

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN &  
TRUYỀN THÔNG VIỆT HÀN

**Khoa Khoa Học Máy Tính**



ĐỒ ÁN CƠ SỞ 1

**TÊN ĐỀ TÀI**

**SỬ DỤNG NGÔN NGỮ JAVA ĐỂ XÂY DỰNG  
CHƯƠNG TRÌNH CHAT APPLICATION THEO  
PHƯƠNG THỨC SOCKET**

Sinh viên thực hiện : **Trần Quang Tuấn – 23IT300**

**Huỳnh Văn Huy – 23IT100**

Lớp : **23IT4**

Giảng viên hướng dẫn: **Ths. Trịnh Thị Ngọc Linh**

Đà Nẵng, tháng 06 năm 2024

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN &  
TRUYỀN THÔNG VIỆT HÀN

**Khoa Khoa Học Máy Tính**



ĐỒ ÁN CƠ SỞ 1

**TÊN ĐỀ TÀI**

**SỬ DỤNG NGÔN NGỮ JAVA ĐỂ XÂY DỰNG  
CHƯƠNG TRÌNH CHAT APPLICATION THEO  
PHƯƠNG THỨC SOCKET**

Sinh viên thực hiện : **Trần Quang Tuấn – 23IT300**

**Huỳnh Văn Huy – 23IT100**

Lớp : **23IT4**

Giảng viên hướng dẫn: **Ths. Trịnh Thị Ngọc Linh**

Đà Nẵng, tháng 06 năm 2024

# LỜI CẢM ƠN

Để đồ án này đạt kết quả tốt đẹp. em đã nhận được sự hỗ trợ giúp đỡ của thầy cô. Với tình cảm sâu sắc, chân thành, cho phép em được bày tỏ lòng biết ơn đến tất cả quý thầy cô đã tạo điều kiện giúp đỡ trong quá trình học tập và nghiên cứu phát triển đề tài.

Trước hết em xin gửi tới các giảng viên trường Đại học Công nghệ thông tin và truyền thông Việt - Hàn lời chào trân trọng, lời chúc sức khỏe và lời cảm ơn sâu sắc. Với sự quan tâm, dạy bảo tận tình chu đáo của các giảng viên đã truyền tải các kỹ năng cần thiết, đến nay em có thể hoàn thành đồ án một cách tốt đẹp.

Đặc biệt, em xin gửi lời cảm ơn chân thành đến Cô giáo – ThS Trịnh Thị Ngọc Linh đã quan tâm, giúp đỡ tận tình cho em để hoàn thành tốt đồ án này trong thời gian qua.

Với điều kiện thời gian cũng như kinh nghiệm còn hạn chế, đồ án này không thể tránh được những thiếu sót. Em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của các thầy cô để em có điều kiện bổ sung, nâng cao ý thức của mình, phục vụ tốt hơn công tác thực tế sau này.

Chúng em xin chân thành cảm ơn!

## NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

# MỤC LỤC

<b>LỜI CẢM ƠN .....</b>	<b>i</b>
<b>NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN.....</b>	<b>ii</b>
<b>MỤC LỤC .....</b>	<b>iii</b>
<b>MỤC LỤC HÌNH ẢNH.....</b>	<b>iv</b>
<b>MỞ ĐẦU.....</b>	<b>1</b>
1.    Giới thiệu đề tài.....	1
2.    Mục tiêu của đề tài .....	1
3.    Phạm vi nghiên cứu.....	2
4.    Nội dung và kế hoạch thực hiện .....	2
5.    Bố cục báo cáo .....	3
<b>CHƯƠNG 1. CƠ SỞ LÝ THUYẾT .....</b>	<b>4</b>
1.1. Giới thiệu sơ lược về LTM.....	4
1.1.1. Giới thiệu LTM: .....	4
1.1.2. Các điều quan trọng về LTM. ....	4
1.2. Giới thiệu về mô hình client – server. ....	5
1.3. Giao thức Socket.IO. ....	5
1.3.1. Giới thiệu về Socket .....	6
1.3.2. Lập trình mạng trong Java.....	6
1.4. Sơ lược về hệ QTCSDL .....	7
1.4.1. MySQL .....	7
1.4.2. HeidiSQL.....	7
<b>CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG .....</b>	<b>8</b>
2.1. Phân tích yêu cầu.....	8
2.2. Phân tích chức năng:.....	8
2.2.1. Trên Client.....	8
2.2.2. Trên Server .....	8
2.3. Phân tích và thiết kế chương trình:.....	9
<b>CHƯƠNG 3. CHƯƠNG TRÌNH DEMO .....</b>	<b>10</b>
3.1. Môi trường triển khai.....	10
3.2. Chương trình:.....	10
3.2.1. Giao diện Server .....	10
3.2.2. Giao diện Client.....	11
<b>KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....</b>	<b>15</b>
<b>TÀI LIỆU THAM KHẢO.....</b>	<b>16</b>

## MỤC LỤC HÌNH ẢNH

Hình 1: Client – Server .....	5
Hình 2: Giao Diện Server.....	10
Hình 3: Trang Đăng Nhập .....	11
Hình 4: Trang Đăng ký .....	11
Hình 5: Trang Chủ .....	12
Hình 6: Gửi Message .....	12
Hình 7: Nhận Emotion .....	13
Hình 8: Gửi Emotion.....	13
Hình 9: Gửi ảnh .....	14
Hình 10: Nhận ảnh .....	14

# MỞ ĐẦU

## 1. Giới thiệu đề tài

Hiện nay mạng internet toàn cầu đã phát triển rất mạnh, đáp ứng rất tốt các nhu cầu về thông tin khoa học, giải trí, liên lạc, mua bán, quảng cáo vv...của con người. Internet đã trở thành một phần không thể thiếu với cuộc sống hiện đại. Về bản chất, Internet chẳng qua là một hệ thống mạng, liên kết các máy tính trên toàn thế giới lại theo những chuẩn chung của nó. Với phạm vi nhỏ hơn nhưng cũng rất tương tự, có những mạng khác đó là WAN, LAN.... mỗi mô hình đều được ứng dụng rất nhiều tiện ích khác nhau.

