**TRƯỜNG ĐẠI HỌC THĂNG LONG**

**BỘ MÔN TIN HỌC**

**---o0o---**

**BÀI TẬP LỚN**

**LẬP TRÌNH PYTHON**

**SỐ LIỆU COVID – 19 TRÊN TOÀN THẾ GIỚI**

**SINH VIÊN THỰC HIỆN:** NGUYỄN QUYẾT TIẾN – A35884

TRẦN MINH HẢI – A36154

**CHUYÊN NGÀNH:** CÔNG NGHỆ THÔNG TIN

**GIÁO VIÊN HƯỚNG DẪN:** HỒ HỒNG TRƯỜNG

HÀ NỘI – 2023

# Tổng quan đề tài

## Giới thiệu đề tài

Từ Tổ chức Y tế Thế giới - Vào ngày 31 tháng 12 năm 2019, WHO đã được cảnh báo về một số trường hợp viêm phổi tại thành phố Vũ Hán, tỉnh Hồ Bắc của Trung Quốc. Loại vi-rút này không giông với bất kỳ loại vi-rút nào khác đã biết. Điều này gây lo ngại vì khi một loại vi-rút mới xuất hiện, chúng ta không biết nó ảnh hưởng đến con người   
như thế nào.

2019 Novel Coronavirus (2019-nCoV) là một loại vi-rút (cụ thể hơn là một loại vi-rút corona) được xác định là nguyên nhân gây ra đợt bùng phát bệnh hô hấp được phát hiện lần đầu tiên ở Vũ Hán, Trung Quốc. Ngay từ đầu, nhiều bệnh nhân trong đợt bùng phát ở Vũ Hán, Trung Quốc được cho là có mối liên hệ nào đó với một chợ hải sản và động vật lớn, cho thấy sự lây lan từ động vật sang người. Tuy nhiên, ngày càng có nhiều bệnh nhân được báo cáo là không tiếp xúc với chợ động vật, cho thấy sự lây lan từ người sang người đang xảy ra. Tại thời điểm này, vẫn chưa rõ mức độ lây lan dễ dàng và bền vững của loại vi-rút này giữa người với người - CDC

Bộ dữ liệu này có thông tin cấp độ hàng ngày về số trường hợp bị ảnh hưởng, tử vong và phục hồi từ vi-rút corona mới năm 2019. Xin lưu ý rằng đây là dữ liệu chuỗi thời gian và do đó, số trường hợp vào bất kỳ ngày cụ thể nào là số tích lũy.

## Thành viên tham gia

* NGUYỄN QUYẾT TIẾN – A35884 – Thành viên nhóm
* Công việc: Thống kê số liệu người chết và bị mắc COVID – 19.
* TRẦN MINH HẢI – A36154 – Trưởng nhóm
* Công việc: Thống kê số liệu người khỏi bệnh COVID – 19, Word, PowerPoint.

# Tài liệu mô tả các endpoint

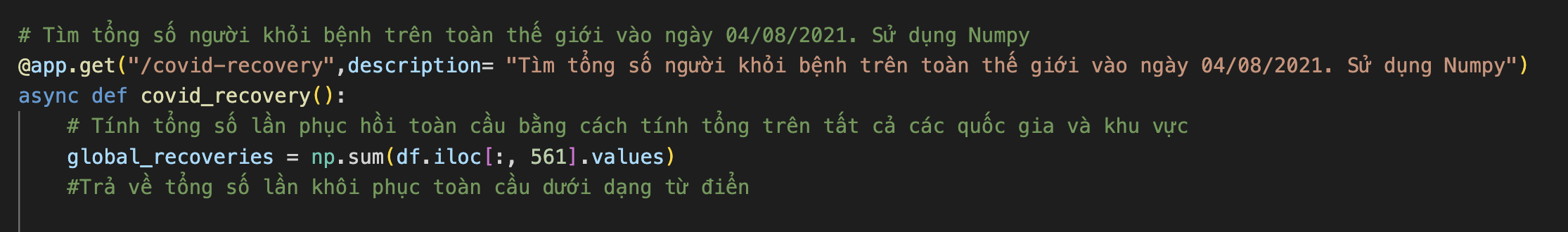
## Bài code của Trần Minh Hải – A36154 (Số ca phục hồi)

Text

Description automatically generated

Các thư viện sử dụng: FastAPI, Query, Pandas, Numpy, json.

