

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH  
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN



BÁO CÁO  
ĐỒ ÁN SOCKET

Môn học: Mạng máy tính

Giáo viên hướng dẫn:

Đồ Hoàng Cường  
Nguyễn Thanh Quân

Sinh viên thực hiện:

Bùi Quốc Trung - 20120023  
Dũ Quốc Huy - 20120101  
Nguyễn Công Đại - 20120448

Thành phố Hồ Chí Minh, tháng 5 năm 2022

# Mục Lục

<b>1 Thông tin nhóm . . . . .</b>	<b>2</b>
1.1 Đề tài thực hiện: . . . . .	2
1.2 Thành viên nhóm . . . . .	2
<b>2 Mức độ hoàn thành . . . . .</b>	<b>2</b>
<b>3 Kích bản giao tiếp của chương trình . . . . .</b>	<b>3</b>
3.1 Giao thức trao đổi giữa Client và Server . . . . .	3
3.2 Cấu trúc thông điệp . . . . .	3
3.3 Kiểu dữ liệu của thông điệp . . . . .	3
3.4 Cách tổ chức cơ sở dữ liệu . . . . .	3
<b>4 Môi trường lập trình và các framework hỗ trợ . . . . .</b>	<b>4</b>
<b>5 Hướng dẫn sử dụng các tính năng chương trình . . . . .</b>	<b>4</b>
5.1 Phía Server . . . . .	4
5.2 Phía Client . . . . .	11
<b>6 Phân chia công việc . . . . .</b>	<b>15</b>
<b>Tài liệu tham khảo . . . . .</b>	<b>16</b>

# 1. Thông tin nhóm

## 1.1. Đề tài thực hiện:

### Đề 1

Viết chương trình gồm 1 ứng dụng server và 1 ứng dụng client để quản lý một “danh bạ số” sử dụng Socket, với giao thức TCP tại tầng Transport. Danh bạ này quản lý các thông tin các thành viên trong 1 tổ chức. Thông tin 1 thành viên gồm: mã số, họ và tên, số điện thoại, email, hình ảnh đại diện (avatar nhỏ), hình ảnh đại diện (avatar lớn). Danh bạ được lưu trữ tại server cùng với các file hình ảnh, server cung cấp các dịch vụ để client có thể truy vấn danh bạ (danh sách các thành viên) và truy vấn thông tin từng thành viên.

## 1.2. Thành viên nhóm

STT	Họ và tên	MSSV
1	Bùi Quốc Trung	20120023
2	DŨ Quốc Huy	20120101
3	Nguyễn Kông Đại	20120448

# 2. Mức độ hoàn thành

STT	Chức năng	Giải thích/Ghi chú	Mức độ hoàn thành
1	Truy vấn thông tin danh bạ	Client truy vấn danh sách các thành viên đang được server quản lý, hiển thị lên giao diện ít nhất các thông tin: mã số, họ và tên	100%
2	Truy vấn thông tin 1 thành viên trong danh bạ	Client truy vấn 1 thành viên trong danh bạ đang được server quản lý, hiển thị lên giao diện ít nhất các thông tin: mã số, họ và tên, số điện thoại, email	100%
3	Quản lý dữ liệu tại server bằng các loại file có cấu trúc như XML, JSON hoặc CSDL quan hệ	Dùng file có cấu trúc: File JSON	100%
4	Mở rộng chức năng số (1). Cho phép tải về hình ảnh đại diện ( <b>avatar ảnh nhỏ</b> ) từ server về client cho tất cả các thành viên trong danh bạ.	File ảnh avatar phải được lưu tại chính server. Hiển thị được hình ảnh avatar trên GUI của ứng dụng client sau khi tải về	100%
5	Mở rộng chức năng số (2). Cho phép tải về hình ảnh đại diện ( <b>avatar ảnh lớn</b> ) từ server về client khi truy vấn 1 thành viên trong danh bạ.	File ảnh avatar phải được lưu tại chính server. Hiển thị được hình ảnh avatar sau khi tải về trên GUI của ứng dụng client	100%
6	Hỗ trợ nhiều client truy cập đồng thời đến server	Nhiều client có thể kết nối đến sever	100%
<b>Tổng kết</b>			100%

Nhóm đã hoàn thành được tất cả các chức năng. Ngoài ra nhóm còn thêm các chức năng thêm, xóa và sửa thành viên tại server. Giao diện cũng khá ổn và đã xử lý các trường hợp có thể xảy ra như client tắt đột ngột, server tắt trước client...

### 3. Kịch bản giao tiếp của chương trình

#### 3.1. Giao thức trao đổi giữa Client và Server

Giao thức được sử dụng là giao thức TCP/IP ở tầng Transport.

#### 3.2. Cấu trúc thông điệp

- Sau khi bật Server, ta tiến hành khởi chạy Client và khi đó, Client gửi thông điệp giao tiếp yêu cầu kết nối đến Server. Sau khi Server chấp nhận yêu cầu thì một kết nối truyền dữ liệu giữa Client và Server được hình thành.
- Về phía Client sẽ có 3 lựa chọn ứng với 3 tính năng, đó là nhận thông tin của tất cả các thành viên, bao gồm cả việc tải về và hiển thị thông tin ảnh nhỏ (Avatar). Ở tính năng này, Client gửi yêu cầu lấy dữ liệu từ Server và từ đó Server sẽ truyền dữ liệu cho Client. Sau khi đã hoàn tất truyền nhận dữ liệu thì GUI của Client sẽ hiển thị danh sách tất cả các thành viên trong danh bạ được Server quản lý. Tính năng thứ 2 là tìm kiếm một thành viên trong danh bạ của Server và mở rộng thêm là tải về và hiển thị hình ảnh lớn (Cover). Việc truyền nhận dữ liệu ở tính năng thứ 2 cũng tương tự như tính năng thứ nhất, và sau khi hoàn tất thì thông tin của thành viên được tìm kiếm sẽ hiện ra. Dữ liệu được truyền theo quy trình

$$Send(Client) \rightarrow Receive(Server) \rightarrow Send(Server) \rightarrow Receive(Client)$$

- Tính năng thứ 3 là tính năng kết thúc phiên làm việc, Client sẽ gửi thông điệp ngắt kết nối đến Server, từ đó Client sẽ ngắt kết nối với Server và không tồn tại trên khung thông tin của Server.

#### 3.3. Kiểu dữ liệu của thông điệp

- Dữ liệu nhận vào và gửi đi từ cả Client và Server đều thuộc kiểu dữ liệu chuỗi (String), riêng phần dữ liệu cho các file ảnh là được Server gửi đi ở dạng nhị phân.
- Các dữ liệu kiểu Unicode (UTF-8) được mã hóa trước khi gửi đi và được trả về dạng ban đầu khi nhận (từ cả hai phía Client và Server).

#### 3.4. Cách tổ chức cơ sở dữ liệu

Dữ liệu được chia làm 2 phần để lưu trữ ở Server, một phần riêng cho dữ liệu dạng chuỗi như ID, name,... và một phần còn lại cho dữ liệu dạng ảnh.

