**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN 2**

🙟🕮🙝

Logo

Description automatically generated

**BÁO CÁO CUỐI KỲ**

**IOT VÀ ỨNG DỤNG**

**ĐỀ TÀI: XÂY DỰNG NODE-RED DASHBOARD SỬ DỤNG DỮ LIỆU CẢM BIẾN OPENEEW**

**Sinh viên thực hiện:  Nhóm 26**

**Nguyễn Quang Nghĩa N20DCCN045**

**Lê Văn Tiến N20DCCN064**

***Giảng viên*: Đàm Minh Lịnh**

---oOo---

**MỤC LỤC**

[**I.** **TÌM HIỂU VỀ ĐỀ TÀI** 3](#_Toc155341050)

[**1.** **Node-red là gì ?** 3](#_Toc155341051)

[**2.** **Node-red Dashboard** 3](#_Toc155341052)

[3. **Dữ liệu cảm biến Openeew** 3](#_Toc155341053)

[4. **Mục tiêu của đề tài** 3](#_Toc155341054)

[**II.** **NỘI DUNG ỨNG DỤNG** 4](#_Toc155341055)

[**1.** **Điều kiện tiên quyết** 4](#_Toc155341056)

[**2.** **Nội dung ứng dụng** 10](#_Toc155341057)

[**3.** **Thuật toán Seismic Activity** 13](#_Toc155341058)

[**III.** **KẾT LUẬN** 14](#_Toc155341059)

1. **TÌM HIỂU VỀ ĐỀ TÀI**
2. **Node-red là gì ?**

Node-RED là một nền tảng lập trình hình đồ hoạ (flow-based) mã nguồn mở, được phát triển dựa trên Node.js. Được thiết kế với mục tiêu đơn giản hóa quá trình phát triển ứng dụng IoT, Node-RED cung cấp một giao diện trực quan dựa trên trình duyệt để kết nối các nút (nodes) và tạo ra các luồng (flows) xử lý dữ liệu. Nhờ vào sự linh hoạt và dễ sử dụng, Node-RED đã trở thành một công cụ quan trọng trong việc phát triển giải pháp IoT và tự động hóa.

1. **Node-red Dashboard**

Node-RED Dashboard là một phần mở rộng của Node-RED, giúp người dùng tạo ra giao diện người dùng web một cách nhanh chóng và hiệu quả. Dashboard này cung cấp các thành phần như biểu đồ, bảng điều khiển và nút điều khiển, giúp biến các dữ liệu phức tạp thành thông tin trực quan và dễ hiểu. Điều này làm cho việc theo dõi và tương tác với dữ liệu trở nên thuận tiện và hấp dẫn hơn.

1. **Dữ liệu cảm biến Openeew**

OpenEEW là một dự án chuyên sâu trong việc xây dựng hệ thống cảnh báo động sớm cho động đất thông qua sử dụng các cảm biến gia đình. Dữ liệu từ các cảm biến này được chuyển đến trung tâm xử lý để phát hiện các biến động đất và cung cấp cảnh báo kịp thời. Thông qua tích hợp dữ liệu từ OpenEEW vào Node-RED, ta có thể thực hiện các xử lý phức tạp và hiển thị thông tin một cách trực quan trên Node-RED Dashboard.

1. **Mục tiêu của đề tài**

Mục tiêu chính của đề tài là xây dựng một hệ thống đồng bộ sử dụng Node-RED và Node-RED Dashboard để tận dụng dữ liệu từ cảm biến OpenEEW. Cụ thể, mục tiêu bao gồm:

* Integrate OpenEEW Data: Kết nối dữ liệu từ cảm biến OpenEEW vào môi trường Node-RED để có thể thực hiện các xử lý dữ liệu, như lọc, phân tích và biểu diễn trực quan.
* Visualization with Node-RED Dashboard: Tạo một giao diện người dùng web trực quan sử dụng Node-RED Dashboard, với các thành phần như biểu đồ thống kê, bảng điều khiển và các nút tương tác.
* Real-time Monitoring: Cập nhật dữ liệu từ OpenEEW một cách liên tục để cung cấp theo dõi thời gian thực về tình trạng động đất.
* Alerting System: Xây dựng một hệ thống cảnh báo để thông báo người dùng khi có sự biến động đáng chú ý trong dữ liệu cảm biến OpenEEW.
* User Interaction: Tạo các chức năng tương tác trên dashboard để người dùng có thể tùy chỉnh và kiểm soát thông tin hiển thị theo nhu cầu của họ.

