

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



BÁO CÁO BÀI TẬP LỚN
MÔN LẬP TRÌNH NÂNG CAO - CO2039

XÂY DỰNG TRANG WEB TRA CỨU
SAO KÊ TỪ DỮ LIỆU CỦA MTTQ VN

NHÓM 491 – HỌC KỲ 241

GVHD: Lê Đình Thuận

STT	Họ và tên sinh viên	MSSV	Nhiệm vụ	Lớp
1	Ngô Quang Thắng	2114831	Lập trình Front-end	L01
2	Hoàng Gia Bảo	2112855	Làm slide và hỗ trợ Front-end	L01
3	Lê Đình Nghĩa	2212220	Lập trình Front-end	L01
4	Nguyễn Ngọc Thành Đạt	2111013	Lập trình Back-end	L01
5	Hà Thúc Thanh Phong	2212549	Lập trình Back-end	L01
6	Trịnh Vũ Đức Hải	2310886	Viết báo cáo và hỗ trợ làm slide	L01
7	Hà Đình Vinh	2313907	Hỗ trợ viết báo cáo và thuyết trình	L01



Mục lục

1	Yêu cầu chung	3
2	Hướng dẫn sử dụng trang web	3
2.1	Setup BackEnd	4
2.2	Setup FrontEnd	4
3	Giao diện người dùng	5
3.1	Giao diện của trang web	5
3.2	Chuyển đổi chế độ sáng/tối	5
3.3	Ẩn và hiện bộ lọc (filter)	6
3.4	Các chức năng hỗ trợ tìm kiếm	7
4	Giao diện lập trình ứng dụng (API)	7
4.1	Tiền xử lý	7
4.2	Xử lý và lọc dữ liệu	7
5	Kết Luận	8



Lời nói đầu

Sự phát triển mạnh mẽ của công nghệ chỉnh sửa hình ảnh đã mang lại nhiều lợi ích cho nhiều lĩnh vực trong cuộc sống. Bên cạnh đó, nó cũng tạo cơ hội để một số đối tượng chỉnh sửa, cắt ghép hình ảnh nhằm tạo tin giả với nhiều mục đích xấu. Gần đây nhất, mạng xã hội Việt Nam đã dậy sóng với nhiều trường hợp người nổi tiếng bị phát hiện đã lợi dụng việc quyên góp từ thiện để "phông bạt", làm hình ảnh cho bản thân, gây nên những lo ngại về xu hướng sống ảo, thiếu trách nhiệm của một bộ phận người trẻ trong các hoạt động cộng đồng.

Tháng 9 vừa qua, Ủy ban Trung ương Mặt trận Tổ quốc Việt Nam đã công bố hơn 20.000 trang sao kê liên quan đến số tiền quyên góp cho đồng bào bị ảnh hưởng bởi bão số 3 (siêu bão Yagi). Ngay sau đó, cộng đồng mạng đã tích cực "check var" thông tin trong bản sao kê. Dễ dàng thấy được khối lượng thông tin rất lớn nhưng cộng đồng mạng vẫn tích cực tra cứu, cho thấy nhu cầu rất lớn.

Nhằm đáp ứng nhu cầu tra cứu thông tin rất lớn của người Việt Nam, nhóm chúng tôi đã thực hiện đề tài: "Xây dựng trang web tra cứu sao kê từ dữ liệu của Mặt trận Tổ quốc Việt Nam" nhằm tạo thuận lợi trong việc tra cứu, giúp người có nhu cầu dễ dàng, nhanh chóng tra được thông tin sao kê mong muốn dựa trên yêu cầu về số tiền gửi, thời gian gửi hoặc nội dung gửi.

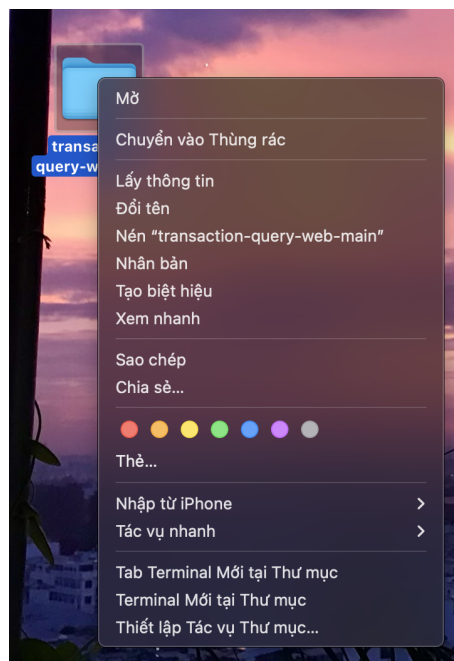
1 Yêu cầu chung

Đề tài: Xây dựng trang web tra cứu sao kê từ dữ liệu của Mặt trận Tổ quốc Việt Nam

