

Thành viên nhóm: Đỗ Tường Lâm - MSV: 1551060732

Nguyễn Việt Hùng - MSV:

BÁO CÁO BÀI TẬP LỚN MÔN HỌC HỆ THỐNG THÔNG TIN ĐỊA LÝ

Đề tài: website du lịch Hà Nội

1. Phân tích đề tài:

- Website chúng em xây dựng nên nhằm mục đích giới thiệu các lịch sử các quận huyện của thành phố Hà Nội; cùng với đó là giới thiệu các địa điểm du lịch trong và gần thành phố cùng với các món ăn truyền thống của Hà Nội mà khách du lịch trong và ngoài nước nên thưởng thức khi đến đây; ngoài ra còn có thông tin của sân bay, bệnh viện và các khách sạn - nhà nghỉ nổi tiếng của Hà Nội.

- Các chức năng của website:

+ Khi người dùng truy cập vào website, người dùng sẽ thấy được các quận huyện của Hà Nội được khoanh vùng và được tô màu khác với bản đồ phía sau. Khi người dùng nhấn vào 1 quận huyện nào đó, vùng đó sẽ được tô màu khác đi và hiện thông tin của khu vực đó ở khung info ở góc trái dưới. Thông tin bao gồm kích thước của quận huyện và link dẫn đến trang wiki tương ứng để đọc thông tin lịch sử, dân số, đường xá và các thông tin khác. Các khu vực có thể đổi màu tùy ý bằng cách chọn trong bảng màu nằm dưới khung info.

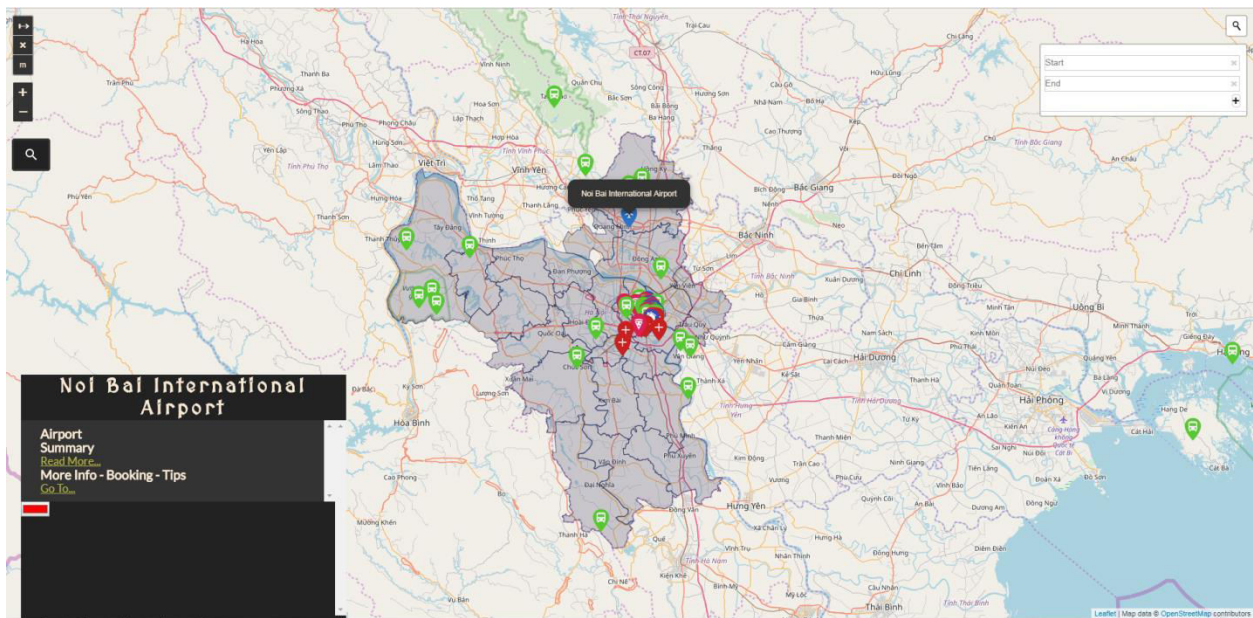
+ Ngoài ra bản đồ còn hiện các map pins riêng biệt cho từng địa điểm được bọn em cung cấp. Các địa điểm này được bọn em chọn lọc ra từ các nguồn thông tin uy tín trên mạng như booking.com, vntrip.vn, foody.vn, maps.google.com, ... và 1 vài nguồn ngoài khác cùng với kinh nghiệm sống. Các map pins được phân biệt bởi màu và các icon khác nhau tương ứng với thể loại của map pin đó. Bao gồm: sân bay, bệnh viện, địa điểm tham quan du lịch, địa điểm ăn ngon, khách sạn - nhà nghỉ. Khi bấm vào 1 map pin, tên của địa điểm sẽ được bật lên và thông tin của địa điểm đó sẽ được hiện ra ở khung info ở góc trái dưới. Thông tin bao gồm: loại địa điểm, trang chủ của địa điểm đó hoặc wiki hoặc bài viết giới thiệu tổng quát địa điểm, trang web đặt đồ ăn tương ứng, trang web đặt

