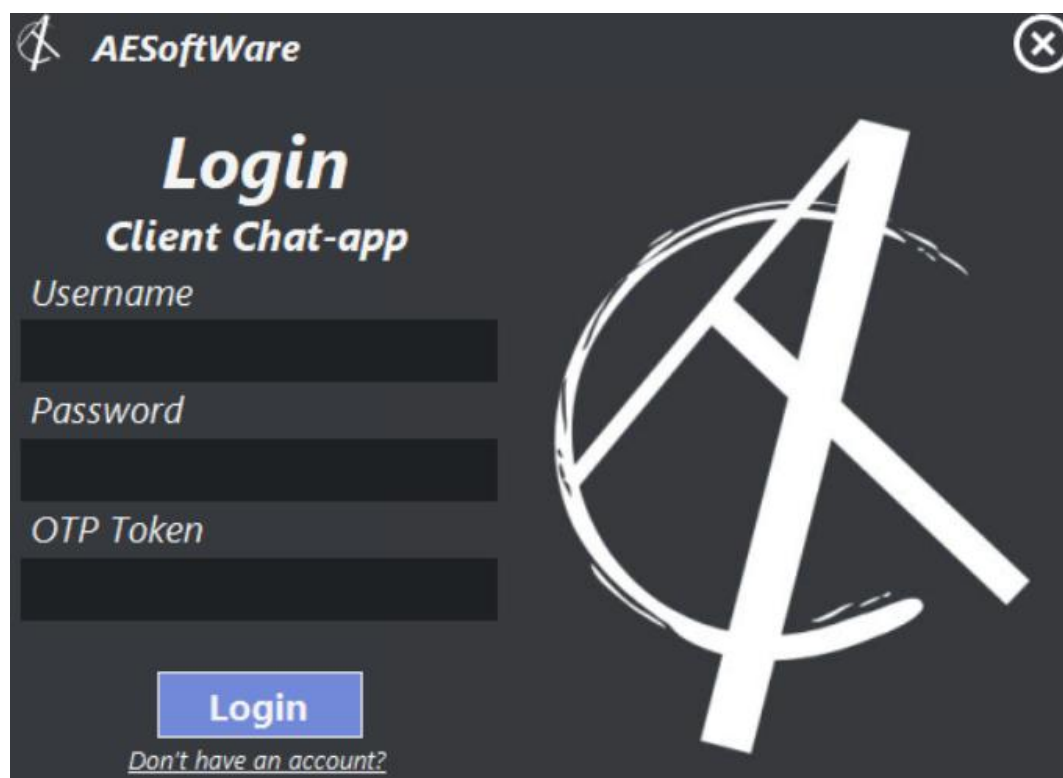


Hình 3.8 Gửi thông tin đăng nhập tới Email

Hình 3.8 Khi đã quét QR xong thì hệ thống sẽ thông báo gửi Email thành công. Ở đây hệ thống sẽ gửi đến Email thông tin bao gồm tên đăng nhập và mật khẩu như hình 3.9.