- *Mục đích:*
Xây dựng một trang web phục vụ việc tra cứu sao kê, có giao diện thân thiện với người dùng.
- *Kiến thức vận dụng:*
Kiến thức được học từ các môn Kỹ thuật lập trình, Cấu trúc dữ liệu và Giải thuật, Lập trình nâng cao,... Và các kiến thức mở rộng phục vụ việc xây dựng trang web.
- *Dữ liệu được sử dụng:*
Dữ liệu sao kê được Ủy ban Trung ương Mặt trận Tổ quốc Việt Nam công bố dưới dạng file CSV gồm 3 cột: date_time (thời gian chuyển khoản), credit (số tiền), detail (thông tin chi tiết).
- *Dữ liệu nhập:*
Thông tin được người sử dụng cung cấp như tên, số tiền, ngày tháng,...
- *Dữ liệu xuất:*
Thông tin sao kê trùng khớp thông tin được người sử dụng nhập.

2 Hướng dẫn sử dụng trang web

Để sử dụng trang web chạy trên local, bạn cần tiến hành setup frontend và backend. Để đảm bảo, ta sẽ setup FE và BE trên hai terminal riêng biệt, vậy việc đầu tiên cần làm là **mở terminal ở thư mục bằng cách click chuột phải vào folder và chọn "Tab Terminal mới tại thư mục"** đối với MacOS hoặc chọn **"Open in Terminal"** đối với Windows như Hình 1 (hãy làm 2 lần để mở 2 tab terminal)



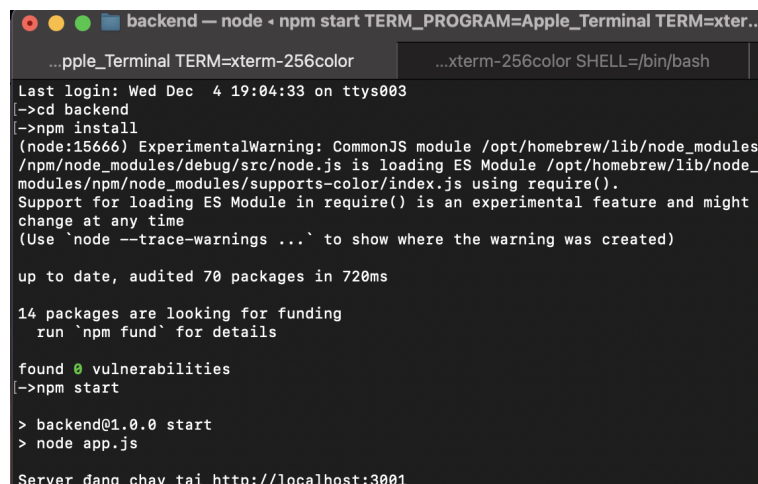
Hình 1: Mở Tab Terminal

2.1 Setup BackEnd

Chọn một tab Terminal để tiến hành setup BackEnd, tại tab này nhập lần lượt ba dòng lệnh sau (nhập riêng biệt):

- `cd backend`
- `npm install`
- `npm start`

Terminal hiển thị như hình 3 là setup BackEnd của bạn đã thành công:



```
backend — node - npm start TERM_PROGRAM=Apple_Terminal TERM=xterm-256color
...xterm-256color SHELL=/bin/bash
Last login: Wed Dec 4 19:04:33 on ttys003
[~]$ cd backend
[~]$ npm install
(node:15666) ExperimentalWarning: CommonJS module /opt/homebrew/lib/node_modules/npm/node_modules/debug/src/node.js is loading ES Module /opt/homebrew/lib/node_modules/npm/node_modules/supports-color/index.js using require().
Support for loading ES Module in require() is an experimental feature and might change at any time
(Use `node --trace-warnings ...` to show where the warning was created)

up to date, audited 70 packages in 720ms

14 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
[~]$ npm start
> backend@1.0.0 start
> node app.js

Server đang chạy tại http://localhost:3001
```

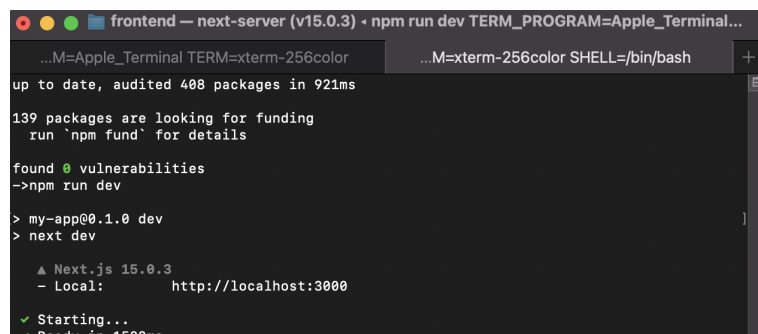
Hình 2: Setup BackEnd

2.2 Setup FrontEnd

Qua tab Terminal còn lại và tiến hành setup FrontEnd tương tự như setup BackEnd, lưu ý là **phải setup BackEnd trước khi tiến hành bước này**:

- `cd frontend`
- `npm install`
- `npm run dev`

Terminal hiển thị như hình 3 là setup FrontEnd của bạn đã thành công



```
frontend — next-server (v15.0.3) - npm run dev TERM_PROGRAM=Apple_Terminal...
...M=Apple_Terminal TERM=xterm-256color
...M=xterm-256color SHELL=/bin/bash
up to date, audited 408 packages in 921ms

139 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
[~]$ npm run dev
> my-app@0.1.0 dev
> next dev

  ▲ Next.js 15.0.3
  - Local:      http://localhost:3000

✓ Starting...
✓ Ready in 1523ms
```

Hình 3: Setup FrontEnd

Sau khi Setup thành công, ta sẽ truy cập vào localhost:3000 để tiến hành tra cứu, lưu ý: trong quá trình tra cứu và sử dụng không nên tắt các tab Terminal vừa thực thi.

3 Giao diện người dùng

3.1 Giao diện của trang web

Giao diện người dùng (User Interface) là một công cụ để người dùng giao tiếp và lấy được thứ mình cần, đây là một "thông dịch viên" giúp gửi yêu cầu của người dùng tới web. Sau khi thực hiện các bước Setup được giới thiệu ở mục trên, đây là giao diện trang web mà người dùng thấy:



Hình 4: Giao diện trang web ở chế độ tối

Giao diện được thiết kế đơn giản và có một vài tính năng cơ bản như **ẩn/hiện bộ lọc (filter)** và **chuyển đổi qua lại giữa chế độ tối (Dark Mode) với chế độ sáng (Light Mode)** để dùng với một cú nhấp chuột, giúp cho người dùng có cảm giác dễ chịu khi trải nghiệm.

3.2 Chuyển đổi chế độ sáng/tối

Chỉ với một cú nhấp, giao diện sẽ chuyển từ chế độ tối (hình 4) qua chế độ sáng (hình 6) hoặc ngược lại bằng **nút bấm ở góc trên bên phải trang web** (hình 5)



Hình 5: Nút bấm điều chỉnh chế độ sáng/tối



Tra cứu dữ liệu "Sao Kê" của MTTQ VN

Toggle: * [Light Mode] ☾

Tìm Kiếm

Hide Filters ^

Start Date: dd/mm/yyyy

End Date: dd/mm/yyyy

Min Credit: \$

Max Credit: \$

Search Details: Search transaction details...

Tìm Kiếm Giao Dịch

Hình 6: Giao diện trang web ở chế độ sáng

Để thực hiện cơ chế này, ban đầu ta chỉ cần định nghĩa giao diện tối (Dark Mode) và giao diện sáng (Light Mode) trên CSS và sử dụng JavaScript để thực hiện việc chuyển đổi, JavaScript sẽ gỡ giao diện tối (Dark Mode) và thay thế bằng sáng (Light Mode) và ngược lại khi nút (tạo bởi HTML) được click vào.

3.3 Ẩn và hiện bộ lọc (filter)

Sau khi tiến hành tra cứu và ra kết quả, nhiều lúc ta muốn các thanh tìm kiếm được thu nhỏ lại để đỡ "chướng mắt", và đó là lý do chức năng "Ẩn/hiện bộ lọc" (Hide Filters) ra đời.

Khi muốn ẩn các thanh tìm kiếm, ta click chuột vào "Hide Filters" và ngược lại, ta click chuột vào "Show Filters" để hiện ra các thanh tìm kiếm.

Toggle: * [Dark Mode] ☾

Tìm Kiếm

Show Filters v

Hình 7: Giao diện sau khi các thanh tìm kiếm được ẩn

Để làm được điều này, ta chỉ cần thay đổi thuộc tính "display" của các filter khi nút được click vào.