phòng khách sạn tương ứng, hướng dẫn thăm quan cùng các tips khi đi thăm quan,... tùy thuộc vào loại địa điểm.

+ Ngoài ra trang web còn có các chức năng phụ như tìm đường đi ngắn nhất, tìm kiếm các địa điểm được cung cấp sẵn, tìm kiếm các địa điểm ở khắp nơi trên thế giới, đo khoảng cách đường chim bay giữa các địa điểm.

- Sau đây là giao diện của website:

+ Giao diện tổng quát:



+ Các map pins tương ứng:



- Sân bay



- Địa điểm ăn uống



- Bệnh viện

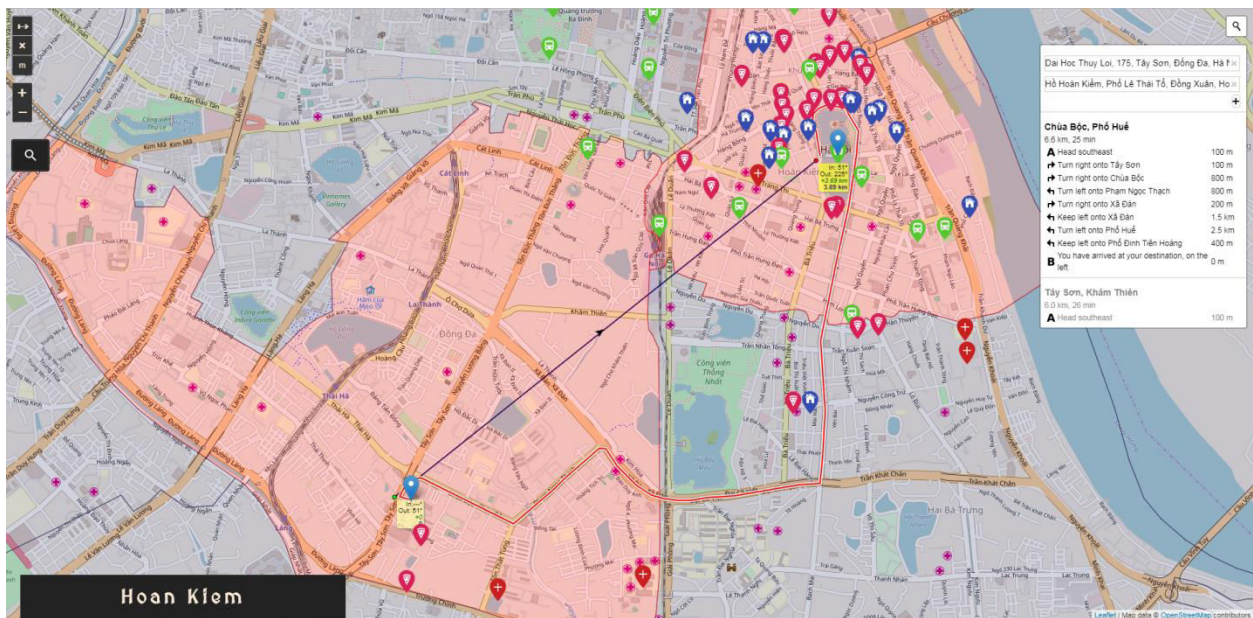


- Khách sạn - nhà nghỉ

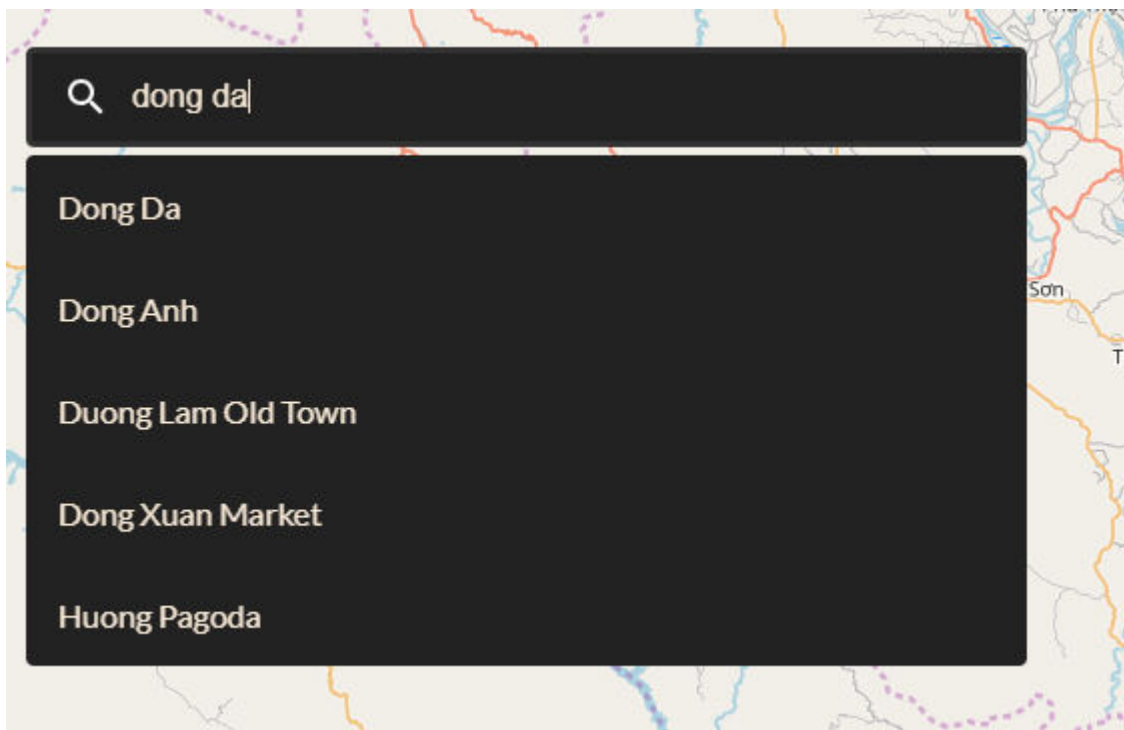


- Địa điểm thăm quan du lịch

+ Chức năng tìm đường đi ngắn nhất và đo khoảng cách bằng đường chim bay:



+ Chức năng tìm kiếm các địa điểm được cung cấp sẵn:



- Các chức năng của backend:

+ Bọn em viết 1 trang back-end nằm riêng biệt với phần front-end.

Back-end của bọn em chia ra làm 2 phần đó là API và phân quản trị cơ sở dữ liệu.

API bao gồm các đường dẫn được viết bằng nodejs có nhiệm vụ trả về dữ liệu

JSON tương ứng khi được truy cập vào. Phần quản trị cơ sở dữ liệu có nhiệm vụ

hiện thị thông tin về cơ sở dữ liệu, thêm sửa xóa cơ sở dữ liệu.

+ Ví dụ: API chứa dữ liệu JSON về khoanh vùng các tỉnh của Hà Nội

+ Phần quản trị cơ sở dữ liệu sẽ bao gồm 1 trang đăng nhập, khi người

dùng nhập đúng username và password sẽ được chuyển đến trang chứa thông tin

về các quận huyện của Hà Nội, dưới cùng là nút thêm địa điểm. Ở trang thêm địa

điêm người dùng có thể thêm sửa xóa bảng dữ liệu chứa thông tin của các địa

điêm. Sau đây là phân quản trị cơ sở dữ liệu:

Name	Type	Latitude	Longitude	Summary	Website		Add
Noi Bai International Airport	Airport	21.2187149	105.8041709	https://en.wikipedia.org/wiki/Noi_Bai_International_Airport	http://noibaiairport.vn/Home/default.aspx	<div>Sup</div> <div>Xoá</div>	
Bach Mai Hospital	Hospital	21.002909	105.839734	https://en.wikipedia.org/wiki/Bach_Mai_Hospital	http://bachmai.gov.vn/	<div>Sup</div> <div>Xoá</div>	
Viet Duc Hospital	Hospital	21.027531	105.84734	https://en.wikipedia.org/wiki/Vietnam_Viet_Duc_Hospital	http://benhvientietduc.org/	<div>Sup</div> <div>Xoá</div>	
103 Military Hospital	Hospital	20.966964	105.790122	https://vi.wikipedia.org/wiki/B%E1%BB%87...	http://www.benhvien103.vn/	<div>Sup</div> <div>Xoá</div>	
108 Center Military Hospital	Hospital	21.018087	105.860998	https://vi.wikipedia.org/wiki/B%E1%BB%87...	http://benhvien108.vn/	<div>Sup</div> <div>Xoá</div>	
Medical University Hospital	Hospital	21.002119	105.83017	https://vi.wikipedia.org/wiki/T%E1%BB%87...	http://benhviendahoc/hanoi.com/	<div>Sup</div> <div>Xoá</div>	
National University Hospital	Hospital	20.990916	105.796532	https://vi.wikipedia.org/wiki/C%E1%BB%87...	http://benhviendahoc/hanoi.com/	<div>Sup</div> <div>Xoá</div>	
Vinmec International General Hospital	Hospital	20.996355	105.86691	https://wikimedia.org/wiki/Vinmec_International_General_Hospital	https://vinmec.com/	<div>Sup</div> <div>Xoá</div>	
Huu Nghi Hospital	Hospital	21.016657	105.861118	https://vi.wikipedia.org/wiki/B%E1%BB%87...	http://huunghihospital.vn/index.php?lang=vi	<div>Sup</div> <div>Xoá</div>	
Hoan Kiem Lake	Tourist Attraction	21.028959	105.85261	https://vi.wikipedia.org/wiki/H%E1%BB%87...	http://www.vietfuntravel.com.vn/blog/gioi-thieu-hoan-kiem-lake	<div>Sup</div> <div>Xoá</div>	
Hanoi Opera House	Tourist Attraction	21.024161	105.857859	https://vi.wikipedia.org/wiki/H%E1%BB%87...	http://www.hanoiopera.com.vn/	<div>Sup</div> <div>Xoá</div>	
The Big Church	Tourist Attraction	21.028615	105.848891	https://vi.wikipedia.org/wiki/H%E1%BB%87...	https://tonggiaophanhanoi.org/	<div>Sup</div> <div>Xoá</div>	
Ho Chi Minh Mausoleum	Tourist Attraction	21.037972	105.833812	https://vi.wikipedia.org/wiki/C%E1%BB%87...	http://www.bqlang.gov.vn/	<div>Sup</div> <div>Xoá</div>	
Hanoi Gas Station	Tourist Attraction	21.024453	105.840812	https://vi.wikipedia.org/wiki/G%E1%BB%87...	http://www.gahanoi.com.vn/	<div>Sup</div> <div>Xoá</div>	
Ly Thai To Monument	Tourist Attraction	21.027453	105.854154	https://vi.wikipedia.org/wiki/V%E1%BB%87...	https://vualythaito.com/	<div>Sup</div> <div>Xoá</div>	
Long Bien Bridge	Tourist Attraction	21.044394	105.861455	https://vi.wikipedia.org/wiki/C%E1%BB%87...	http://www.vietfuntravel.com.vn/blog/gioi-thieu-long-bien-bridge	<div>Sup</div> <div>Xoá</div>	
Vietnam National Museum of History	Tourist Attraction	21.024319	105.859656	https://vi.wikipedia.org/wiki/B%E1%BB%87...	http://www.bactanglichsu.vn/	<div>Sup</div> <div>Xoá</div>	
Dong Xuan Market	Tourist Attraction	21.038066	105.849712	https://vi.wikipedia.org/wiki/C%E1%BB%87...	http://www.chodongxuan.org/	<div>Sup</div> <div>Xoá</div>	