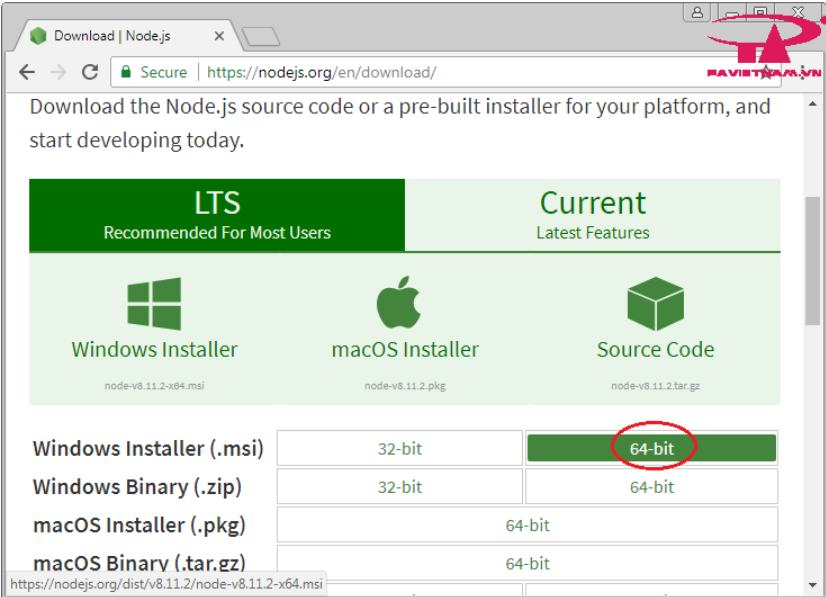
Bằng cách này, đề tài sẽ mang lại một giải pháp toàn diện cho việc giám sát và đánh giá tình hình động đất, đồng thời cung cấp cơ hội để người dùng tương tác và phản hồi trong thời gian thực. Từ đó cho thấy mục tiêu chính của đề tài này là xây dựng quy trình hiển thị cảnh báo cảm biến trên bản đồ.

1. **NỘI DUNG ỨNG DỤNG**
2. **Điều kiện tiên quyết**

Để thực hiện đề tài này, ta cần phải có các điều kiện tiên quyết như sau:

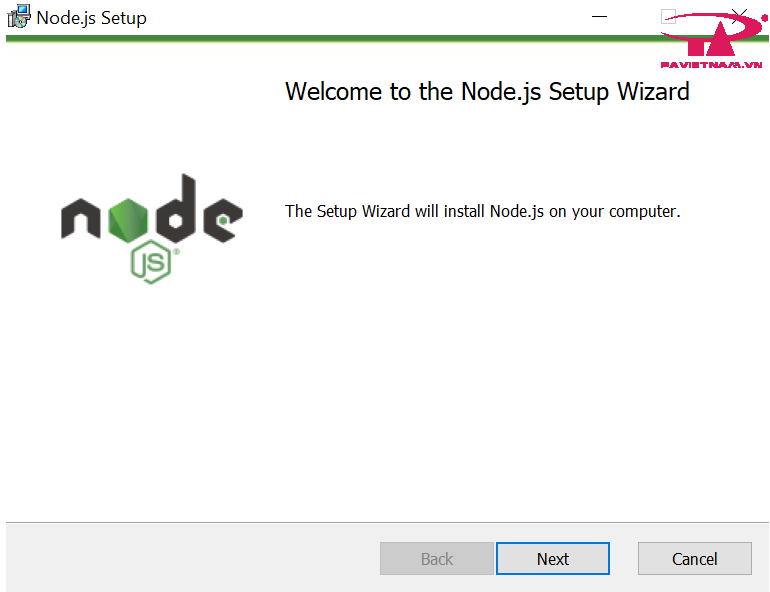
* Cài đặt NodeJS

+) Đây là link download NodeJS: [*https://nodejs.org/en/download/*](https://nodejs.org/en/download/)



