**42**

**Họ và tên: Nguyễn Phi Long Ký tên: Long**

**MSSV: N21DCCN142**

**Trường: Học viện Công nghệ Bưu chính Viễn thông**

**Môn: Nhập môn Công nghệ phần mềm**

**Tên chủ đề (CNPM mới): Tìm hiểu & ứng dụng NodeJS với framework Express tích hợp Tailwind.**

**Nguồn sản phẩm:** [**https://tailwind-nodejs.onrender.com/**](https://tailwind-nodejs.onrender.com/)

**Mục lục**

[**I.** **Giới thiệu sơ lược:** 3](#_Toc166658257)

[**II.** **Tổng lược các yếu tố kỹ thuật & công nghệ:** 3](#_Toc166658258)

[**1.** **NodeJS:** 4](#_Toc166658259)

[**2.** **Express Framework (Epressjs):** 4](#_Toc166658260)

[**3.** **Tailwind:** 4](#_Toc166658261)

[**III.** **Hướng dẫn một ứng dụng đơn giản:** 5](#_Toc166658262)

[**IV.** **Hướng dẫn setup và chạy sản phẩm:** 15](#_Toc166658263)

[**V.** **Ưu điểm và nhược điểm:** 17](#_Toc166658264)

[**1.** **NodeJS:** 17](#_Toc166658265)

[**2.** **Tailwind CSS:** 18](#_Toc166658266)

[**VI.** **Tổng kết** 19](#_Toc166658267)

1. **Giới thiệu sơ lược:**

Với sự phát triển của mô hình website hiện nay, việc đáp ứng yêu cầu và xu hướng mới là điều cần thiết. Dưới đây là một số xu hướng hiện nay:

* Tăng cường trải nghiệm người dùng (User Experience - UX): Các website ngày càng tập trung vào việc cải thiện trải nghiệm người dùng, bao gồm tối ưu hóa tốc độ tải trang, phản hồi nhanh chóng và thiết kế đáp ứng cho các thiết bị di động.
* Tiếp cận đa nền tảng (Cross-platform Accessibility): Sự phát triển của các công nghệ như Progressive Web Apps (PWA) và Responsive Web Design (RWD) cho phép website hoạt động mượt mà trên nhiều thiết bị và nền tảng, từ máy tính đến điện thoại di động và thiết bị thông minh.
* Tích hợp trí tuệ nhân tạo (Artificial Intelligence - AI) và học máy (Machine Learning - ML): Các công nghệ này được sử dụng để cá nhân hóa nội dung, cải thiện dự đoán và tăng cường tương tác với người dùng thông qua chatbots và hệ thống gợi ý.

Cả Node.js, framework Express và Tailwind đều là các công nghệ phổ biến và phát triển mạnh mẽ, và chúng đều rất thích hợp cho sự phát triển website hiện nay. Dưới đây là lý do tại sao:

* Node.js: Hiệu suất cao và không đồng bộ, Node.js cho phép xây dựng các ứng dụng web có khả năng xử lý hàng ngàn kết nối đồng thời mà vẫn duy trì hiệu suất cao.
* Express.js:
* Đơn giản và linh hoạt.
* Hỗ trợ middleware mạnh mẽ.
* Hỗ trợ tốt cho các công nghệ phổ biến khác như (MongoDB, MySQL, Socket.io, …).
* Tailwind:
  + Tối ưu hóa cho tốc độ phát triển: Tailwind CSS giúp tăng tốc độ phát triển thông qua các lớp CSS được định nghĩa trước, giúp giảm thời gian viết CSS và tối ưu hóa quy trình thiết kế.
  + Thiết kế linh hoạt và tùy biến: Với Tailwind CSS, bạn có thể dễ dàng tùy chỉnh giao diện người dùng của mình mà không cần phải viết CSS tùy chỉnh, từ đó giúp tiết kiệm thời gian và công sức.

Trong sản phẩm demo này, ta sử dụng NodeJS với framework Express để xây dựng phần backend và Tailwind CSS để xây dựng phần giao diện web (frontend). Ở đây sản phẩm sẽ là một trang web Portfolio hay còn gọi là CV.

1. **Tổng lược các yếu tố kỹ thuật & công nghệ:**
2. **NodeJS:**

NodeJS là một mã nguồn được xây dựng dựa trên nền tảng Javascript V8 Engine, nó được sử dụng để xây dựng các ứng dụng web. NodeJS là một mã nguồn mở được sử dụng rộng rãi bởi hàng ngàn lập trình viên trên toàn thế giới. NodeJS có thể chạy trên nhiều nền tảng hệ điều hành khác nhau từ Window cho tới Linux, OS X nên đó cũng là một lợi thế. NodeJS cung cấp các thư viện phong phú ở dạng Javascript Module khác nhau giúp đơn giản hóa việc lập trình và giảm thời gian ở mức thấp nhất.

Tính chất của NodeJS:

* Không đồng bộ: Tất cả các API của NodeJS đều không đồng bộ (none-blocking), nó chủ yếu dựa trên nền của NodeJS Server và chờ đợi Server trả dữ liệu về. Việc di chuyển máy chủ đến các API tiếp theo sau khi gọi và cơ chế thông báo các sự kiện của Node.js giúp máy chủ để có được một phản ứng từ các cuộc gọi API trước (Realtime).
* Kiến trúc đơn luồng: NodeJS hoạt động trên một luồng duy nhất. Dựa trên kiến trúc “Mô hình vòng lặp sự kiện đơn luồng” (Single Threaded Event Loop Model), NodeJS có thể xử lý nhiều yêu cầu của máy khách. Vòng lặp sự kiện được thực thi trên một luồng chính duy nhất. Vòng lặp sự kiện cho phép NodeJS thực hiện tất cả các hoạt động I/O không chặn mặc dù JavaScript là đơn luồng.

1. **Express Framework (Epressjs):**

Expressjs là một framework được xây dựng trên nền tảng của Nodejs. Nó cung cấp các tính năng mạnh mẽ để phát triển web hoặc mobile. Expressjs hỗ trợ các method HTTP và midleware tạo ra API vô cùng mạnh mẽ và dễ sử dụng.

Các tính năng của Express framework:

* Cho phép thiết lập các lớp trung gian để trả về các HTTP request.
* Định nghĩa routing có thể được sử dụng với các hành động khác nhau dựa trên phương thức HTTP và URL.
* Cho phép trả về các trang HTML dựa vào các tham số truyền vào đến template.

1. **Tailwind:**

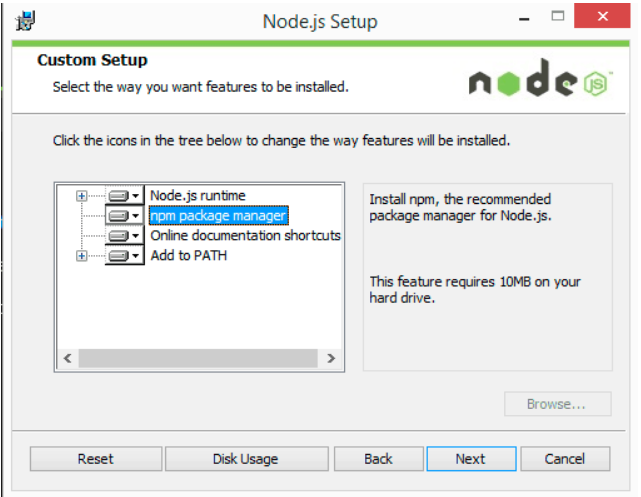
Tailwind CSS là một trong những framework phát triển web được nhiều lập trình viên ưa chuộng hiện nay.

Được ra mắt vào năm 2017, Tailwind CSS đã nhanh chóng trở thành một trong những công cụ quan trọng giúp việc xây dựng giao diện web trở nên dễ dàng hơn.

