

## PHẦN II: YÊU CẦU

### ĐỀ 3

#### COVID Việt Nam

#### Nội dung:

Chương trình mô phỏng ứng dụng trực tuyến gồm một server và nhiều client. Server sẽ lưu trữ thông tin COVID giúp client có thể tra cứu. Mọi quá trình client đăng nhập, client đăng xuất, .... đều được thể hiện trên màn hình của server.

Nếu Server lưu trữ COVID của Việt Nam: Thì cho phép nhập tỉnh thành để tra cứu (có thể viết tắt)

Nếu Server lưu trữ COVID của thế giới: Thì cho phép nhập tên nước (có thể viết tắt)



#### Yêu cầu:

| CHỨC NĂNG       | Ý NGHĨA   | MỞ RỘNG  |
|-----------------|---|--|
| KẾT NỐI         | <b>0,5 điểm</b><br>Cho phép client kết nối đến server thông qua kết nối TCP   | <b>0,5 điểm</b><br>Cho phép client và server đặt tại các host khác nhau<br>(cho phép client nhập IP của server để kết nối)   |
| QUẢN LÝ KẾT NỐI | <b>0,5 điểm</b><br>Khi client hoặc server mất kết nối đột ngột, không làm chương trình treo hay xảy ra lỗi  | Nếu một client mất kết nối không làm ảnh hưởng đến các client khác <ul style="list-style-type: none"> <li>Quản lý kết nối đa tiểu trình <b>0,5 điểm</b></li> </ul> |
| ĐĂNG NHẬP       | <b>0,5 điểm</b><br>Client đăng nhập bằng cách gửi username, password cho server<br>Server nhận thông tin username, password từ client và kiểm tra với thông tin đã lưu trữ tại server |  |

|                              |   |   |
|------------------------------|---|---|
| <b>ĐĂNG KÝ</b>               | <p><b>0,5 điểm</b></p> <p>Client đăng ký bằng cách gửi username, password cho server</p> <p>Server nhận thông tin username, password từ client và kiểm tra với thông tin đã lưu trữ tại server, nếu đã tồn tại, gửi thông báo đến client, yêu cầu đăng ký tài khoản khác</p>      |   |
| <b>TRA CỨU</b>               | <p><b>1.5 điểm</b></p> <p>Cho phép Client tra cứu theo ngày với quốc gia (thế giới) hoặc theo tỉnh thành (Việt Nam)</p> <p>Server có thể tự tạo ra dữ liệu mẫu (nếu không làm phần nâng cao)</p>  | <p><b>2 điểm</b></p> <p>Server sẽ kết nối tới một website khác (third party APIs/Web services) để lấy thông tin (JSON hoặc HTML), sau đó rút trích thông tin và lưu trữ liệu dưới Server để phục vụ request của Client.</p> <p>Có thể dùng thư viện để kết nối và gửi các HTTP request đến các 3rd party APIs/Web services này.</p> <p><b>0.5 điểm</b></p> <p>Server cập nhật thông tin liên tục 60 phút 1 lần của ngày hôm đó</p> <p>Ví dụ link:</p> <p><a href="https://github.com/javieraviles/covidAPI">https://github.com/javieraviles/covidAPI</a></p> <p><a href="https://vi.wikipedia.org/wiki/B%E1%BA%A3n_m%E1%BA%ABu:D%E1%BB%AF_l%E1%BB%87u_%C4%91%E1%BA%A1i_d%E1%BB%8Bch_COVID-19/S%E1%BB%91_ca_nhi%E1%BB%85m_theo_t%E1%BB%89nh_th%C3%A0nh_t%E1%BA%A1i_Vi%E1%BB%87t_Nam#cite_note-1">https://vi.wikipedia.org/wiki/B%E1%BA%A3n_m%E1%BA%ABu:D%E1%BB%AF_l%E1%BB%87u_%C4%91%E1%BA%A1i_d%E1%BB%8Bch_COVID-19/S%E1%BB%91_ca_nhi%E1%BB%85m_theo_t%E1%BB%89nh_th%C3%A0nh_t%E1%BA%A1i_Vi%E1%BB%87t_Nam#cite_note-1</a></p> |
| <b>QUẢN LÝ CƠ SỞ DỮ LIỆU</b> | <p>Điểm max cho phần này là 1 điểm</p> <ul style="list-style-type: none"> <li>Sử dụng dữ liệu lưu trữ tĩnh trong code <b>0 điểm</b></li> <li>Nếu sử dụng dữ liệu lưu trữ trong file <b>0,5 điểm</b></li> <li>Sử dụng file cấu trúc: xml, json, sql, .... <b>1 điểm</b></li> </ul> |   |
| <b>THOÁT</b>                 | <p><b>0,5 điểm</b></p> <ul style="list-style-type: none"> <li>Client được phép gửi thông báo ngừng kết nối đến server</li> <li>Server có thể gửi thông báo ngừng kết nối đến tất cả client đang hoạt động</li> </ul>  |   |
| <b>GIAO DIỆN</b>             | <p>Điểm max cho phần này là 1.5 điểm</p> <p>Console Application <b>0.5 điểm</b></p>   | <p>Có thiết kế giao diện đồ họa cho chương trình (GUI)</p> <ul style="list-style-type: none"> <li>Client <b>0,5 điểm</b></li> <li>Server <b>0,5 điểm</b></li> </ul> <p>Có cả Client - Server: 1.5 điểm</p>  |