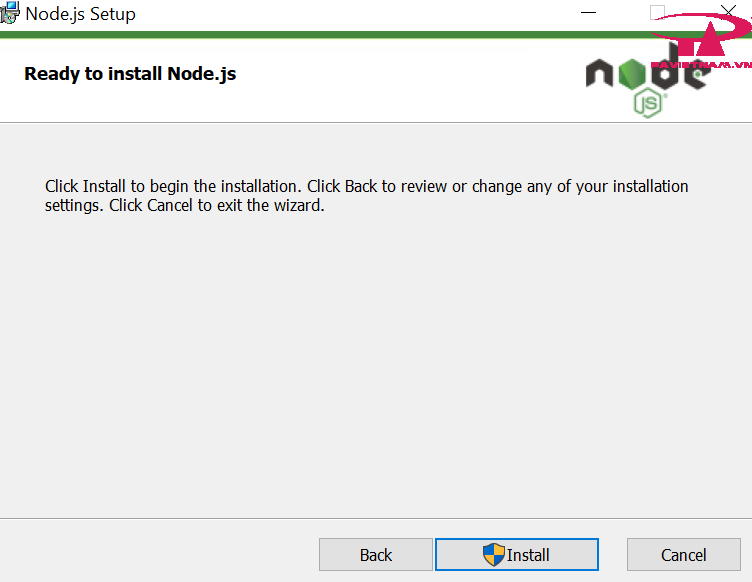
+) Để cài đặt NodeJS, bản mở file cài đặt mà bạn vừa tải ở bước trên về.