Tailwind css là một utility-first CSS framework nó hỗ trợ phát triển xây dựng nhanh chóng giao diện người dùng, nó cũng có điểm chung giống như Bootstrap & điểm làm nó nổi bật hơn cả đó là chúng ta có thể tùy biến phát triển css theo cách mà chúng ta định nghĩa ra.

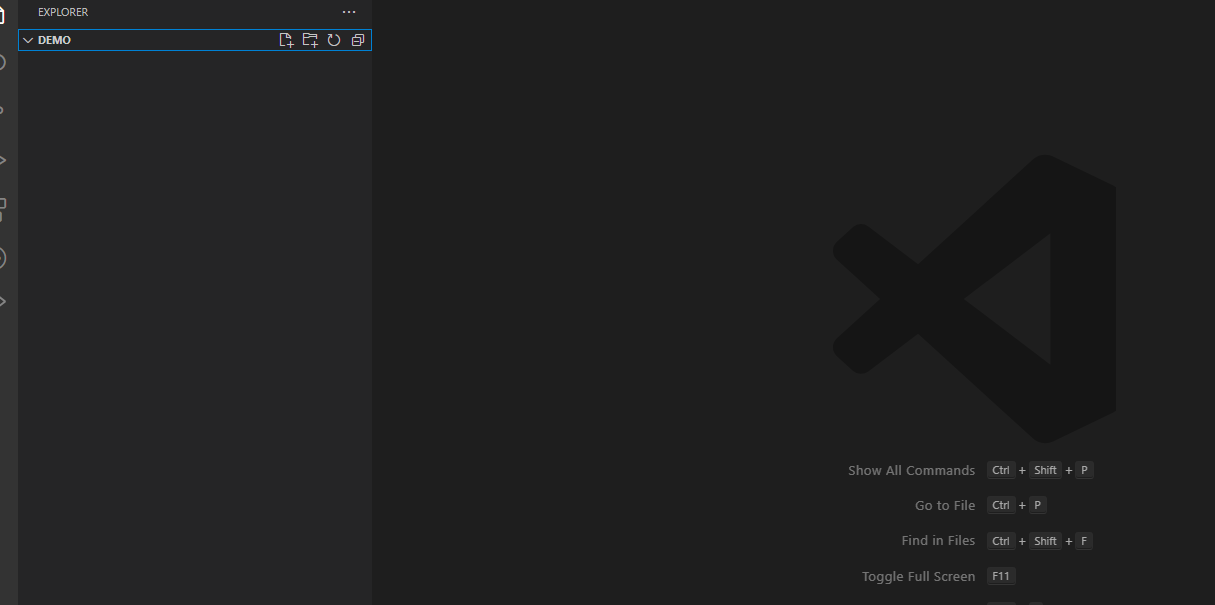
1. **Hướng dẫn một ứng dụng đơn giản:**

Đầu tiên ta phải cài đặt NodeJS:

1. Tải tại website chính thức của Node.js <https://nodejs.org/en/download>
2. Chạy file .msi đã tải
3. Tiếp theo setup như bình thường, cho đến bước này:

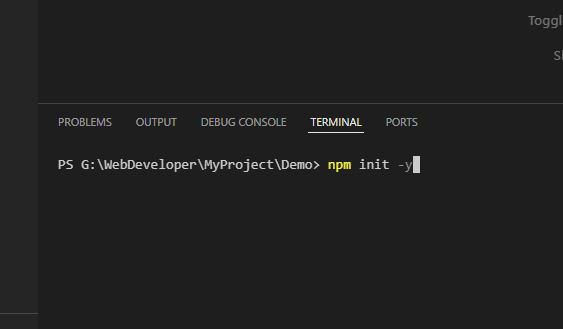
Setup như hình và bấm next.

1. Restart máy tính.

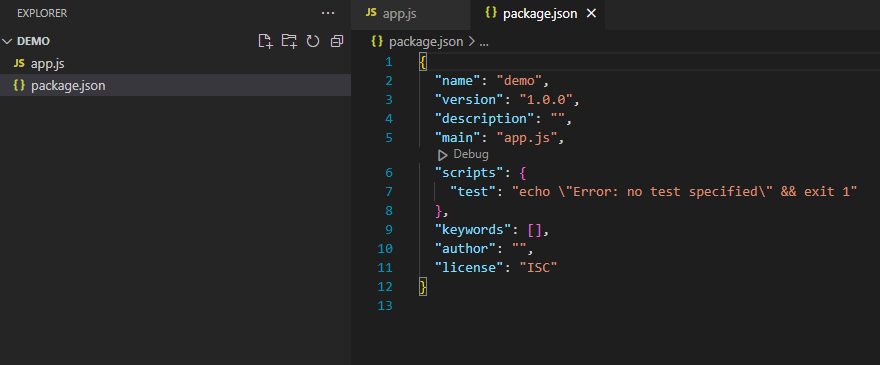
Tiếp theo ta tạo một thư mục trống, ở đây ta dùng Visual Studio Code để thực hiện, ta vào Visual Studio Code và add thư mục ấy vào. Như hình dưới:

Ta tạo file app.js, sau đó, ta mở cửa sổ Terminal lên bằng tổ hợp phím Ctrl + ` hoặc bấm trên thanh tab.

Tại cửa sổ Terminal, ta gõ: npm init –y để khởi tạo project về nodejs với file app.js làm gốc.



Nó sẽ tự sinh file như này:

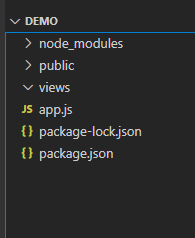


Tiếp theo ta cài đặt framework Express, mở Terminal lên và gõ:

npm install express

Nó sẽ tự sinh file node\_modules và sau này cài đặt cái gì sẽ nằm trong file này hết.

Tiếp đến ta tạo thư mục views để chứa phần code giao diện trong này, tạo thư mục public nơi chứa các file css, script, images, …



Sau đó mở file app.js và code như sau:

const express = require('express');

const path = require('path');

const app = express();

const port = 1009;

app.set("view engine", "ejs");

app.set('views', path.join(\_\_dirname, 'views'));

app.use(express.static(path.join(\_\_dirname, "public")));

app.get('/', (req, res) => {

    res.render('home');

});

app.listen(port);

Ở đây ta sử dụng view engine là ejs. Ejs là template phổ biến của Nodejs với cú pháp thân thiện và tương đồng HTML. Cài đặt ejs: npm install ejs

Trong thư mục views ta tạo file home.ejs và code như sau:

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Hướng dẫn một ứng dụng đơn giản</title>