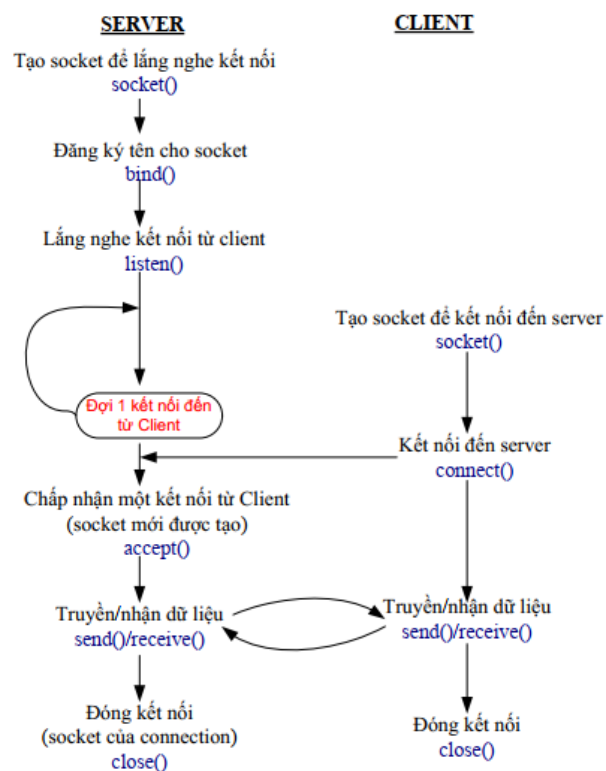
## PHẦN III: GIỚI THIỆU ĐỒ ÁN

### I. TỔNG QUAN CHƯƠNG TRÌNH

#### 1. CÔNG CỤ HỖ TRỢ

|                    |                  |
|--------------------|------------------|
| Ngôn ngữ lập trình | Python 3         |
| Framework          | Python socket    |
|                    | Python Threading |
|                    | Pyqt5            |

#### 2. MÔ HÌNH SOCKET CLIENT- SERVER



Hình 1: Sơ đồ tương tác giữa server-client theo giao thức TCP

Phân thành 3 giai đoạn như sau:

Giai đoạn 1: Server tạo Socket – gán số hiệu cổng.

- `socket.socket(socket.AF_INET, socket.SOCK_STREAM)`: Server yêu cầu tạo một socket để có thể sử dụng các dịch vụ của tầng vận chuyển, giao thức mặc định được chọn là TCP (`socket.SOCK_STREAM`).
- `bind(ADDR)`: Server yêu cầu gán số hiệu cổng cho socket.
- `Listen()`: Server lắng nghe những yêu cầu kết nối từ client.

Giai đoạn 2: Client tạo Socket.

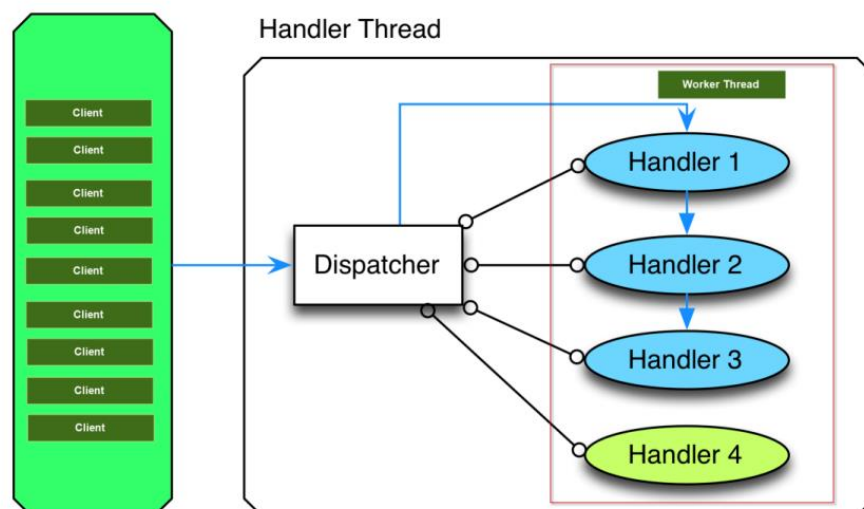
- `socket.socket(socket.AF_INET, socket.SOCK_STREAM)`: Client yêu cầu tạo một socket để có thể sử dụng các dịch vụ của tầng vận chuyển, giao thức mặc định được chọn là TCP, thông thường hệ thống tự động gán một số hiệu cổng còn rảnh cho socket của Client.

Giai đoạn 3: Trao đổi thông tin giữa Client và Server.

- Sau khi tạo Socket xong, client kết nối đến server qua hàm `.connect(SVaddr)`.
- Client và Server có thể trao đổi thông tin qua lại với nhau thông qua hai hàm `send()` và `receive()`.
- Đơn vị dữ liệu trao đổi giữa Client và Server là các stream of bytes liên tục.
- Mỗi gói tin có chứa thông tin về địa chỉ người gửi và người nhận (IP, Port).