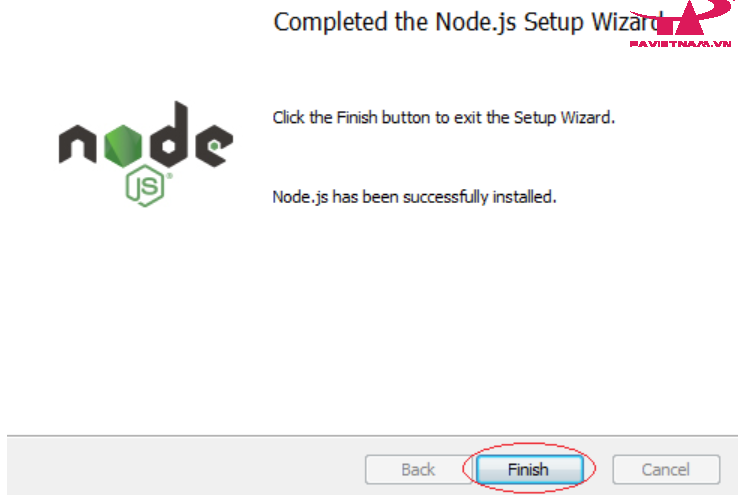
Tại màn hình cài đặt bạn bấm **Next**



+) Chúng ta tiếp tục nhấn accept và next để tiếp tục download

+) Và cuối cùng là bấm Install và Finish





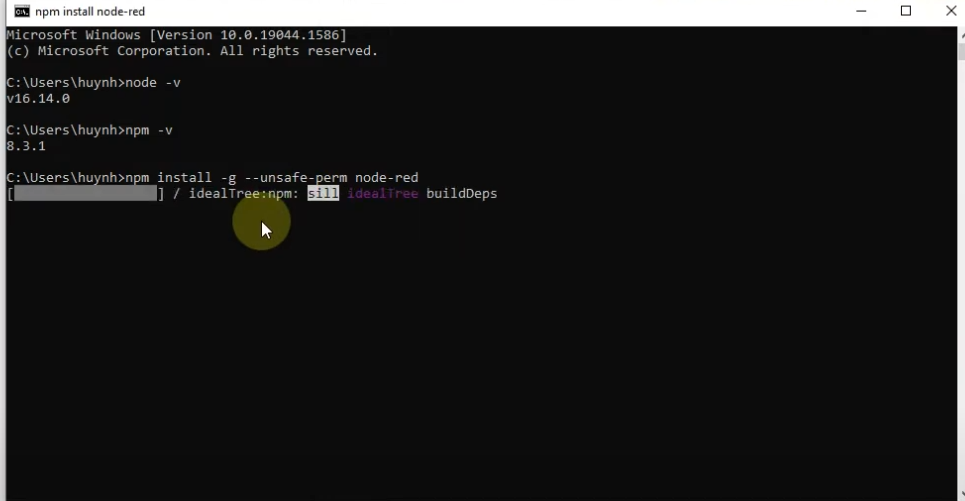
* Cài đặt Node-red

+) Đầu tiên cần mở cửa sổ “Command Prompt” lên

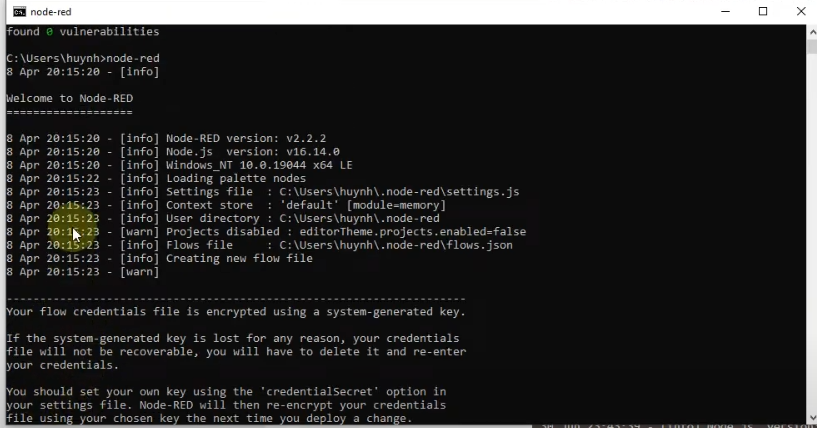
+) Nhập : node -v : Kiểm tra máy mình đã cài nodeJs gì

*npm -v*  : Kiểm tra phiên bản npm

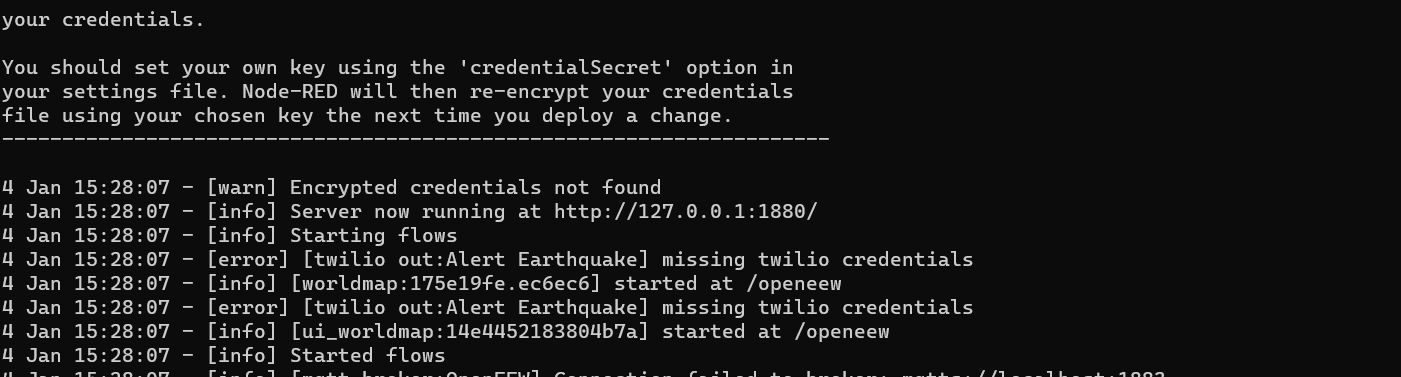
***npm install -g --unsafe-ferm node-red***  : Để cài đặt Node-red