Với mục tiêu để cho các sinh viên khi học xong môn học với một kiến thức tốt về lập trình hướng đối tượng, cơ sở dữ liệu...có thể củng cố kiến thức và vận dụng, phát huy kiến thức đã học. Vì vậy, ngày hôm nay, lúc em đang học về lập trình Java Socket. đã tìm hiểu được đề tài " Sử dụng ngôn ngữ JAVA để xây dựng chương trình Chat Application theo phương thức SOCKET ", điều đó giúp em thực hành để có cơ hội rõ hơn về mạng, một lĩnh vực lập trình thú vị.

Ứng dụng hoạt động dựa vào giao thức Socket.IO để truyền dữ liệu giữa hai máy khác nhau, cụ thể ở đây là một máy Server để cung cấp dịch vụ và nhiều máy client sử dụng dịch vụ của server cung cấp để cho người sử dụng.

Ngôn ngữ lập trình ở đây là ngôn ngữ Java vì Java là ngôn ngữ có thể được sử dụng để viết ứng dụng trên hầu hết các thiết bị (ở đây ta quan tâm đến thiết bị có hỗ trợ mạng). Trọng điểm, có mấy khả năng nổi trội do Java cung cấp mà ta có thể sử dụng là:

- Lập trình đa tuyến trình hay threads (Cùng 1 ứng dụng nhưng có thể chạy nhiều tiến trình song song với nhau thay vì cứ phải tiến trình này đợi tiến trình kia kết thúc mới có thể thực hiện).

- Java là một nền tảng có thể chạy trên nhiều loại thiết bị với nhiều hệ điều hành khác nhau nên có phạm vi ứng dụng rất rộng rãi.

## 2. Mục tiêu của đề tài

Sử dụng thành thạo ngôn ngữ lập trình Java để tạo ra một ứng dụng chat có sử dụng đến các giao thức đã học đặc biệt ở đây là sử dụng giao thức Socket.IO. Hiểu được cách vận hành của giao thức, hiểu được sự tương tác giữa server và client.

### 3. Phạm vi nghiên cứu

- Thiết lập kết nối Socket.IO.
- Gửi và nhận tin nhắn real-time
- Bảo mật và xác thực.
- Tạo giao diện người dùng.
- Tích hợp các tính năng bổ sung.
- Kiểm thử và tối ưu

### 4. Nội dung và kế hoạch thực hiện

Thời gian	Nội dung thực hiện
Từ 1/4 đến 7/4	Xác định mục tiêu và yêu cầu
Từ 8/4 đến 20/4	Thiết kế hệ thống, thiết kế giao diện
Từ 21/4 đến 4/5	Xây dựng hướng kết nối
Từ 5/5 đến 11/5	Xây dựng hoàn thiện ban đầu
Từ 12/5 đến 18/5	Kiểm thử và điều chỉnh
Từ 19/5 đến 25/5	Hoàn thiện toàn bộ ứng dụng



## **5. Bố cục báo cáo**

Sau phần *Mở đầu*, báo cáo được trình bày trong ba chương, cụ thể như sau:

Chương 1. *Tổng quan về các cơ sở lý thuyết*

Chương 2. *Phân tích thiết kế các yêu cầu và thiết kế hệ thống cho chương trình*

Chương 3. *Chương trình DEMO*

Cuối cùng là *Kết luận*, *Tài liệu tham khảo* và *Phụ lục* liên quan đến đề tài.

# CHƯƠNG 1. CƠ SỞ LÝ THUYẾT

## 1.1. Giới thiệu sơ lược về LTM

### 1.1.1. Giới thiệu LTM:

Lập trình mạng là một lĩnh vực quan trọng trong công nghệ thông tin và phát triển phần mềm, tập trung vào việc thiết lập, quản lý và tương tác với các kết nối mạng, gửi và nhận dữ liệu qua mạng.

### 1.1.2. Các điều quan trọng về LTM.

Mục tiêu của lập trình mạng: Lập trình mạng nhằm phát triển các ứng dụng và dịch vụ có khả năng giao tiếp qua mạng. Điều này bao gồm việc truyền tải dữ liệu, thiết lập kết nối, xử lý giao tiếp giữa các máy tính và các thiết bị trong mạng.

Giao thức mạng: Để truyền thông qua mạng, lập trình mạng thường sử dụng các giao thức mạng như TCP/IP, UDP, HTTP, FTP, và nhiều giao thức khác. Các giao thức này định rõ cách dữ liệu được đóng gói, gửi và nhận trên mạng.

Socket Programming: Một phần quan trọng của lập trình mạng là sử dụng socket. Một cơ chế cho phép các máy tính trò chuyện và truyền dữ liệu qua mạng. Lập trình socket là cách cơ bản để xây dựng ứng dụng mạng.

Các ngôn ngữ lập trình mạng: Các ngôn ngữ phổ biến để lập trình mạng bao gồm C, C++, Python, Java, Ruby, và nhiều ngôn ngữ khác. Mỗi ngôn ngữ cung cấp thư viện và API để tạo ứng dụng mạng.

Các loại ứng dụng mạng: Lập trình mạng có thể được áp dụng vào nhiều loại ứng dụng, bao gồm ứng dụng web (web development), ứng dụng máy chủ (server applications), ứng dụng trò chuyện (chat applications), trò chơi trực tuyến (online gaming), IoT (Internet of Things), và nhiều ứng dụng khác.

Bảo mật mạng: An ninh mạng (network security) là một khía cạnh quan trọng của lập trình mạng. Phải có các biện pháp để đảm bảo tính bảo mật và bảo vệ dữ liệu khi truyền qua mạng.

Debugging và Testing: Việc kiểm tra và gỡ lỗi (debugging) trong lập trình mạng thường khá phức tạp do tình hình mạng thay đổi và các vấn đề về đồng thời hóa (concurrency). Cần phải sử dụng các công cụ và phương pháp đặc biệt để kiểm tra và sửa lỗi.

Phát triển ứng dụng đa nền tảng: Lập trình mạng có thể được thực hiện trên nhiều nền tảng khác nhau, bao gồm ứng dụng máy tính, ứng dụng di động, và ứng dụng web.