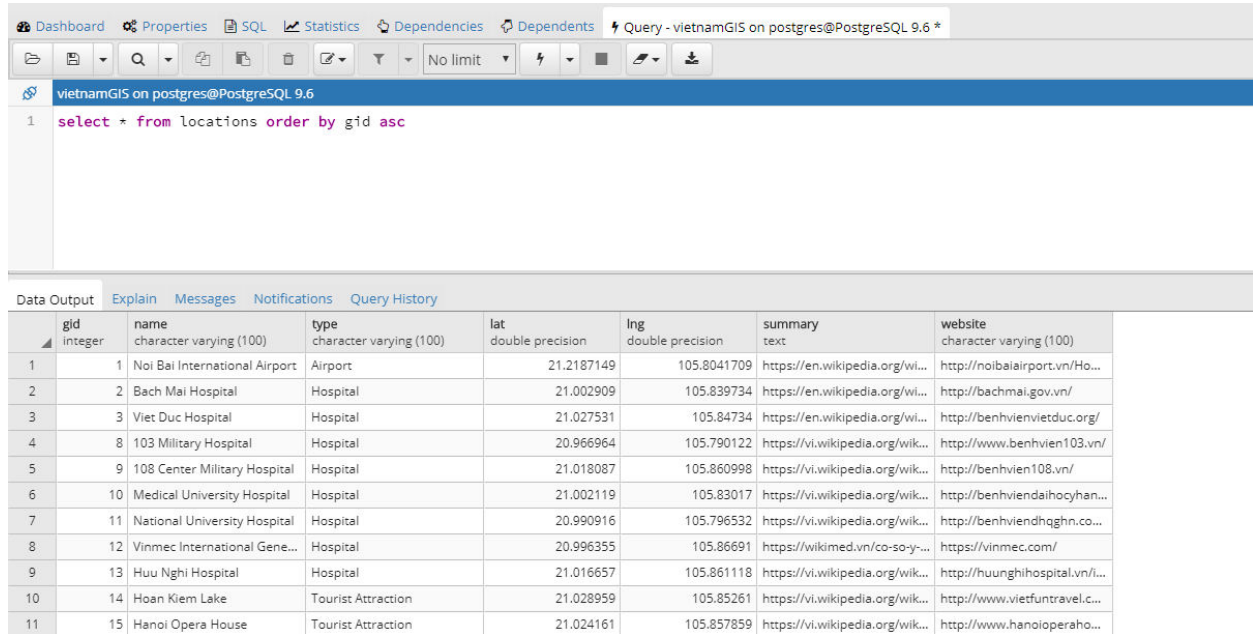
- Cuối cùng em xin nói đến phần cơ sở dữ liệu của chúng em: bọn em sử dụng PostgreSQL cùng với add-on là PostGIS.

+ Bảng districts bọn em download từ trên mạng về do bên trong đó có chứa dữ liệu geom bao gồm vành đai của các quận huyện Hà Nội. Bảng có rất nhiều trường mặc định tuy nhiên các trường bọn em sử dụng trong code là: gid chứa id riêng biệt của quận huyện đó, varname_2 chứa tên quận huyện, geom chứa thông tin về vành đai của quận huyện và summary chứa trang wiki tương ứng.

VietnamGIS on postgres@PostgreSQL 9.6											
Query - vietnamGIS on postgres@PostgreSQL 9.6 *											
1 select * from districts											
Data Output	Explain	Messages	Notifications	Query History							
gid	name	type	geom	summary	varname_2	nl_name_2	engtype_2	type_2	name_2	id_2	id_1
23	Hà Nội	Huyện	010600002...	https://vi.wikipedia.org/wiki/H%E1%BB%87...	Thanh Oai	[null]	District	Huyện	Thanh Oai	238	23
23	Hà Nội	Huyện	010600002...	https://vi.wikipedia.org/wiki/U%E1%BB%87...	Ứng Hòa	[null]	District	Huyện	Ứng Hòa	211	23
23	Hà Nội	Huyện	010600002...	https://vi.wikipedia.org/wiki/D%E1%BB%87...	Đông Anh	[null]	District	Huyện	Đông Anh	212	23
23	Hà Nội	Huyện	010600002...	https://vi.wikipedia.org/wiki/D%E1%BB%87...	Đan Phượng	[null]	District	Huyện	Đan Phượng	214	23
23	Hà Nội	Huyện	010600002...	https://vi.wikipedia.org/wiki/G%E1%BB%87...	Gia Lâm	[null]	District	Huyện	Gia Lâm	219	23
23	Hà Nội	Huyện	010600002...	https://vi.wikipedia.org/wiki/H%E1%BB%87...	Hai Bà Trưng	[null]	District	Huyện	Hai Bà Trưng	221	23
23	Hà Nội	Huyện	010600002...	https://vi.wikipedia.org/wiki/H%E1%BB%87...	Hoài Đức	[null]	District	Huyện	Hoài Đức	222	23
23	Hà Nội	Quận	010600002...	https://vi.wikipedia.org/wiki/H%E1%BB%87...	Hoàng Mai	[null]	Urban District	Quận	Hoàng Mai	224	23
23	Hà Nội	Huyện	010600002...	https://vi.wikipedia.org/wiki/M%E1%BB%87...	Mê Linh	[null]	District	Huyện	Mê Linh	229	23
23	Hà Nội	Huyện	010600002...	https://vi.wikipedia.org/wiki/P%E1%BB%87...	Phù Xuyên	[null]	District	Huyện	Phù Xuyên	230	23
23	Hà Nội	Huyện	010600002...	https://vi.wikipedia.org/wiki/P%E1%BB%87...	Phúc Thọ	[null]	District	Huyện	Phúc Thọ	231	23
23	Hà Nội	Huyện	010600002...	https://vi.wikipedia.org/wiki/T%E1%BB%87...	Từ Liêm	[null]	District	Huyện	Từ Liêm	236	23