</head>

<body>

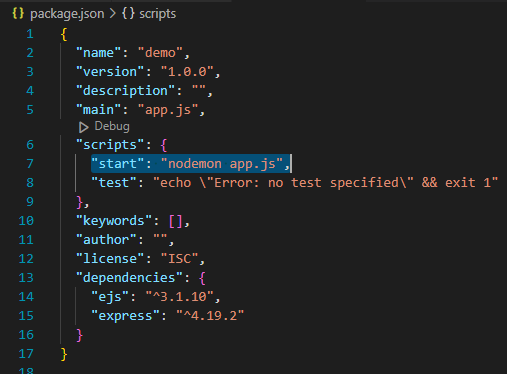
    <h1>This is a demo</h1>

</body>

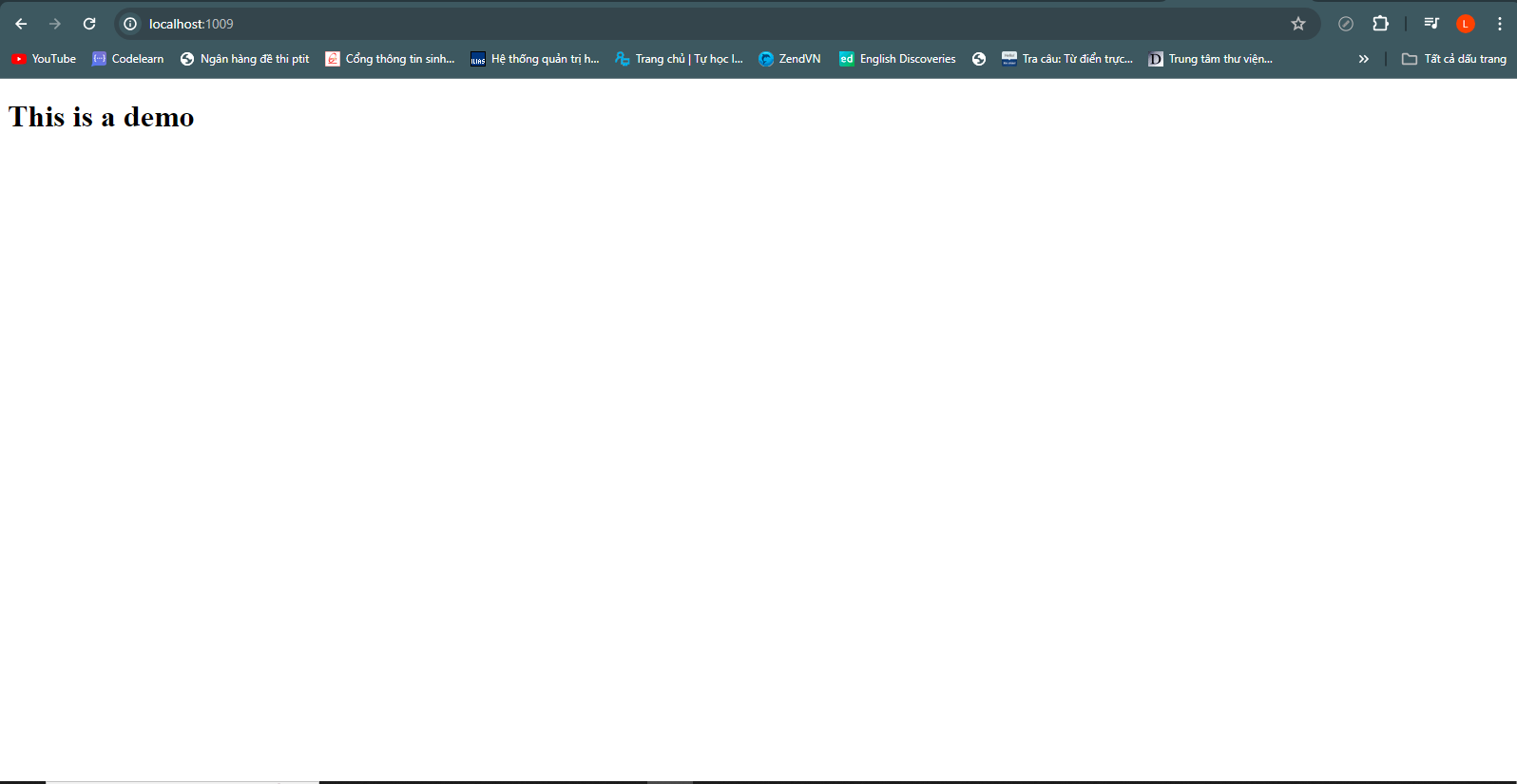
</html>

Tại terminal ta gõ: npm install nodemon đây là package giúp tiết kiệm thời gian lập trình, khi ta sửa đoạn code server sẽ tự động làm mới và không cần phải chạy lại server.

Sau đó, tại file package.json, ta thêm đoạn code sau:



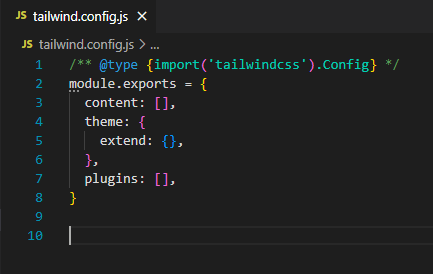
Khi đó, khi ta chạy dự án, chỉ cần gõ npm start

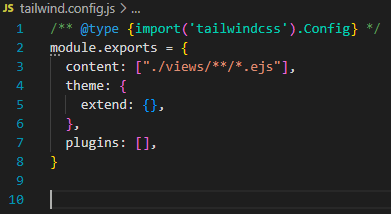
Ta chạy dự án và mở browser lên, trên url gõ localhost:1009 sẽ được như hình

Tiếp theo, ta sẽ tích hợp Tailwind CSS để thiết kế phần giao diện.

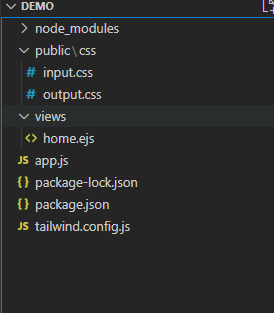
Tại terminal, ta gõ: npm install -D tailwindcss

npx tailwindcss init

Sau đó, nó sẽ tự sinh file tailwind.config.js

Ta thêm dòng này vào content: 

Để set thư mục views nhận được Tailwind CSS.

Ta tạo thư mục css trong đó tạo 2 file input.css và output.css

Trong file input.css ta dán đoạn code này vào:

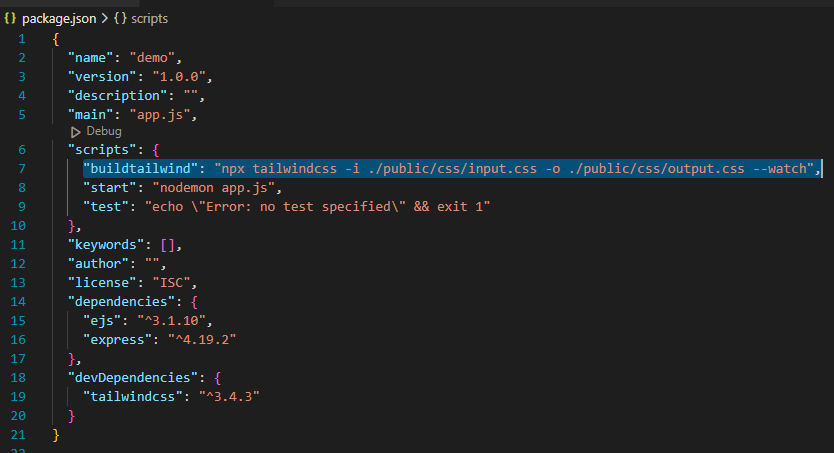
@tailwind base;

@tailwind components;

@tailwind utilities;

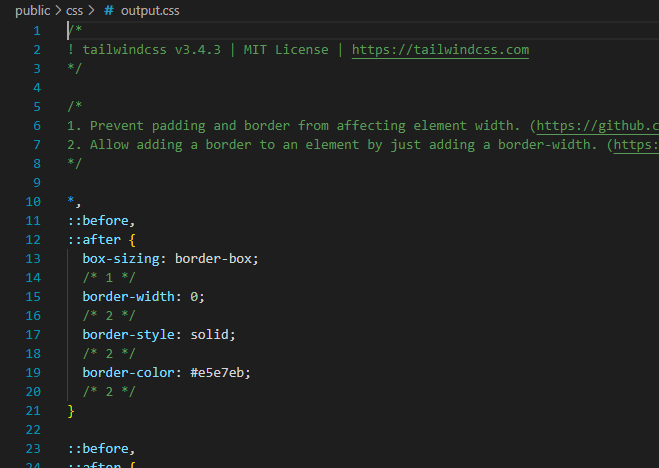
Tiếp tục ta thêm dòng này vào file package.json:

"buildtailwind": "npx tailwindcss -i ./public/css/input.css -o ./public/css/output.css --watch"



