

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH



LẬP TRÌNH NÂNG CAO

BÁO CÁO BÀI TẬP LỚN

HK241 - Nhóm: 495

ĐỀ TÀI:

THIẾT KẾ WEBSITE TRA CỨU SAO KÊ

GV hướng dẫn: LÊ ĐÌNH THUẬN

Danh sách thành viên:

STT	Họ và tên	MSSV	Tỷ lệ đóng góp	Email
1	Nguyễn Anh Khoa	2053134	100%	khoa.nguyen0101ag@hcmut.edu.vn
2	Từ Mai Thế Nhân	2114277	100%	nhan.tu2107@hcmut.edu.vn
3	Hoàng Lê Đông Hà	2210843	95%	ha.hoangledongbrvt0@hcmut.edu.vn
4	Nguyễn Chí Trung	2213696	100%	trung.nguyenbkhcm@hcmut.edu.vn
5	Huỳnh Huy Mân	2433003	100%	man.huynh2433003@hcmut.edu.vn

Mục lục

1	Phân công công việc	3
1.1	Bảng phân công công việc	3
2	Giới thiệu	4
3	Công cụ hỗ trợ	5
3.1	Visual Studio Code	5
3.2	Thư viện venv	5
3.3	Thư viện FastAPI	6
3.4	Tailwind CSS	6
3.5	Thư viện Pandas	7
4	Thiết kế figma và UI web	9
5	Cấu trúc dữ liệu và giải thuật tìm kiếm, thiết kế ứng dụng	11
5.1	Cấu trúc dữ liệu và giải thuật tìm kiếm	11
5.2	Cấu trúc mã nguồn	13
5.3	các tính năng xây dựng	15
5.4	Flowchart của website	17
6	Hướng dẫn sử dụng	19
6.1	Run project by via DockerHub	19
6.2	NORMAL WAY RUN PROJECT	19
6.2.1	Step 1: Clone the repository	19
6.2.2	Step 2: Build the Docker image	19
6.2.3	Step 3: Start the application	19
6.2.4	Step 4: Access the application:	19
6.2.5	Step 5: Stop the application:	19
6.3	Local Development (Without Docker)	19
7	Tài liệu tham khảo	21

Danh sách hình vẽ

1	Visual Studio Code	5
2	Python virtual environment	6
3	Thư viện FastAPI	6
4	Tailwind CSS	7
5	Thư viện Pandas	8
6	Giao diện khởi động	9
7	Hiển thị bộ lọc “Filters”	9
8	Kết quả tìm kiếm với từ khóa “phuong anh”	9
9	Kết quả lọc Start Date, End Date và số tiền “Less than 4 million”	9
10	Kết quả lọc Start Date, End Date, Min Amount và Max Amount	10
11	Kết quả lọc trả về “No transactions found”	10
12	Cấu trúc mã nguồn	13
13	Flowchart của ứng dụng	17



1 Phân công công việc

1.1 Bảng phân công công việc

STT	Họ và tên	MSSV	Nhiệm vụ
1	Nguyễn Anh Khoa	2053134	Backend, thiết kế database, frontend, chỉnh sửa, tổng duyệt các tính năng, lên list công việc
2	Từ Mai Thế Nhân	2115036	Backend, thiết kế database
3	Hoàng Lê Đông Hà	2210843	Frontend, Figma
4	Nguyễn Chi Trung	2213696	Phân loại data, truy vấn data, test performance
5	Huỳnh Huy Mân	2433003	Phân loại data, viết báo cáo

Link github source code: https://github.com/anhKh0a0902/state_webv3

2 Giới thiệu

Trang web tra cứu sao kê ủng người dân bị thiệt hại do bão lũ được xây dựng với mục tiêu cung cấp thông tin minh bạch và dễ dàng cho toàn dân về các khoản ủng hộ dành cho những người gặp khó khăn trong thiên tai.

Người nhận các khoản ủng hộ là tài khoản ngân hàng của Ủy Ban Mặt Trận Tổ Quốc, nơi tiếp nhận đóng góp từ cộng đồng. Tất cả mọi người dân đều có thể gửi tiền ủng hộ, thể hiện tinh thần tương thân tương ái trong những lúc khó khăn.

