



VacciTrack - Hệ thống theo dõi

Phân tích dữ liệu tiêm chủng

Tên: Nguyễn Minh Kha

Lớp: TC-PY4E03

Người hướng dẫn : Lý Quốc Lâm

Mục Lục

01 Giới thiệu dự án

04 Phân luồng

02 Công nghệ

05 Quản Lý

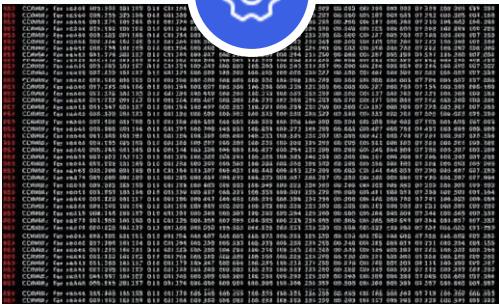
03 Vai Trò

06 Phát triển

01

Tổng quan dự án VacciTrack

Giải quyết vấn đề dữ liệu



Vấn đề dữ liệu thô

Dữ liệu COVID-19 tồn tại dạng CSV khổng lồ, khó đọc và khó so sánh.



Thách thức phân tích

Hàng nghìn dòng số liệu rời rạc
không thể hỗ trợ ra quyết định
nhanh chóng.



Giải pháp VacciTrack

Web App tự động xử lý, làm sạch và trực quan hóa dữ liệu thành Dashboard dễ hiểu.

Ứng dụng Python Flask

Xử lý dữ liệu

Python và Pandas xử lý file CSV phức tạp hiệu quả.

Micro-framework

Flask nhẹ, triển khai nhanh, phù hợp xây dựng RESTful API.

Tư duy Fullstack

Kết hợp Data Engineering, Backend và Frontend tạo hệ thống hoàn chỉnh.

Kiến trúc Client-Server

Tách biệt tầng dữ liệu và hiển thị, tối ưu luồng thông tin.

Rank	No file	Organization	Version	Type	Events
1. 8	JS-26	DigitalOcean	0.10.0	Project Management	61,100, 0,000
1. 9	JS-69	Dynamiq	0.2.0	Project Management	43,000, Project Selected
1. 9	JS-59	Devstack	0.1.0	Project Management	19,000, Project Selected
1. 9	JS-98	Gatsby	0.10.0	Project Management	19,100, Project Selected
1. 9	JS-06	Dropbox	Project	Project Management	12,400, Project Selected
1. 9	JS-42	Dropbox	Project	Project Management	12,100, Project Selected
1. 9	JS-59	Dropbox	Project	Project Management	12,000, Project Selected
1. 9	JS-13	Dropbox	Project	Project Management	8,000, Project Selected
1. 9	JS-95	DuckDuckGo	Project	Project Management	7,000, Project Selected
1. 9	JS-20	Elm.js	Project	Project Management	2,100, Project Selected
1. 9	JS-48	Enzyme	Project	Project Management	2,000, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	1,900, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	1,800, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	1,700, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	1,600, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	1,500, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	1,400, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	1,300, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	1,200, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	1,100, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	1,000, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	900, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	800, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	700, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	600, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	500, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	400, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	300, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	200, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	100, Project Selected
1. 9	JS-59	EventSource	Project	Project Management	0, Project Selected

Trực quan hóa dữ liệu

Biểu đồ động

Chuyển đổi dữ liệu thô thành biểu đồ trực quan, dễ hiểu và tương tác.