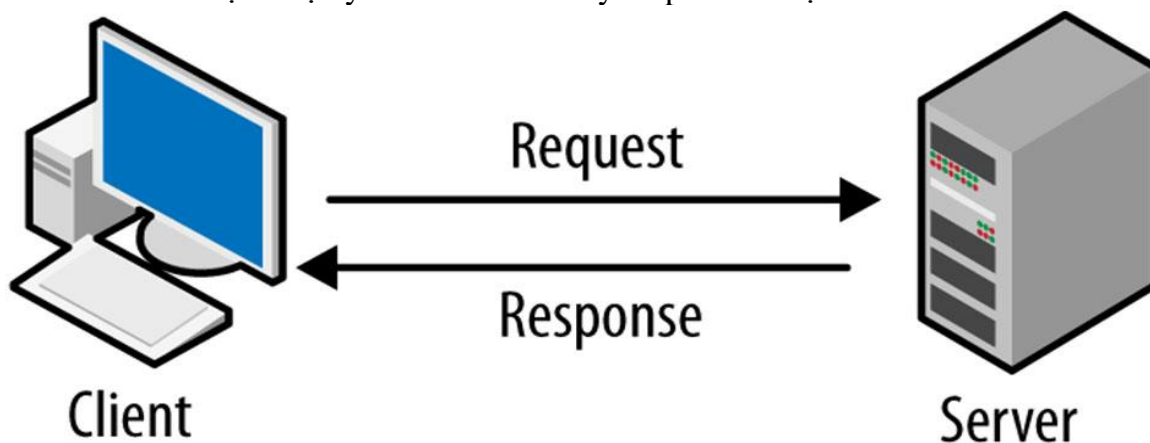
## 1.2. Giới thiệu về mô hình client – server.

Server được hiểu là máy chủ, thường là một hệ thống máy lớn, có bộ xử lý mạnh. có khả năng hoạt động đáng tin cậy, có khả năng lưu trữ dữ liệu lớn, nó chuyên quản lý tài nguyên ( chủ yếu là cơ sở dữ liệu), cung cấp các dịch vụ mạng cho các máy khách hàng (client) sử dụng. Bình thường nó chạy suốt thời gian thực và sẵn sàng chấp nhận các yêu cầu kết nối và các yêu cầu dịch vụ khác từ máy khách (Client).

Client là máy khách, nó thường được sử dụng bởi người dùng cuối. Nó hoạt động dựa trên việc sử dụng dịch vụ mà máy server cung cấp để thực hiện các công việc mà người dùng cuối mong muốn.

Quá trình hoạt động của mô hình này là lặp lại 2 quá trình sau:

- Client gửi yêu cầu lên server.
- Server nhận được yêu cầu thì sẽ xử lý và phản hồi lại client.



*Hình 1: Client – Server*

- Mỗi cổng trong mỗi thời điểm được sử dụng cho tối đa là 1 ứng dụng. Ví dụ như một ứng dụng nào đó đã sử dụng cổng 55 thì ứng dụng khác không thể sử dụng cổng 55 đó nữa, khi nào ứng dụng trước đó đóng cổng 55 lại thì ứng dụng sau mới có thể sử dụng được.

## 1.3. Giao thức Socket.IO.

Socket.IO là một thư viện và giao thức dùng để phát triển ứng dụng web thời gian thực (real-time) thông qua việc truyền dữ liệu giữa máy chủ và trình duyệt hoặc giữa các máy chủ. Socket.IO cung cấp một cách để thiết lập kết nối WebSocket hoặc sử dụng các phương thức thay thế nếu WebSocket không khả dụng, như long polling.

Socket.IO hỗ trợ cả phía máy chủ và phía trình duyệt, cho phép bạn xây dựng các ứng dụng web thời gian thực như trò chơi trực tuyến, ứng dụng trò chuyện, thông báo thời gian thực, và các ứng dụng khác cần gửi và nhận dữ liệu ngay lập tức.

### 1.3.1. Giới thiệu về Socket

Socket là một khái niệm để định vị một dịch vụ của một máy tính trên mạng khi kết hợp hai khái niệm IP và Port lại.

Socket có những nhiệm vụ sau:

- Gắn một cổng trên máy.
- Lắng nghe các kết nối.
- Kết nối/Đóng kết nối đến máy tính ở xa qua cổng đã gắn.
- Gửi và nhận dữ liệu.
- Lắng nghe dữ liệu đến.

Thực chất, Socket chẳng qua là sự kết hợp giữa địa chỉ IP của máy tính và cổng Port mà ứng dụng sử dụng. Chính vì sự kết hợp đó mà nó trở thành một khái niệm mà từ đó các ngôn ngữ lập trình có chứa các gói, các giao diện lập trình (API) để hỗ trợ các nhà lập trình dễ dàng trong việc lập trình liên quan đến mạng qua giao thức Socket.IO. Điển hình là ngôn ngữ Java mà em sẽ giới thiệu ngay sau đây.

### 1.3.2. Lập trình mạng trong Java

#### 1.3.2.1. Gói ServerSocket

##### **Nhiệm vụ chính:**

- Mở một cổng kết nối trên server, sẵn sàng hoạt động trên cổng đó.

***Public Socket(int port): Tạo cổng kết nối trên cổng port***

- Lắng nghe, chờ đợi các yêu cầu kết nối trên cổng đó và kết nối khi có yêu cầu

***Public accept(): Chấp nhận một kết nối và trả về một đối tượng Socket mô tả nguồn đã được kết nối***

- Tạo luồng dữ liệu vào và ra để có thể truyền dữ liệu trên cổng đến một socket ở máy khách. (Đây là chức năng quan trọng nhất).

***getInputStream(): Trả về luồng mà có thể nhập dữ liệu vào***

***getOutputStream(): Trả về một luồng ghi dữ liệu***

- Đóng kết nối khi không sử dụng.

***close(): Đóng kết nối khi không cần***

### 1.3.2.2. Gói Socket

#### **Nhiệm vụ chính:**

- Kết nối tới một máy chủ thông qua địa chỉ IP và số liệu cổng trên máy chủ đó.

***Public void Socket(InetAddress address, int port)***

- Tạo luồng dữ liệu ra vào:

***Public OutputStream getOutputStream();***

***Public InputStream getInputStream();***

- Đóng kết nối

***Public void Close();***

## 1.4. Sơ lược về hệ QTCSDL

### 1.4.1. MySQL

MySQL là một hệ quản trị cơ sở dữ liệu phổ biến được sử dụng trên toàn thế giới. Nó hỗ trợ nhiều loại dữ liệu, bao gồm dữ liệu cấu trúc và dữ liệu phi cấu trúc, và được sử dụng trong các ứng dụng web, ứng dụng doanh nghiệp, và nhiều ứng dụng khác.

MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở, có nghĩa là bạn có thể tải về, sử dụng và sửa đổi nó mà không cần phải mua giấy phép. Tuy nhiên, Oracle cũng cung cấp một phiên bản MySQL Enterprise với các tính năng và hỗ trợ mở rộng cho các doanh nghiệp lớn.

### 1.4.2. HeidiSQL

HeidiSQL là công cụ quản lý cơ sở dữ liệu mã nguồn mở miễn phí chạy trên Windows được ra đời năm 2002 bởi Ansgar. Nó cho phép bạn xem, chỉnh sửa dữ liệu cũng như cấu trúc từ các máy tính chạy các hệ thống cơ sở dữ liệu MariaDB, MySQL, Microsoft SQL, PostgreSQL và SQLite.

HeidiSQL có một giao diện web dễ sử dụng. Nó cung cấp một giao diện đồ họa để quản lý cơ sở dữ liệu một cách dễ dàng và hiệu quả. Sử dụng GUI trực quan của nó để duyệt, tạo và chỉnh sửa dữ liệu từ cơ sở dữ liệu cũng như để tạo và chỉnh sửa bảng, lượt xem, quy trình, kích hoạt và các sự kiện theo lịch trình. Dễ dàng so sánh và đồng bộ hóa dữ liệu và cấu trúc của bạn giữa cơ sở dữ liệu cục bộ và từ xa. Đây là một công cụ hữu ích và đáng tin cậy được thiết kế cho các nhà phát triển web sử dụng máy chủ MySQL phổ biến, Microsoft SQL và PostgreSQL.

## CHƯƠNG 2. PHÂN TÍCH THIẾT KẾ HỆ THỐNG

### 2.1. Phân tích yêu cầu

Viết chương trình cho phép chat giữa các người dùng qua mạng.

### 2.2. Phân tích chức năng:

#### 2.2.1. *Trên Client*

Tạo một username do người dùng đăng ký.

Đăng nhập vào server với tài khoản đã tạo.

Nhận danh sách các nick đang online trên server.

Gửi tin nhắn tới tài khoản đang online trên server.

Thực hiện gửi hình ảnh đến một client khác trên server.

Nhận tin nhắn từ client khác.

#### 2.2.2. *Trên Server*

Tạo cổng kết nối sẵn sàng đón nhận các kết nối từ client.

Xử lý yêu cầu đăng nhập từ client.

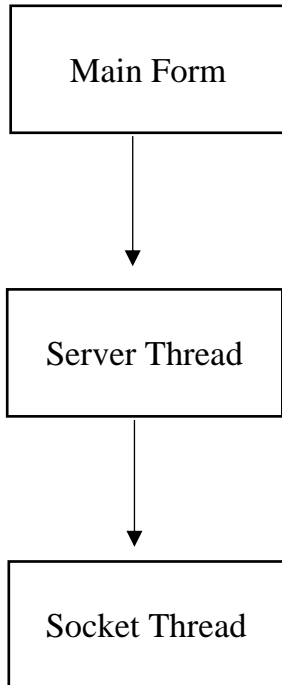
- Nếu chưa có tài khoản thì phải yêu cầu tạo tài khoản.
- Nếu có tài khoản rồi thì chỉ cho phép đăng nhập.

Trả về danh sách các user đang hoạt động cho user yêu cầu.

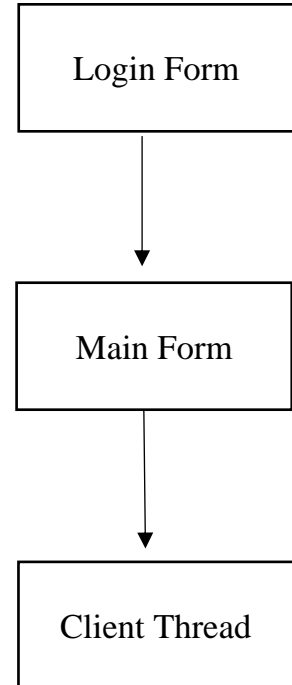
Nhận tin nhắn từ một user khác.