Trang web cho phép người dùng tra cứu thông tin theo nhiều tiêu chí khác nhau, bao gồm:

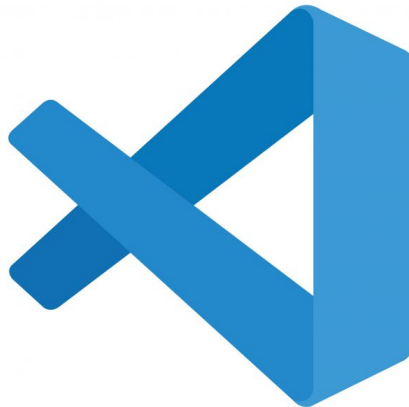
- Theo ngày gửi
- Theo tên người gửi
- Theo số tiền gửi

Thông qua nền tảng này, chúng tôi mong muốn tạo ra sự minh bạch trong việc quản lý và sử dụng các khoản đóng góp, đồng thời khuyến khích mọi người cùng chung tay hỗ trợ bà con bị ảnh hưởng bởi bão lũ.

3 Công cụ hỗ trợ

3.1 Visual Studio Code

- Visual Studio Code là gì ?
Visual Studio Code (VS Code) là một trình biên tập mã nguồn được phát triển bởi Microsoft. Nó cung cấp một môi trường làm việc mạnh mẽ và linh hoạt cho lập trình viên.
- Một số tính năng tiêu biểu của VS Code
 - Hỗ trợ Git: Tích hợp sẵn công cụ quản lý phiên bản.
 - Debugging: Tính năng gỡ lỗi giúp phát hiện và sửa lỗi trong mã.
 - Syntax Highlighting: Làm nổi bật cú pháp cho nhiều ngôn ngữ lập trình.
 - Snippets: Tính năng hỗ trợ mã để tiết kiệm thời gian viết code.
 - VS Code hỗ trợ nhiều ngôn ngữ lập trình như: HTML, CSS, JavaScript, C/C++, C, F, JSON.



Hình 1: Visual Studio Code

3.2 Thư viện venv

- Thư viện venv là gì?
venv là một công cụ trong Python dùng để tạo môi trường ảo. Nó cho phép bạn tạo ra một không gian riêng biệt cho mỗi dự án, giúp quản lý các gói và thư viện mà không gây xung đột với các dự án khác.
- Một số tính năng tiêu biểu của venv
 - **Tạo môi trường ảo:** Mỗi môi trường ảo có thể chứa các phiên bản thư viện khác nhau.
 - **Quản lý phụ thuộc:** Dễ dàng cài đặt và quản lý các gói cần thiết cho dự án.
 - **Cô lập:** Giúp cô lập các dự án Python, đảm bảo rằng các thay đổi trong một dự án không ảnh hưởng đến dự án khác.



Hình 2: *Python virtual environment*

3.3 Thư viện FastAPI

- Thư viện FastAPI là gì?
FastAPI là một thư viện Python hiện đại dùng để xây dựng các API nhanh chóng và hiệu quả. Nó dựa trên tiêu chuẩn ASGI (Asynchronous Server Gateway Interface) và hỗ trợ việc viết các ứng dụng web không đồng bộ.
- Tính chất tiêu biểu của FastAPI
 - Tốc độ cao: FastAPI được thiết kế để có hiệu suất cao, gần bằng với Node.js và Go, nhờ vào việc sử dụng Starlette cho các tính năng mạng.
 - Hỗ trợ tự động hóa tài liệu: FastAPI tự động tạo tài liệu API bằng OpenAPI và Swagger UI, giúp lập trình viên dễ dàng thử nghiệm và khám phá API.
 - Kiểm tra và xác thực: Hỗ trợ kiểm tra kiểu dữ liệu và xác thực đầu vào một cách dễ dàng thông qua Pydantic.
- Tính năng nổi bật của thư viện phù hợp cho:
 - Xây dựng API RESTful: Dễ dàng tạo các API RESTful cho ứng dụng web.
 - Hỗ trợ Async/Await: Cho phép xử lý đồng thời nhiều yêu cầu mà không làm chậm ứng dụng.
 - Tích hợp dễ dàng với các thư viện khác: Có thể tích hợp với cơ sở dữ liệu, xác thực, và các thư viện khác một cách dễ dàng.