Chart.js

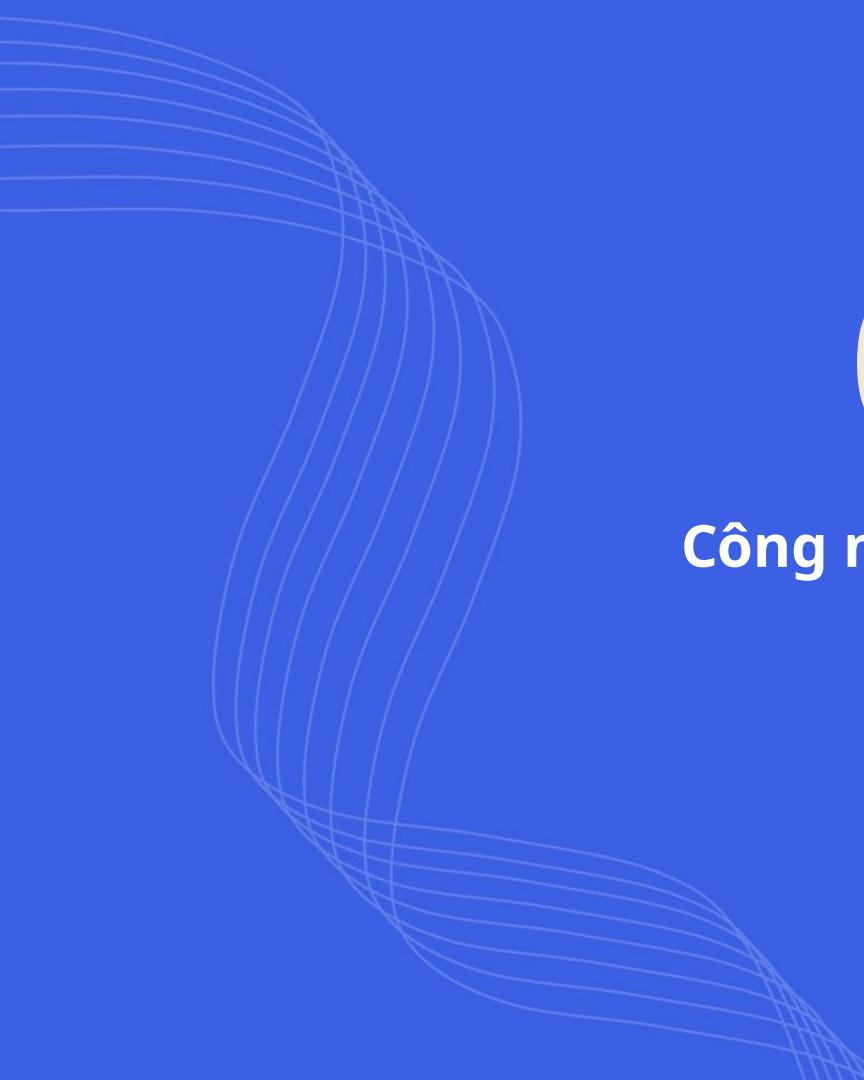
Thư viện JavaScript tạo biểu đồ tương tác đa dạng: đường, tròn, cột.



Dashboard hiện đại

Giao diện Glassmorphism với Tailwind CSS, hiển thị đồng bộ nhiều loại biểu đồ.

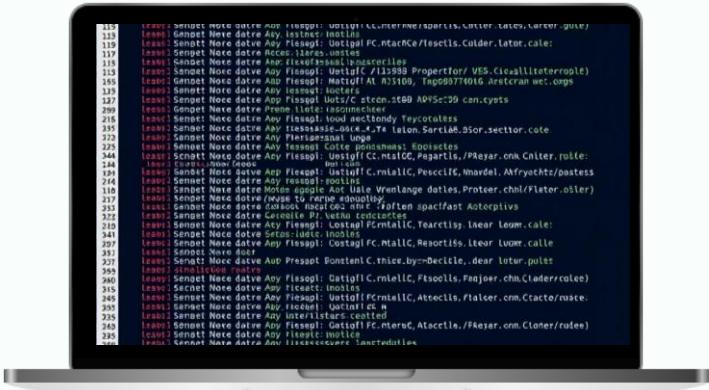




02

Công nghệ sử dụng

Python và Pandas



1

Xử lý dữ liệu mạnh mẽ

Thư viện tốt nhất xử lý CSV phức tạp, dễ thao tác hàng triệu dòng dữ liệu.

2

Flask framework nhẹ

Triển khai nhanh, phù hợp xây dựng RESTful API cho dự án phân tích dữ liệu.