### 2.3. Phân tích và thiết kế chương trình:

#### SERVER



#### CLIENT



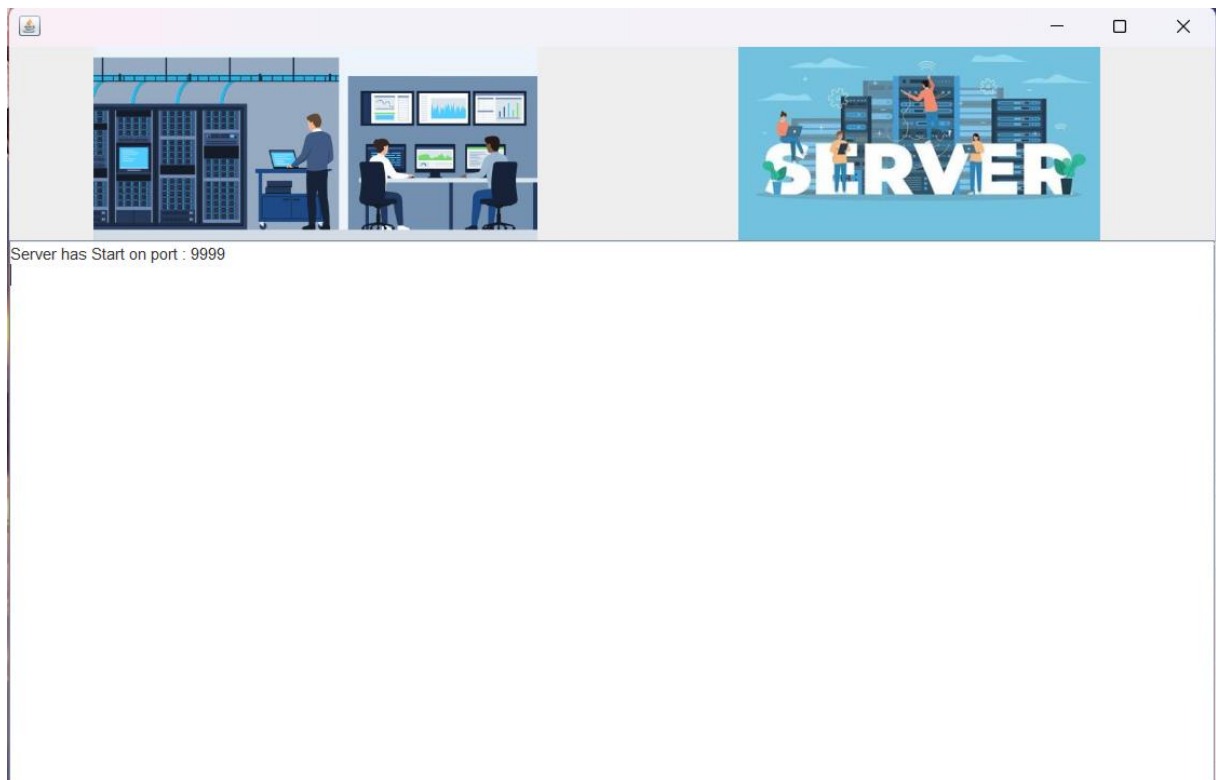
## CHƯƠNG 3. CHƯƠNG TRÌNH DEMO

### 3.1. Môi trường triển khai

- Sử dụng Netbean để xây dựng chương trình.
- Sử dụng HeidiSQL để thiết kế cơ sở dữ liệu.