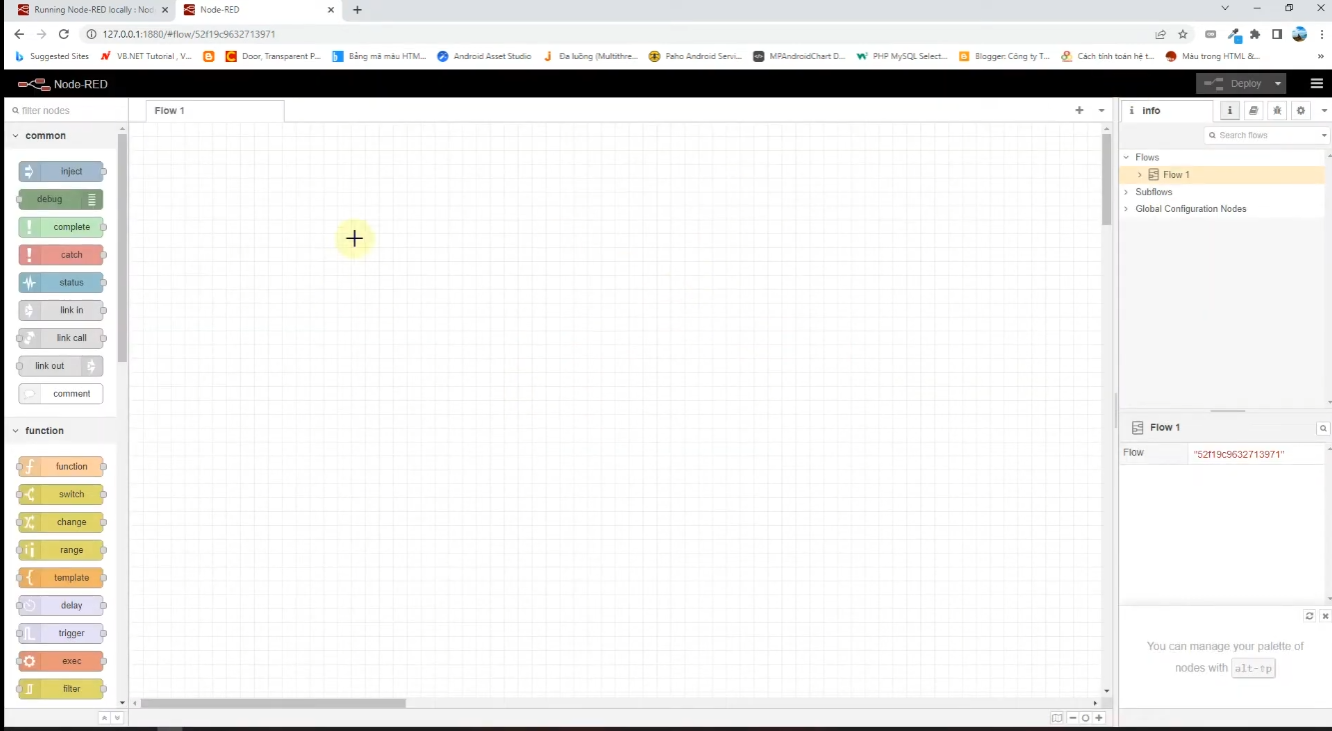
+) Nhập : *node-red*  : Để running



+) Chúng ta tìm kiếm đến dòng “ Server now running at “ để xem nó chạy ở trang nào