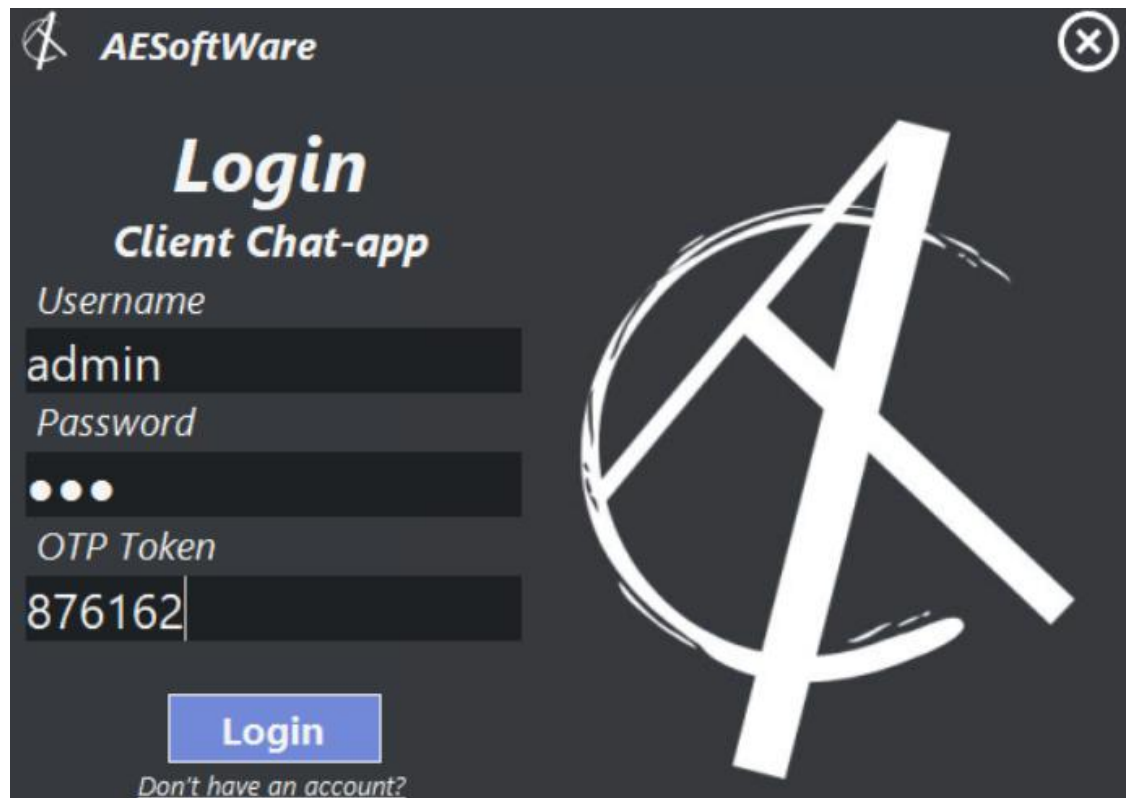


Hình 3.9 Thông tin gửi đến Email



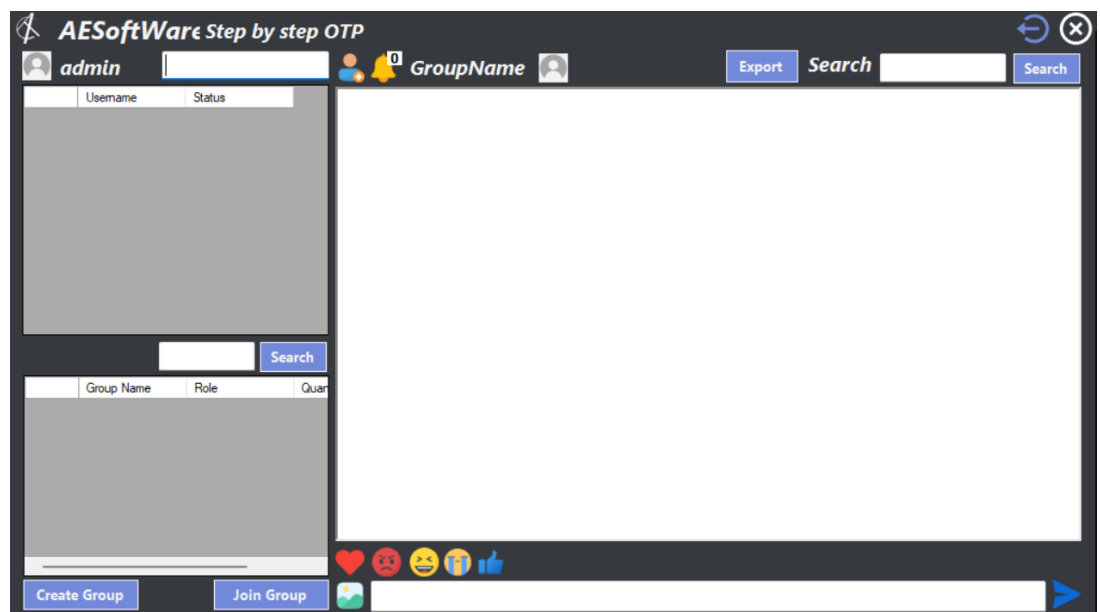
Hình 3.10 Giao diện đăng nhập

Hình 3.10: Sau khi đăng ký thành công thì hệ thống sẽ chuyển qua giao diện đăng nhập để người dùng tiến hành nhập tài khoản bao gồm tên tài khoản và mật khẩu. Bên cạnh đó người dùng cũng cần phải theo dõi mã OTP ở ứng dụng Authenticator như hình 3.7 để theo dõi mã OTP của mình vì mỗi mã OTP này nó chỉ tồn tại 30 giây.



Hình 3.11 Đăng nhập thành công

Hình 3.11: Sau khi nhập đúng tên tài khoản và mật khẩu, cùng với nhập đúng mã OTP trong thời gian 30 giây thì nhấn login hệ thống sẽ đưa người dùng đến giao diện chính.



Hình 3.12 Giao diện chính

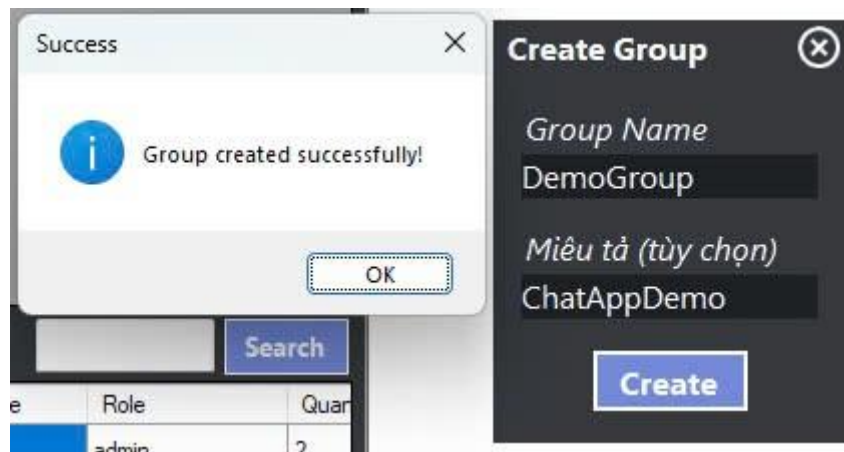
Hình 3.12: Giao diện chính của ứng dụng Chat App sẽ bao gồm các chức năng như sau:

- Thanh menu: có các chức năng như avatar và username, thanh tìm kiếm tên người dùng và nút kết bạn, nút thông báo gửi kết bạn, tên nhóm và thanh tìm kiếm trong hộp thoại, nút xuất ra bảng báo cáo thành viên trong nhóm.
- Datagridview hiển thị danh sách bạn bè và danh sách nhóm, ô tìm kiếm tên nhóm, nút tạo nhóm và tham gia nhóm.
- Phần lớn nhất chính là khung chat



Hình 3.13 Tạo nhóm

Hình 3.13 Khi nhấn vào nút Create Group, hệ thống sẽ mở giao diện tạo nhóm cho phép người dùng làm admin. Sau khi nhập tên nhóm và nhập thông tin miêu tả, người dùng nhấn Create thì hệ thống sẽ thông báo tạo nhóm thành công như hình 3.14.



Hình 3.14 Tạo nhóm thành công

	Group Name	Role	Quar
	Tech Talk	admin	2
▶	DemoGroup	admin	1

Below the table, there are two buttons: 'Create Group' and 'Join Group'.