3

Làm sạch dữ liệu

Xử lý giá trị rỗng, ép kiểu dữ liệu, tạo cấu trúc sẵn sàng cho API.

Flask Micro-framework



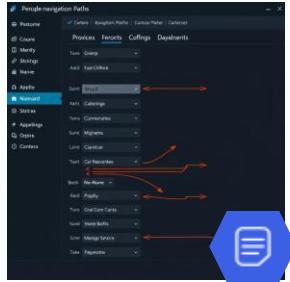
Micro-framework nhẹ

Triển khai nhanh, phù hợp cho dự án data-driven.



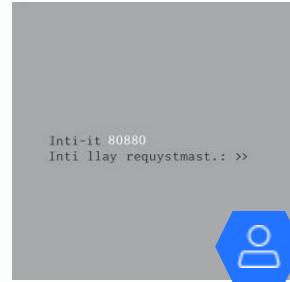
RESTful API

Xây dựng API hiệu quả cho giao tiếp client-server.



Routing linh hoạt

Cấu hình đường dẫn điều hướng dễ dàng.



Web Server

Khởi tạo server tại port 8080, xử lý request.

HTML5/Tailwind CSS



Giao diện Glassmorphism

Sử dụng Tailwind CSS tạo hiệu ứng kính mờ, tối ưu hiển thị mọi thiết bị.

Responsive Design

Thiết kế thích ứng tự động với mọi kích thước màn hình, từ điện thoại đến máy tính.

Chart.js

Thư viện vẽ biểu đồ tương tác, hiển thị dữ liệu trực quan sinh động.

Chart.js và GenAI



01

Biểu đồ tương tác

Chart.js tạo đồ thị động, trực quan từ dữ liệu JSON, hỗ trợ nhiều loại biểu đồ.



02

Hỗ trợ AI

GenAI sinh mã CSS nhanh chóng, tăng tốc phát triển giao diện, giảm lỗi cú pháp.



03

Tối ưu UX

AI gợi ý cấu trúc MVC, debug JavaScript, cải thiện trải nghiệm người dùng.



03

Vai trò và quy trình

Xử lý dữ liệu thô

01

Thách thức dữ liệu

File CSV khổng lồ với hàng nghìn dòng số liệu rời rạc, khó đọc và phân tích.

02

Công cụ Pandas

Thư viện mạnh mẽ xử lý file CSV phức tạp, dễ dàng thao tác hàng triệu dòng dữ liệu.

03

Quy trình làm sạch

Đọc file an toàn, xử lý lỗi, chuẩn hóa giá trị rỗng và ép kiểu dữ liệu.

04

Kết quả đầu ra

Dữ liệu sạch dạng List of Dictionaries, sẵn sàng cho API và hiển thị.



Xây dựng API Backend

Khởi tạo Flask

Tạo web server tại port 8080, cấu hình routing và API endpoints.

1

Xử lý dữ liệu

Sử dụng Pandas làm sạch dữ liệu CSV, chuyển thành JSON.

2

API Endpoints

Tạo /api/vaccines phân phối dữ liệu cho client.

3

Mô hình MVC

Tách biệt logic xử lý và hiển thị dữ liệu.

4

Trực quan hóa Frontend

- 
- 01 Giao diện Glassmorphism**
Sử dụng Tailwind CSS tạo hiệu ứng kính mờ, tối ưu hiển thị mọi thiết bị.
 - 02 GenAI Support**
AI sinh mã CSS nhanh, tăng tốc phát triển 50%, giảm lỗi cú pháp.
 - 03 Hiển thị dữ liệu**
Fetch API gọi dữ liệu, chuyển JSON thành biểu đồ Chart.js đa dạng.



04

Triển khai dự án

Xử lý Raw Data

Input: File **country_vaccinations.csv** chứa nhiều nhiễu (NaN, null, sai format ngày tháng). dẫn đến sập server

Hành Động:

Khi bao cho biết file csv sử dụng ; : sep=';'

Đọc file an toàn bằng engine python : (engine='python')

Xử lý dòng khi xuất hiện lỗi: (on_bad_lines='skip').