### a. SERVER

Đề án xây dựng mô hình server phục vụ song song:



Hình 2: Multiple clients

Server chia thành 3 phần thực hiện song song nhau:

Phần 1: Xử lý các yêu cầu nối kết.

Lập lại các công việc sau:

- Lắng nghe yêu cầu nối kết của Client và chấp nhận yêu cầu kết nối.
- Tạo luồng phục vụ mới khi kết nối với một client mới.

Phần 2 (Worker Thread) : Xử lý các thông điệp yêu cầu từ khách hàng.

Lập lại các công việc sau:

- Chờ nhận thông điệp yêu cầu của client.
- Phân tích và xử lý yêu cầu.
- Gửi thông điệp trả lời cho client.

Phần 2 sẽ kết thúc khi server chấm dứt kết nối với client.

Phần 3 tự động cập nhật server.

Cứ mỗi 60 phút thì server sẽ gửi request GET tới api có url: <https://api.apify.com/v2/key-value-stores/EaCBL1JNtjR3EakU/records/LATEST?disableRedirect=true>.

database của server sẽ được cập nhật dữ liệu mới mỗi 60 phút.

Với mỗi Client, trên Server sẽ có một Phần 2 để xử lý yêu cầu của khách hàng. Như vậy tại một thời điểm bất kỳ luôn tồn tại 1 Phần 1, 1 phần 3 và 0 hoặc nhiều Phần 2.

## ***b. CLIENT***

Client xử lý socket:

- Yêu cầu kết nối đến server.
- Thực hiện gửi yêu cầu và nhận hồi đáp từ server.
- Xử lý dữ liệu.

## **II.GIAO TIẾP CLIENT-SERVER.**

Giao thức trao đổi giữa server và client: TCP (Transmission Control Protocol).

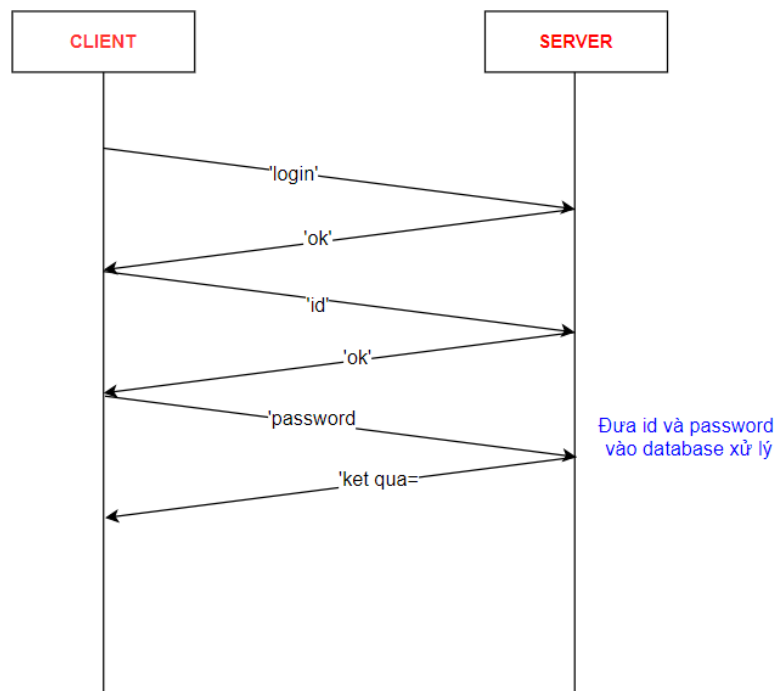
Cấu trúc thông điệp:

Muốn truy vấn 1 yêu cầu đầu tiên client gửi cho server loại truy vấn muốn thực hiện : 'login', 'signup', 'search'....

Tiếp theo sau khi nhận được phản hồi từ server, client tiếp tục gửi những thông tin thêm để server xử lý.

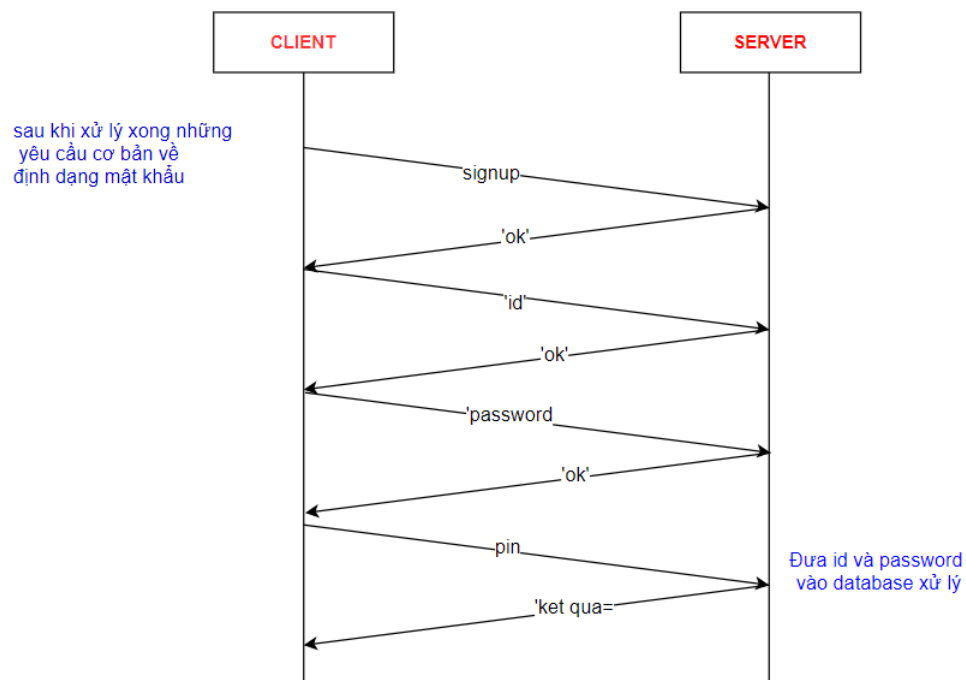
Cuối cùng server trả về kết quả sau khi xử lý cho client.