Hình 3: *Thư viện FastAPI*

3.4 Tailwind CSS

- Tailwind CSS là gì?
Tailwind CSS là một framework CSS tiện ích giúp xây dựng giao diện người dùng một cách nhanh

chóng và linh hoạt. Thay vì sử dụng các lớp CSS truyền thống, Tailwind cung cấp một bộ các lớp tiện ích để tạo kiểu cho các thành phần trên trang web.

- Một số tính năng tiêu biểu của Tailwind CSS
 - Tiện ích (Utility-first): Cung cấp hàng trăm lớp tiện ích để bạn có thể xây dựng giao diện mà không cần viết CSS tùy chỉnh.
 - Tùy biến dễ dàng: Hỗ trợ cấu hình và tùy biến thông qua tệp cấu hình, cho phép bạn điều chỉnh màu sắc, kích thước và nhiều thuộc tính khác.
 - Responsive Design: Dễ dàng tạo các bố cục responsive bằng cách sử dụng các lớp tiện ích cho các kích thước màn hình khác nhau.
- Ứng dụng của Tailwind CSS
 - Tăng tốc độ phát triển: Giúp lập trình viên xây dựng giao diện mà không cần phải viết nhiều CSS.
 - Dễ bảo trì: Mã nguồn trở nên dễ hiểu hơn nhờ vào việc sử dụng các lớp tiện ích có tên rõ ràng.
 - Xử lý trạng thái phức tạp: Dễ dàng thêm các trạng thái như hover, focus, và active chỉ bằng cách thêm các lớp tiện ích tương ứng.



Hình 4: *Tailwind CSS*

3.5 Thư viện Pandas

- Thư viện Pandas là gì?

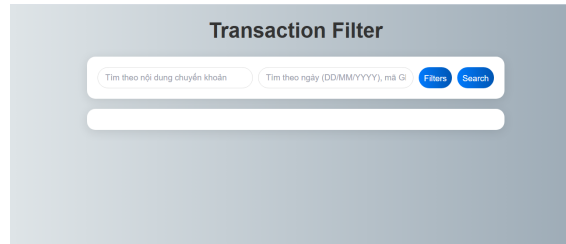
Pandas là một thư viện mã nguồn mở trong Python, được thiết kế để xử lý và phân tích dữ liệu. Nó cung cấp các cấu trúc dữ liệu và chức năng để làm việc với dữ liệu có tổ chức, như bảng và chuỗi thời gian.
- Một số tính năng tiêu biểu của Pandas
 - Cấu trúc dữ liệu mạnh mẽ: Pandas cung cấp hai cấu trúc chính là DataFrame và Series, giúp dễ dàng quản lý và phân tích dữ liệu.
 - Xử lý dữ liệu linh hoạt: Hỗ trợ nhiều thao tác như lọc, nhóm, và biến đổi dữ liệu một cách dễ dàng.
 - Hỗ trợ nhiều định dạng dữ liệu: Có thể đọc và ghi dữ liệu từ nhiều nguồn khác nhau như CSV, Excel, SQL, và JSON.

- Ứng dụng của Pandas
 - Xử lý dữ liệu: Cung cấp các công cụ để xử lý và thay thế dữ liệu nếu thiếu.
 - Thao tác thời gian: Hỗ trợ các kiểu dữ liệu thời gian, cho phép phân tích và xử lý dữ liệu theo thời gian hiệu quả.
 - Tích hợp với các thư viện khác: Tương thích tốt với các thư viện như NumPy, Matplotlib và Scikit-learn, giúp mở rộng khả năng phân tích dữ liệu.