### 3.2. Chương trình:

#### 3.2.1. Giao diện Server

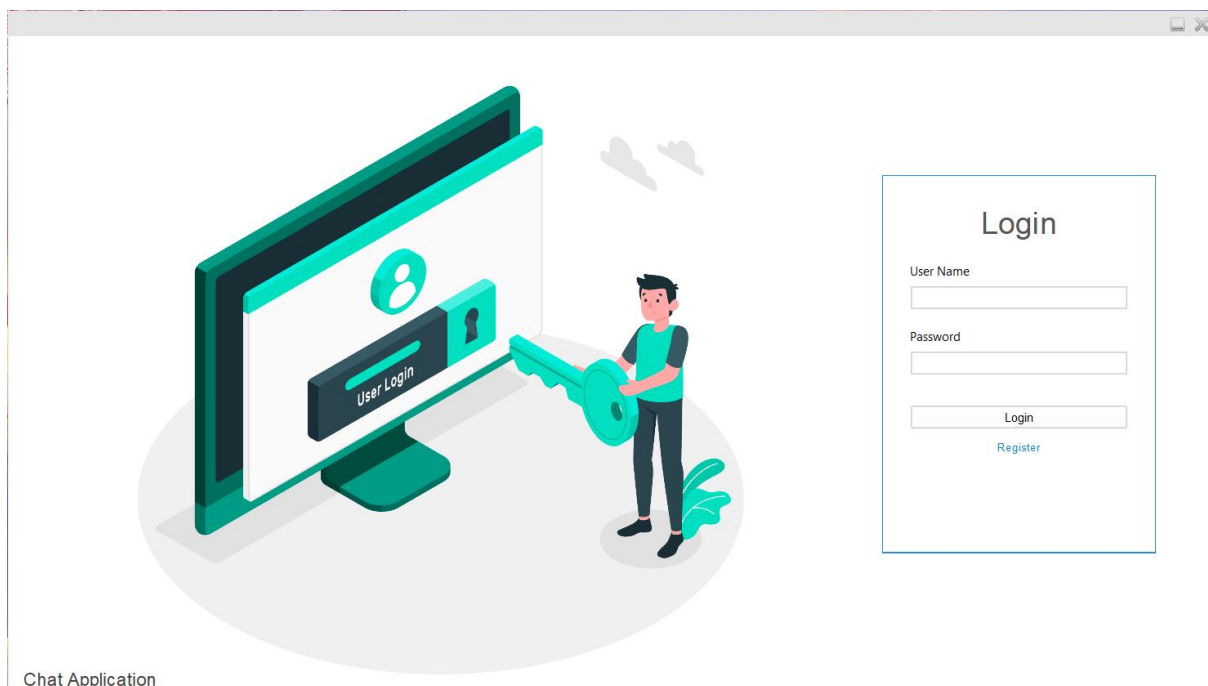


*Hình 2: Giao Diện Server*



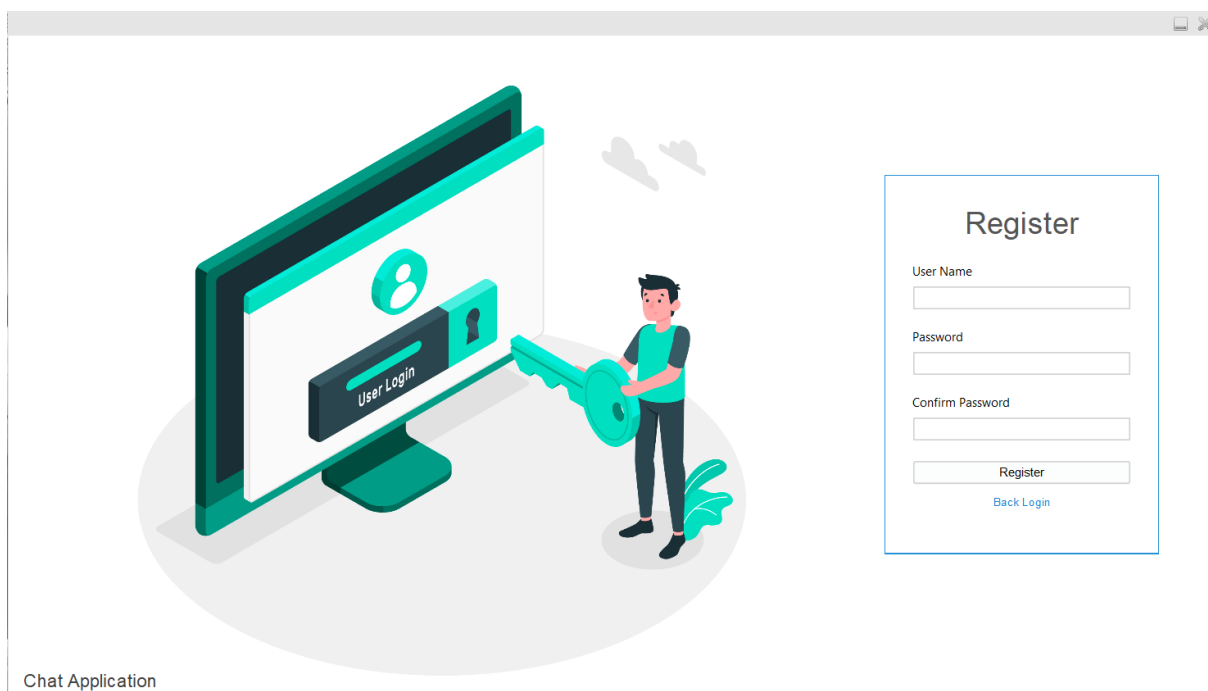
### 3.2.2. Giao diện Client

#### 3.2.2.1. Trang Đăng Nhập



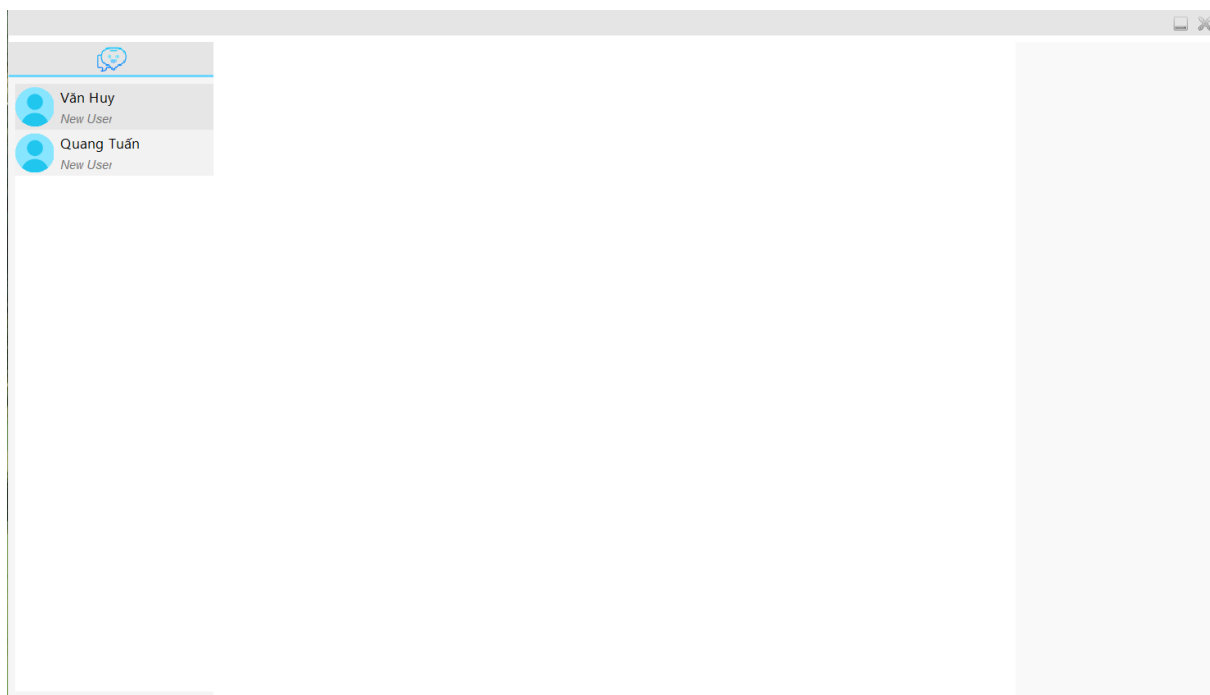
**Hình 3: Trang Đăng Nhập**

#### 3.2.2.2. Trang Đăng Ký



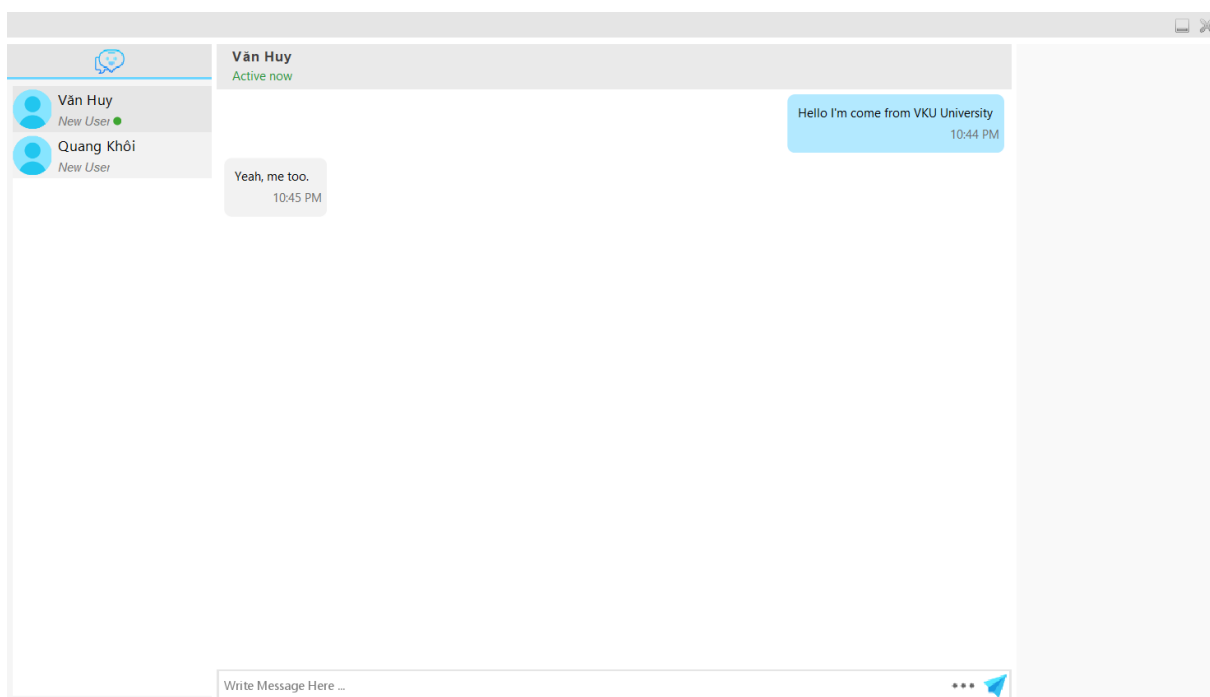
**Hình 4: Trang Đăng ký**

### 3.2.2.3. Trang chủ



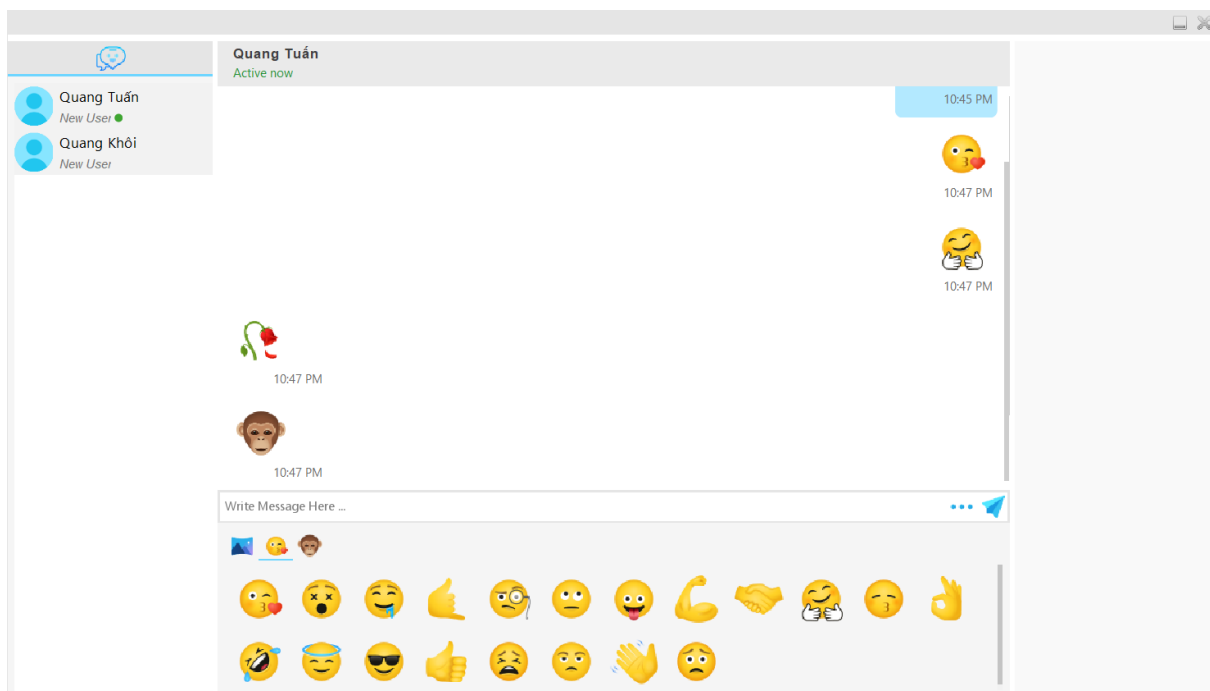
**Hình 5: Trang Chủ**

### 3.2.2.4. Gửi message

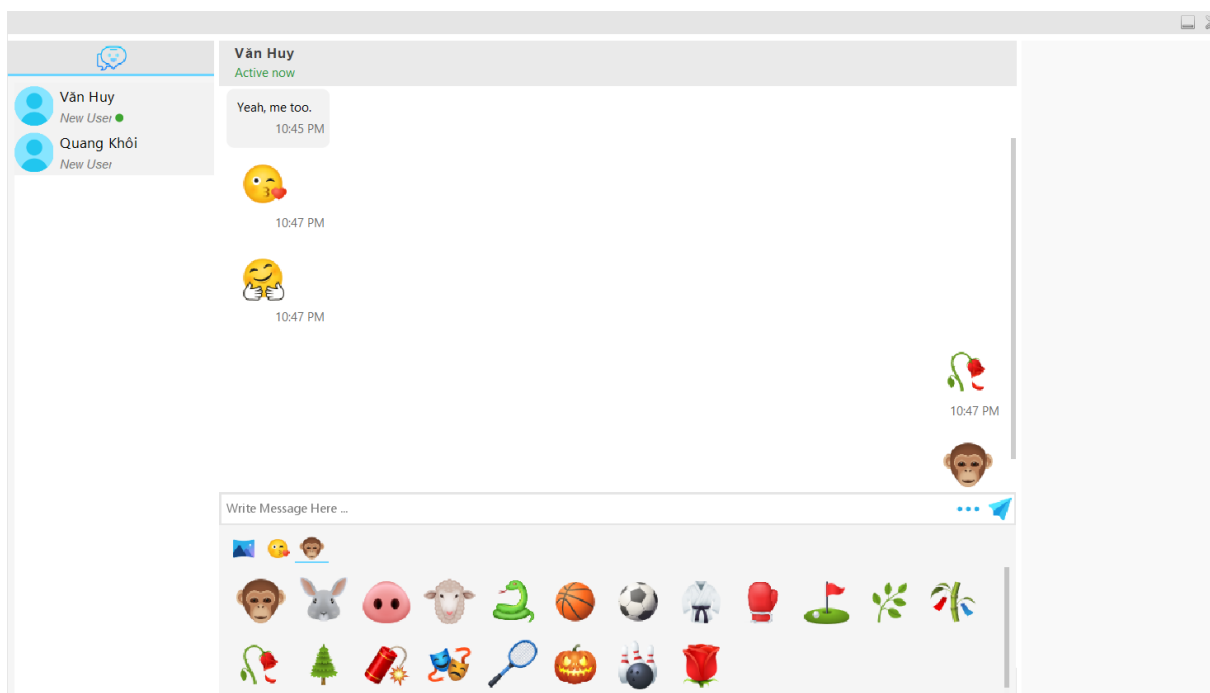