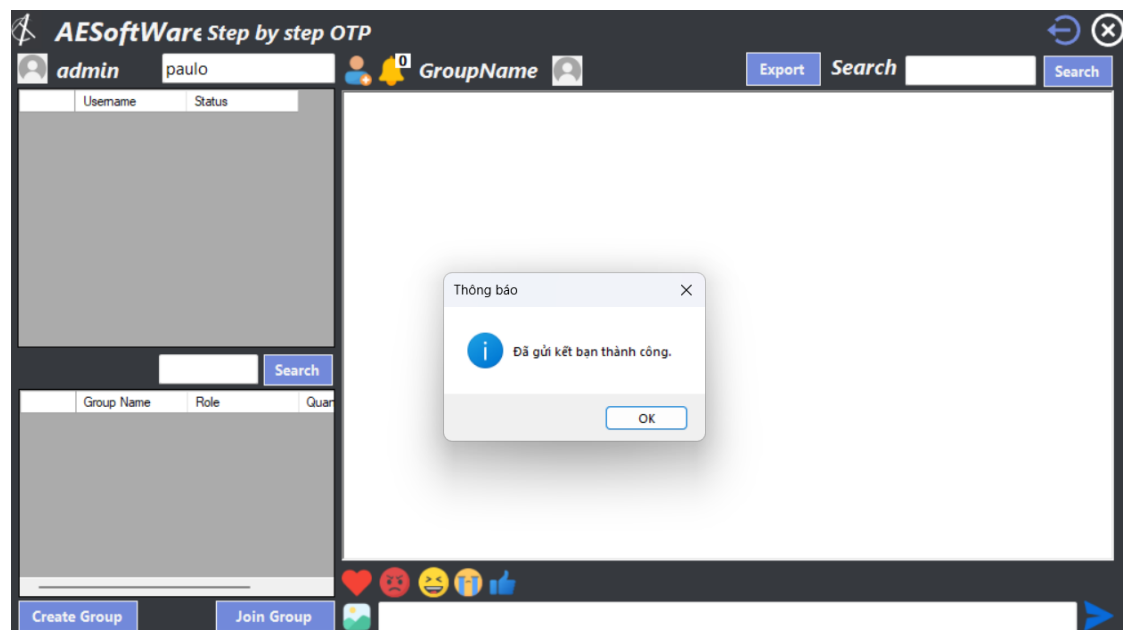
Hình 3.15 Các group của người dùng đã tạo hoặc tham gia

Hình 3.15 thể hiện danh sách các nhóm mà người dùng đó đã tạo, thông tin hiển thị bao gồm tên nhóm, vai trò admin của người dùng hoặc là vai trò thành viên và số lượng thành viên trong nhóm.