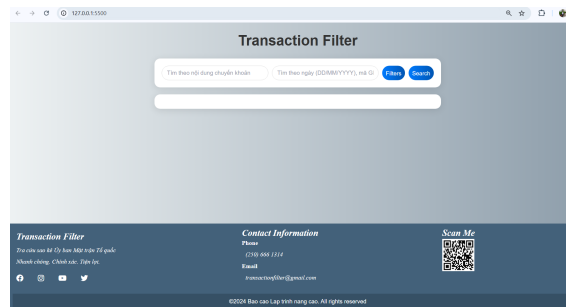


Hình 5: *Thư viện Pandas*

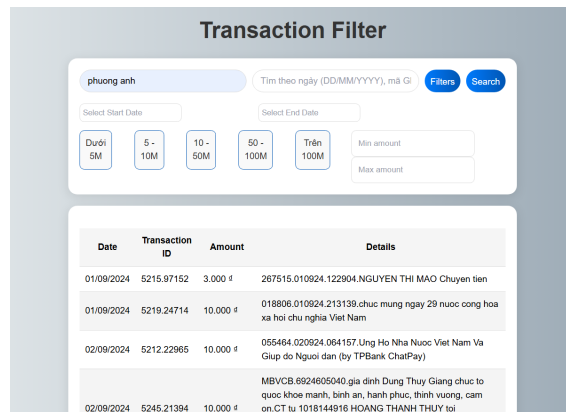
4 Thiết kế figma và UI web



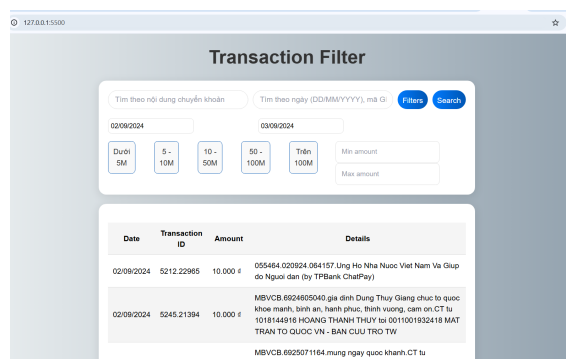
Hình 6: Giao diện khởi động



Hình 7: Hiển thị bộ lọc “Filters”



Hình 8: Kết quả tìm kiếm với từ khóa “phuong anh”



Hình 9: Kết quả lọc Start Date, End Date và số tiền “Less than 4 million”

Transaction Filter

Filters Search

Dưới 5M

5 - 10M

10 - 50M

50 - 100M

Trên 100M

Date	Transaction ID	Amount	Details
02/09/2024	5212.22965	10,000 đ	055464.020924.064157.Ứng Ho Nhà Nước Viet Nam Va Glup do Nguoi dan (by TPBank ChatPay)
02/09/2024	5245.21394	10,000 đ	MBVCB.6924605040.gia dinh Dung Thuy Giang chuc to quoc khoe manh, binh an, hanh phuc, thinh vuong, cam on.CT tu 1018144916 HOANG THANH THUY tai 0011001932418 MAT TRAN TO QUOC VN - BAN CUU TRO TW
02/09/2024	5078.73943	50,000 đ	MBVCB.6925071164.mung ngay quoc khanh.CT tu 1028808193 toi 0011001932418 Uy Ban Trung uong Mat tran

Hình 10: Kết quả lọc Start Date, End Date, Min Amount và Max Amount

Transaction Filter

Filters Search

Dưới 5M

5 - 10M

10 - 50M

50 - 100M

Trên 100M

Loading...

No transactions found

Hình 11: Kết quả lọc trả về “No transactions found”

5 Cấu trúc dữ liệu và giải thuật tìm kiếm, thiết kế ứng dụng

5.1 Cấu trúc dữ liệu và giải thuật tìm kiếm

Thư viện pandas là gì?

- Thư viện pandas trong python là một thư viện mã nguồn mở, hỗ trợ đắc lực trong thao tác dữ liệu. Đây cũng là bộ công cụ phân tích và xử lý dữ liệu mạnh mẽ của ngôn ngữ lập trình python. Thư viện này được sử dụng rộng rãi trong cả nghiên cứu lẫn phát triển các ứng dụng về khoa học dữ liệu
- Thư viện này sử dụng một cấu trúc dữ liệu riêng là Dataframe. Pandas cung cấp rất nhiều chức năng xử lý và làm việc trên cấu trúc dữ liệu này. Chính sự linh hoạt và hiệu quả đã khiến cho pandas được sử dụng rộng rãi.