- Dữ liệu dạng chuỗi được lưu trong 1 file JSON duy nhất là *member.json* nằm trong folder *database* được đặt chung thư mục với file thực thi của Server, file JSON này có cấu trúc rõ ràng để dễ tra cứu cũng như cập nhật.
- Dữ liệu dạng ảnh sẽ được lưu trong 1 folder và folder này nằm chung thư mục với file thực thi của Server dưới dạng *.jpg*, *.png*, và thông tin đường dẫn đến ảnh đó được lưu ở dạng chuỗi chung file JSON với các dữ liệu chuỗi ở trên.

```
{
    "001": {
        "fullname": "Đỗ Quốc Huy",
        "phone": "0123456789",
        "email": "huy2k2@gmail.com",
        "imageDir_small": "Image/small001.jpg",
        "imageDir_big": "Image/big001.jpg"
    },
    "002": {
        "fullname": "Nguyễn Công Đại",
        "phone": "0123456790",
        "email": "dai2k2@gmail.com",
        "imageDir_small": "Image/small002.jpg",
        "imageDir_big": "Image/big002.jpg"
    },
    "003": {
        "fullname": "Bùi Quốc Trung",
        "phone": "0123456791",
        "email": "trung2k2@gmail.com",
        "imageDir_small": "Image/small003.jpg",
        "imageDir_big": "Image/big003.jpg"
    }
}
```

Hình 1: Tổ chức dữ liệu trong file *member.json*

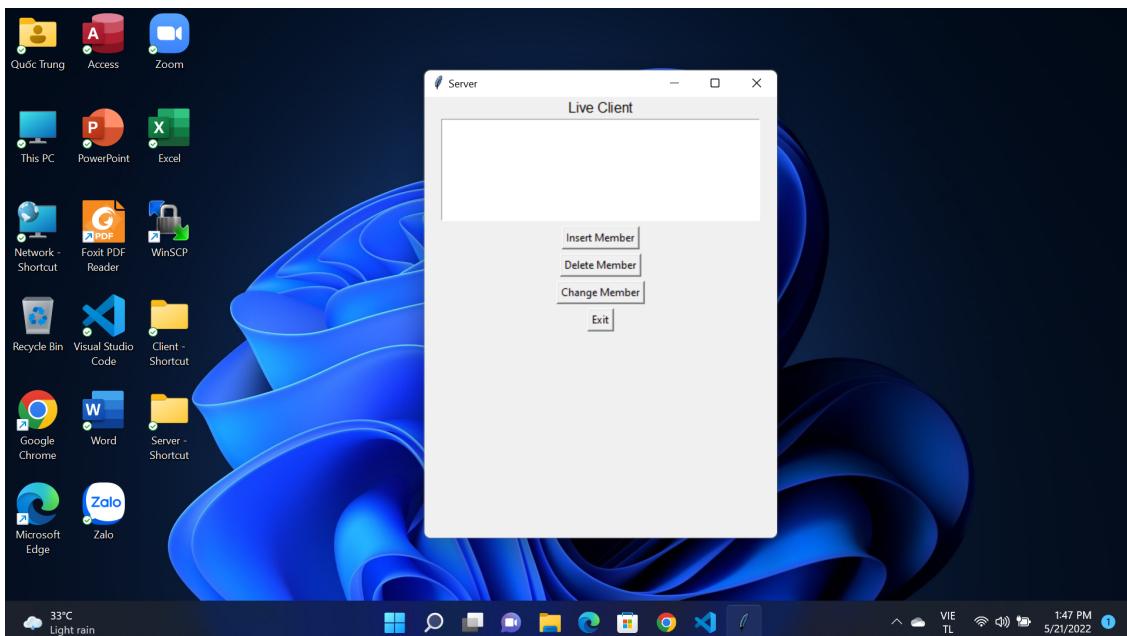
## 4. Môi trường lập trình và các framework hỗ trợ

- Ứng dụng được viết và thực thi trên nền hệ điều hành Window 10, 11
- Ngôn ngữ lập trình được sử dụng là Python (phiên bản sử dụng là Python 3.10)
- Các thư viện hỗ trợ lập trình: socket, json, os, threading, tkinter (GUI Frameworks), PIL, base64, ctypes, faulthandler và turtle.
- Chương trình được viết bằng Python phiên bản 3.10 chỉ hỗ trợ cho window 64-bit nên các hệ điều hành 32-bit thế hệ trước như window 7, windows XP,... sẽ không sử dụng được

## 5. Hướng dẫn sử dụng các tính năng chương trình

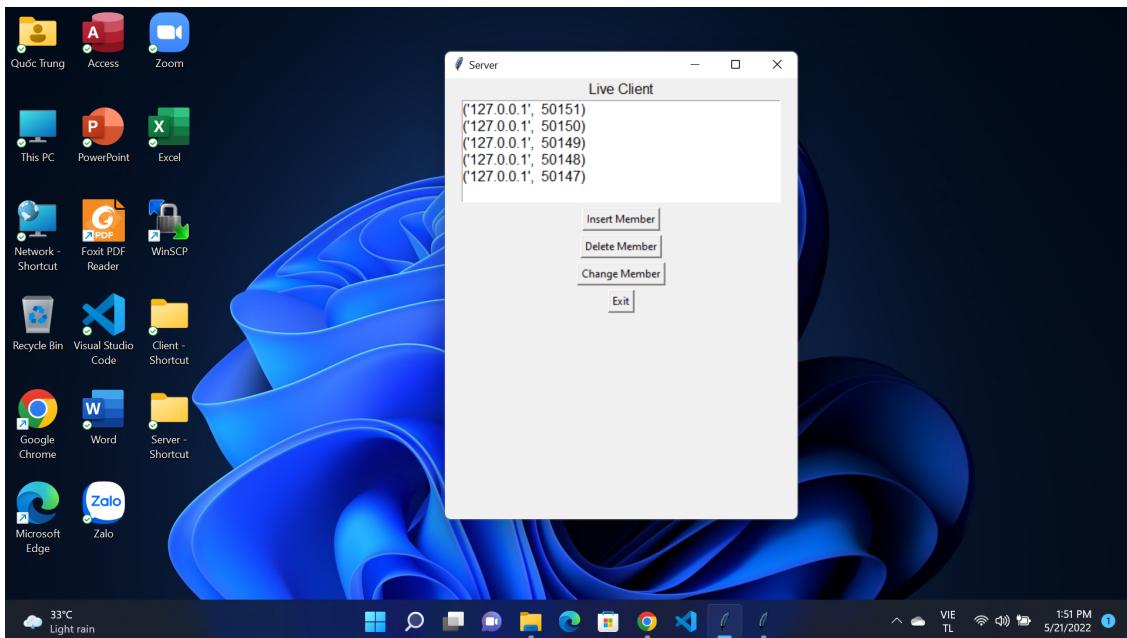
### 5.1. Phía Server

Sau khi khởi chạy file thực thi của Server, màn hình sẽ hiển thị GUI của Server sẽ được hiển thị như hình bên dưới.



Hình 2: Giao diện ban đầu của Server

Tiếp đó, ta khởi chạy file thực thi của Client, thì GUI của Server sẽ hiển thị địa chỉ IP và Port của Client vừa kết nối ở khung *LiveClient*, nếu mở nhiều Client thì bên phía Server sẽ hiển thị địa chỉ IP và Port của tất cả các Client đó, theo thứ tự từ dưới lên, nghĩa là khi có Client mới kết nối đến server thì nó sẽ hiển thị trên đầu danh sách.