Đăng nhập



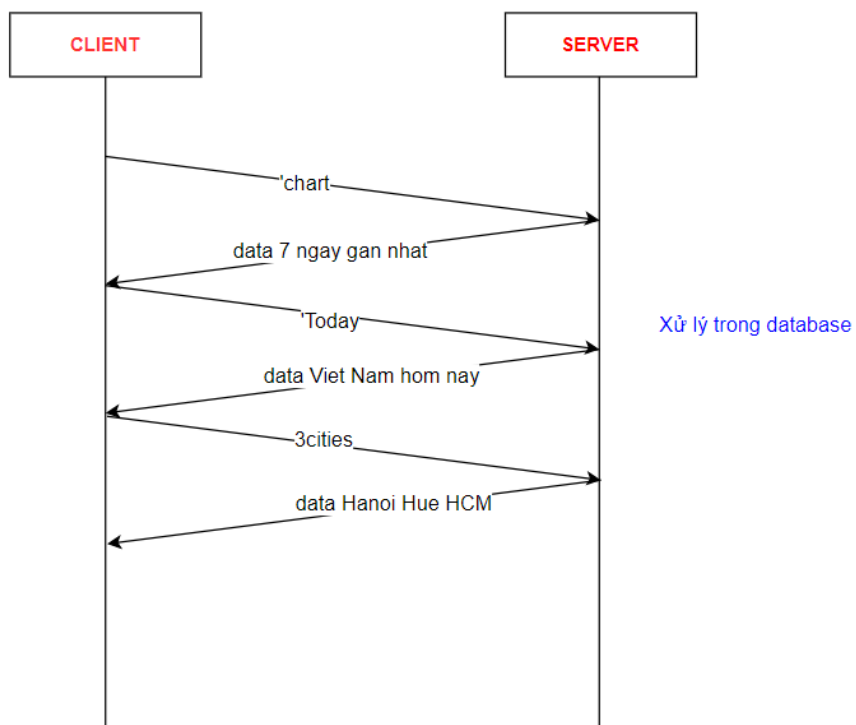
Hình 3: Đăng nhập

Đăng kí tài khoản



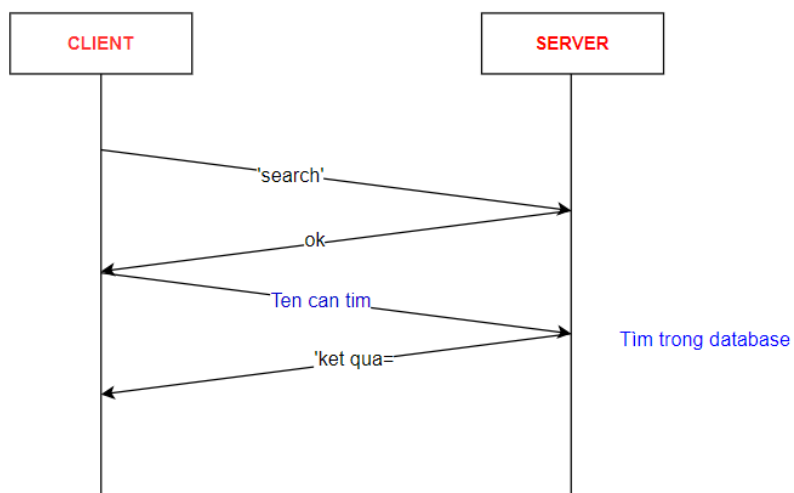
Hình 4: Đăng kí tài khoản

Cập nhật client GUI



Hình 5: Cập nhật UI

Tìm kiếm



Hình 6: Tìm kiếm

### III. QUẢN LÝ CƠ SỞ DỮ LIỆU.

Cơ sở dữ liệu được lưu trữ bằng các file cấu trúc json:

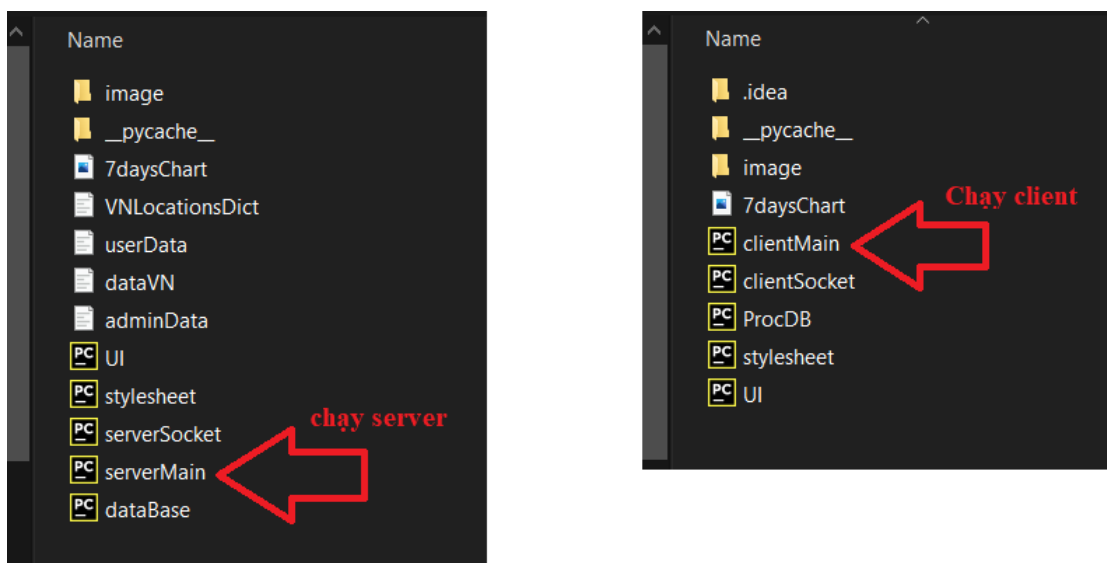
- adminData.json: lưu trữ các tài khoản admin server.
- userData.json: lưu trữ các thông tin người dùng.
- VNLocations.json: lưu trữ tên các tỉnh/thành phố ở Việt Nam và các loại từ khoá để tìm kiếm của nó.
- dataVN.json: lưu trữ dữ liệu chính của chương trình, gồm thông tin về covid-19 ở Việt Nam.

Tất cả cơ sở dữ liệu đều được quản lý ở server, client chỉ nhận được data nếu server gửi cho.

### PHẦN IV: HƯỚNG DẪN SỬ DỤNG IN4 COVID-19

Vì chương trình viết bằng python không yêu cầu release file \*.exe nên để chạy chương trình người dùng phải cài đặt ngôn ngữ python 3 và các framework, library cần thiết.

- Để chạy server vào folder server và chạy file serverMain.py.
- Để chạy client vào folder client và chạy file clientMain.py.



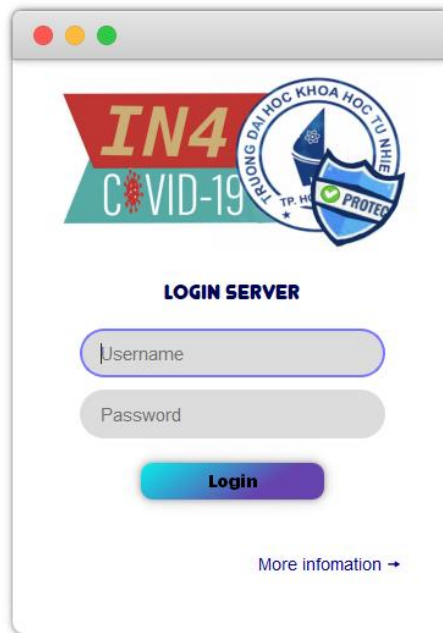
Hình 7: Hướng dẫn chạy file chương trình

#### I. Server

Khởi động server người dùng được yêu cầu phải đăng nhập username và password.

Username và password đăng nhập ở server được cài đặt do người lập trình viên server, tài khoản không được đăng kí thủ công hay có quyền lấy lại mật khẩu.





Hình 8: Giao diện đăng nhập server

Khi đã đăng nhập vào được server giao diện sẽ hiển thị:

- Thời gian cập nhật gần nhất của data base.
- Số client đang kết nối đến server.
- Địa chỉ của những client đang kết nối.