Cấu Trúc Dữ Liệu Chính

DataFrame Pandas: Là cấu trúc dữ liệu chính

- Xử lý các hàng, cột trong file csv và lưu trữ toàn bộ giao dịch
- Các cột chính:
 - credit: Số tiền giao dịch (float64)
 - datetime: Chuỗi ngày và mã giao dịch
 - detail: Chi tiết giao dịch
 - date: Ngày giao dịch (datetime)
 - transid: Mã giao dịch

Giải Thuật Xử Lý Dữ Liệu

- Nạp và Làm Sạch Dữ Liệu:

Listing 1: Nạp và Làm Sạch Dữ Liệu

```
def load_data(self):

    self.df = pd.read_csv('chuyen_khoan.csv', dtype={
        'credit': 'float64',
        'date_time': 'str',
        'detail': 'str',
    })

    self.df[['date', 'trans_id']] = self.df['date_time'].
        str.split('_', expand=True)

    self.df['date'] = pd.to_datetime(self.df['date'],
        format='%d/%m/%Y', errors='coerce')
```

```
self.df = self.df.dropna(subset=['credit', 'date'])

}
```

- Giải Thuật Tìm Kiếm Đa Trường:

Listing 2: Giải Thuật Tìm Kiếm Đa Trường

```
def search_multiple_fields(self, search_text: str) -> pd.
    DataFrame:
    df['credit_str'] = df['credit'].astype(str)
    df['date_str'] = df['date'].dt.strftime('%d/%m/%Y')

    mask = (
        df['detail'].str.lower().str.contains(search_text, na=
            False) |
        df['trans_id'].str.lower().str.contains(search_text, na
            =False) |
        df['credit_str'].str.contains(search_text, na=False) |
        df['date_str'].str.contains(search_text, na=False)
    )

    if any(x in search_text for x in ['tr', 'm', 'trieu-
        tiengviet', 'trieu']):
        amount = self._convert_amount_text(search_text)
        if amount:

            lower_bound = amount * 0.9
            upper_bound = amount * 1.1
            amount_mask = (df['credit'] >= lower_bound) & (df['
                credit'] <= upper_bound)
            mask = mask | amount_mask
```

- Giải Thuật Chuyển Đổi Số Tiền:

Listing 3: Giải Thuật Chuyển Đổi Số Tiền:

```
def _convert_amount_text(self, text: str) -> float:
    patterns = {
        r'(\d+)tr': lambda x: float(x) * 1000000,
        r'(\d+)m': lambda x: float(x) * 1000000,
        r'(\d+)trieu': lambda x: float(x) * 1000000,
        r'(\d+)t': lambda x: float(x) * 1000000
    }

    for pattern, converter in patterns.items():
        match = re.search(pattern, text)
        if match:
            return converter(match.group(1))
    return None
```

Chiến Lược Lọc Dữ Liệu

- Hỗ trợ nhiều tiêu chí lọc đồng thời:
 - Lọc theo cột detail
 - Lọc theo khoảng ngày
 - Lọc theo từ khóa
 - Lọc theo khoảng số tiền
 - Kết hợp nhiều bộ lọc

Kỹ Thuật Phân Trang

Điểm Nổi Bật về Giải Thuật

- Tìm kiếm mờ (Fuzzy Search - Fuzzy Seach (tìm kiếm "mờ"), hay còn hay được gọi là Approximate Search (tìm kiếm gần đúng))
- Linh hoạt trong việc xử lý đầu vào
- Xử lý ngoại lệ và làm sạch dữ liệu
- Hiệu suất cao nhờ sử dụng Pandas
- Ưu điểm:
 - Xử lý nhanh với DataFrame
 - Tìm kiếm linh hoạt
 - Hỗ trợ nhiều tiêu chí lọc
 - Mở rộng dễ dàng
- Nhược điểm:
 - Phụ thuộc nhiều vào thư viện Pandas
 - Hiệu suất có thể giảm với dữ liệu rất lớn

5.2 Cấu trúc mã nguồn

 _pycache_	update main.py and style.css	last week
 .gitignore	assignment advanced programing	2 weeks ago
 README.md	first commit	2 weeks ago
 chuyen_khoan.csv	assignment advanced programing	2 weeks ago
 index.html	assignment advanced programing	2 weeks ago
 main.py	update main.py and style.css	last week
 requirements.txt	assignment advanced programing	2 weeks ago
 script.js	assignment advanced programing	2 weeks ago
 style.css	update main.py and style.css	last week