+ Bảng locations chứa thông tin của các địa điểm, bao gồm: gid chứa id riêng biệt của các địa điểm, name chứa tên của địa điểm, type chứa loại địa điểm, lat và lng chứa thông tin kinh độ vĩ độ của địa điểm đó, summary chứa trang

wiki giới thiệu hoặc bài viết giới thiệu địa điểm, website chứa website tương ứng hoặc website book phòng hoặc website ship đồ ăn hoặc hướng dẫn thăm quan du lịch tương ứng với từng loại địa điểm.



The screenshot shows a PostgreSQL query editor interface. The query being executed is: `select * from locations order by gid asc`. The results are displayed in a table with 11 rows and 8 columns.

	gid	name	type	lat	lng	summary	website
	integer	character varying (100)	character varying (100)	double precision	double precision	text	character varying (100)
1	1	Noi Bai International Airport	Airport	21.2187149	105.8041709	https://en.wikipedia.org/wiki/Noi_Bai_International_Airport	http://noibaiairport.vn/Ho...
2	2	Bach Mai Hospital	Hospital	21.002909	105.839734	https://en.wikipedia.org/wiki/Bach_Mai_Hospital	http://bachmai.gov.vn/
3	3	Viet Duc Hospital	Hospital	21.027531	105.84734	https://en.wikipedia.org/wiki/Viet_Duc_Hospital	http://benhvienvietduc.org/
4	8	103 Military Hospital	Hospital	20.966964	105.790122	https://vi.wikipedia.org/wiki/103_Military_Hospital	http://www.benhvien103.vn/
5	9	108 Center Military Hospital	Hospital	21.018087	105.860998	https://vi.wikipedia.org/wiki/108_Center_Military_Hospital	http://benhvien108.vn/
6	10	Medical University Hospital	Hospital	21.002119	105.83017	https://vi.wikipedia.org/wiki/Medical_University_Hospital	http://benhviendaihocynhan...
7	11	National University Hospital	Hospital	20.990916	105.796532	https://vi.wikipedia.org/wiki/National_University_Hospital	http://benhviendhqh.vn...
8	12	Vinmec International Gene...	Hospital	20.996355	105.86691	https://wikimed.vn/co-so-y-...	https://vinmec.com/
9	13	Huu Nghi Hospital	Hospital	21.016657	105.861118	https://vi.wikipedia.org/wiki/Huu_Nghi_Hospital	http://huongnghi-hospital.vn/...
10	14	Hoan Kiem Lake	Tourist Attraction	21.028959	105.85261	https://vi.wikipedia.org/wiki/Hoan_Kiem_Lake	http://www.vietfuntravel.c...
11	15	Hanoi Opera House	Tourist Attraction	21.024161	105.857859	https://vi.wikipedia.org/wiki/Hanoi_Opera_House	http://www.hanoiopera.ho...

+ Cuối cùng là bảng users chứa thông tin đăng nhập của người dùng.

2. Phân tích code:

- Đầu tiên em viết folder server và folder views chứa các file backend của website. File đầu tiên trong folder server là index.js. File này em sử dụng library là Koa để viết các middleware dùng để lấy response time và các lỗi xảy ra trong quá trình truy cập vào server và viết vào file tạo bởi module logger của Winston. Ngoài ra em còn sử dụng package Body-parser để lấy thông tin được POST từ các form đăng nhập - thêm sửa dữ liệu. Và library Pg để đăng nhập vào PostgreSQL, từ đó dùng library ExpressJS cùng package EJS để và render các trang đăng nhập - thêm sửa xóa database. Ngoài ra server API sẽ được chạy ở port 5000 còn trang quản trị cơ sở dữ liệu được chạy ở port 3000.

- File logger.js dùng package Winston và path để viết log.