+) Đây là giao diện của trang node Red



* Tải và thêm các node sau vào node-red

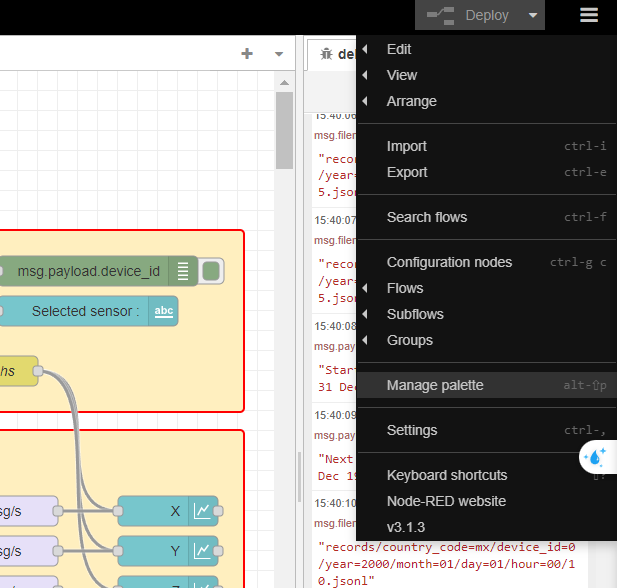
+ node-red-dashboard

+ node-red-contrib-web-worldmap

+ node-red-node-twilio

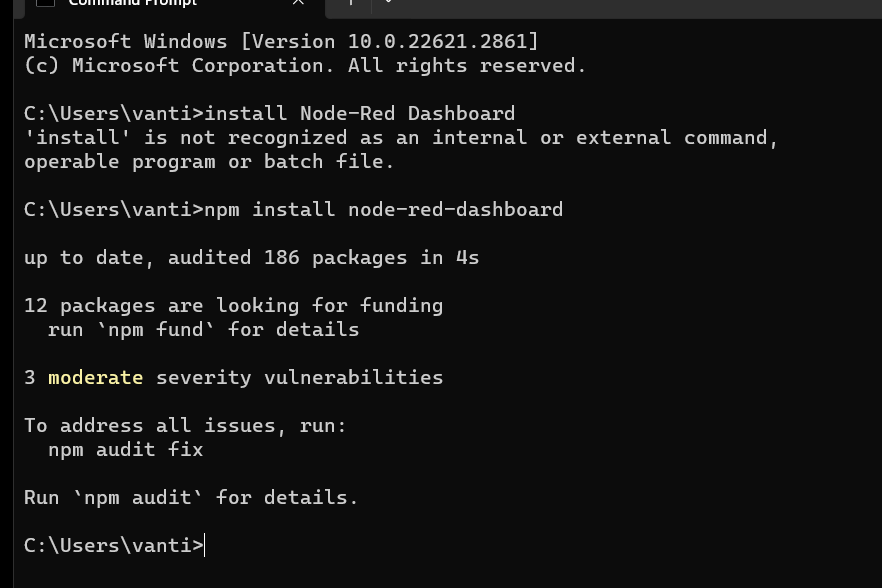
+ node-red-node-ui-table

Click vào dấu 3 sọc bên góc phải màn hình, sau đó chọn “Manage palette” nhập 4 thư viện bên trên vào:



* Cài đặt Node-red-dashboard

+) Nhập “ *npm install node-red-dashboard* “ để cài đặt.



- Cuối cùng là cách mở trang web

+) Nhập đường dẫn vào node-red +”/ui”



- Đây là giao diện



1. **Nội dung ứng dụng**

* Sơ lược về các node:

Theo các hướng dẫn và demo từ mã nguồn mở tại IBM, các node được setup như hình bên dưới:

A diagram of a diagram

Description automatically generated with medium confidence

Luồng này có 4 Phần:

Luồng 1: Thêm chân cảm biến vào bản đồ:

*Mục tiêu*: Đặt ghim trên bản đồ tại vị trí vĩ độ/kinh độ của mạng cảm biến Grillo OpenEEW.

*Cách thực hiện*: Sử dụng thông tin về vĩ độ và kinh độ của mạng cảm biến để thêm chân các cảm biến trên bản đồ.

Luồng 2: Báo cáo Trạng thái Cảm biến:

*Mục tiêu:* Kiểm tra định kỳ xem các cảm biến gần đây có báo cáo kết quả địa chấn qua MQTT hay không.

*Cách thực hiện:* Nếu không nhìn thấy cảm biến trong chu kỳ trước, hãy đánh dấu thiết bị ngoại tuyến bằng cách thay đổi chốt bị rơi thành màu đen bằng biểu tượng cảnh báo. Khi cảm biến kết nối lại và trực tuyến trở lại, hãy đánh dấu thiết bị màu xanh lá cây trên bản đồ.

Luồng 3: Đăng ký Mạng cảm biến OpenEEW / Trạng thái cảm biến:

*Mục tiêu*: Đăng ký với nhà môi giới MQTT và giám sát trạng thái cảm biến, ghi lại thời gian khi cảm biến báo cáo kết quả đo địa chấn.

*Cách thực hiện*: Gửi thông tin đăng ký và kết quả đo địa chấn qua MQTT, và ghi lại dấu thời gian của các sự kiện quan trọng.

Luồng 4: Nghe các trận động đất:

*Mục tiêu*: Chạy thuật toán địa chấn để dự đoán khả năng xảy ra động đất.

*Cách thực hiện*: Nếu phát hiện có khả năng xảy ra động đất, gửi cảnh báo qua Twilio và đánh dấu điểm ghim bị rơi màu đỏ trên bản đồ.

* Giao diện:

Giao diện của bài này được hiển thị như hình bên dưới. Đó là bản đồ và các chấm màu xanh là vị trí các cảm biến. Màu xanh tượng trưng cho việc các thông số bình thường và không có động đất. Nếu nút màu xanh biến thành màu đỏ có nghĩa là cảm biến ở đó cho ta biết có động đất xảy ra.