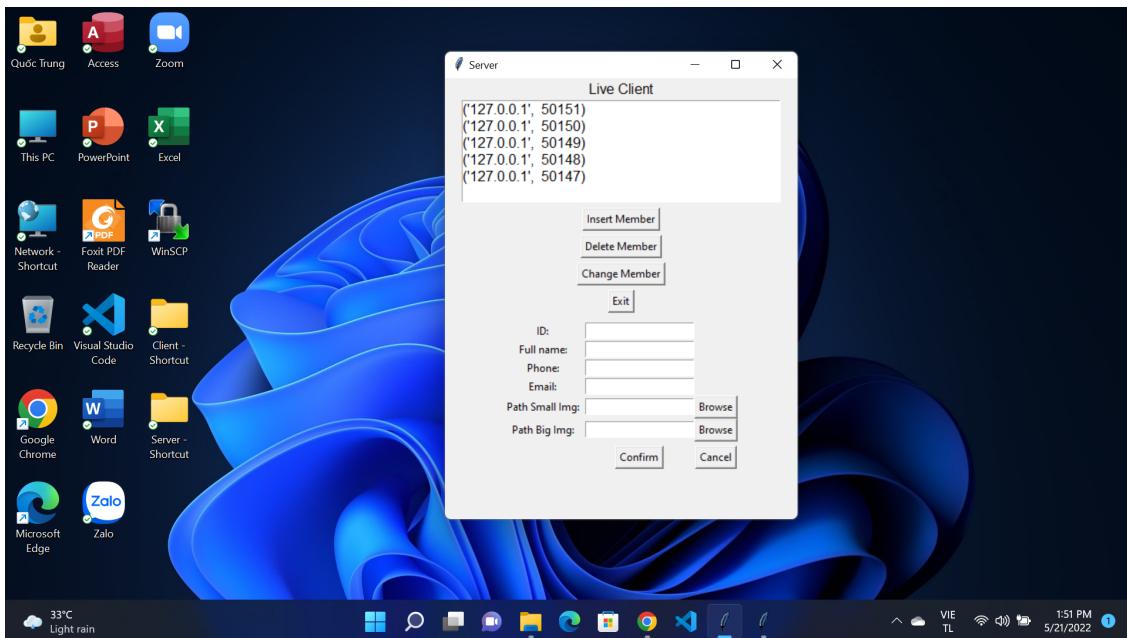


Hình 3: Thông tin của các Client đang kết nối với Server

Ngoài việc hiển thị các Client đã kết nối thì Server còn có các tính năng khác như thêm thành viên vào danh bạ (InsertMember), xóa một thành viên khỏi danh bạ (DeleteMember), Sửa đổi thông tin của một thành viên đã có sẵn trong danh bạ (ChangeMember) và cuối cùng là thoát khỏi chương trình (Exit).

### Thêm thành viên:

Dối với tính năng thêm thành viên vào danh bạ, thì sau khi chọn nút *InsertMember* trên GUI của Server sẽ xuất hiện giao diện thêm thành viên như sau:



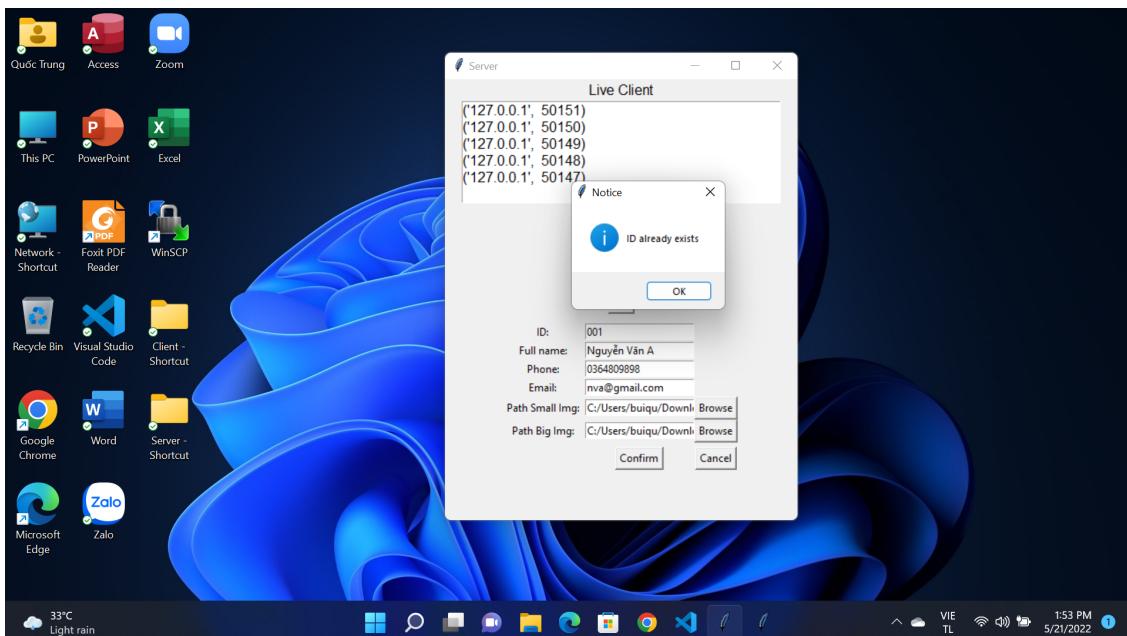
Hình 4: Giao diện thêm thành viên

Về phần này bao gồm các thông tin:

- ID: Là mã thành viên (từ 0 - 8 chữ số). Ví dụ: 001
- Full name: Là họ và tên của thành viên cần thêm. Ví dụ: Nguyễn văn A
- Phone: Số điện thoại của thành viên đó. ví dụ: 0364809898
- Email: là địa chỉ email của thành viên. ví dụ: nva@gmail.com
- Path Small img: là hình đại diện (hình nhỏ) ở đây ta chỉ có thể chọn các file hình có định dạng .jpg và .png
- Path Big img: là hình lớn cách chọn và ràng buộc như chọn hình nhỏ

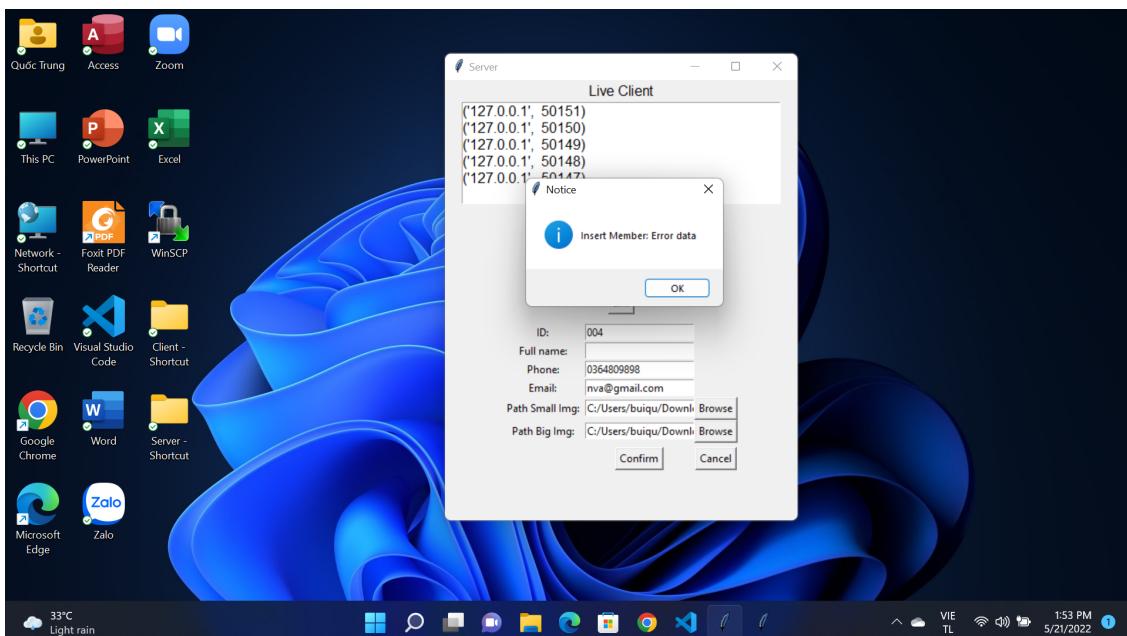
Khi thêm thành viên chúng ta cần **lưu ý**:

- ID của thành viên mới thì không trùng với các thành viên cũ. Ví dụ, ở đây chúng ta thêm thành viên mới có ID 001 bị trùng thì sẽ thông báo đã tồn tại ID.