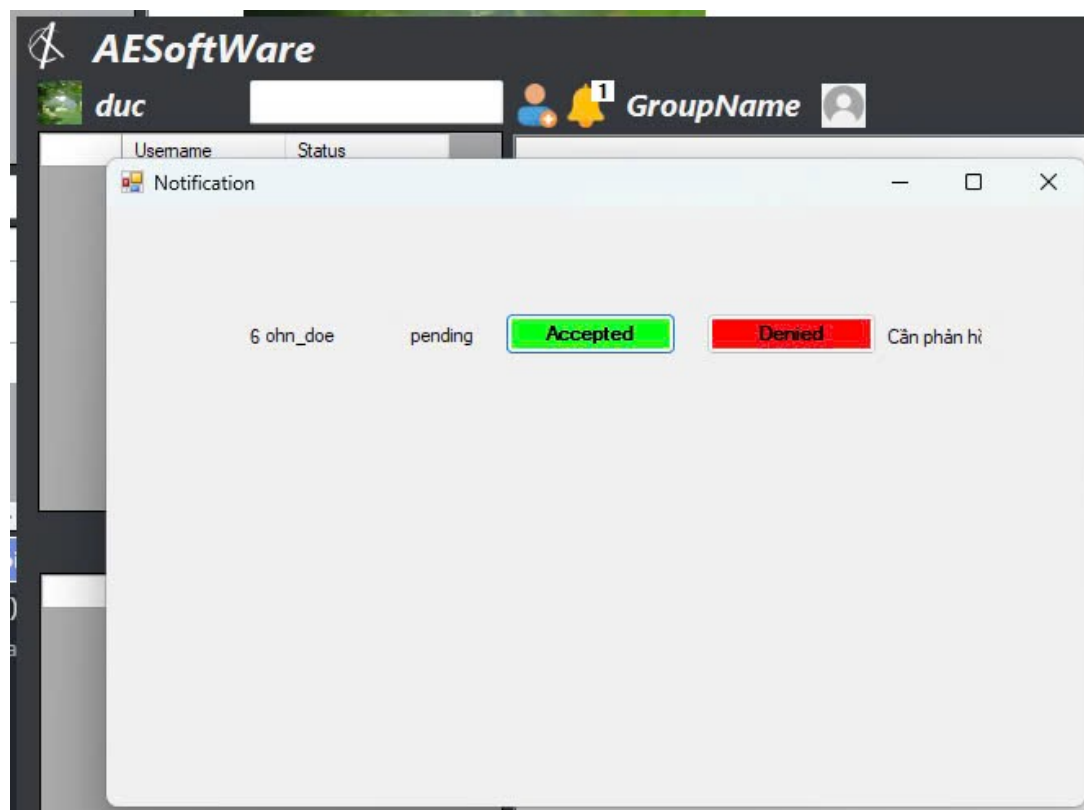
Hình 3.16 Tham gia vào group

Hình 3.16: Khi người dùng nhấn vào nút Join Group, hệ thống sẽ mở lên giao diện tham gia vào nhóm và ở đây người dùng chỉ cần nhập mã nhóm thì tên nhóm sẽ tự hiện ra. Sau khi nhập đầy đủ các thông tin, người dùng nhấn nút Join thì sẽ được tham gia vào nhóm đó, nếu đã tham gia rồi thì hệ thống sẽ báo bạn đã tham gia vào nhóm, vui lòng nhập nhóm khác.