A map of a country

Description automatically generated

* Chức năng:
* Khi click vào các nút cảm biến, ta sẽ thấy được các thông số như sau:

A map of a country

Description automatically generated

* Khi có động đất, nút cảm biến ở vị trí có động đất sẽ chuyển sang màu đỏ. Tôi có một node Fake Earthquake để làm giả động đất. Để thấy ví dụ khi có động đất, ta cần phải disable node grillo-openeew/traces/+ để các dữ liệu thực từ các cảm biến không làm nhiễu dữ liệu giả từ nút Fake Earthquake

A blue rectangle with black text

Description automatically generated

Sau khi kích hoạt node Fake Earthqquake, ta sẽ thấy được nút cảm biến chuyển sang màu đỏ như sau:

A map of a country

Description automatically generated

* Ta có thể phóng to, thu nhỏ bản đồ tùy theo ý muốn.
* Góc trên bên phải của bản đồ trên có một mục dùng để setup bản đồ theo ý muốn

A map of a country with points of location

Description automatically generated

1. **Thuật toán Seismic Activity**

Một cách đơn giản để phát hiện hoạt động địa chấn là đo tốc độ lớn từ các loại cảm biến tốc độ mở ra, dựa trên ngưỡng đã đặt cao hơn xung nhịp của kế hoạch gia tốc. Gal được định nghĩa là 1 cm trên giây bình phương (1 cm/s2). Hoạt động địa chấn đáng lo ngại có thể vượt quá 3 gals (3 cm/giây2).

Các luồng Node-RED này quan sát dữ liệu gia tốc kế openeew theo thời gian thực và tính toán xem cảm biến có thể gặp phải hoạt động địa chấn hay không bằng thuật toán sau. Mỗi giây, hàm này nhận được mảng x/y/z dữ liệu rung động trên giây. Các mảng dữ liệu được truyền vào hàm bên trong msg.payload.traces[0]

Thuật toán này được ứng dụng trong node Alert on possible earthquake, Các dữ liệu từ node mqtt hay dữ liệu giả từ Fake Earthquake sẽ thông qua node function này để tính toán gia tốc và cảnh báo nếu vượt mức. Hàm javascript lặp qua dữ liệu rung để tìm kiếm gia tốc vượt quá 3 cm/giây2 như sau:

A computer screen shot of a program

Description automatically generated

1. **KẾT LUẬN**

Trước hết, chúng tôi đã thành công trong việc phát triển một hệ thống có khả năng hiển thị cảnh báo từ các cảm biến trên một bản đồ tương tác. Sự tích hợp giữa dữ liệu cảm biến và hiển thị trực quan giúp người dùng dễ dàng theo dõi và phân tích thông tin một cách thuận tiện.

Chúng tôi cũng đã đạt được mục tiêu tăng cường khả năng phản ứng trước những tình huống cảm biến quan trọng. Việc xây dựng một ứng dụng cảnh báo động đất sử dụng dữ liệu từ các cảm biến theo thời gian thực sẽ giúp nhanh chóng phát hiện và thông báo khi mức độ rung chuyển vượt mức, giúp chúng ta cải thiện khả năng đưa ra được những quyết định để có thể giảm thiểu thiệt hại có thể xảy ra.

Tuy nhiên vẫn còn một số mục tiêu chúng tôi chưa thực hiện được bởi một số nguyên nhân. Đó là việc gửi tin nhắn thông báo khi có động đất đến số điện thoại hoặc email thông qua twilio. Việc này có thể thực hiện nếu chúng tôi có đầy đủ các điều kiện tương ứng.

Cuối cùng, đề tài này tuy có một vài thiếu sót nhưng nó lại đạt được những kết quả kỹ thuật tích cực và có thể mở ra những triển vọng nghiên cứu và ứng dụng trong tương lai. Sự phát triển của các công nghệ cảm biến và xử lý dữ liệu liên tục mở rộng các khả năng của hệ thống, làm cho nó trở thành một công cụ quan trọng trong việc giải quyết các thách thức liên quan đến theo dõi và quản lý cảnh báo từ cảm biến trong thế giới ngày nay.