### Endpoint về Numpy



* Dùng phương thức get để lấy dữ liệu của endpoint “/covid-recovery”.
* Sử dụng Pandas để đọc dữ liệu số ca đã khỏi bệnh
* Khai báo biến “global\_recoveries” để tìm ra số tổng số người đã khỏi bệnh vào ngày 04/08/2021 vì dữ liệu của các ca đã khỏi bệnh hiện tại mới cập nhật đến 04/08/2021
* Trả về giá trị: {"global\_recoveries": int(global\_recoveries)}

Kết quả:

Graphical user interface, text, application, email

Description automatically generated

### Endpoint về Get

Text

Description automatically generated

* Sử dụng thư viện pandas để đọc file “data.csv” để đọc file dữ liệu về ca bệnh đã khỏi bởi Covid – 19.

Text

Description automatically generated

* Sử dụng phương thức get với endpoint là “/filter\_recovery\_data”.
* Khai biến hàm “filter\_recovery\_data” với tham số truyền vào là “country” được truy vấn.
* Khai báo “filtered\_df” là bảng dữ liệu được truy vấn theo tham số “country” trong dữ liệu “data.csv”.
* Khai báo “data” để chuyển “filtered\_df” sang kiểu “dict”.
* Và trả về dữ liệu “data”.

Kết quả:

Graphical user interface, text, application, email

Description automatically generated

### Endpoint về Pandas

#### Tìm quốc gia và tổng số ca đã khỏi bệnh backup

Text

Description automatically generated

* Tạo DataFrame mới với cột “Country” và cột thứ 561 vì dữ liệu của các ca khỏi bệnh được cập nhật đến ngày 04/08/2021.
* Khai báo biến “data\_dict\_backup” để chuyển dữ liểu thành kiểu “dict”.

Text

Description automatically generated

* Sử dụng phương thức get với endpoint là “filter\_recovery\_total\_data\_backup”.
* Khai báo hàm filter\_recovery\_total\_data\_backup với tham số là “country”.
* Chạy vòng lặp lần lượt từng dữ liệu trong “data\_dict\_backup” khi nào giá trị “Country” bằng với giá trị tham số truyền vào thì trả về {"message": f"{country} có số ca khỏi bệnh do Covid 19 là {str(item['Total'])}"}.

Kết quả:

Graphical user interface, text, application, email

Description automatically generated

#### Tìm quốc gia với dữ liệu cập nhật bởi file json

Text

Description automatically generated

* Sử dụng phương thức get và endpoint “filter\_recovery\_total\_dataJSON”.
* Khai báo hàm “filter\_recovery\_total\_dataJSON” với tham số truyền vào là country.
* Cho chạy vòng lặp trên file “data\_dict”.
* Khi nào chạy đến giá trị Country bằng tham số truyền vào thì in ra kết quả {"message": f"{country} có số ca khỏi bệnh do Covid 19 là {str(item['Total'])}"}.

Kết quả:

Graphical user interface, text, application, email

Description automatically generated

Text

Description automatically generated

### Endpoint sử dụng Post

Text

Description automatically generated

* Sử dụng phương thức post và endpoint “update\_total” để cập nhật số ca khỏi bệnh do COVID – 19.
* Khởi tạo hàm “update\_total” với hai tham số truyền vào là “country” và “new\_total”.
* Chạy vòng lặp trong data\_dict. Nếu Country chạy đến giá trị tham số “country” thì giá trị Total sẽ là giá trị tham số “new\_total”.
* Rồi cập nhật vào file Json.

## Bài code của Nguyễn Quyết Tiến – A35884 (Số ca tử vong)

Thư viện sử dụng: numpy, pandas, fastapi, json, query.

### Tạo dataframe

A screen shot of a computer program

Description automatically generated

### Endpoint về Numpy

Text

Description automatically generated

* Dùng phương thức get để lấy dữ liệu từ data.
* Tạo endpoint: "/highest\_cases\_country".
* Lấy dữ liệu mới nhất của cột cuối cùng của dữ liệu số người chết.
* Chuyển dữ liệu pandas sang numpy.
* Sử dụng số thứ tự của cột đưa vào biến ‘highest\_cases\_ind’.
* Dùng df.loc tìm quốc gia có số người chết nhiều nhất.
* Đầu ra: {"Đất nước có số người chết nhiều nhất là: ": highest\_cases\_country}.

Kết quả:

A screenshot of a computer

Description automatically generated

### Endpoint về Pandas

Text

Description automatically generated

* Dùng phương thức get để lấy dữ liệu từ data.
* Tạo endpoint: "/filter\_death\_total\_dataJSON".
* Tạo hàm "/filter\_death\_total\_dataJSON" với tham số truyền vào là ‘country’.
* Đọc dữ liệu của file data\_death\_dict.
* Nếu dữ liệu đọc đến trùng với tham số truyền vào, trả về: {"message": f"{country} có số ca chết do Covid 19 là {str(item['Total'])}"}.
* Nếu không trùng thì trả về: return {"message": f"{country} not found"}.