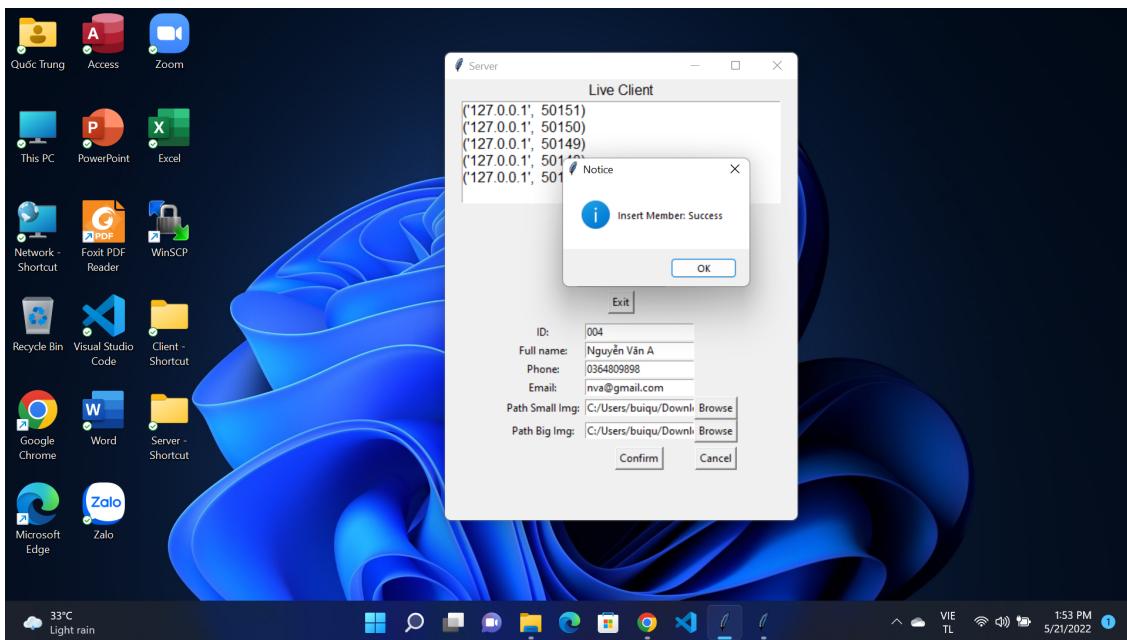
Hình 5: Thông báo ID đã tồn tại

- Phải điền đầy đủ các thông tin của thành viên. Nếu không điền đầy đủ chương trình sẽ thông báo lỗi data sai:



Hình 6: Thông báo sai data

Sau khi điền đúng các thông tin ta cần bấm *Confirm* để thêm thành viên. Ví dụ về thêm thành viên hợp lệ:

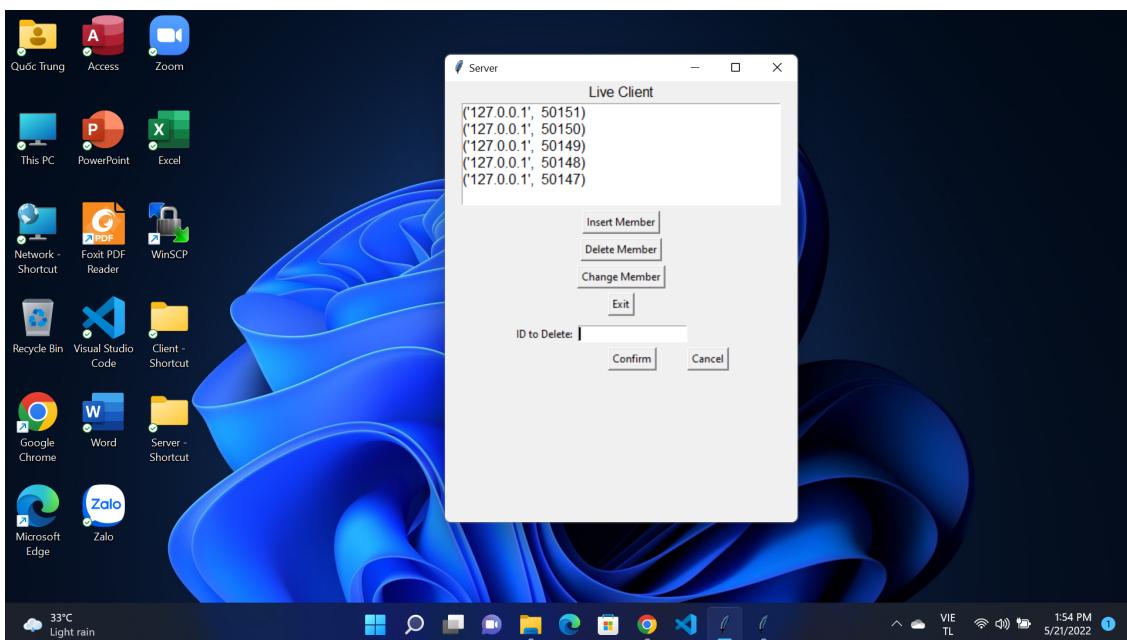


Hình 7: Thông báo thêm thành viên thành công

Còn nếu không muốn thêm thành viên thì ta bấm vào *Cancel* để quay lại giao diện chính của Server.

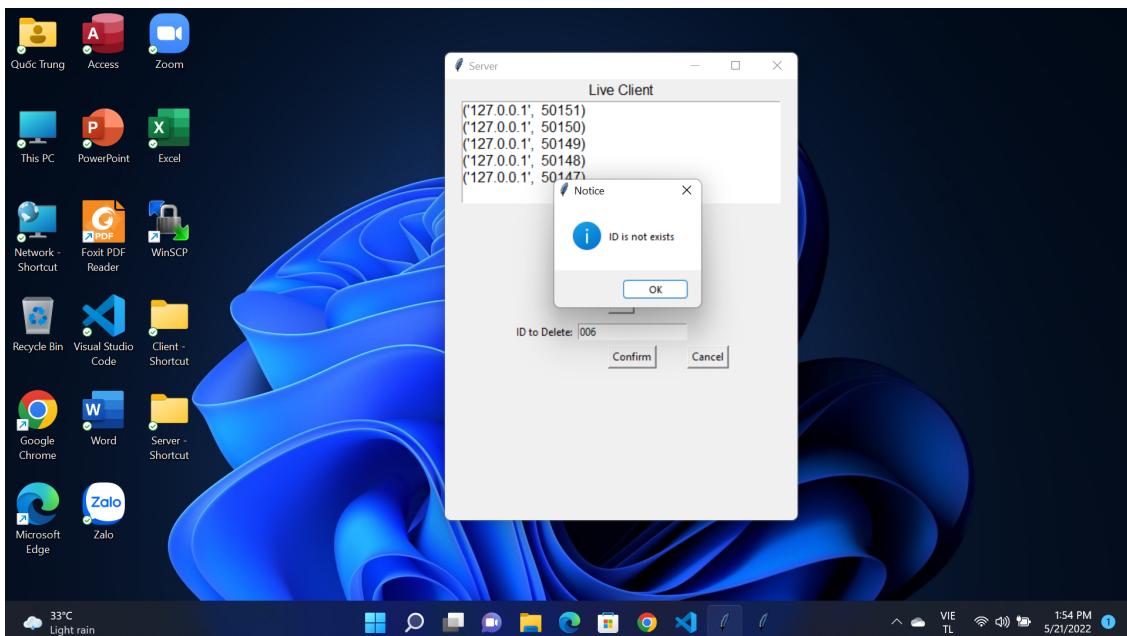
### Xóa thành viên:

Đối với tính năng xóa thành viên trong danh bạ, thì sau khi chọn nút *DeleteMember* trên GUI của Server sẽ xuất hiện giao diện xóa thành viên:



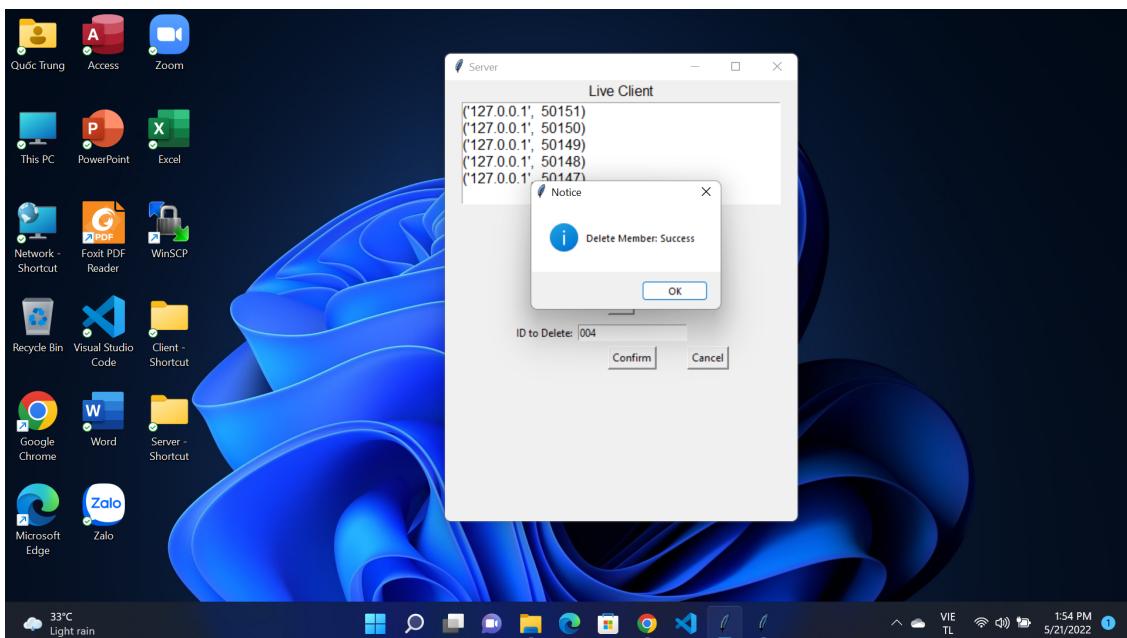
Hình 8: Giao diện xóa thành viên

ID to Delete: là nhập id của thành viên muốn xóa. Để xóa ta nhấn vào nút *Confirm*. Khi không muốn xóa thì ấn vào nút *Cancel* để quay lại giao diện chính của Sever. Ở đây nếu ta nhập ID không đúng (nghĩa là ID này không có trong danh sách thành viên của server) thì chương trình sẽ thông báo.



Hình 9: Thông báo ID nhập không tồn tại

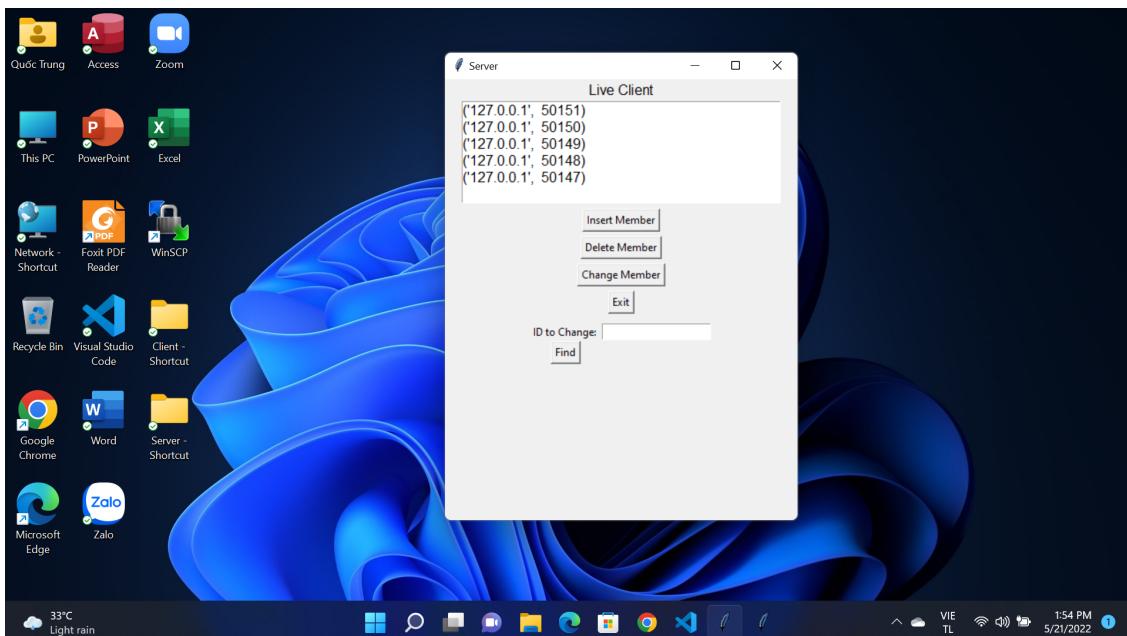
Còn đây là ví dụ về xóa hợp lệ. Chúng ta sẽ xóa thành viên Nguyễn Văn A có id = 004 mà chúng ta vừa thêm ở trên.



Hình 10: Thông báo xóa thành viên thành công

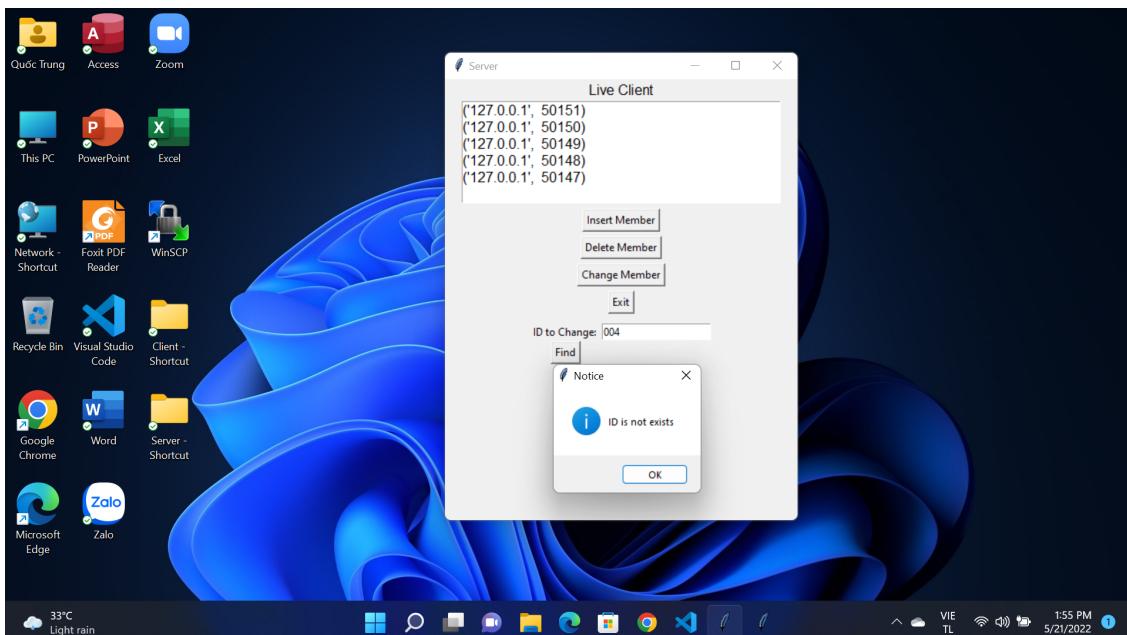
### Sửa thông tin của thành viên

Đối với tính năng sửa thông tin thành viên trong danh bạ, thì sau khi chọn nút *ChangeMember* trên GUI của Server, Đầu tiên sẽ hiện giao diện tìm thành viên cần sửa.