Hình 12: Cấu trúc mã nguồn

- `__pycache__`/: Thư mục chứa file bytecode được biên dịch tự động bởi Python.
- `.gitignore`: File khai báo các file/thư mục không được đẩy lên Git.
- `README.md`: Tài liệu mô tả ứng dụng: mục đích, cách cài đặt và sử dụng.
- `chuyen_khoan.csv`: File CSV chứa dữ liệu sao kê các giao dịch.
- `index.html`: Giao diện chính của ứng dụng, xây dựng bằng HTML.
- `main.py`: File chính triển khai backend bằng FastAPI. `requirements.txt`: Danh sách các thư viện cần thiết cho ứng dụng.
- `script.js`: File JavaScript hỗ trợ các chức năng tương tác trên giao diện.
- `style.css`: File CSS định dạng giao diện, cải thiện trải nghiệm người dùng.

Chi tiết các thành phần

- **Thư mục `__pycache__`**: Thư mục này tự động được tạo bởi Python để lưu trữ các file bytecode. Thường được bỏ qua khi đẩy lên Git.
- **File `.gitignore`**: Chứa danh sách các file/thư mục không cần đưa lên Git, ví dụ: `__pycache__`, file môi trường ảo, hoặc các file tạm thời.
- **File `README.md`**: Mô tả chi tiết ứng dụng, cách sử dụng và các bước cài đặt.
- **File `chuyen_khoan.csv`**: File dữ liệu đầu vào chứa thông tin các giao dịch cần kiểm tra, được xử lý bằng thư viện **Pandas**.
- **File `index.html`**: Giao diện chính của ứng dụng, cho phép người dùng tải lên file sao kê hoặc xem kết quả xử lý.
- **File `main.py`**: File chính của ứng dụng:
 - Triển khai các API endpoint bằng **FastAPI**.
 - Sử dụng thư viện **Pandas** để đọc và xử lý file CSV.
 - Trả về kết quả cho giao diện người dùng.
- **File `requirements.txt`**: Danh sách các thư viện cần thiết, ví dụ:

```
fastapi
uvicorn
pandas
```

Người dùng cài đặt bằng lệnh: `pip install -r requirements.txt`.

- **File `script.js`**: Chứa mã JavaScript để xử lý các tương tác trên giao diện, ví dụ: gửi yêu cầu API, cập nhật nội dung trang.
- **File `style.css`**: File CSS định dạng giao diện, giúp cải thiện trải nghiệm người dùng bằng cách tùy chỉnh màu sắc, bố cục, và kiểu chữ.

Hoạt động của ứng dụng

- **Frontend:** Người dùng tải lên file sao kê qua giao diện web (`index.html`). Giao diện được định dạng bằng `style.css`, trong khi `script.js` xử lý các tương tác.
- **Backend:** File `main.py` nhận file từ người dùng, sử dụng **Pandas** để xử lý dữ liệu và trả lại kết quả.
- **Dữ liệu:** File `chuyen_khoan.csv` là đầu vào để kiểm tra các giao dịch.

5.3 các tính năng xây dựng

Tính năng chính

- **Tìm kiếm từ khóa**
 - Một ô nhập liệu (`input`) cho phép người dùng nhập từ khóa để tìm kiếm giao dịch.
 - Nút `Search` để thực hiện tìm kiếm dựa trên từ khóa đã nhập.
- **Bộ lọc nâng cao**
 - **Bộ lọc theo ngày:**
 - * Hai trường nhập liệu ngày (`input`) cho phép người dùng chọn ngày bắt đầu (`Start Date`) và ngày kết thúc (`End Date`).
 - * Giao diện chọn ngày được hỗ trợ bởi thư viện **Flatpickr**.
 - **Bộ lọc theo số tiền:**
 - * Các nút nhanh cho phép người dùng chọn các khoảng giá trị cố định:
 - Dưới 5 triệu.
 - 5-10 triệu.
 - 10-50 triệu.
 - 50-100 triệu.
 - Trên 100 triệu.
 - * Hai trường nhập liệu số (`input`) cho phép người dùng nhập khoảng giá trị tùy chỉnh (`Min amount` và `Max amount`).
- **Bảng kết quả giao dịch**
 - Hiển thị các giao dịch được tìm kiếm hoặc lọc.
 - Bảng bao gồm các cột:
 - * `Date`: Ngày giao dịch.
 - * `Transaction ID`: Mã giao dịch.
 - * `Amount`: Số tiền giao dịch.
 - * `Details`: Thông tin chi tiết.
- **Thông báo trạng thái**
 - Thông báo `Loading` hiển thị khi hệ thống đang xử lý dữ liệu.