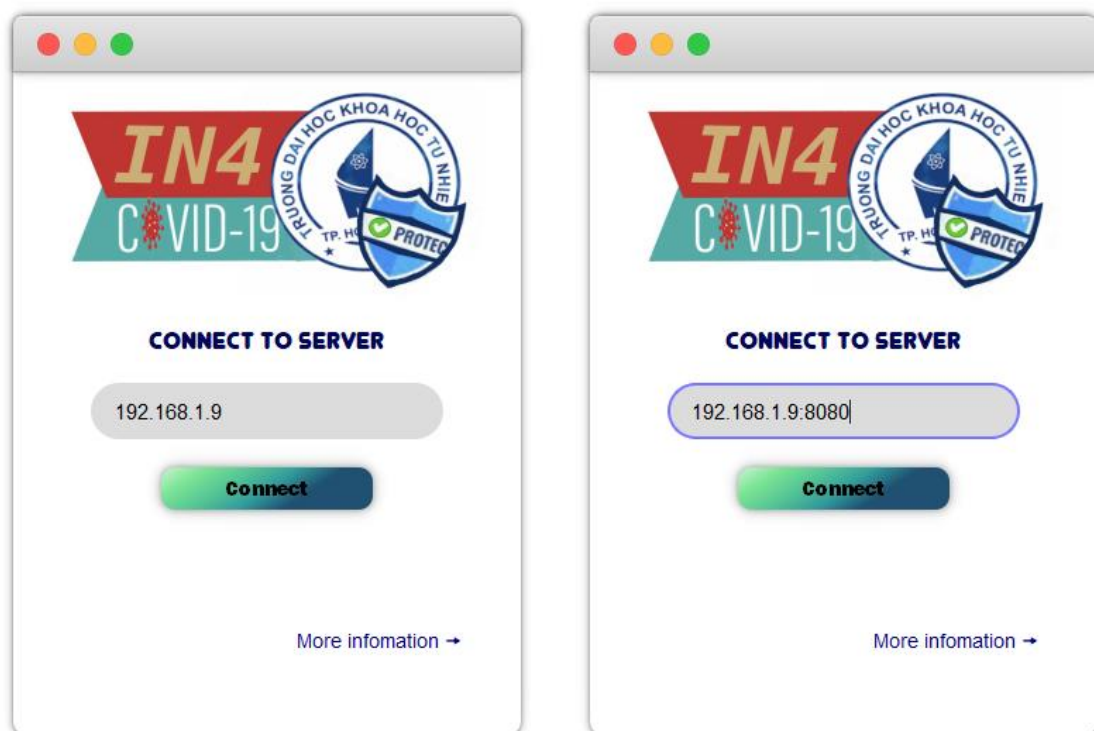
Hình 9: Giao diện khi chạy server

Người thao tác có thể thực hiện 3 chức năng là:

- Cập nhật dữ liệu ngay lập tức (update server).
- Ngắt kết nối với tất cả client (disconnect all)
- Tắt server và giao diện (exit)

## **II. Client**

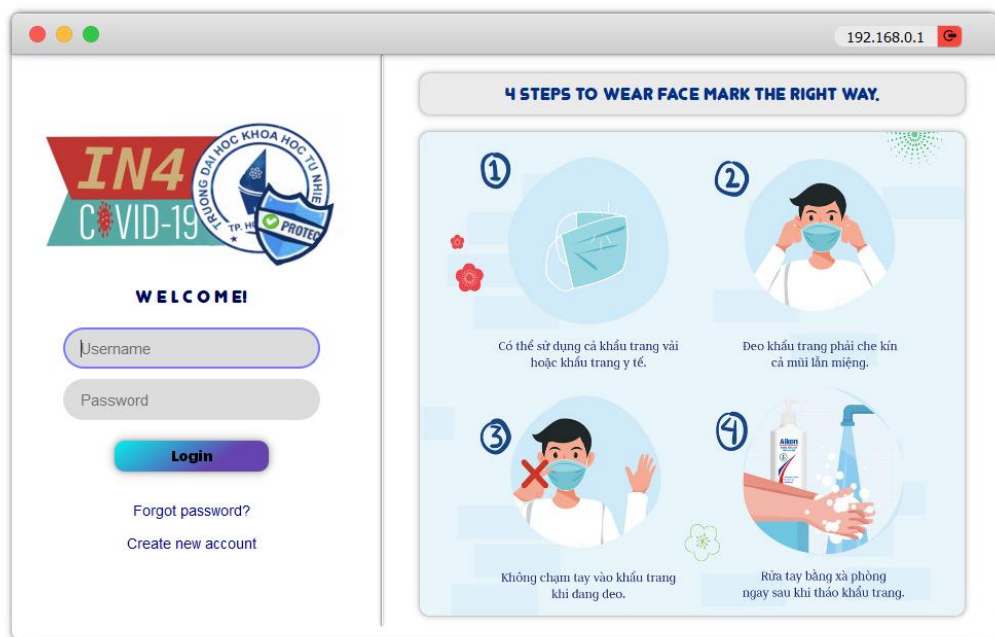
Để kết nối tới server người dùng cần nhập đúng địa chỉ của server (IP và Port) nếu không nhập port chương trình sẽ dùng port mặc định 8888, nếu nhập không đúng hoặc không thể kết nối sẽ hiện cảnh báo không thể kết nối đến server.



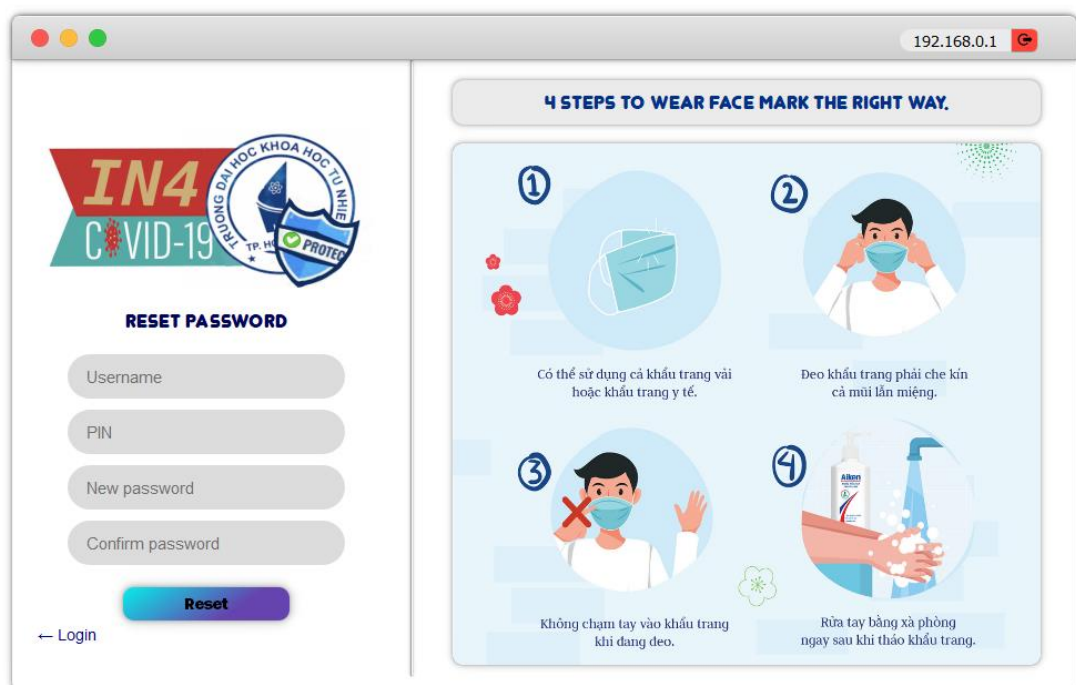
*Hình 10: Giao diện kết nối client đến server*