3.4 Các chức năng hỗ trợ tìm kiếm

Trang web cũng cung cấp một vài chức năng cơ bản cho việc tìm kiếm trở nên thuận lợi:

- *Chọn Start Date (ngày bắt đầu) và End Date (ngày kết thúc):*
Được tích hợp Data Picker để người dùng có thể chọn ngày tháng năm cụ thể một cách dễ dàng mà không cần phải "nhập chày" dữ liệu, điều này cũng giúp cho việc xử lý dữ liệu dễ hơn vì có thể người dùng sẽ nhập không đúng "Data Format" mà Developer xử lý.
- *Nhập Min Credit (số tiền nhỏ nhất) và Max Credit (số tiền lớn nhất):*
Ngoài việc nhập số tiền, còn có thêm hai nút ↑ và ↓ ở bên phải nơi nhập để tăng hoặc giảm một đơn vị. Những người dùng đang ở tư thế ngả lưng và không muốn với tới các nút số ở trên cùng bàn phím có thể sử dụng Arrow Key ở trên bàn phím để thực hiện tăng giảm (không khuyến nghị).
Việc cho phép tìm kiếm bằng khoảng thời gian (Start Date và End Date) hay khoảng tiền (Min Credit và Max Credit) rất hữu ích khi người dùng không thể nhớ được các thông tin chi tiết của tài khoản mà mình mong muốn.
- *Nhập Search Details (thông tin chi tiết):*
Chức năng cực kỳ cần thiết khi người dùng muốn tìm kiếm bằng các thông tin chi tiết khác như: tên, thông tin chuyển khoản,... Giúp người dùng nhanh chóng có được kết quả mong muốn.
- *Tải kết quả tìm kiếm dưới dạng file CSV:*
Kết quả sau khi tìm kiếm sẽ có thể được tải về dưới dạng file CSV trong phạm vi tìm kiếm mong muốn.

4 Giao diện lập trình ứng dụng (API)

API được xem như công cụ giao tiếp giữa web và server. Ở đây, ta chỉ cần lọc data theo dữ kiện mà người dùng nhập và trả về.

4.1 Tiền xử lý

Dữ liệu ban đầu sẽ được chuyển thành list các Dictionary có các key là "date", "credit", "detail",.... và các value là các giá trị tương ứng.

Ta sẽ khởi tạo hai HashMap gồm DataByAmount (để lưu theo cặp Key - Value là Credit - Data) và DataByDetail (để lưu theo cặp Detail - Data) và Detail được chuyển thành chữ viết thường hết để hỗ trợ việc tìm kiếm.

Sau khi tải tất cả dữ liệu vào Map, ta sẽ tiến hành sort DataByAmount theo key của nó, việc làm này sẽ giúp tiết kiệm thời gian lọc dữ liệu trong khoảng [MinCredit;MaxCredit]

4.2 Xử lý và lọc dữ liệu

Dữ liệu nhập từ UI sẽ được chuyển thành các biến: StartDate, EndDate, MinCredit, MaxCredit và Detail.

Nếu MinCredit hoặc MaxCredit được nhập vào, tức là người dùng có nhu cầu tìm kiếm theo số tiền trong khoảng đó. Ta sẽ lấy dữ liệu từ DataByAmount và đưa vào kết quả (Result) nếu dữ liệu có key thỏa: $\text{MinCredit} \leq \text{key} \leq \text{MaxCredit}$ (DataByAmount đã được sắp xếp dựa trên credit từ trước). Trường hợp nếu số tiền không được nhập vào, ta lấy toàn bộ dữ liệu từ DataByDetail đưa vào kết quả (Result).



Nếu StartDate và EndDate được nhập, dữ liệu từ kết quả (Result) sau đó sẽ được so sánh với StartDate và EndDate để tiến hành lọc, dữ liệu sau khi lọc tiếp tục đưa vào Kết quả (Result).

Nếu Detail được nhập, ta sẽ chuyển Detail được nhập về chữ thường và so sánh với Detail của từng dữ liệu trong kết quả (Result), nếu Detail của dữ liệu trong kết quả (Result) không chứa Detail được nhập thì ta sẽ tiến hành loại bỏ.

Sau đó, ta sẽ trả về kết quả (Result) hiển thị trên trang Web.

5 Kết Luận

Với giao diện đơn giản, dễ hiểu, tích hợp các chức năng hỗ trợ tìm kiếm các giao dịch với những tiêu chí mà người dùng mong muốn, trang web tra cứu sao kê là một công cụ cực kỳ hữu ích cho người dùng. Trang web đã đáp ứng nhu cầu tìm kiếm thông tin phục vụ tổ chức những hành vi xấu, góp phần xây dựng một xã hội văn minh.

Qua bài tập lớn, nhóm đã có cơ hội vận dụng những kiến thức đã được học và tìm hiểu được những kiến thức mới trong xây dựng trang web. Đây là một đề tài thú vị khi đã giúp sinh viên vận dụng các kiến thức để tạo ra một sản phẩm có tác dụng gần gũi trong cuộc sống, có thể dễ dàng tái sử dụng với các bộ dữ liệu khác.