**Hình 6: Gửi Message**

### 3.2.2.5. Gửi Emotion

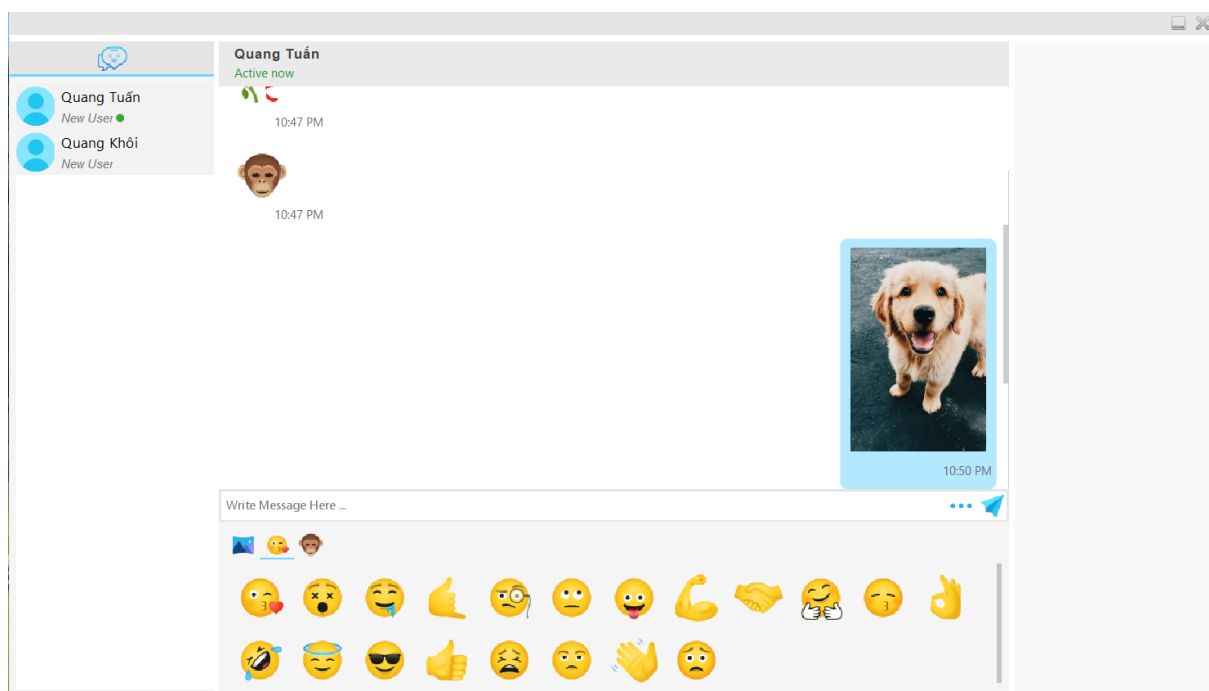


**Hình 7: Nhận Emotion**

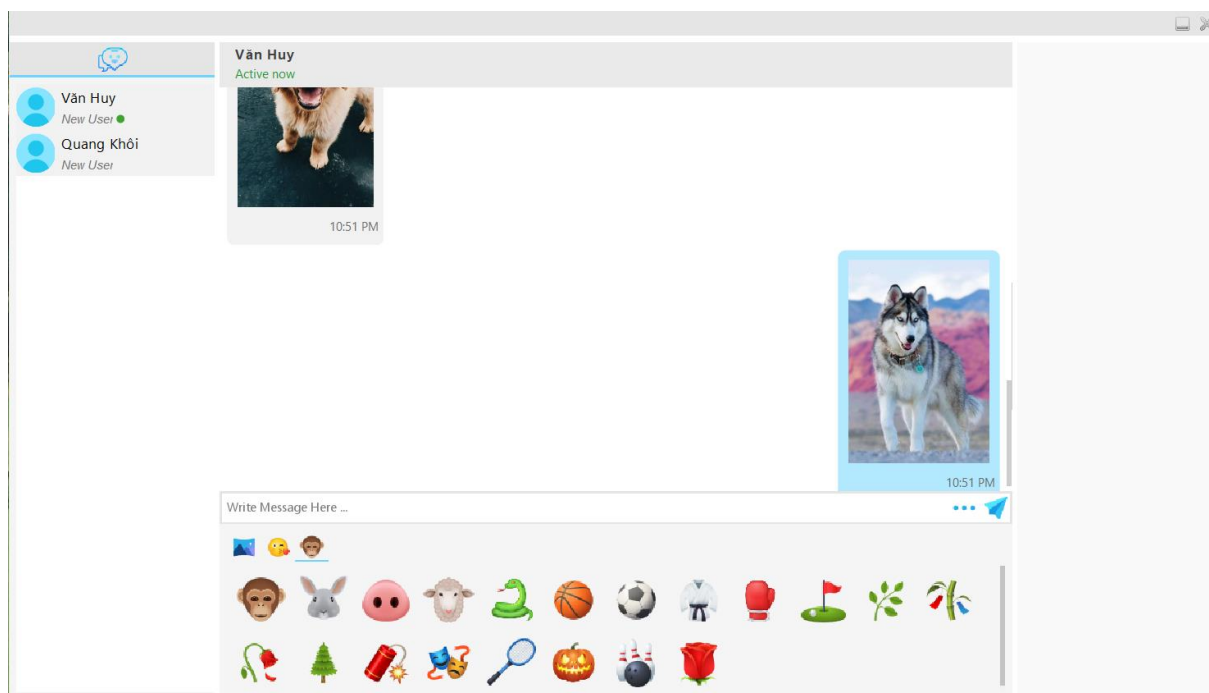


**Hình 8: Gửi Emotion**

### 3.2.2.6. Gửi ảnh



**Hình 9: Gửi ảnh**



**Hình 10: Nhận ảnh**

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## 1. Kết luận

- Khi thực hiện chương trình chúng em có một cái nhìn trực quan hơn về việc kết nối giữa Client thông qua Server. Biết cách hoạt động khi Client gửi ảnh hoặc tin nhắn qua Client khác.

## 2. Hướng phát triển

- Hạn chế của chương trình là chưa thể gửi được file dạng PDF, DOCX... chúng em sẽ tìm hiểu và cải thiện chương trình để được tối ưu hơn.
- Tiếp tục mở rộng các tính năng của ứng dụng như hỗ trợ gửi/nhận các định dạng file phổ biến khác, tính năng tải ảnh về máy, call video, v.v.
- Tối ưu hóa hiệu năng, tốc độ và trải nghiệm người dùng của ứng dụng.
- Xem xét triển khai ứng dụng trên đa nền tảng (mobile, web, desktop) để tăng khả năng tiếp cận và tiện dụng.
- Đảm bảo an toàn, bảo mật thông tin người dùng và tuân thủ các quy định về quyền riêng tư.

## TÀI LIỆU THAM KHẢO

Thư viện trường Công Nghệ Thông tin và Truyền Thông Việt – Hàn.

Đồ án môn học Lập trình Java nâng cao.

W3schools, F8, Stack Overflow,...