Sau khi kết nối được tới server người dùng cần nhập đúng username và password để đăng nhập vào chương trình tra cứu IN4 COVID-19.

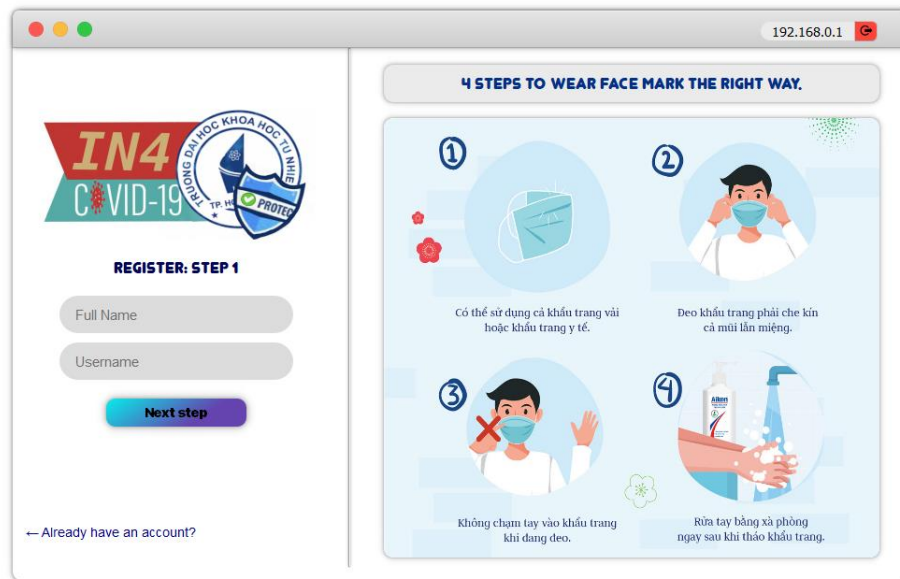
Người dùng có thể chọn đăng ký tài khoản mới (Creat new account) hoặc lấy lại mật khẩu (Forgot password) để xử lý tài khoản đăng nhập.



Hình 11: Giao diện đăng nhập client



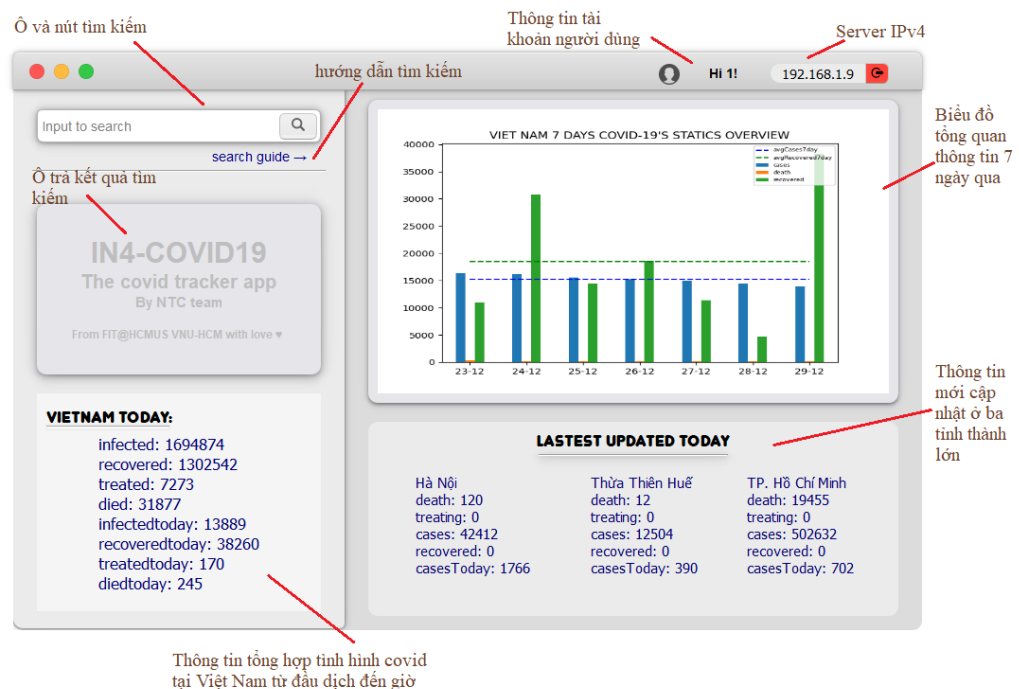
Hình 12: Giao diện đặt lại mật khẩu



Hình 13: Giao diện đăng kí tài khoản

Sau khi đăng nhập thành công vào giao diện chính của ứng dụng người dùng sẽ thấy các khu vực các thông tin về covid-19:

- Thông tin các thống kê trong ngày tại Việt Nam.
- Biểu đồ các ca nhiễm, ca phục hồi và ca tử vong trong vòng 7 ngày qua.
- Thông tin thống kê mới nhất của 3 tỉnh thành lớn : Hà Nội, TTHuế, TP Hồ Chí Minh.



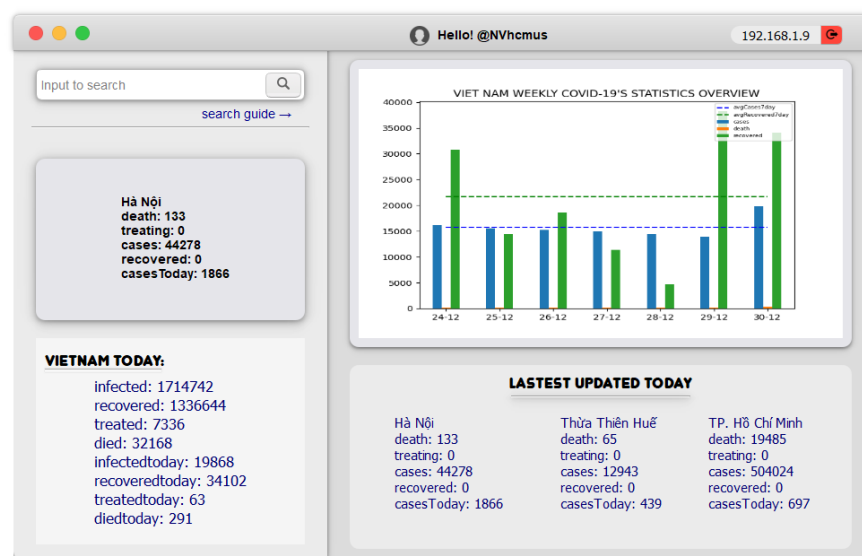
Hình 14: Giao diện chính của ứng dụng

Khu vực thao tác chính là ô tìm kiếm. Để tra cứu thông tin về một tỉnh/thành phố người dùng có thể nhập tên tỉnh/thành phố có dấu hoặc không dấu, viết tắt theo qui ước...

Ví dụ:

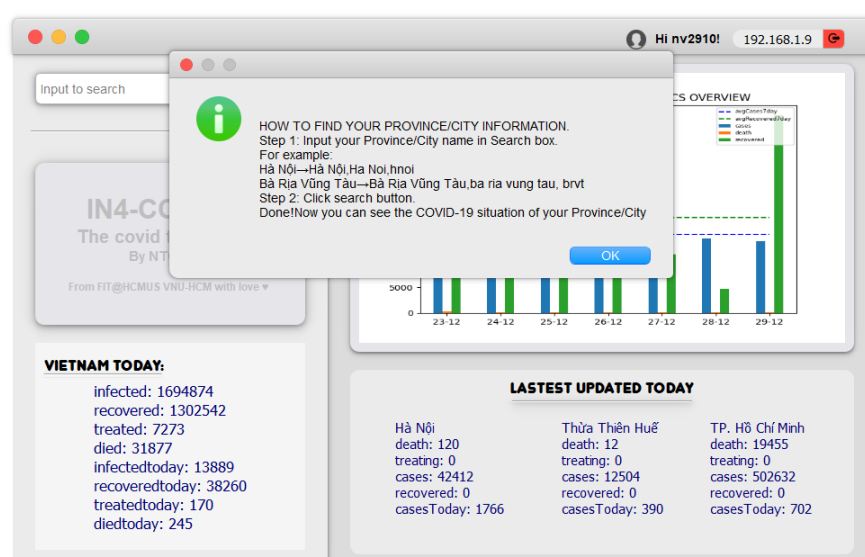
Muốn tìm thông tin về “Bà Rịa Vũng Tàu” người dùng có thể nhập : “Bà Rịa Vũng Tàu”, “ba ria vung tau”, “Ba Ria Vung Tau”, “BRVT”, “brvt”, “ba ria”, “vung tau”...

Muốn tìm thông tin về “Hà Nội” người dùng có thể nhập: “Hà Nội”, “Ha Noi”, “ha noi”, “hnoi”...



Hình 15: Giao diện trả về kết quả tìm kiếm

Người dùng có thể nhấn vào *search guide* để xem hướng dẫn tìm kiếm.



Hình 16: Giao diện hướng dẫn tìm kiếm

---

## ***PHẦN V: TÀI LIỆU THAM KHẢO***

---

<https://www.tutorialspoint.com/pyqt5/index.htm>

<https://docs.python.org/3/library/socket.html>

<https://docs.python.org/3/library/threading.html>

<https://realpython.com/python-sockets/>