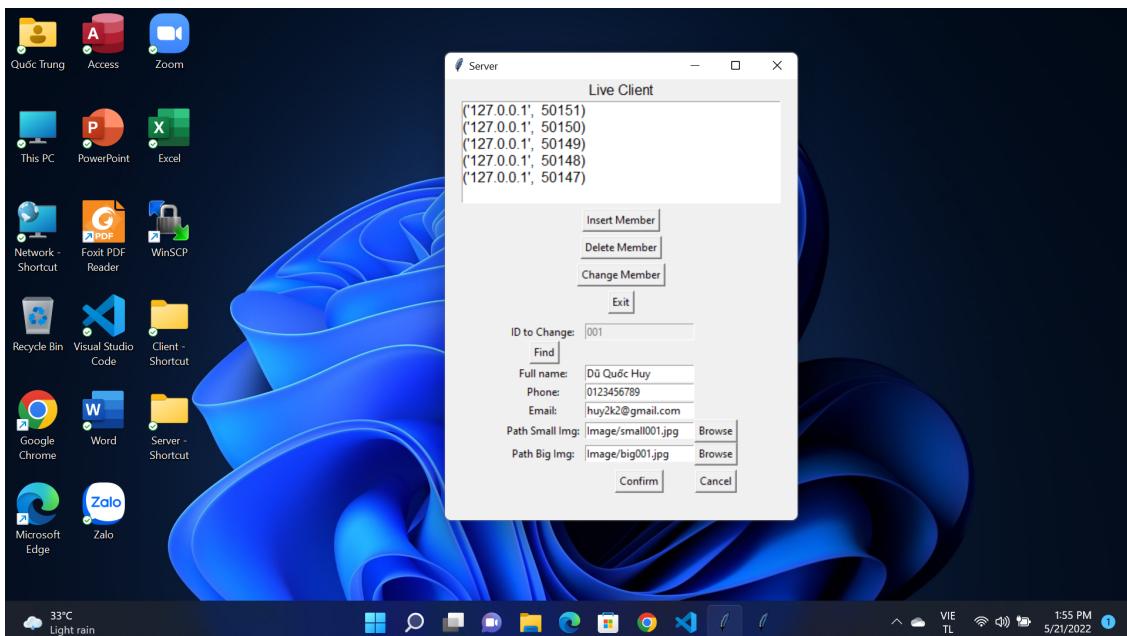
Hình 11: Giao diện tìm thành viên cần sửa

ID to change: là id của thành viên cần sửa. Ta nhập ID vào nhấn nút *Find* để tìm. Nếu ID không có trong danh sách thì sẽ được thông báo. Ví dụ tìm thành viên id = 004 (Đã xóa ở trên)



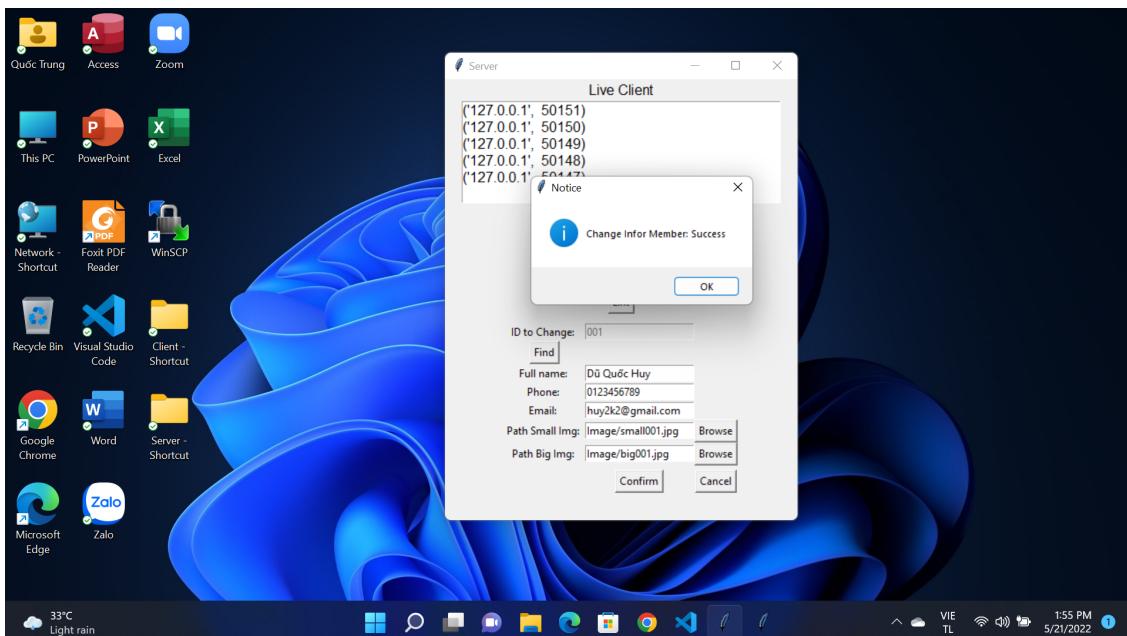
Hình 12: Thông báo ID vừa nhập không tồn tại

Còn nếu ta nhập đúng id thì giao diện các thông tin của thành viên đó để chỉnh sửa.



Hình 13: Hiển thị thông tin của thành viên cần sửa

Sau khi ta chỉnh sửa xong ta nhấn *Confirm* để thay đổi thông tin.

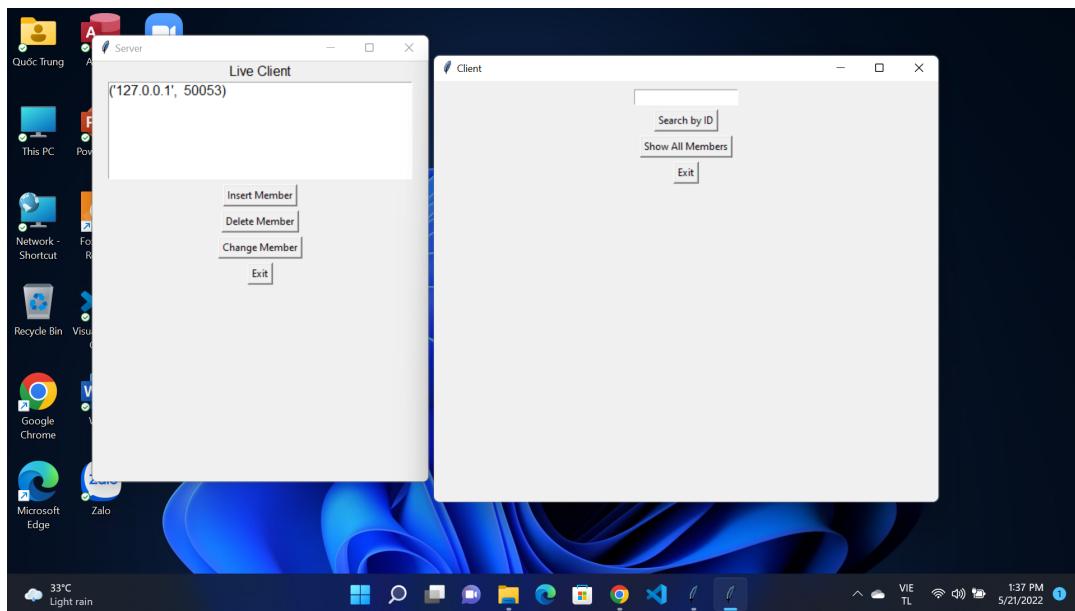


Hình 14: Thông báo sửa thông tin thành viên thành công

Còn nếu bạn không muốn chỉnh sửa thì nhấn nút *Cancel* để quay về giao diện chính của Server.