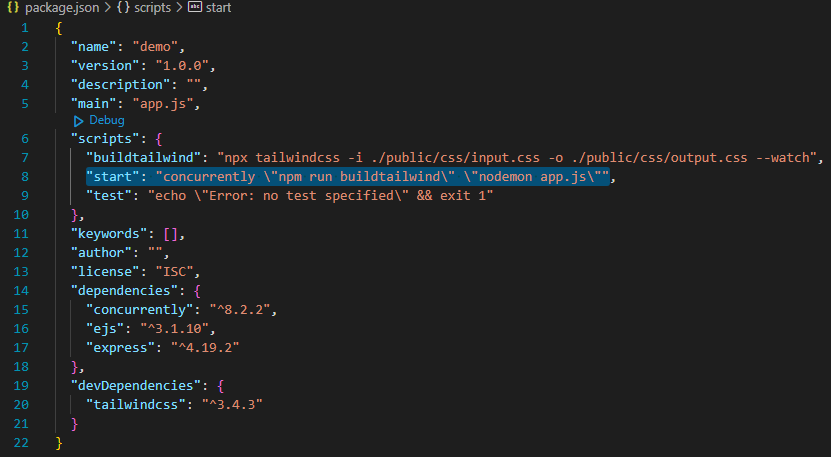
Sau đó, terminal ta gõ: npm run buildtailwind để build file css vào file output.css

Khi tải xong thì trong file output.css sẽ như thế này



Ta sửa phần start của file package.json như sau:

"start": "concurrently \"npm run buildtailwind\" \"nodemon app.js\""



Làm như trên để thuận tiện trong việc run dự án, khi chạy ta chỉ cần gõ npm start thì nó sẽ tự build css bên tailwindcss và chạy file app.js cho chúng ta. Nhưng trước hết ta phải cài đặt module concurrently: npm install concurrently

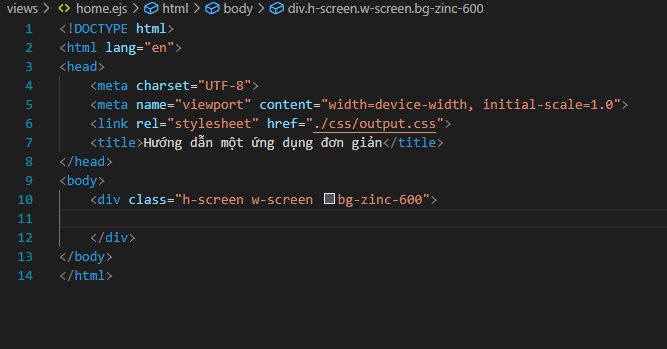
Như vậy, ta đã tích hợp thành công Tailwind CSS vào NodeJs thành công.

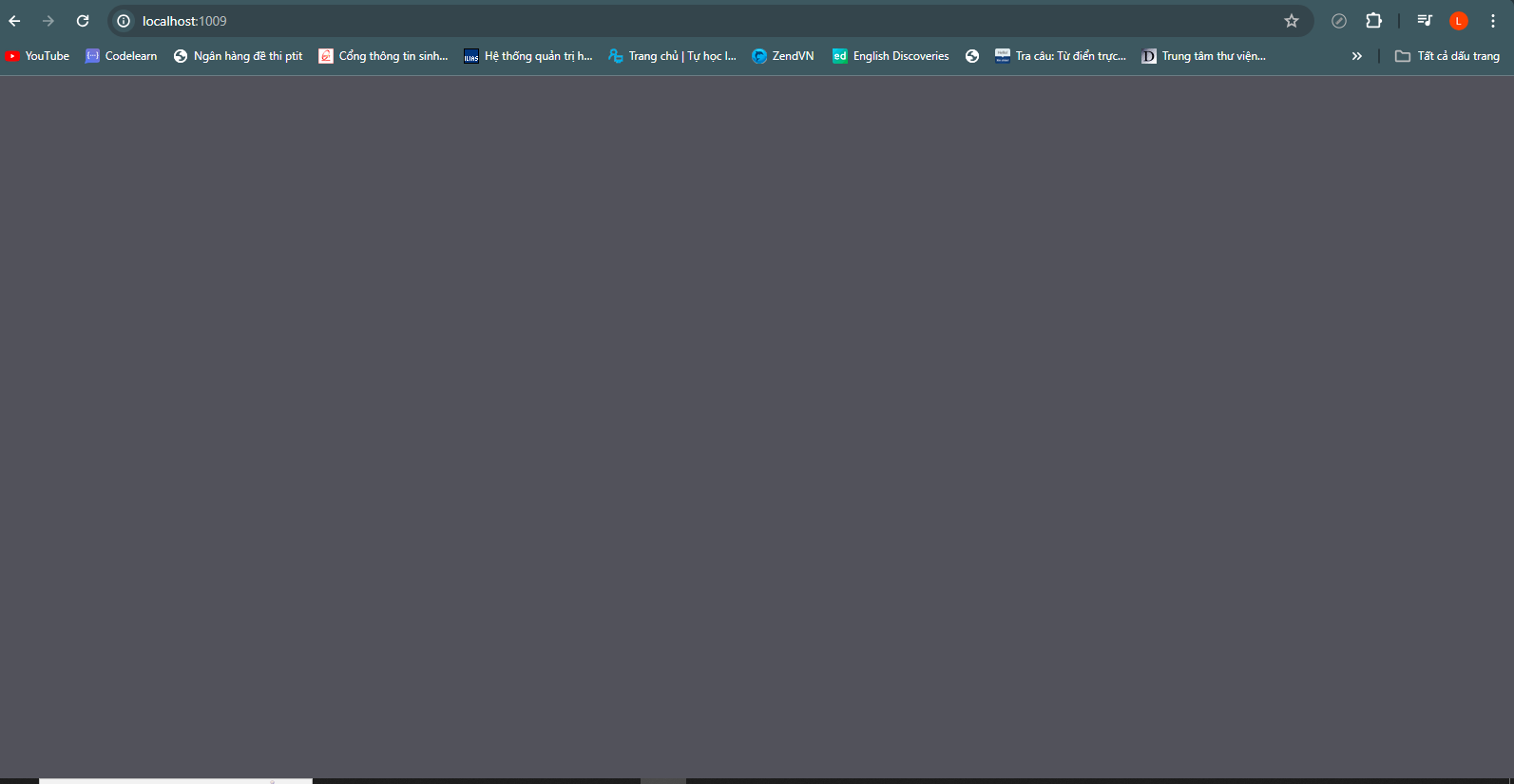
Tiếp theo ta sẽ link file này vào phần giao diện, và thực hiện một ví dụ đơn giản

<link rel="stylesheet" href="./css/output.css">

Để nó ở phần head của file home.ejs

Thêm thẻ div, với class h-screen là chiều cao full màn hình, w-screen là chiều rộng full màn hình, bg-zinc-600 là set màu nền



Kết quả:

Tiếp theo, ta thêm một hộp ở giữa màn hình với class,

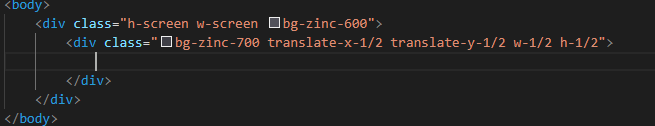
bg-zinc-700 là set màu nền,

translate-x-1/2: di chuyển hộp sang giữa màn hình theo chiều ngang,

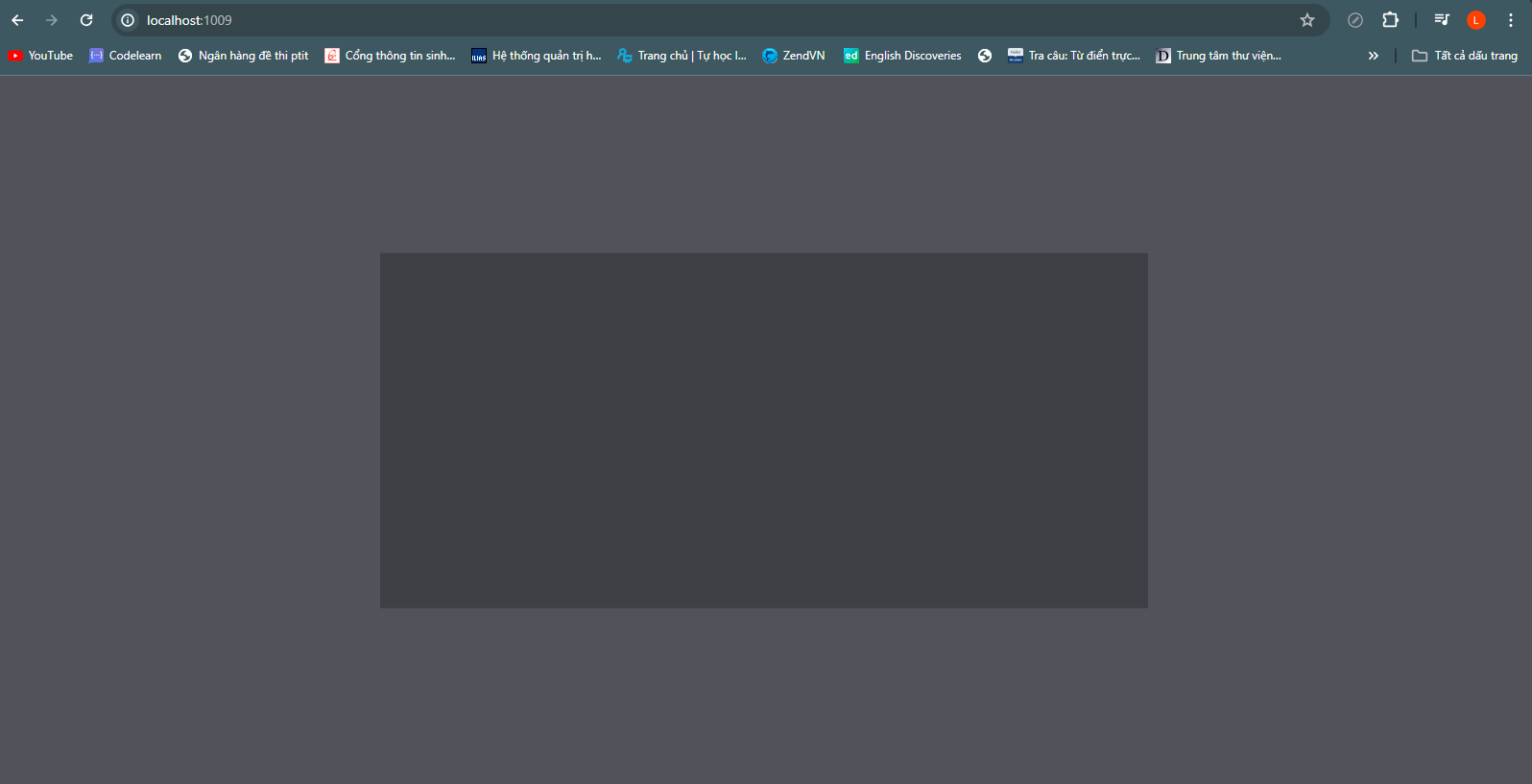
translate-y-1/2: di chuyển hộp xuống giữa màn hình theo chiều dọc,

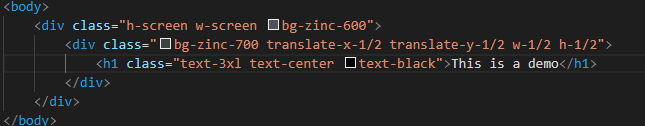
w-1/2: set chiều rộng của hộp bằng nửa màn hình

h-1/2: set chiều cao của hộp bằng nửa màn hình



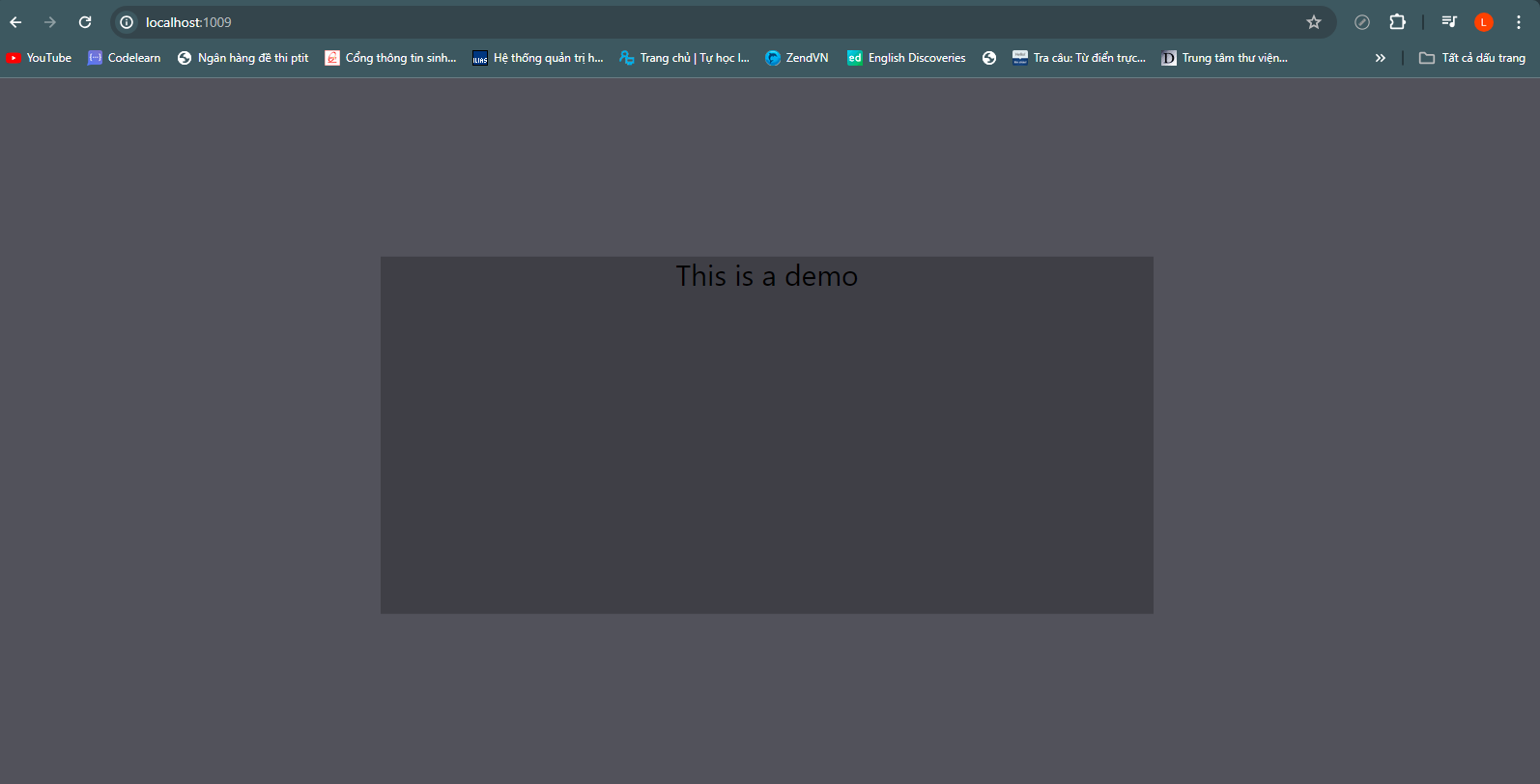
Kết quả:



Tiếp theo, ta thêm dòng chữ vào hộp:

Với class text-3xl: set font-size, text-center: căn giữa, text-black: chữ màu đen

Kết quả:



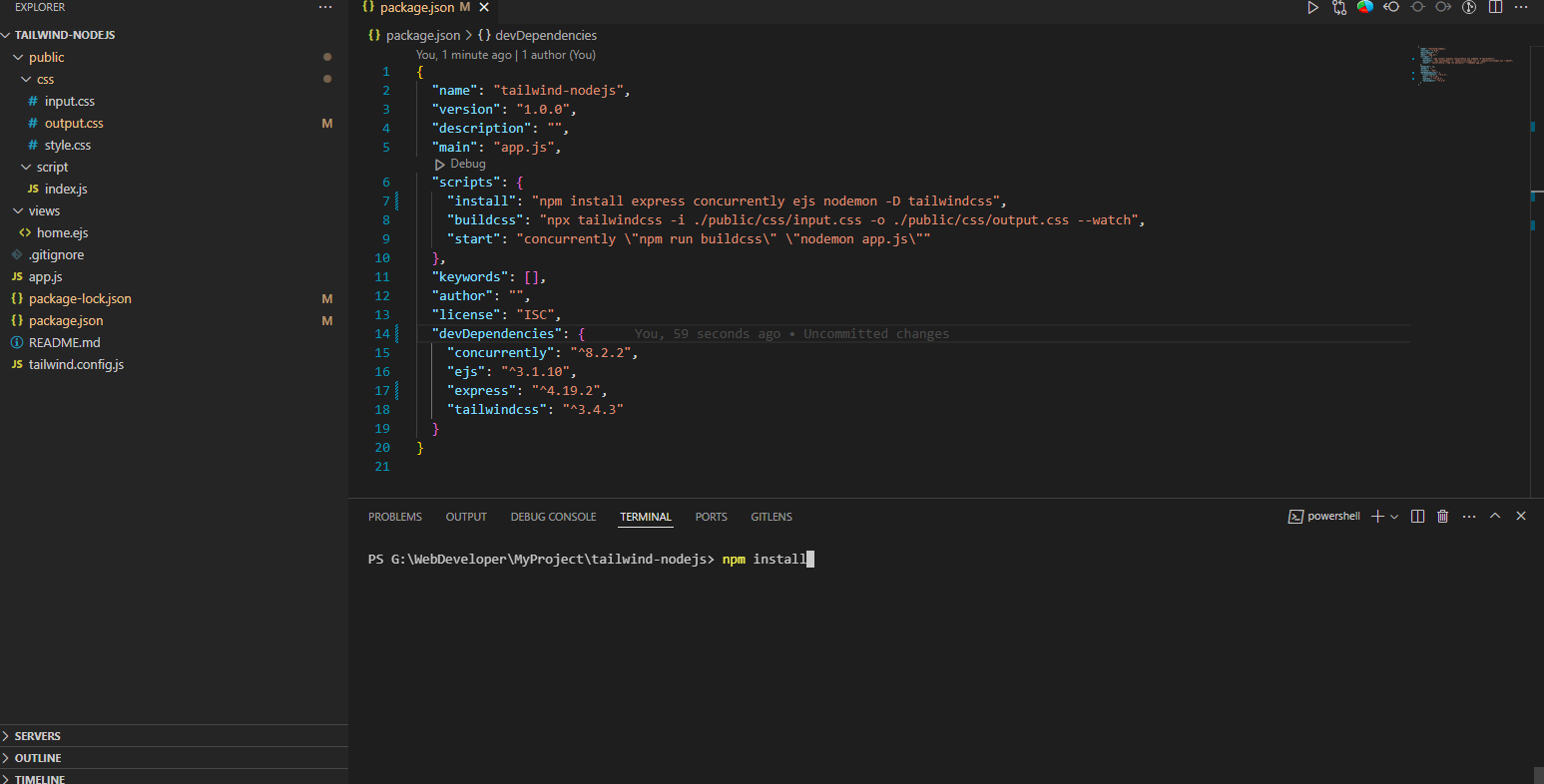
Tổng kết, trên đây là cách setup tích hợp Node.js với TailwindCSS.

1. **Hướng dẫn setup và chạy sản phẩm:**

Sau khi giải nén thư mục sản phẩm ra, ta mở thư mục project trên Visual Studio Code

Như đã hướng dẫn ở phần III, ta sẽ tải Nodejs.