- File database.js sử dụng library Pg để kết nối vào database và truy xuất các dữ liệu tương ứng ra. Sau đó nó truyền dữ liệu này cho file api.js. File này sử dụng

library Koa-router để viết các trang trả lại dữ liệu JSON tương ứng của các quận huyện và địa điểm khi người dùng dùng AJAX gọi đến. Ngoài ra nó còn sử dụng Koa-static để render ra file SVG chứa các icon tương ứng của các địa điểm, các file SVG được chứa trong folder public.

- Folder views ở ngoài chứa các file .ejs để render các trang HTML để Express truyền dữ liệu đến.

- Xong phần back-end giờ em xin nói đến front-end.

- Folder public ở ngoài cùng chứa các file HTML, CSS, JS của trang web. File HTML chỉ chứa các đường dẫn tới các file CSS và JS của các thư viện cần thiết, và ngoài ra là 1 div trống để chứa tất cả nội dung của trang web. Phần CSS em sử dụng SASS để viết các file SCSS của từng thành phần và sau đó dùng package để import nó vào file CSS tổng quát. Phần JS em sử dụng webpack để convert ES5-6-7-8 về với vanilla Javascript và viết vào file JS tổng quát.

- Folder app dùng để chứa tất cả code trong quá trình viết của trang web và sau đó em dùng các package như trên đã nói để viết code vào các file tổng quát. Trong này bao gồm main1.js là file JS chính, main.scss là file CSS được viết bằng SASS, _variables.scss chứa các biến CSS, folder services chứa code API và search service, folder component chứa các component của trang web.

- Đầu tiên là file component.js trong folder component. File này sẽ export ra 1 class là Component là class gốc để các component khác extends. Component này bao gồm placeholderId để chứa id của div cần select, props để chứa các thông tin của component, template để chứa HTML chèn vào div tương ứng. Trong hàm tạo của Component ta có: componentElem để select div tương ứng và load HTML từ template vào, sau đó lấy tất cả nội dung của các ref (ref là 1 attribute được viết tắt chức className của ReactJS) và bỏ vào 1 object là refs. Sau đó là hàm createEvents để addEventListener vào ref tương ứng khi có event xảy ra. Cuối cùng là hàm triggerEvent để tạo event và lưu thông tin cần truyền vào hàm khác khi event xảy ra.

- Trước khi nói đến component tiếp theo là InfoPanel, em sẽ nói đến api.js trong folder services. File này em sử dụng AJAX, cụ thể là package axios để lấy JSON từ server và trả lại các thông tin tương ứng cho front-end.

- Component Infopanel được extends từ Component và nó dùng để hiển thị thông tin của thứ mình muốn xem. Đầu tiên component này lấy dữ liệu từ file api.js và lưu vào biến api. Tiếp theo em addEventListener vào ref có giá trị là title và toggle class "info-active" mỗi khi bấm vào Infopanel, ban đầu Infopanel được dịch chuyển xuống ra ngoài màn hình 1 đoạn và khi toggle class này thì nó sẽ trở lại vị trí ban đầu, hiển thị đầy đủ thông tin. Hàm showInfo() để load HTML vào Infopanel tùy vào thứ mình muốn xem thông tin là quận huyện hay địa điểm. Nếu nó là quận huyện thì hàm getDistrictDetails() sẽ được thực thi và hiển thị thông tin tương ứng được lấy từ api getAllDistrictDetails. Còn nếu nó là địa điểm thì hàm getLocationDetail sẽ được thực thi và nó lấy thông tin từ api getLocationSummary.

- Component Map cũng được extends từ Component và nó dùng để hiển thị bản đồ cùng các chức năng tương ứng.

+ Đầu tiên là thuộc tính map, thuộc tính này sử dụng lệnh L.map để render map có tâm là tâm của Hà Nội, độ zoom và TileLayer là openstreetmap. Ngoài ra còn có plugin Polyline Measure tích hợp vào để có chức năng đo khoảng cách. Và còn có plugin Leaflet Routing Machine để phục vụ chức năng tìm kiếm đường đi ngắn nhất, plugin này sử dụng plugin Leaflet Control Geocoder để phục vụ việc tìm kiếm, và em sử dụng thêm server Mapbox để việc tìm kiếm nhanh hơn.

+ Hàm addLocationGeojson() để vẽ các map pins tương ứng với các địa điểm. Mỗi layer là các loại địa điểm khác nhau và em sử dụng lệnh L.geoJSON để lấy dữ liệu JSON từ api truyền về và vẽ, em sử dụng thuộc tính pointToLayer của leaflet để trả lại 1 map pin tại vị trí đó vs icon tương ứng đã lấy từ server. Sau đó là thuộc tính onEachFeature để gán event vào khi mình bấm vào map pin.