## 5.2. Phía Client

Về phía Client, sau khi đã tạo kết nối với Server thì GUI của Client sẽ hiển thị như sau:



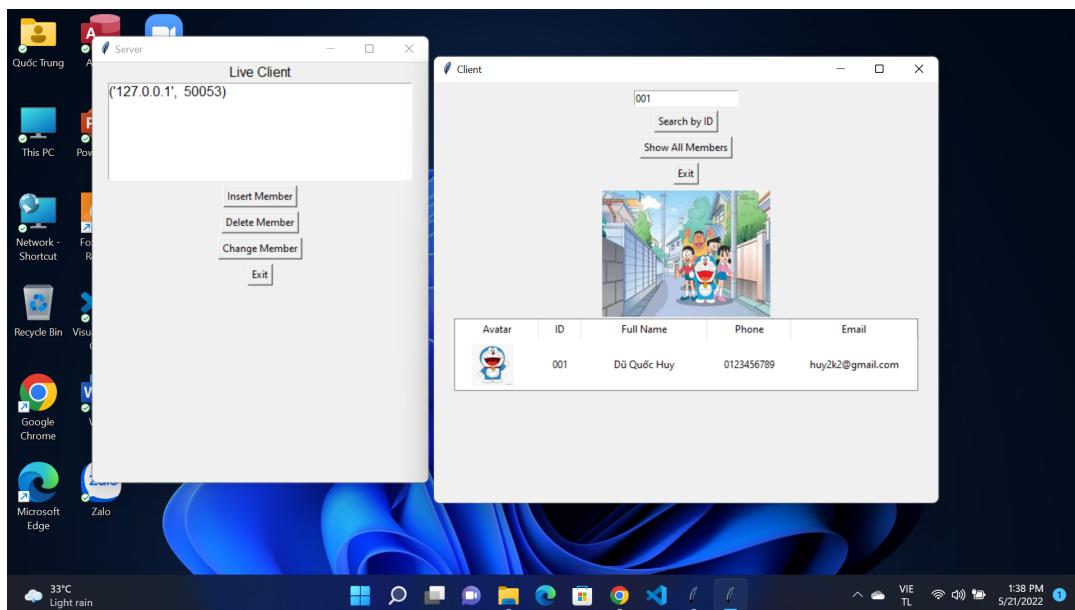
Hình 15: Giao diện ban đầu của Client

Sẽ có 3 tính năng chính ứng với 3 lựa chọn trên GUI của Client, tương ứng là tìm kiếm thông tin một thành viên trong danh bạ của Server (Search by ID), xem thông tin của tất cả các thành viên trong danh bạ của Server (Show all Members) và cuối cùng cũng là ngắt kết nối với Server, thoát khỏi Client (Exit).

### Tìm kiếm một thành viên (Search by ID)

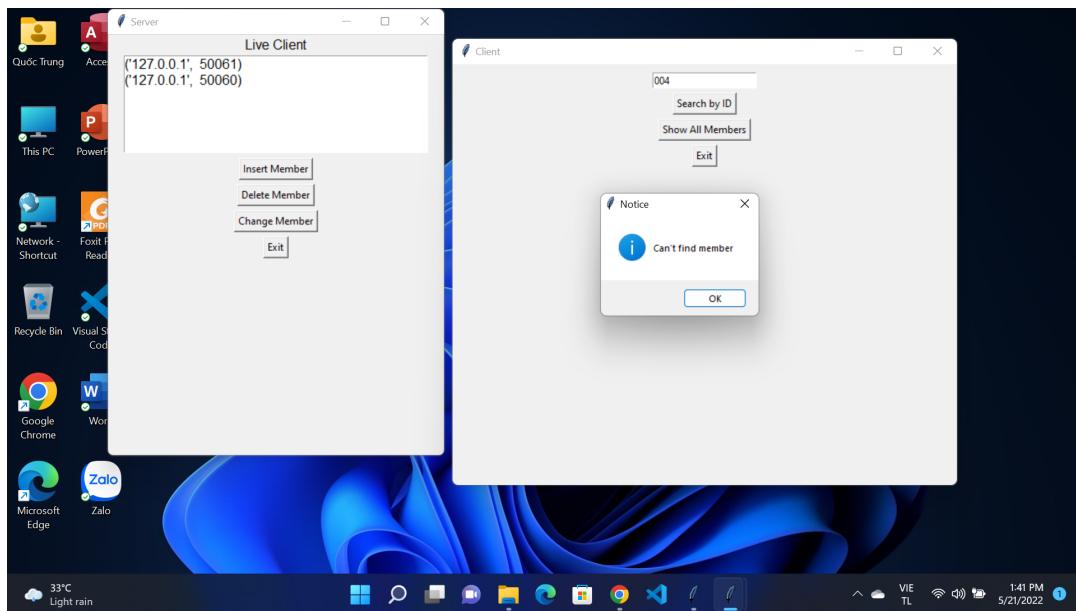
Ở tính năng này, Client sẽ yêu cầu người dùng nhập vào ID của thành viên cần tìm, sẽ có 2 trường hợp xảy ra sau khi người dùng xong ID và ấn vào nút *SearchbyMember*:

- Nếu ID của người dùng nhập vào có trong danh bạ của Server thì thông tin của thành viên có ID đó sẽ hiện ra, bao gồm ảnh đại diện (Avatar), ID, Fullname, Phone, Email và đặc biệt là ảnh Cover (mở rộng chức năng (2))



Hình 16: Thông tin của thành viên được tìm kiếm

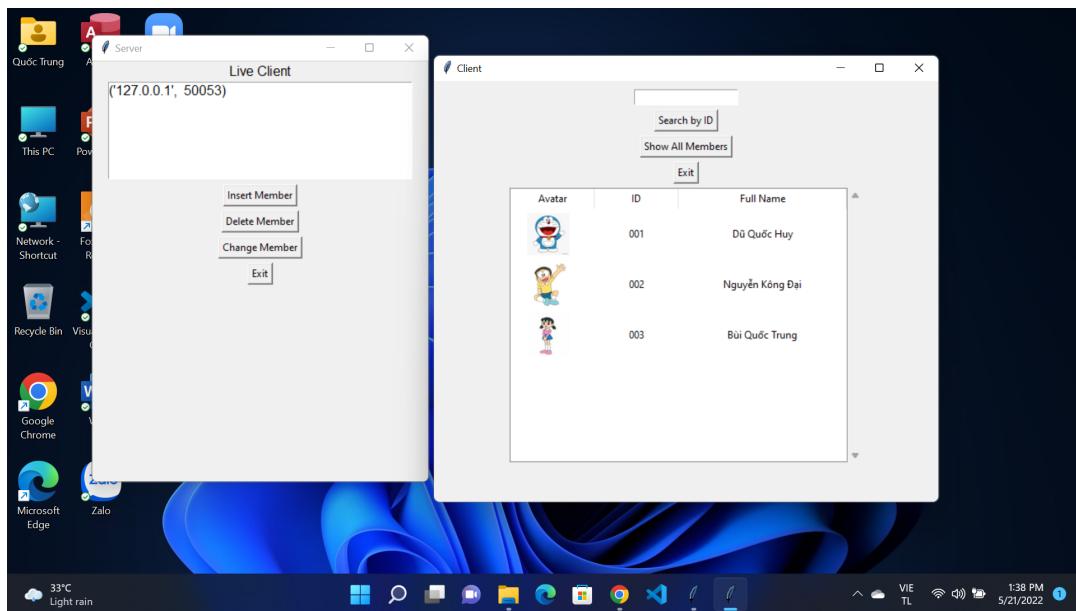
- Nếu ID của người dùng nhập vào không có trong danh bạ của Server, Client sẽ thông báo không thể tìm thấy ID đó