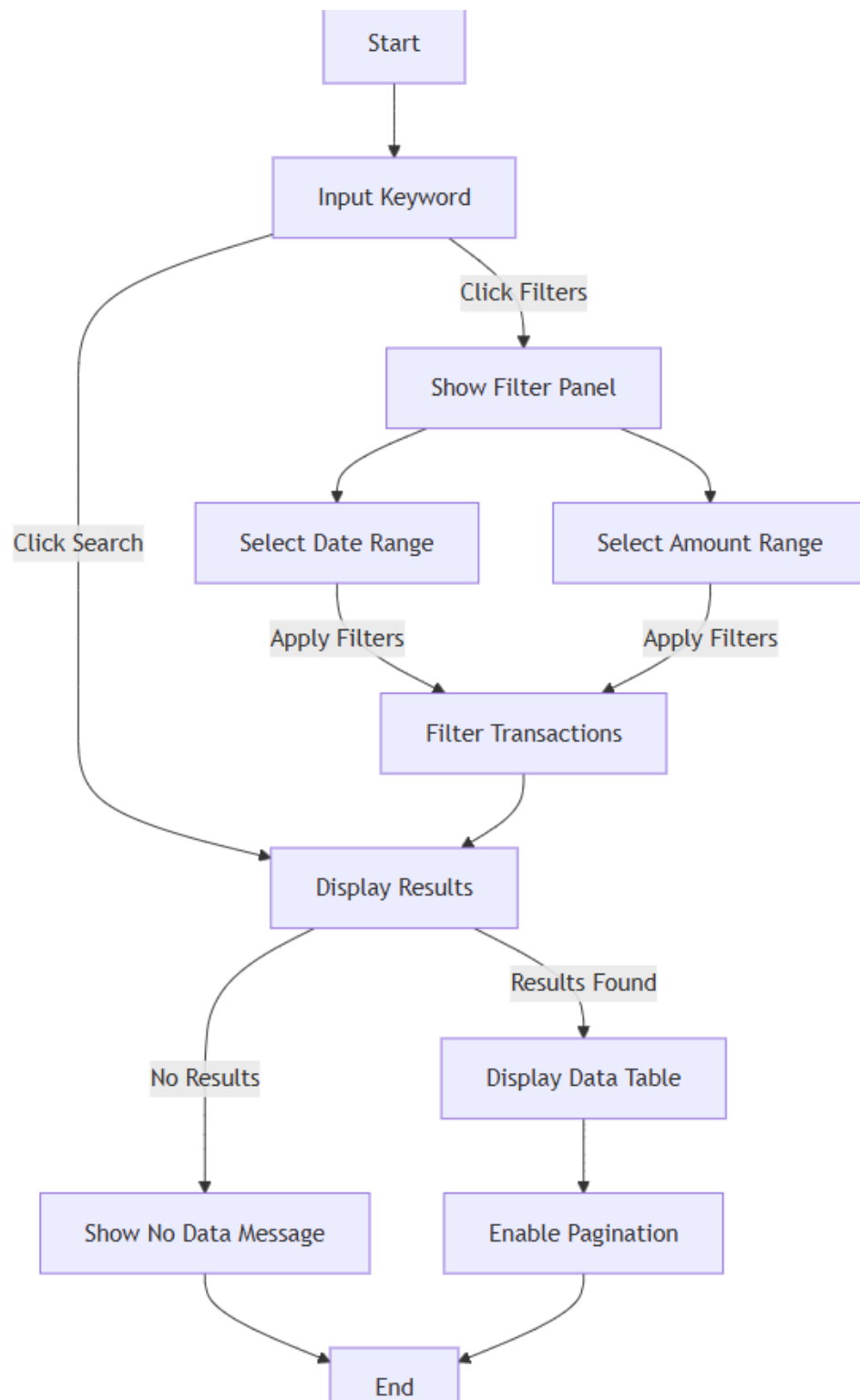


- Thông báo `No transactions found` hiển thị khi không có giao dịch nào phù hợp với tiêu chí tìm kiếm hoặc lọc.