+ Hàm onEachLocation() được gọi khi người dùng bấm vào map pin và nó sẽ hiển thị tên của map pin bên trên khi mình bấm vào, và khi bấm vào sẽ triggerEvent có tên là locationSelected, event này truyền dữ liệu của map pin vào Infopanel.

+ Hàm addDistrictGeojson() tương tự hàm addLocationGeojson() sẽ sử dụng lệnh L.geoJSON để vẽ khu vực quận huyện ra và add vào layer districts. Layer này sẽ là các mảnh thể hiện diện tích của quận huyện có màu xanh nước biển mờ. Và cũng như trên em sử dụng thuộc tính onEachFeature để gán event vào khi mình bấm vào.

+ Hàm `onEachDistrict()` được gọi khi người dùng bấm vào khu vực quận huyện và đổi màu của quận đó cũng như hiển thị thông tin ở Infopanel. Hàm `changeColor` để đổi màu layer tùy thuộc vào màu đang được chọn trong bảng màu.

+ Hàm `toggleLayer()` để bật layer đó lên.

+ Hàm `selectLocation()` để phục vụ cho việc tìm kiếm và hiển thị thông tin. Hàm này tìm trong tất cả các layer của map và trả lại layer có id trùng với id thứ mà người dùng tìm kiếm. Nếu layer đó có kiểu geometry là Point thì sử dụng lệnh `map.flyTo` để zoom tới map pin đó và lệnh `fireEvent` để click vào map pin đó. Tương tự nếu nó là quận huyện thì `flyToBounds` và click.

- Trước khi nói đến SearchBar em xin nói đến service cuối cùng là `search.js`. File này em sử dụng library FuseJS để tạo ra 1 Fuzzy Search với thông tin là tất cả những layers của map. Đầu tiên là cài đặt 1 vài options của FuseJS, `searchbase` ban đầu là 1 mảng rỗng và Fuse ban đầu cũng rỗng. Sau đó hàm `addGeoJsonItems()` để truyền các layers vào trong `searchbase` và bắt đầu thực hiện tìm kiếm trên `searchbase` này. Và hàm `search()` để trả lại nội dung đang search.

- Search Component lấy data từ `searchService` bên `search.js` và `addEventListener` `keyup` để lấy dữ liệu từ khung input truyền vào hàm `search()`.

+ Hàm `onSearch()` sẽ gọi hàm `search()` sau mỗi 0.5s sau khi 1 chữ cái mới được bấm vào.

+ Hàm `search()` sẽ lấy 10 giá trị đầu tiên tìm được và với mỗi giá trị đó sẽ gọi 1 hàm `displaySearchResult()`. Hàm này sẽ tạo 1 div tương ứng với 1 giá trị tìm được và `appendChild` vào div `results`, ngoài ra nó còn `addEventListener` `click` khi người dùng click vào và gọi hàm `searchResultSelected()`. Hàm này sẽ clear div `results` và `triggerEvent` `resultSelected`.

- Cuối cùng là `main1.js`, đây là nơi tập trung xử lý tất cả sự kiện và truyền thông tin cho các component khác.

+ Hàm tạo của nó sẽ truyền template bao gồm tất cả các div vào div có id `app` là div chờ sẵn ở HTML. Và sau đó gọi các services cần thiết ra, cuối cùng là gọi 2 hàm là `initializeComponents()` và `loadMapData()`.

+ Hàm `initializeComponents()`: tạo `infoComponent` mới với props bao gồm data lấy được từ `apiService`, tạo `mapComponent` có props bao gồm events `locationsSelected` (event này sẽ lấy thông tin của nơi người dùng bấm vào và truyền nó cho hàm `showInfo` của `Infopanel`), tạo `searchBar` có props bao gồm data lấy từ `searchService` và events là `resultSelected` (event này lấy id và tên layer và truyền vào hàm `selectLocation` của `Map`).

+ Hàm `loadMapData()` lấy tất cả dữ liệu của tất cả các layer và truyền vào `searchbase` của `searchService`, lấy các icon và bật các map layer lên.

Báo cáo của nhóm chúng em đến đây là kết thúc. Em xin cảm ơn thầy vì đã bỏ thời gian ra đọc báo cáo của nhóm chúng em a.