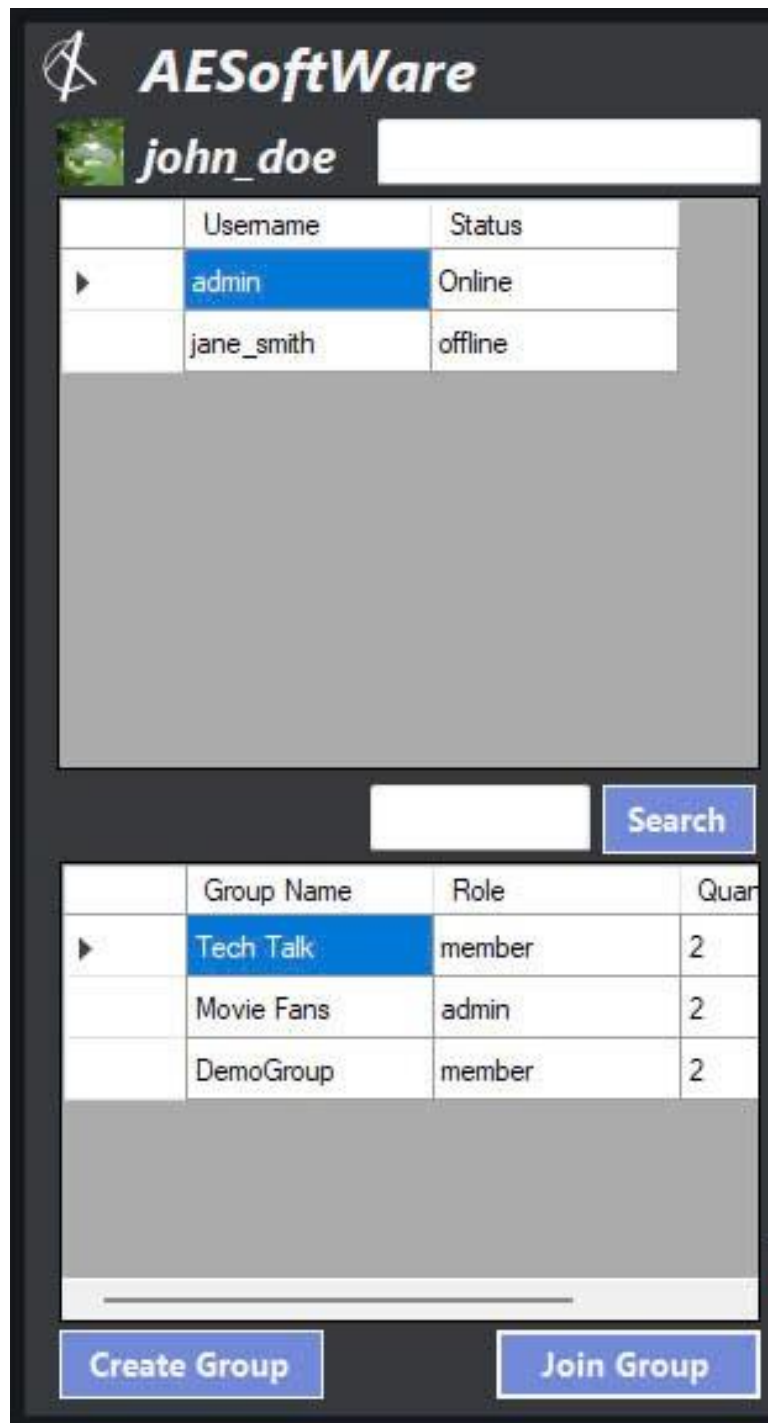
Hình 3.17 Gửi kết bạn

Hình 3.17: Khi người dùng nhập tên của một người dùng khác trên thanh tìm kiếm bạn bè và nhấn vào nút có icon user thì hệ thống sẽ báo đã gửi kết bạn thành công nếu người dùng đó tồn tại, ngược lại sẽ báo người dùng không tồn tại.



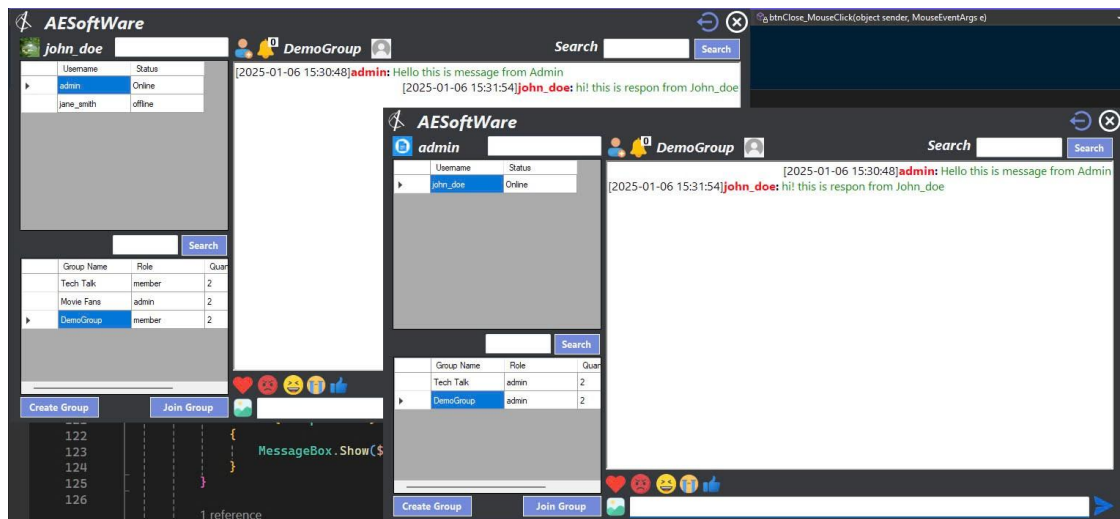
Hình 3.18 Chờ chấp nhận lời mời kết bạn

Hình 3.18: Khi được một người dùng khác gửi lời mời thì ở phần nút chuông sẽ hiện thông báo lên 1, nếu có 2 thì sẽ tăng lên 2. Khi nhấn vào nút chuông hệ thống sẽ hiện ra tên người dùng muốn gửi kết bạn và trạng thái đang là chờ đợi. Bạn có thể nhấp Accepted để chấp nhận hoặc nhấn Denied để từ chối.