- **Phân trang**

- Cung cấp hệ thống phân trang giúp người dùng dễ dàng duyệt qua các kết quả giao dịch.

5.4 Flowchart của website



Hình 13: Flowchart của ứng dụng

Trang web Transaction Filter bắt đầu bằng việc người dùng truy cập vào giao diện chính. Tại đây, họ có thể nhập từ khóa vào ô tìm kiếm để tìm các giao dịch phù hợp. Sau khi nhập từ khóa, người dùng có thể nhấn nút Search để tìm kiếm trực tiếp hoặc nhấn nút Filters để mở bảng bộ lọc nâng cao. Khi



bảng bộ lọc được hiển thị, người dùng có thể chọn khoảng thời gian bằng cách nhập ngày bắt đầu và kết thúc hoặc chọn khoảng giá trị giao dịch từ các mức có sẵn, như "Dưới 5 triệu", "5-10 triệu", hoặc nhập giá trị tùy chỉnh trong mục Min Amount và Max Amount.

Sau khi thiết lập bộ lọc, người dùng nhấn nút áp dụng để hệ thống xử lý và hiển thị kết quả. Nếu không có giao dịch nào thỏa mãn điều kiện, trang web sẽ hiển thị thông báo "No transactions found". Ngược lại, nếu có kết quả, danh sách các giao dịch sẽ được hiển thị trong bảng với các thông tin như ngày giao dịch, mã giao dịch, số tiền, và chi tiết bổ sung. Nếu số lượng giao dịch lớn, trang web sẽ kích hoạt tính năng phân trang, cho phép người dùng dễ dàng xem các giao dịch khác trên các trang tiếp theo. Khi hoàn tất, người dùng có thể dừng thao tác hoặc thực hiện tìm kiếm mới.

6 Hướng dẫn sử dụng

6.1 Run project by via DockerHub

- Step 1: Pull project

```
docker pull anhkhhoa01010902/kai_t367_statewebv3:tagname
```

- Step 2: Start container from image and Run project

```
docker run -p 8000:8000 anhkhhoa01010902/kai_t367_statewebv3:v1.2
```

6.2 NORMAL WAY RUN PROJECT

6.2.1 Step 1: Clone the repository

```
git clone <repository-url>
cd <repository-folder>
```

6.2.2 Step 2: Build the Docker image

```
docker-compose build
```

6.2.3 Step 3: Start the application

```
docker-compose up
```

6.2.4 Step 4: Access the application:

API: `http://localhost:8000/docs` (Swagger documentation).

Frontend: Open `index.html` in a browser or serve it with a static file server.

6.2.5 Step 5: Stop the application:

```
docker-compose down
```

6.3 Local Development (Without Docker)

Install dependencies: Cài đặt các phụ thuộc bằng lệnh:

```
pip install -r requirements.txt
```

Run the application: `bash unicorn main:app --reload` Access the application:



API: `http://127.0.0.1:8000/docs`

Frontend: Open `index.html` in a browser or serve it with a static file server.

7 Tài liệu tham khảo

- [1] ASOS “<https://www.asos.com/women/>”, Tham khảo front-end, cập nhật lần cuối 25/11/2021
- [2] Dev Ed “<https://www.youtube.com/watch?v=FK4YusHIIj0>”, Hướng dẫn thiết kế website dưới 1 tiếng với Figma, cập nhật lần cuối: 09/03/2019
- [3] Jay “<https://stackoverflow.com/questions/11488960>”, How do I put my website’s logo to be the icon image in browser tabs?, cập nhật lần cuối: 15/07/2012.
- [4] Website chính thức của W3School. Truy cập tại: <https://www.w3schools.com/>.
- [5] Document ngôn ngữ PHP <https://www.php.net/manual/en/>
- [6] Goossens, M., Mittelbach, F., Samarin, A LaTeX Companion, Addison-Wesley, Reading, MA, 1994.
- [7] Kopka, H., Daly P.W., A Guide to LaTeX, Addison-Wesley, Reading, MA, 1999.
- [8] Pan, D., “A Tutorial on MPEG/Audio Compression,” IEEE Multimedia, Vol.2, pp.60- 74, Summer 1998
- [9] Slice bài giảng môn học Lập trình Nâng Cao.