Hình 17: Thông báo không thể tìm thấy ID

### Xem thông tin của tất cả thành viên (Show all Members)

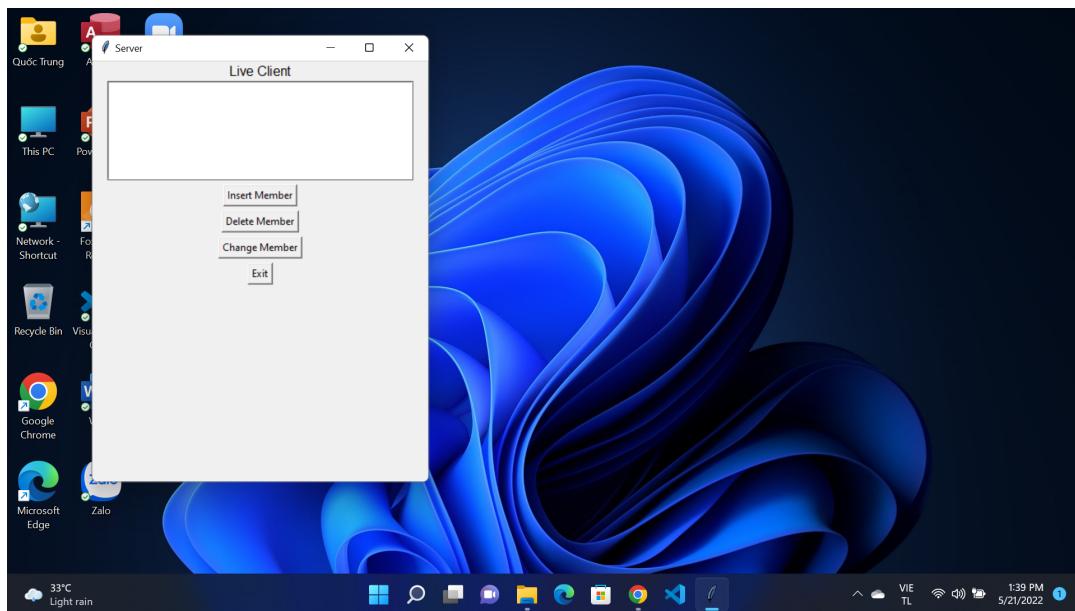
Với tính năng này của Client, khi người dùng ấn vào *ShowallMembers* thì GUI của Client sẽ hiện ra danh sách toàn bộ các thành viên đang nằm trong danh bạ của Server, bao gồm các thông tin là ảnh đại diện (Avatar), ID và Fullname (phần thông tin có cả ảnh đại diện là tính năng mở rộng của chức năng (1)).



Hình 18: Thông tin tất cả thành viên thuộc danh bạ Server

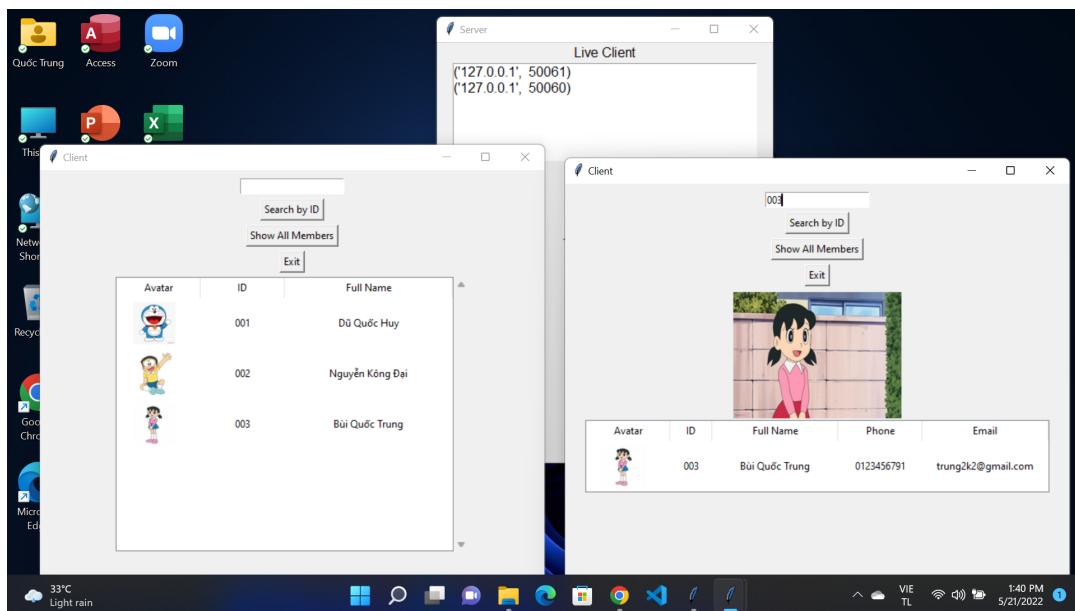
### Ngắt kết nối với Server và thoát khỏi Client (Exit)

Khi không còn muốn thao tác với Server, người dùng có thể chọn Exit để thoát khỏi Client. Khi đó, GUI của Client sẽ tự động đóng lại và thông tin (bao gồm địa chỉ IP và Port) của Client đó sẽ tự động xóa tại khung *LiveClient* của Server.



Hình 19: Thông tin của Client đã thoát không còn ở Server

Ngoài các tính năng đã nêu ở trên, Chương trình còn cho phép nhiều Client truy cập cùng lúc đến Server, các Client này có thể thực hiện các chức năng một cách thoải mái và khi Server cập nhật dữ liệu thì các Client này ngay lập tức sẽ nhận được các cập nhật đó.



Hình 20: Các Client khác nhau cùng truy cập đến Server

## 6. Phân chia công việc

Thành viên	Công việc	Phần trăm đóng góp
Bùi Quốc Trung 20120023	<ul style="list-style-type: none"> <li>• Chương trình trao đổi thông tin của Client và Server</li> <li>• Giao diện hiển thị thông tin chi tiết một thành viên bên Client</li> <li>• Kiểm lỗi cho chương trình và giao diện</li> <li>• Viết báo cáo</li> </ul>	30%
Dũ Quốc Huy 20120101	<ul style="list-style-type: none"> <li>• Thu thập và xử lý dữ liệu</li> <li>• Chương trình trao đổi thông tin của Client và Server</li> <li>• Giao diện hiển thị thông tin toàn bộ thành viên bên Client, giao diện Server</li> <li>• Chức năng thêm và sửa thông tin thành viên của Server</li> <li>• Xử lý đọc và ghi file json khi mở và thoát chương trình Server</li> <li>• Viết báo cáo</li> </ul>	40%
Nguyễn Công Đại 20120448	<ul style="list-style-type: none"> <li>• Giao diện hiển thị và lựa chọn hình ảnh</li> <li>• Chức năng nhiều Client kết nối đến Server</li> <li>• Chức năng xóa thành viên của Server</li> <li>• Viết báo cáo</li> </ul>	30%

Trong quá trình thực hiện, các thành viên có hỗ trợ và giúp đỡ cho nhau để đồ án được hoàn thiện nhất có thể. Cả 3 thành viên cùng tham gia viết báo cáo ứng với phần mình được phân công.

## Tài liệu tham khảo

- [1] Tài liệu, video hướng dẫn, code mẫu của thầy Nguyễn Thanh Quân
- [2] Lập trình socket bằng Python: <https://viblo.asia/p/lap-trinh-socket-bang-python-jvEla084Zkw>
- [3] Hướng dẫn đọc ghi file json với dữ liệu utf8: <https://stackoverflow.com/questions/18337407/saving-utf-8-texts-with-json-dumps-as-utf8-not-as-u-escape-sequence>
- [4] Tkinter Treeview <https://www.pythontutorial.net/tkinter/tkinter-treeview/>
- [5] Hướng dẫn lập trình giao diện GUI bằng Tkinter trong ngôn ngữ Python <https://www.vnitech.com/huong-dan-lap-trinh-giao-dien-gui-bang-tkinter-trong-ngon-ngu-python/>