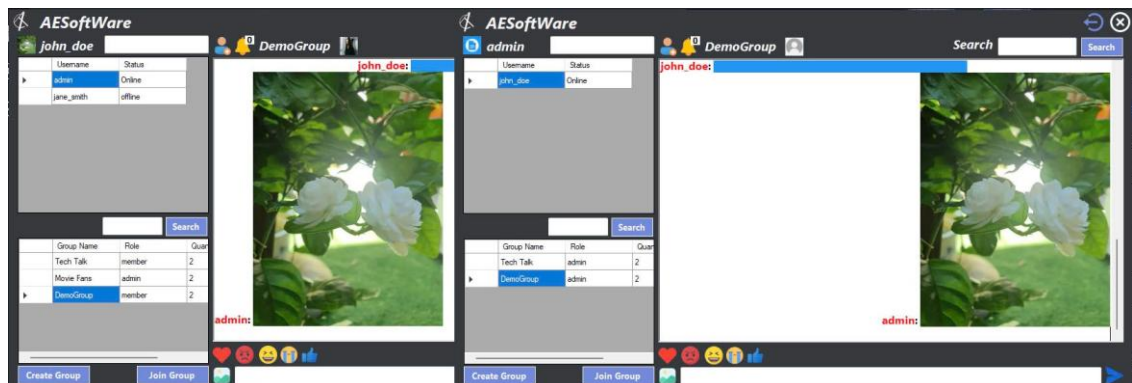
Hình 3.19 Danh sách bạn bè và nhóm

Hình 3.19: Sau khi thực hiện các bước tạo nhóm, tham gia nhóm và kết bạn thì ta có được danh sách bạn bè và nhóm như hình 3.19.



Hình 3.20 Tương tác giữa 2 người dùng

Hình 3.20: Khi 2 người dùng cùng kết nối 1 server với nhau và đã kết bạn hay tham gia chung nhóm thì sẽ nhắn tin, tương tác qua lại với nhau như mọi ứng dụng mạng xã hội khác. Người dùng có thể gửi tin nhắn, hoặc hình ảnh, file như hình 3.20 và hình 3.21.



Hình 3.21 Gửi hình ảnh cũng như các tệp

The screenshot shows a web browser window with the title 'frm_Report'. The address bar contains '1 of 1' and navigation icons. The main content area displays a table with the following data:

Group ID	Group Name	Created By
1	FCO	paulo

Hình 3.22 Giao diện Export

Hình 3.22: Khi nhấn nút Export trên thanh Menu, hệ thống sẽ xuất ra bảng báo cáo danh sách các nhóm mà bạn đang làm admin.

CHƯƠNG 4: KẾT LUẬN

Ứng dụng Chat App được xây dựng trên nền tảng Windows Forms đã hoàn thành với đầy đủ các chức năng cần thiết, bao gồm đăng ký, đăng nhập, gửi/nhận tin nhắn, quản lý nhóm, và chia sẻ tệp tin. Với việc kết hợp giữa giao diện thân thiện, trực quan và backend mạnh mẽ thông qua giao thức TCP/IP, hệ thống đảm bảo khả năng giao tiếp thời gian thực, bảo mật dữ liệu, và quản lý hiệu quả các phiên làm việc của người dùng. Hệ thống server không chỉ xử lý logic phức tạp mà còn lưu trữ thông tin quan trọng trên cơ sở dữ liệu, giúp đồng bộ hóa dữ liệu giữa các phiên và cung cấp trải nghiệm liền mạch. Đây là một ứng dụng phù hợp cho nhu cầu giao tiếp cá nhân và nhóm, có tiềm năng mở rộng thêm nhiều tính năng nâng cao trong tương lai, như tích hợp video call, mã hóa tin nhắn, hoặc triển khai trên các nền tảng khác. Việc hoàn thiện ứng dụng này không chỉ góp phần giải quyết nhu cầu giao tiếp hiệu quả mà còn là minh chứng cho việc ứng dụng linh hoạt các công nghệ lập trình vào thực tiễn.

TÀI LIỆU THAM KHẢO

Nguyễn Hà Giang, Mai Ngọc Thu, Nguyễn Đình Ánh, Nguyễn Huy Cường
(2023). *Tài liệu học tập phần “ Lập trình trên môi trường windows”* Hutech

Source code tham khảo ở <https://github.com/dihlogg/ChatApplication>