Kết quả:

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

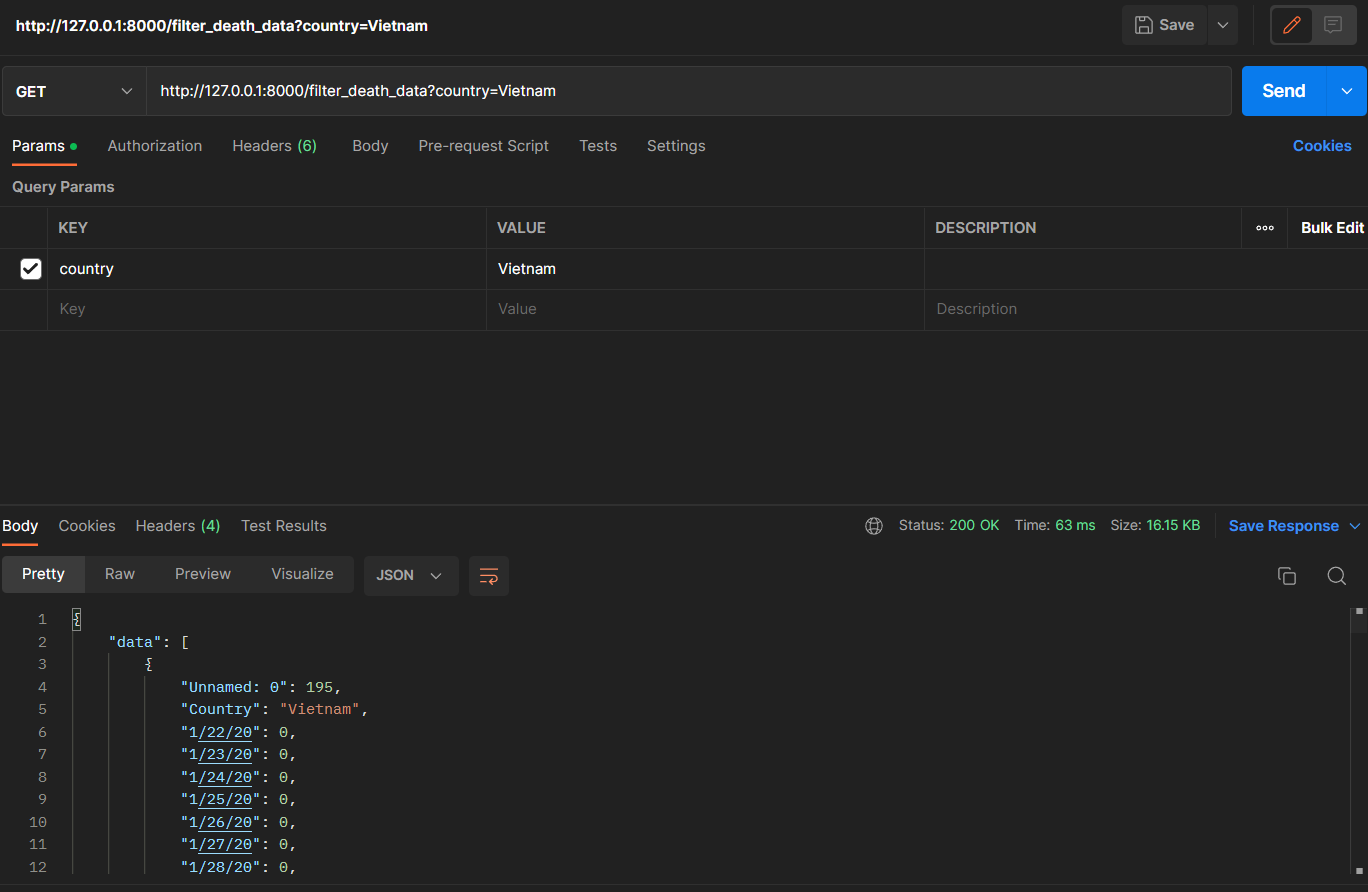
### Endpoint về Get

Text

Description automatically generated

* Dùng phương thức get để lấy dữ liệu từ data.
* Tạo endpoint: "/filter\_death\_data".
* Tạo hàm "/filter\_death\_data" với tham số truyền vào là ‘country’.
* Lấy thông tin từ tham số truyền vào.
* Trả về thông tin ca tử vong của quốc gia đó theo từng ngày: return {"data":data}.