ParserError

pandas.errors.ParserError: Error tokenizing data. C error: Expected 4 fields in line 2850, saw 6

Traceback (most recent call last)

```
File "C:\Users\Administrator\AppData\Local\Programs\Python\Python311\lib\site-packages\fastapi\app.py", line 1406, in __call__
    return self._wsgi_app(environ, start_response)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^
File "C:\Users\Administrator\AppData\Local\Programs\Python\Python311\lib\site-packages\fastapi\app.py", line 1476, in wsgi_app
    response = self._handle_exception(e)
               ^^^^^^^^^^^^^^^^^^
File "C:\Users\Administrator\AppData\Local\Programs\Python\Python311\lib\site-packages\fastapi\app.py", line 1472, in wsgi_app
    response = self._full_dispatch_request()
               ^^^^^^^^^^^^^^^^^^
File "C:\Users\Administrator\AppData\Local\Programs\Python\Python311\lib\site-packages\fastapi\app.py", line 802, in _full_dispatch_request
    rv = self._handle_user_exception(e)
           ^^^^^^^^^^^^^^^^^^
File "C:\Users\Administrator\AppData\Local\Programs\Python\Python311\lib\site-packages\fastapi\app.py", line 800, in _full_dispatch_request
    rv = self._dispatch_request()
           ^^^^^^^^^^^^^^
File "C:\Users\Administrator\AppData\Local\Programs\Python\Python311\lib\site-packages\fastapi\app.py", line 805, in _dispatch_request
    return self._ensure_sync(self._view_functions[rvs.endpoint])(**view_args) # type: ignore[no-any-return]
File "D:\Administrador\Documents\code\fastCovid\testCovid\readvaccines.py", line 21, in get_vaccine
    df = pd.read_csv(
          ^^^^^^
File "C:\Users\Administrator\AppData\Local\Programs\Python\Python311\lib\site-packages\pandas\io\parsers\readers.py", line 1026, in read_csv
    return _read_file(f, sep=sep, header=header, chunksize=chunksize, memory=memory)
           ^^^^^^^^^^
File "C:\Users\Administrator\AppData\Local\Programs\Python\Python311\lib\site-packages\pandas\io\parsers\readers.py", line 626, in _read
    return parse(read_csv())
           ^^^^^^^^^^
File "C:\Users\Administrator\AppData\Local\Programs\Python\Python311\lib\site-packages\pandas\io\parsers\readers.py", line 1925, in read
    ) + self._engine.read() # type: ignore[attr-defined]
File "C:\Users\Administrator\AppData\Local\Programs\Python\Python311\lib\site-packages\pandas\io\parsers\generic_parser_wrapper.py", line 234, in read
    chunks + self._reader.read_low_memory(mrows)
           ^^^^^^^^^^
File "pandas\_parser.pyx", line 832 in pandas._libs.parsers.TextReader.read_low_memory
File "pandas\_parser.pyx", line 902 in pandas._libs.parsers.TextReader._read_low_memory
File "pandas\_parser.pyx", line 674 in pandas._libs.parsers.TextReader._tokenize_rows
File "pandas\_parser.pyx", line 654 in pandas._libs.parsers.TextReader._tokenize_rows
File "pandas\_parser.pyx", line 654 in pandas._libs.parsers.TextReader._tokenize_rows
```

Làm sạch dữ liệu

làm sạch dữ liệu(Data Cleaning):

Chuẩn hóa: Xử lý giá trị rỗng bằng `df.where(pd.notnull(df), None)`

Ép kiểu: Chuyển đổi dữ liệu sang `object` hoặc `datetime`.



Xây dựng Server

Xây dựng web server:

khởi tạo flask app tại port **8080**:

Routing & API:

Routes: Cấu hình đường dẫn **@app.route** để điều hướng (Home, Analytics)

API Endpoint: Tạo **/api/vaccines** để phân phối dữ liệu JSON cho Client.