Mở terminal lên, và gõ: npm install

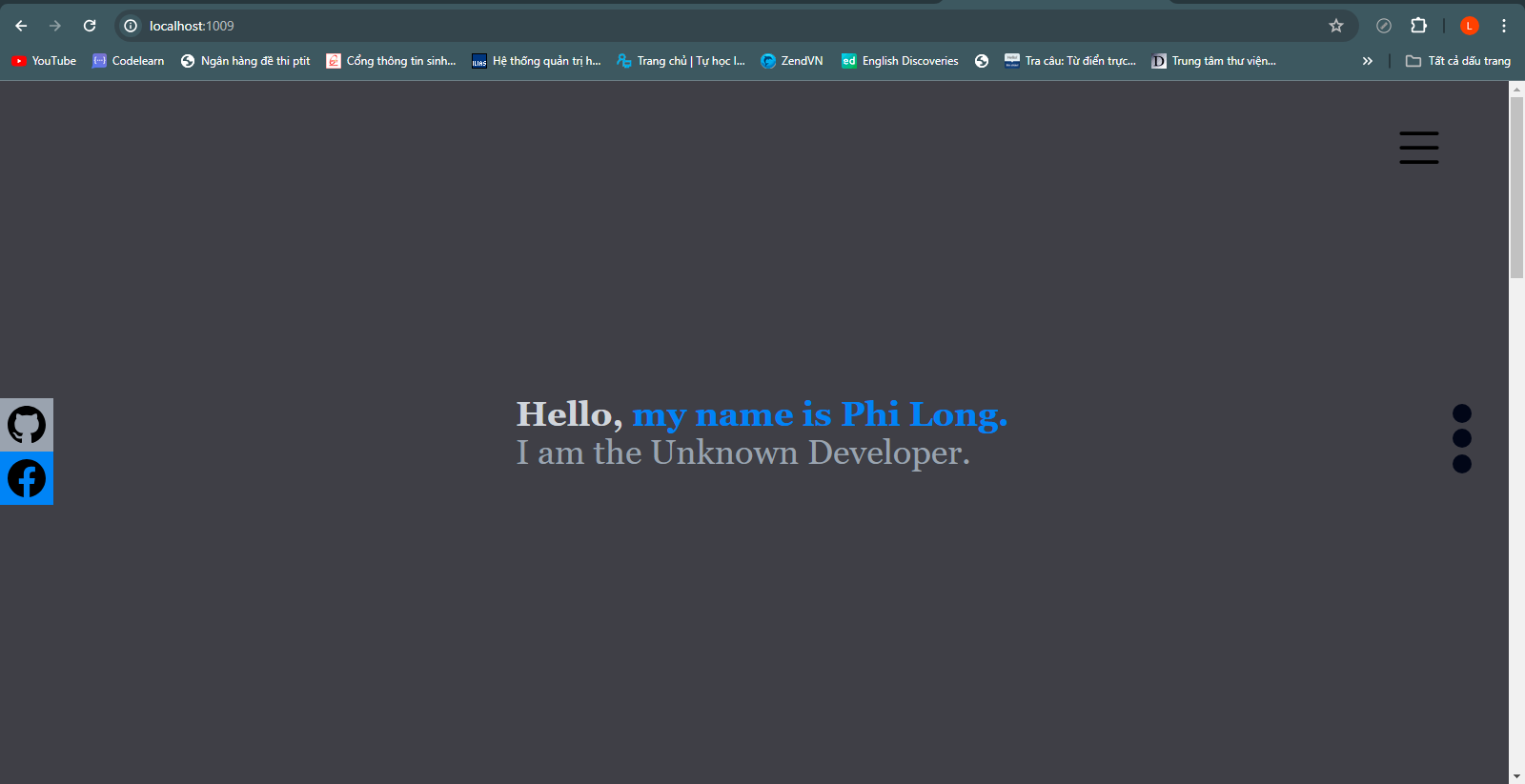
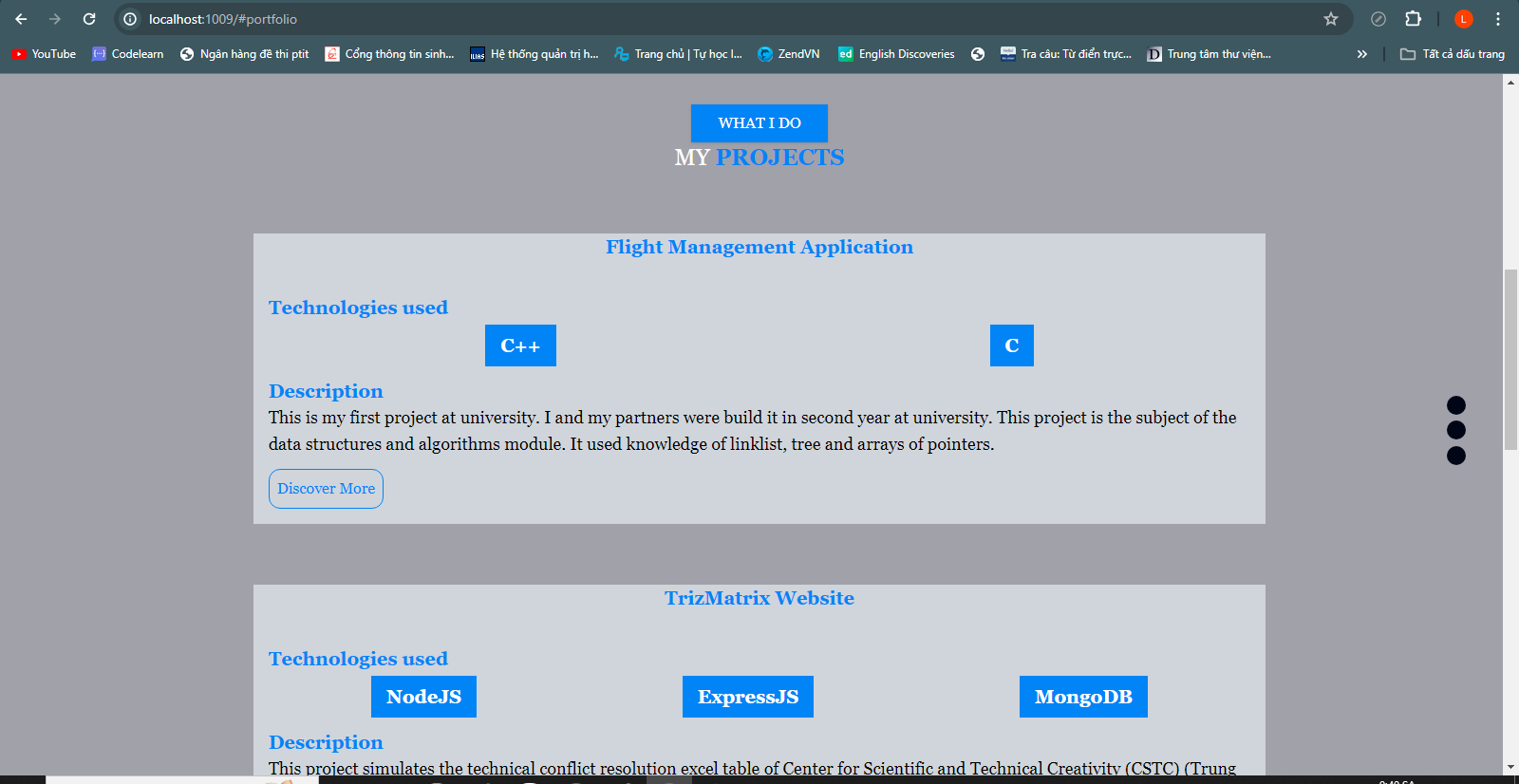