Kết quả:



### Endpoint về Post

Text

Description automatically generated

* Dùng phương thức get để lấy dữ liệu từ data.
* Tạo endpoint: "/update\_death\_total".
* Tạo hàm "/update\_death\_total"với tham số truyền vào là ‘country’ và ‘new\_total’.
* Đọc dữ liệu của file data\_death\_dict.
* Nếu thông tin từ cột ‘country’ trùng với dữ liệu đầu vào, cập nhật số từ tham số ‘new\_total’ vào số ca tử vong.
* Ghi dữ liệu mới vào file JSON: data\_death\_dict.
* Trả về tổng số ca tử vong của quốc gia đó: return {"message": f"{country} có số ca chết do Covid 19 là {new\_total}"}.
* Nếu không tìm thấy ‘country’, trả về: return {"message": f"{country} not found"}.

Kết quả:

* Dữ liệu ban đầu:

Text

Description automatically generated

* Dữ liệu sau khi cập nhật:

A screenshot of a computer

Description automatically generated

* Dữ liệu mới trong file JSON:

Text

Description automatically generated

## Bài code của Nguyễn Quyết Tiến – A35884 (Số ca nhiễm bệnh)

A screen shot of a computer code

Description automatically generated

Các thư viện sử dụng:Fastapi,numpy,pandas,Query.