Giao diện hiển thị dữ liệu

Hiển thị dữ liệu (Visualization):

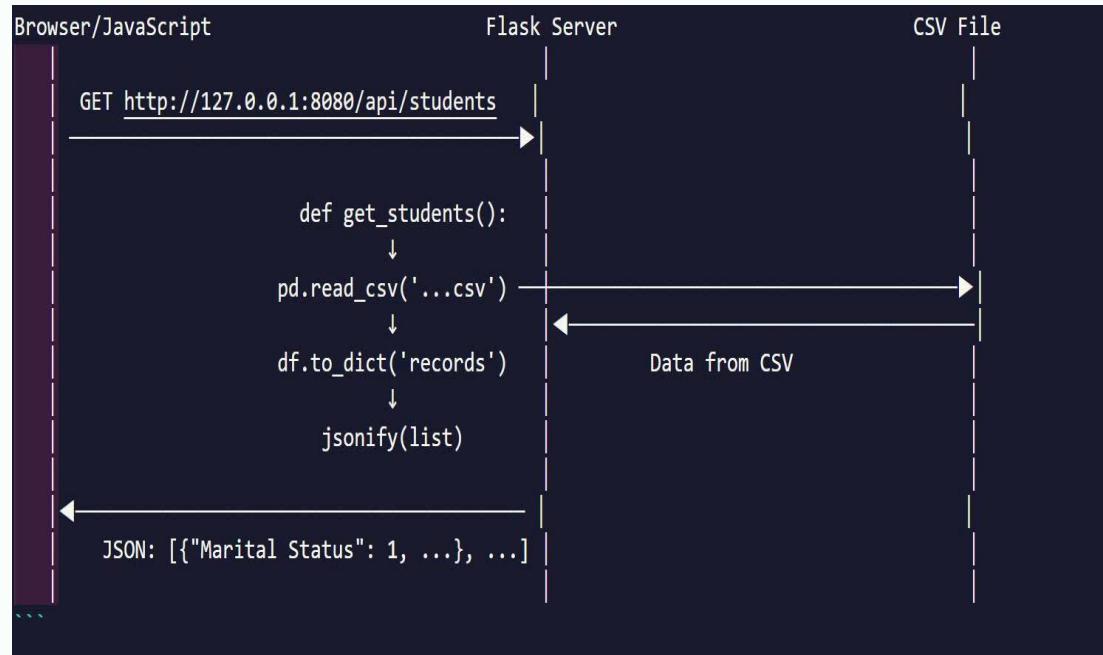


Client-Server :Sử dụng **Fetch API** để gọi dữ liệu từ Backend
Chart.js: Convert dữ liệu JSON thành các biểu đồ tương tác (Line, Doughnut, Polar Area)

Kiến trúc hệ thống

Mô hình Client-Server:

1. **Client:** Gửi Request
2. **Flask Server:** Nhận lệnh → Gọi Pandas
3. **Pandas:** Đọc & Làm sạch
4. **Server:** Trả về JSON
5. **Client:** Hiển thị biểu đồ



Quản lý mã nguồn Git

Quản lý phiên bản

Sử dụng Git theo dõi thay đổi mã nguồn, dễ dàng quay lại phiên bản trước.

Commit thường xuyên

Áp dụng Feature-based Commits, ghi chú rõ ràng cho từng tính năng mới.

Đảm bảo an toàn

Bảo vệ mã nguồn, cho phép rollback nhanh chóng khi gặp lỗi.

Triển khai qua Ngrok

localhost

Chạy thử nghiệm và Debug trên máy cá nhân.



01

Demo từ xa

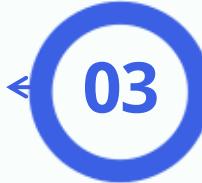
Trình diễn sản phẩm trực tiếp từ bất kỳ đâu.



02

Ngrok Tunnel

Tạo đường hầm bảo mật để public web server ra Internet.



03

Thanks !