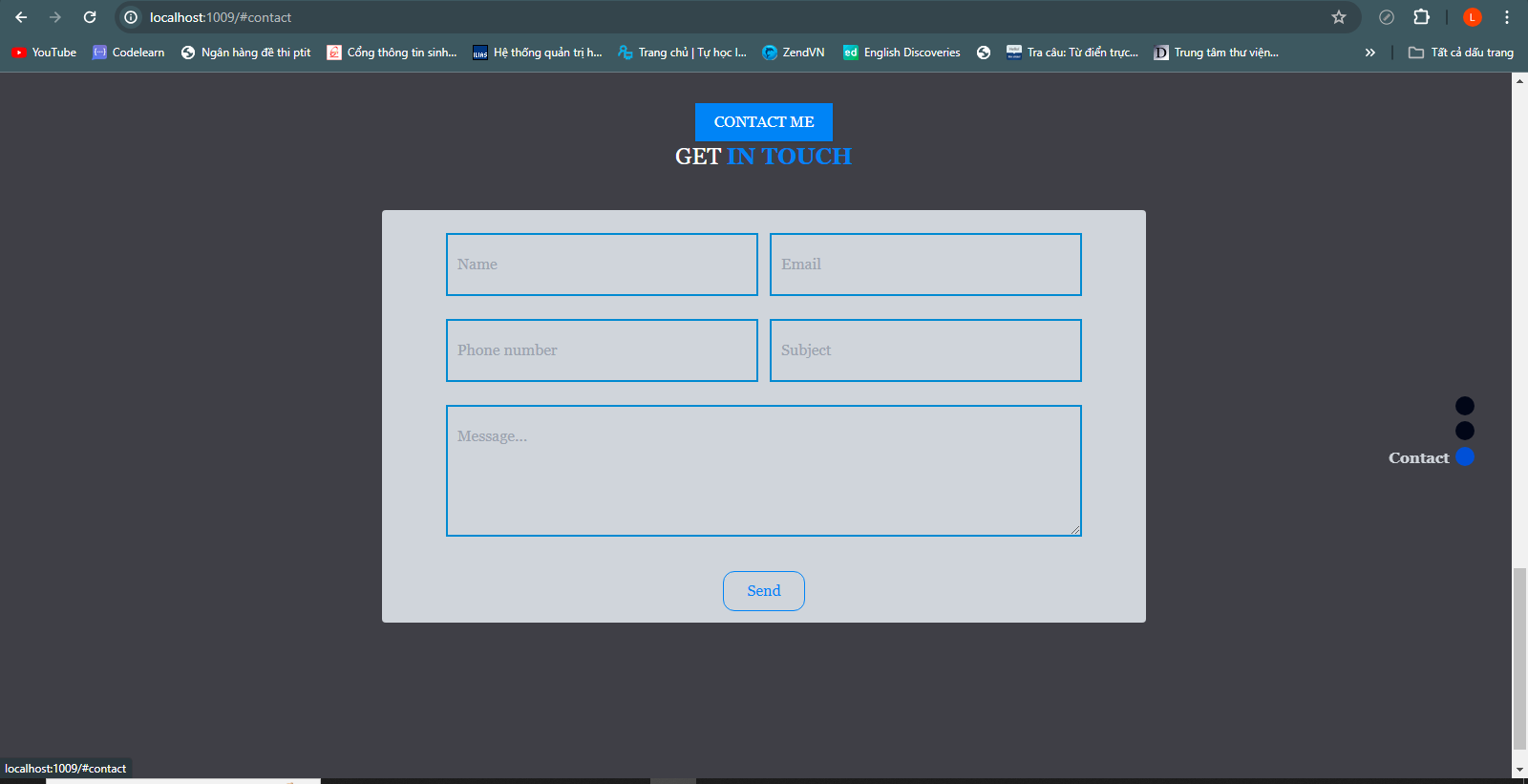
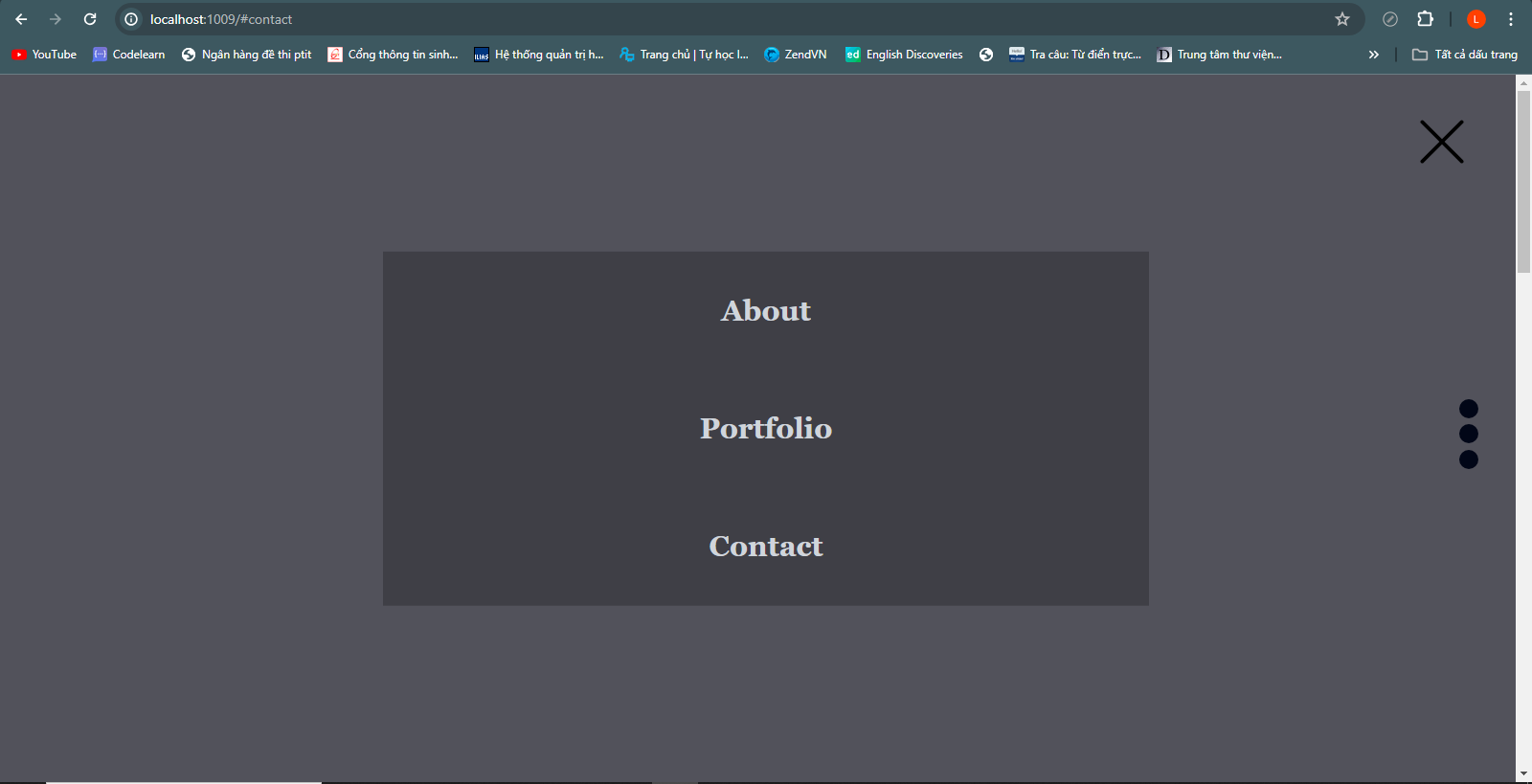
Để cài đặt những module cần thiết để chạy sản phẩm.

Sau khi cài xong, vẫn trên cửa sổ terminal, ta gõ: npm start

Mở browser lên, trên thanh URL gõ: localhost:1009

Kết quả sản phẩm:





1. **Ưu điểm và nhược điểm:**
2. **NodeJS:**

* **Ưu điểm:**

**NodeJS có thể mở rộng**

* Các ứng dụng NodeJS có khả năng mở rộng cao vì chúng hoạt động không đồng bộ vì các yêu cầu đồng thời có thể được xử lý rất hiệu quả bằng NodeJS.
* NodeJS hoạt động trên một luồng đơn nên khi có một yêu cầu đến, NodeJS sẽ bắt đầu xử lý yêu cầu đó và sẵn sàng xử lý yêu cầu tiếp theo

**Thời gian thực thi code nhanh**

* Công cụ thời gian chạy (runtime motor) JavaScript V8 được NodeJS sử dụng và cũng được Google Chrome sử dụng. Một trình bao bọc được trung tâm cung cấp cho JavaScript và vì lý do đó, công cụ thời gian chạy trở nên nhanh hơn.

**Khả năng tương thích trên nhiều nền tảng**

* Các loại hệ điều hành khác nhau như Windows, UNIX, LINUX, MacOS và các thiết bị di động khác đều có thể sử dụng NodeJS.

**Một ngôn ngữ**

* Một trong những lợi ích chính của NodeJS là khả năng viết toàn bộ cơ sở hạ tầng của bất kỳ ứng dụng web nào chỉ bằng một ngôn ngữ JavaScript. Do đó, lập trình viên không cần tốn nhiều thời gian học các ngôn ngữ khác nhau để đáp ứng.
* **Nhược điểm:**
* Nodejs được viết bằng C++ và JavaScript nên khi xử lý cần phải trải qua một quá trình biên dịch. Nếu bạn cần xử lý những ứng dụng tốn tài nguyên CPU thì không nên sử dụng Nodejs.
* Nodejs so với các ngôn ngữ khác như PHP, Ruby và Python sẽ không có sự chênh lệch quá nhiều. Nodejs có thể sẽ phù hợp với việc phát triển ứng dụng mới. Tuy nhiên khi xây dựng và triển khai dự án quan trọng thì Nodejs không phải là sự lựa chọn hoàn hảo nhất.

1. **Tailwind CSS:**

* **Ưu điểm:**
* Không cần viết một dòng CSS nào mà chỉ cần code html thôi
* Các class dễ sử dụng, dễ làm quen
* Tài liệu chi tiết, đầy đủ, dễ hiểu
* Giảm thiểu việc đặt tên class không đúng khi làm teamwork
* Chỉnh sửa code nhanh
* Hỗ trợ cài đặt với nhiều framework front-end khác như react, vuejs, …
* **Nhược điểm:**
* Số lượng code html trên một dòng cực kỳ dài.
* Cần có kiến thức về CSS thì mới sử dụng tốt được.
* Tốn khá nhiều thời gian sử dụng để làm quen, nhớ các tên class.

1. **Tổng kết**

Các ứng dụng được xây dựng bằng NodeJS cực kỳ nhanh và phản hồi nhanh. Ngoài ra, NodeJS còn thúc đẩy việc tái sử dụng code và tính module hóa thông qua trình quản lý gói, NPM, nơi lưu trữ một hệ sinh thái rộng lớn gồm các thư viện nguồn mở. Điều này thúc đẩy sự phát triển nhanh chóng, tiết kiệm cả thời gian và công sức.

Về Tailwind CSS với những dự án nhanh, có độ phức tạp về layout, độ mở rộng lớn, nhiều người cùng code dự án thì nên sài framework Tailwind CSS. Vì nó có độ linh động cao, chỉnh sửa code nhanh, việc đặt tên đã có một nguyên tắc nhất định và sẽ không có trường hợp bị xung đột code CSS.

Nguồn tham khảo:

<https://itviec.com/blog/nodejs-la-gi/>

<https://tailwindcss.com/docs/installation/>