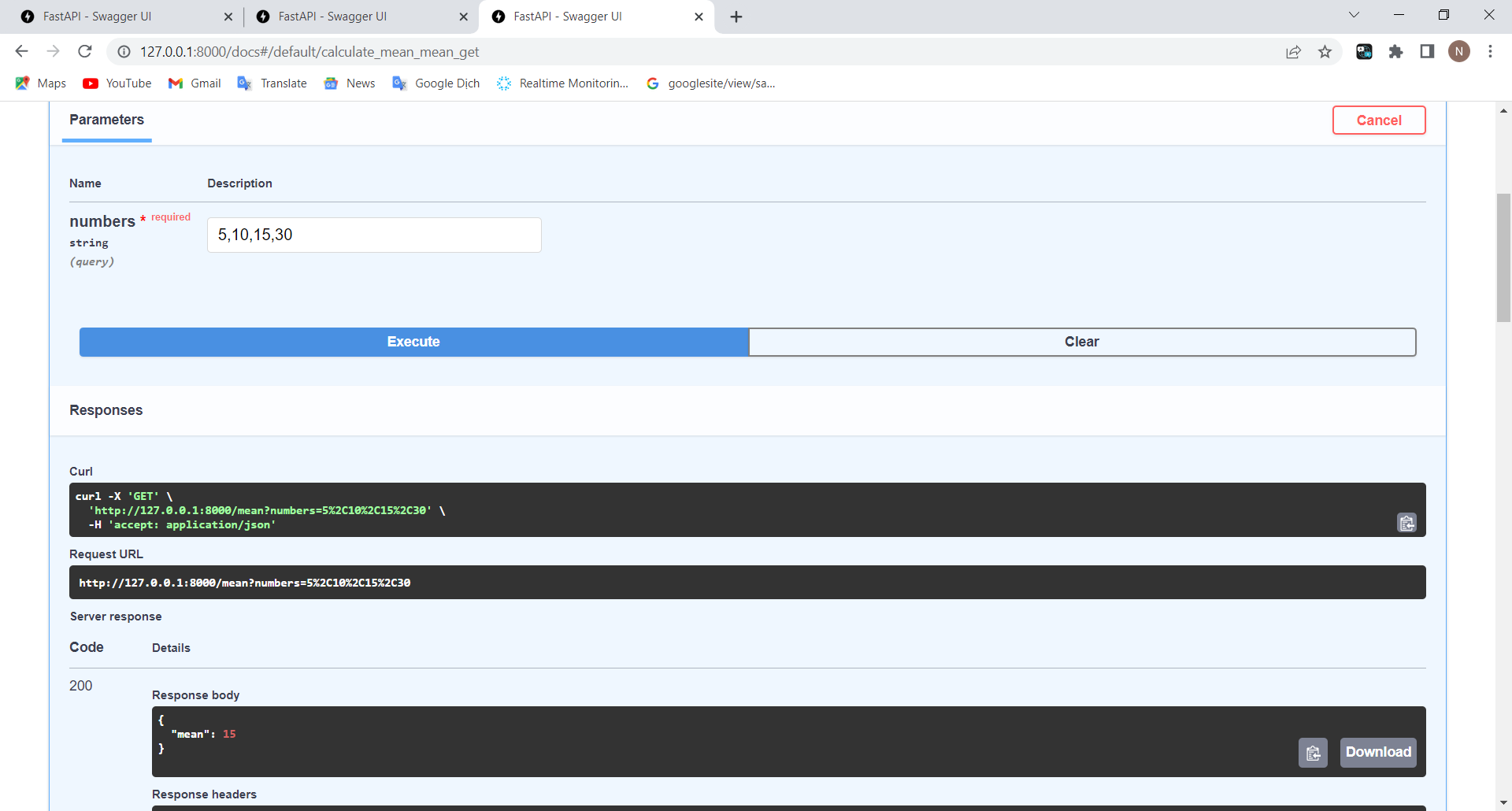
### Endpoint sử dụng Numpy

Text

Description automatically generated

* Dùng phương thức get để nhập dữ liệu
* Tính trung bình của một tập
* Trả về giá trị {“mean”:np.mean(arr)}

Kết quả:



### Endpoint sử dụng Get

A screen shot of a computer code

Description automatically generated

Sử dụng pandas để đọc file: “confirmed\_data” để đọc file dữ liệu về số ca nhiễm bệnh bởi covid-19.

#### Truy vấn cột

Text

Description automatically generated

* Sử dụng phương thức get với endpoint là “/filter\_confirmed\_data” để biết số ca nhiễm bệnh trong 365 ngày đầu tiên kể từ khi phát hiện ca nhiễm bệnh tại quốc gia được chỉ định.
* Khai biến hàm “filter\_confirmed\_data” với tham số truyền vào là “country”,cột bắt đầu và cột kết thúc được truy vấn.
* Khai báo “selected\_cols” là bảng dữ liệu được truy vấn theo cột trong “confirmed\_df”, cột bắt đầu, cột kết thúc
* Và trả về dữ liệu của cột được tham chiếu.

Kết quả:

Graphical user interface, text

Description automatically generated

#### Lấy số liệu ca nhiễm từ Việt Nam

Text

Description automatically generated

* Khai báo vn\_confirmed với tham số truyền vào là quốc gia việt nam
* Sử dụng “vn\_confirmed.drop…” để xóa cột.
* Sử dụng min\_confirmed=np.min(vn\_confirmed.values) để lấy ra số liệu ca nhiễm ít nhất của Việt Nam.
* Sử dụng max\_confirmed=np.max(vn\_confirmed.values) để lấy ra số liệu ca nhiễm cao nhất của Việt Nam.
* Và trả về kiểu dữ liệu là int.

Kết quả:

Graphical user interface, application

Description automatically generated

### Endpoint sử dụng Post

Text

Description automatically generated

* Sử dụng phương thức get\_country\_confirmed để lấy dữ liệu các quốc gia ở trong data.
* Sử dụng confirmed\_data để lấy tên quốc gia và kiểm tra có phải trong danh sách các quốc gia đó hay không.
* Total\_confirmed để lấy dữ liệu số ca nhiễm bệnh kể từ ngày bắt đầu cho đến ngày cuối cùng phát hiện ra ca nhiễm và tính tổng.
* Trả về dữ liệu {“confirmed\_cases”: total\_confirmed. to \_ dict()} và chuyển dữ liệu thành kiểu dict.

Kết quả:

Graphical user interface

Description automatically generated with medium confidence

# Swagger UI

## Swagger UI là gì?

Swagger UI là một công cụ mã nguồn mở được sử dụng để tạo tài liệu API tự động và cung cấp cho người dùng một giao diện đồ họa để khám phá các API đó. Nó được   
xây dựng trên cơ sở của các tài liệu API được định dạng bằng Swagger hoặc OpenAPI,   
cho phép người dùng dễ dàng xem và tương tác với các API đó thông qua giao diện web.

Swagger UI cung cấp một giao diện đơn giản và trực quan, cho phép người dùng khám phá tài liệu API, kiểm tra các yêu cầu và phản hồi, và thử nghiệm các API đó   
bằng cách gửi các yêu cầu HTTP tương ứng. Nó cung cấp cả một số tính năng tiện ích như   
auto-complete và syntax highlighting để giúp người dùng viết yêu cầu HTTP chính xác và   
nhanh chóng hơn.

## Thực hành về Swagger UI

### Swagger của Trần Minh Hải – A36154

Graphical user interface, text, application, email

Description automatically generated

### Swagger của Nguyễn Quyết Tiến – A35884

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated