

Giáo viên hướng dẫn
(Ký tên và ghi rõ họ tên)

dẫn

This image shows a full page of a handwriting practice worksheet. It consists of numerous horizontal rows, each defined by two parallel dotted lines. The rows are evenly spaced and extend across the entire width of the page, providing a guide for letter height and placement. There is no text or other markings on the page.

Nguyễn Minh Hải Đăng

LỜI CẢM ƠN

Tôi xin gửi lời cảm ơn sâu sắc và chân thành nhất của mình đến với quý thầy, cô Trường đại học Trà Vinh nói chung và quý thầy, cô giảng viên của Khoa Kỹ thuật và Công nghệ, Bộ môn Công nghệ thông tin nói riêng. Cũng như là sự hỗ trợ và giúp đỡ tận tình của cô Phạm Thị Trúc Mai đã trực tiếp chỉ bảo, giải đáp thắc mắc, định hướng và hướng dẫn tôi trong cả quá trình làm đồ án cơ sở ngành. Sự hỗ trợ và động viên không ngừng từ cô đã giúp tôi vượt qua được những thử thách và phát triển kỹ năng một cách đáng kể

Tôi cũng muốn gửi lời cảm ơn đến tất cả các bạn bè, những người đồng đội đã cùng tôi học tập. Sự khích lệ, lời khuyên và sự hỗ trợ của các bạn đã giúp tôi vượt qua được những khó khăn cản trở trong quá trình thực hiện nghiên cứu đồ án.

Cuối cùng tôi muốn gửi lời tri ân đến những nguồn tài liệu, các dự án đã được tham khảo và mọi nguồn lực hữu ích mà tôi đã sử dụng để hoàn thành đồ án này.

Một lần nữa, xin chân thành cảm ơn tất cả sự giúp đỡ và ủng hộ của mọi người. Đây là kết quả của sự hỗ trợ và đóng góp của một đội ngũ rất lớn và tôi cảm thấy rất may mắn, biết ơn vì điều đó.

Chân thành cảm ơn!

MỤC LỤC

LỜI CẢM ƠN	3
MỤC LỤC	4
DANH MỤC HÌNH ẢNH – BẢNG BIỂU	6
KÍ HIỆU CÁC CỤM TỪ VIẾT TẮT	8
TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH.....	9
MỞ ĐẦU	Error! Bookmark not defined.
CHƯƠNG 1: TỔNG QUAN	12
CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT.....	14
2.1 GIỚI THIỆU VỀ NGÔN NGỮ HTML VÀ CSS	14
2.1.1 Giới thiệu về HTML.....	14
2.1.2 Giới thiệu về CSS.....	15
2.2 GIỚI THIỆU VỀ NGÔN NGỮ JAVASCRIPT VÀ NODEJS.....	16
2.2.1 Giới thiệu về ngôn ngữ JS	16
2.2.2 Giới thiệu về ngôn ngữ NodeJS	16
2.3 Giới thiệu về RESTful API và MySQL.....	16
2.3.1 API (Application Programming Interface).....	16
2.3.2 Giới thiệu về MySQL.....	17
2.4 Mô hình MVC	18
CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU	19
3.1 Thiết kế dữ liệu cho website	19
3.1.1 Cơ sở dữ liệu cho dự án:.....	19
3.1.2 Mô tả các bảng:	19
3.1.3 Các dữ liệu trong MySQL	21
3.1.4. Yêu cầu chức năng:	22
3.2. Tạo và thiết kế tổ chức các thư mục cho trang quản trị	22
3.3. Thiết kế trang quản trị cho website	23
3.4. Thiết kế và tạo giao diện người dùng	26
CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU	28
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	30

DANH MỤC TÀI LIỆU THAM KHẢO	31
PHỤ LỤC.....	32

DANH MỤC HÌNH ẢNH – BẢNG BIỂU

Hình 1: Tổng quan về RESTful API	17
Hình 2: Mô hình ERD	19
Hình 3: Ảnh dữ liệu trong bảng Nhà cung cấp trong MySQL	21
Hình 4: Ảnh dữ liệu bản sản phẩm trong MySQL	22
Hình 5: Các thông tin cần điền vào NodeJS	22
Hình 6: Tổ chức cây thư mục cho dự án	23
Hình 7: Giao diện Admin của website	24
Hình 8: Giao diện hiển thị thông tin sản phẩm	24
Hình 9: Giao diện thêm sản phẩm	25
Hình 10: Giao diện chỉnh sửa sản phẩm	25
Hình 11: Đưa dự án lên Github	26
Hình 12: Giao diện người dùng	26
Hình 13: Các sản phẩm được trình bày trên website	27
Hình 14: Giao diện hoàn chỉnh của thêm sản phẩm	28
Hình 15: Giao diện hoàn chỉnh của chỉnh sửa thông tin sản phẩm	28
Hình 16: Giao diện trang quản trị cửa hàng	29
Hình 17: Giao diện của người dùng	29

Bảng 1: Bảng Sản phẩm	19
Bảng 2: Bảng Nhà cung cấp	20
Bảng 3: Bảng Danh mục sản phẩm	21

KÍ HIỆU CÁC CỤM TỪ VIẾT TẮT

Tên Viết Tắt	Tên Tiếng Anh
API	Application Programming Interface
HTML	Hypertext Markup Language
CSS	Cascading Style Sheets
JS	JavaScript
W3C	World Wide Web Consortium

TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH

Vấn đề nghiên cứu:

Nghiên cứu và tìm hiểu về API, nghiên cứu và tạo API dữ liệu về quản lý của hàng thiết bị điện tử và ứng dụng vào dự án website quản lý của hàng thiết bị điện tử.

Tìm hiểu về nghiên cứu về HTML, CSS, Javascript, NodeJS để xây dựng front-end và back-end.

Các hướng tiếp cận:

Tìm hiểu và nghiên cứu về cách API hoạt động, các sử dụng API để tương tác với các dịch vụ, phần mềm khác nhau. Hiểu rõ hơn về các loại API, cách chúng hoạt động và cách tích hợp API vào trang web. Xác định được cách lưu trữ được những cơ sở dữ liệu và liên kết chúng với API. Tạo ra giao diện người dùng dễ sử dụng. Xây dựng một trang web quản lý bán hàng, tạo ra một nền tảng kinh doanh trực tuyến để quản lý quá trình bán hàng.

Cách giải quyết vấn đề:

Nghiên cứu sâu hơn về API, tìm hiểu và đào sâu vào tài liệu API, thực hiện các tương tác với các API để hiểu rõ cách hoạt động của chúng. Sử dụng những ngôn ngữ lập trình phù hợp để tạo ra trang web quản lý bán hàng, tích hợp các API đã tạo và xây dựng cơ sở dữ liệu cần thiết. Thử nghiệm tích hợp API, sửa lỗi và điều chỉnh trang web để đảm bảo hoạt động ổn định và hiệu quả.

Kết quả đạt được:

Tìm hiểu và nghiên cứu đề tài này đã mang lại cho tôi thêm kiến thức và kỹ năng về mảng lập trình. Hiểu rõ cách tạo và ứng dụng API vào website, xây dựng được một website quản lý của hàng thiết bị điện tử hoàn chỉnh và tích góp được nhiều kiến thức và kỹ năng chuyên môn cho bản thân.

MỞ ĐẦU

Lí do chọn đề tài:

Thị trường thương mại điện tử là một thị trường phát triển ngày càng mạnh mẽ và đang trở thành một phần quan trọng trong hệ thống kinh doanh hiện nay. Đồng thời tạo ra một website bán hàng hiệu quả không những là nơi quảng cáo và trưng bày sản phẩm mà còn là một cửa hàng giao dịch trực tuyến, giúp người tiêu dùng tiếp cận đến sản phẩm nhanh chóng và tiện lợi. Nghiên cứu về cách sử dụng API để xây dựng nên một website bán hàng giúp hiểu rõ hơn về cách tích hợp các công nghệ mới vào mô hình kinh doanh trực tuyến. Sử dụng API để tạo ra một trang web không chỉ tạo ra tính tương tác cao mà còn mở ra khả năng kết nối với nhiều dịch vụ khác nhau như thanh toán trực tuyến, quản lý hàng hóa, quảng cáo và dịch vụ vận chuyển.

Việc xây dựng một website thương mại sẽ đi đôi với việc có một lượng người dùng rất lớn, cần phải có một hệ thống hỗ trợ tài nguyên và vận hành ổn định, giúp xử lý các yêu cầu đồng thời một cách hiệu quả từ người dùng nên sử dụng môi trường NodeJS là một sự lựa chọn đúng đắn.

NodeJS là một môi trường thực thi mã nguồn mở được xây dựng dựa trên Javascript và đã trở thành một trong những công nghệ phổ biến trong việc xây dựng ứng dụng web hiện đại và có sức ảnh hưởng lớn trong cộng đồng phát triển phần mềm.

Với sự phát triển của lĩnh vực thương mại điện tử nên những kiến thức hiểu biết và kỹ năng liên quan đến thiết kế website bán hàng sẽ mang đến nhiều cơ hội phát triển trong tương lai nên em đã quyết định chọn đề tài “Nghiên cứu API và NodeJS để xây dựng website quản lý cửa hàng thiết bị điện tử”.

Đối tượng nghiên cứu:

Nghiên cứu các API liên quan đến lĩnh vực thương mại điện tử như API hiển thị sản phẩm, API cập nhật sản phẩm, API sửa sản phẩm và API xóa sản phẩm. Tập trung vào quá trình tích hợp các API này vào một trang web bán hàng, gọi các endpoint, xử lý dữ liệu và hiển thị thông tin cần thiết cho người dùng.

Sử dụng các ngôn ngữ như HTML, CSS, JS và môi trường NodeJS để tạo ra một giao diện trang web phù hợp với API đã tạo. Tạo ra một giao diện trang web thân thiện với người dùng.

CHƯƠNG 1: TỔNG QUAN

API (Application Programming Interface) là một cách thức cho phép các ứng dụng hoặc phần mềm khác tương tác với nhau. Nó cung cấp một bộ các quy tắc, giao thức và công cụ để cho phép các phần mềm khác nhau giao tiếp và trao đổi dữ liệu với nhau. Tạo API và xây dựng một website quản lý bán hàng, sử dụng các kiến thức từ việc tìm hiểu và nghiên cứu API để phát triển một trang web, xác định được những cách tương tác với API, gọi các endpoint và xử lý những dữ liệu được trả về. Xây dựng được một giao diện thân thiện với người dùng, sử dụng API để hiển thị các thông tin một cách rõ ràng và thuận tiện cho người dùng. Tối ưu hóa website để đảm bảo có thể tải trang nhanh, tương tác mượt mà và có khả năng mở rộng và phát triển thêm.

Sử dụng môi trường NodeJS để xây dựng các ứng dụng web đa nền tảng, từ các website đơn giản đến các ứng dụng web phức tạp. NodeJS thường được sử dụng để xây dựng các ứng dụng và website có thể tương tác và kết nối với nhau thông qua các phương thức như là HTTP. NodeJS có thể cung cấp khả năng đa luồng và đa sự kiện và làm cho nó rất phù hợp cho việc phát triển các ứng dụng thời gian thực hay là các ứng dụng cập nhật dữ liệu liên tục. Với NodeJS, bạn có thể xây dựng và quản lý các dịch vụ nhỏ và độc lập, giúp dễ dàng bảo trì hoặc mở rộng hơn. NodeJS còn có thể cung cấp các thư viện để hỗ trợ cho việc phát triển các ứng dụng IoT, cho phép kết nối và quản lý các thiết bị thông minh.

Hiện nay có thể sử dụng đồng thời cả NodeJS và API với nhau để thiết kế các dự án trang web bán hàng. NodeJS có thể sử dụng để tạo ra các API backend và có thể sử dụng NodeJS để xử lý những yêu cầu từ người dùng, truy vấn cơ sở dữ liệu, xác thực và trả dữ liệu theo yêu cầu. Có thể sử dụng NodeJS để tạo ra các ứng dụng hoặc website kết nối với các dịch vụ bên ngoài thông qua các API. Sử dụng NodeJS để xử lý và gửi các dữ liệu đến API. Điều này bao gồm cả việc lấy dữ liệu từ các nguồn khác nhau, xử lý dữ liệu và trả về kết quả cho ứng dụng hoặc là trang web của chúng ta. NodeJS cho phép mở rộng ứng dụng một cách linh hoạt, đặc biệt là có thể giúp ứng dụng dễ dàng tương tác với các hệ thống khác và mở rộng chức năng một cách linh hoạt. Kết hợp giữa NodeJS với API cũng cho phép tối ưu hóa việc giao tiếp giữa dữ

liệu tạm thời để tăng tốc độ truy cập và cải thiện hiệu suất của một ứng dụng hay là một trang web.

Sự kết hợp giữa NodeJS và API mở ra rất nhiều khả năng, từ việc xây dựng các dịch vụ backend cho ứng dụng của chúng ta đến việc kết nối và tương tác với các dịch vụ và nguồn dữ liệu bên ngoài.

Thông qua dự án này với sự kết hợp giữa NodeJS và API, tôi hi vọng bản thân có thể phát triển hơn về kiến thức lẫn kỹ năng lập trình một trang web bán hàng và tôi cũng hy vọng có thể đóng góp một phần nhỏ nào đó vào sự phát triển của lĩnh vực thương mại điện tử nói chung và lĩnh vực công nghệ thông tin nói riêng.

CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT

2.1 GIỚI THIỆU VỀ NGÔN NGỮ HTML VÀ CSS

2.1.1 Giới thiệu về HTML

HTML, viết tắt của HyperText Markup Language, là một ngôn ngữ đánh dấu sử dụng để xây dựng và cấu trúc trang web. HTML là một phần quan trọng của bất kỳ trang web nào trên Internet và đóng vai trò quan trọng trong việc định nghĩa cấu trúc và nội dung của trang web.

HTML (HyperText Markup Language) và CSS (Cascading Style Sheets) thường được kết hợp để điều chỉnh giao diện của web, cùng với Javascript để thêm tính năng tương tác động cho trang web cùng nhau tạo ra trải nghiệm web tốt hơn.

Ngôn ngữ HTML đã trải qua một quá trình phát triển rất dài từ khi nó được phát minh và dưới đây là một số các điểm chính trong lịch sử phát triển của ngôn ngữ:

HTML 1.0: Là phiên bản đầu tiên của HTML được tạo ra vào năm 1991 bởi Tim Berner-Lee, người được coi là cha đẻ của World Wide Web. HTML 1.0 đã định nghĩa được cách tạo ra các trang web đơn giản với các phần tử cơ bản như là tiêu đề, đoạn văn bản, liên kết và hình ảnh.

HTML 2.0: Được ra mắt vào năm 1995 và được bổ sung thêm một số tính năng mới như là biểu mẫu và hỗ trợ ký tự quốc tế.

HTML 3.2 : Xuất hiện vào năm 1997, HTML 3.2 đã mang đến nhiều cải tiến như các bảng, màu sắc và hình ảnh nền. Điều này khiến cho việc thiết kế web trở nên phong phú hơn.

HTML 4.01 : Ra mắt vào năm 1999, HTML 4.01 tiếp tục mở rộng và cung cấp các tiêu chuẩn mới như CSS để quản lý giao diện của trang web cũng như với các phần tử mới.

XHTML (HTML 5.0): XHTML hay Extensible HTML được ra đời để tạo ra một phiên bản nghiêm ngặt hơn của HTML. HTML 5.0 mang đến nhiều cải tiến lớn như việc hỗ trợ video, audio, canvas (dùng để vẽ đồ họa), local storage và các API cho ứng dụng web.

HTML5 và phiên bản hiện tại: HTML5 không chỉ là một phiên bản của HTML mà còn là một tiêu chuẩn mà web hiện đại đang sử dụng. Nó liên tục được cập nhật và mở rộng thêm nhiều tính năng mới, cung cấp cơ sở cho việc phát triển các ứng dụng web đa dạng và phức tạp.

2.1.2 Giới thiệu về CSS

CSS là viết tắt của Cascading Style Sheets, một ngôn ngữ thiết kế đơn giản, xử lý một phần giao diện của trang web. CSS mô tả cách các phần tử HTML hiển thị trên màn hình và các phương tiện khác.

Sử dụng CSS, chúng ta có thể kiểm soát màu chữ, cỡ chữ, kiểu chữ, khoảng cách giữa các đoạn văn bản, kích thước của các thành phần trên trang web, màu nền, thiết kế bố cục và cách trang web hiển thị trên những màn hình có kích thước khác nhau cũng như hàng loạt hiệu ứng khác. CSS rất hữu ích và tiện lợi. Nó có thể kiểm soát tất cả các trang trên một website. Các stylesheet ngoài được lưu trữ dưới dạng các tập tin .CSS. CSS được kết hợp với ngôn ngữ đánh dấu HTML và XHTML.

Bố cục CSS thường chủ yếu dựa vào hình hộp và mỗi hộp đều chiếm những khoảng trống trên trang web các thuộc tính như:

Padding: Gồm không gian xung quanh nội dung (ví dụ: xung quanh đoạn văn bản).

Border: Là đường viền nằm ngay bên ngoài phần đệm.

Margin: Là khoảng cách xung quanh bên ngoài của phần tử.

Đây là một số mốc quan trọng trong lịch sử phát triển của CSS;

CSS Level 1 (1996): CSS được giới thiệu lần đầu tiên bởi W3C vào năm 1996 với CSS Level 1, cung cấp một số thuộc tính cơ bản cho việc kiểm soát kiểu dáng của một trang web.

CSS Level 2 (1998): Phiên bản CSS này mở rộng khả năng kiểm soát và bổ sung nhiều tính năng mới như vị trí, định vị và giao diện cho người dùng.

CSS Level 2 Revision 1 (2004): Bản CSS sửa đổi này bổ sung thêm một số các tính năng mới và sửa lỗi từ CSS Level 2.

CSS Level 3 (2001 - hiện tại): CSS Level 3 không phải là một phiên bản cụ thể mà là một loạt các bản mở rộng của CSS, mỗi bản mở rộng cung cấp nhiều tính năng mới.

CSS4 và Beyond: CSS4 đang dần được phát triển để thay thế CSS3 với việc cung cấp các tính năng mới và nâng cấp cho CSS3. Tuy nhiên thay vì là một phiên bản cụ thể thì CSS4 được coi là một tập hợp các module độc lập với nhau, mỗi module điều chỉnh và bổ sung các tính năng cụ thể cho CSS.

2.2 GIỚI THIỆU VỀ NGÔN NGỮ JAVASCRIPT VÀ NODEJS

2.2.1 Giới thiệu về ngôn ngữ JS

JS (JavaScript) là một ngôn ngữ lập trình đa dụng, thường được sử dụng trong việc phát triển ứng dụng web. Được tạo ra ban đầu để làm cho trang web trở nên tương tác hơn và động hơn, JS đã phát triển thành một trong những ngôn ngữ phổ biến nhất trên thế giới và được sử dụng trong nhiều môi trường khác nhau.

JS được tạo trong mười ngày bởi Brendan Eich, một nhân viên của Netscape, vào tháng 9 năm 1995. Được đặt tên đầu tiên là Mocha, tên của nó được đổi thành Mona rồi LiveScript trước khi thật sự trở thành JavaScript nổi tiếng như bây giờ. Phiên bản đầu tiên của ngôn ngữ này bị giới hạn độc quyền bởi Netscape và chỉ có các tính năng hạn chế, nhưng nó tiếp tục phát triển theo thời gian, nhờ một phần vào cộng đồng các lập trình viên đã liên tục làm việc với nó.

2.2.2 Giới thiệu về ngôn ngữ NodeJS

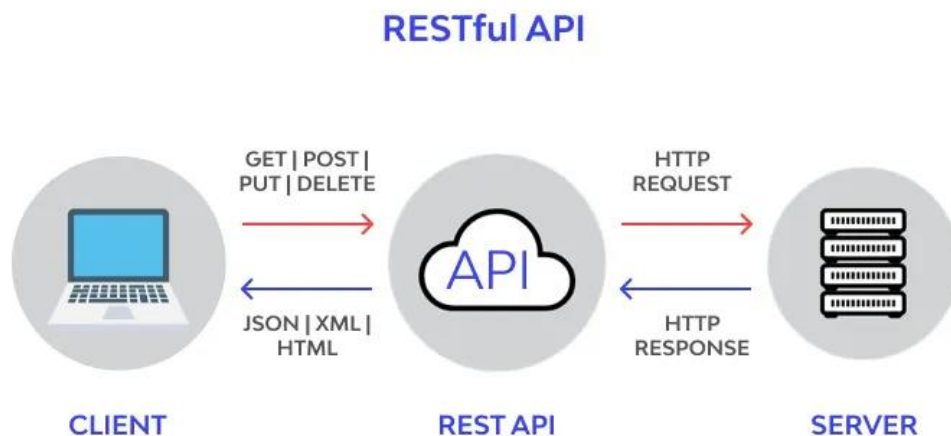
NodeJS là một môi trường thực thi mã JavaScript nằm ở phía máy chủ, cho phép chạy mã JavaScript không chỉ trên trình duyệt mà còn trên máy chủ. Được xây dựng dựa trên JavaScript Engine của Google Chrome (V8 JavaScript Engine), NodeJS là một công cụ mạnh mẽ cho việc phát triển ứng dụng web và các ứng dụng máy chủ hiệu quả.

Theo số liệu thống kê của Stack Overflow về các Framework, Nodejs đã xuất sắc giành vị trí số một với số lượng người dùng lên tới 50%. NodeJS thường được sử dụng để xây dựng các ứng dụng web thời gian thực như ứng dụng chat, trò chơi trực tuyến, ứng dụng IoT (Internet of Things), API, và các ứng dụng mạng phức tạp khác. Sức mạnh của nó nằm ở khả năng xử lý đa nhiệm và đáp ứng yêu cầu một cách nhanh chóng và linh hoạt.

2.3 Giới thiệu về RESTful API và MySQL

2.3.1 API (Application Programming Interface)

API (Application Programming Interface) là một cách thức cho phép các ứng dụng hoặc phần mềm khác tương tác với nhau. Nó cung cấp một bộ các quy tắc, giao thức và công cụ để cho phép các phần mềm khác nhau giao tiếp và trao đổi dữ liệu với nhau. API có thể được sử dụng để truy cập các dịch vụ, chia sẻ dữ liệu giữa các ứng dụng, và thực hiện các chức năng cụ thể. REST (Representational State Transfer) là một kiến trúc phần mềm để xây dựng các dịch vụ web dựa trên các giao thức web chuẩn như HTTP. RESTful API sử dụng các nguyên tắc của REST để thiết kế và triển khai các API, cho phép các ứng dụng giao tiếp với nhau một cách đơn giản, linh hoạt và hiệu quả.



Hình 1: Tổng quan về RESTful API

2.3.2 Giới thiệu về MySQL

MySQL là một hệ quản trị cơ sở dữ liệu (Hệ quản trị cơ sở dữ liệu quan hệ - RDBMS) mã nguồn mở phổ biến được sử dụng rộng rãi trên toàn thế giới. Được phát triển và duy trì bởi Oracle Corporation, MySQL cung cấp một hệ thống quản lý cơ sở dữ liệu mạnh mẽ, ổn định và linh hoạt. MySQL là một trong số các phần mềm RDBMS. RDBMS và MySQL thường được cho là một vì độ phổ biến quá lớn của MySQL. Các ứng dụng web lớn nhất như Facebook, Twitter, YouTube, Google, và Yahoo! đều dùng MySQL cho mục đích lưu trữ dữ liệu.

2.4 Mô hình MVC

Mô hình MVC (Model-View-Controller) là một kiến trúc phần mềm phổ biến được sử dụng trong việc phát triển ứng dụng web và các ứng dụng khác để tách biệt và tổ chức code một cách rõ ràng.

Model (Mô hình): Đây là phần của ứng dụng chịu trách nhiệm xử lý dữ liệu và logic. Model biểu diễn cấu trúc dữ liệu, tương tác với cơ sở dữ liệu, xử lý logic kinh doanh và cung cấp dữ liệu cho phần View hoặc Controller. Nó không biết gì về cách hiển thị dữ liệu hay người dùng sử dụng nó như thế nào.

View (Giao diện): View là phần hiển thị dữ liệu cho người dùng cuối. Nó có trách nhiệm hiển thị thông tin từ Model một cách trực quan và giao diện người dùng thân thiện. View nhận dữ liệu từ Model và hiển thị nó một cách phù hợp với định dạng, trình bày cho người dùng.

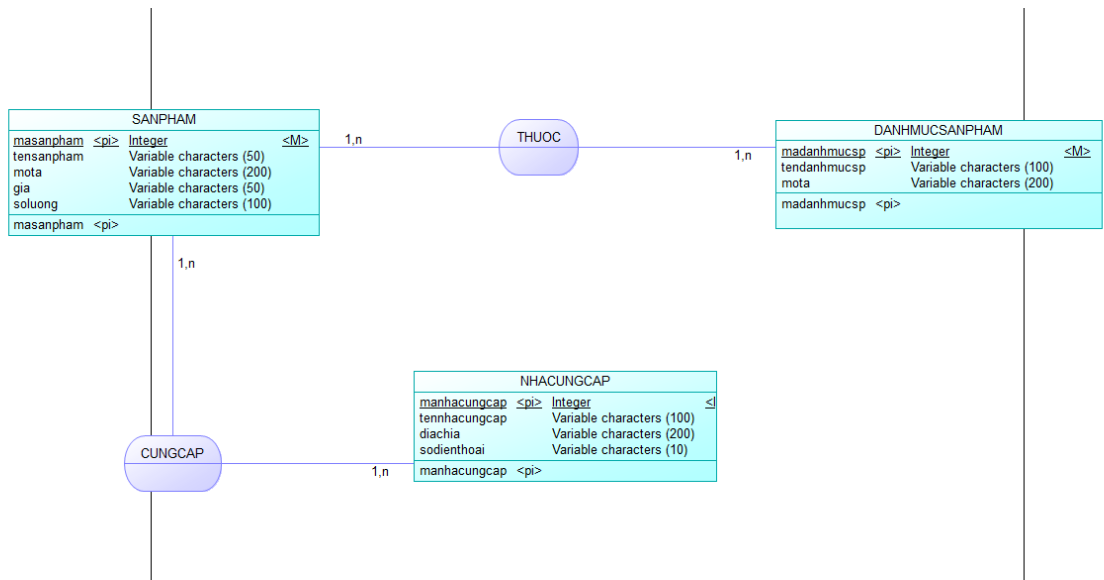
Controller (Bộ điều khiển): Controller là phần trung gian giữa Model và View. Nó nhận các yêu cầu từ người dùng thông qua giao diện (View), xử lý các sự kiện và tương tác với Model để lấy dữ liệu cần thiết và sau đó cập nhật lại View để hiển thị dữ liệu cho người dùng. Controller làm trung gian để điều hướng luồng dữ liệu và tương tác giữa Model và View.

Mô hình MVC giúp tách biệt logic ứng dụng ra thành các phần riêng biệt và có tổ chức. Điều này giúp dễ dàng bảo trì, mở rộng ứng dụng và cải thiện tái sử dụng code. Đồng thời, nó cũng tạo điều kiện thuận lợi cho việc làm việc nhóm, vì mỗi thành viên có thể tập trung vào một phần cụ thể của ứng dụng mà không ảnh hưởng quá nhiều đến các phần khác.

CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU

3.1 Phân tích và thiết kế dữ liệu cho website

3.1.1 Cơ sở dữ liệu cho dự án:



Hình 2: Mô hình ERD

SANPHAM (masanpham, tensanpham, mota, gia, soluong)

NHACUNGCAP (manhacungcap, tennhacungcap, diachi, sodienthoai)

DANHMUCSANPHAM(madanhmucsp, tendanhmucsp, mota)

3.1.2 Mô tả các bảng:

Tên bảng: SANPHAM

Bảng 1: Bảng Sản phẩm

Tên tắt thuộc tính	Diễn giải	Loại giá trị	Kiểu dữ liệu	Miền giá trị	Chiều dài	Ghi chú
masanpham	Mã sản phẩm	Bắt buộc	Nvarchar	Khóa chính	255	
tensanpham	Tên sản phẩm		Nvarchar		255	

mota	Mô tả		Text		255	
gia	Giá		Decimal		(10,2)	
soluong	Số lượng		Int			

Tên bảng: NHACUNGCAP

Bảng 2: Bảng Nhà cung cấp

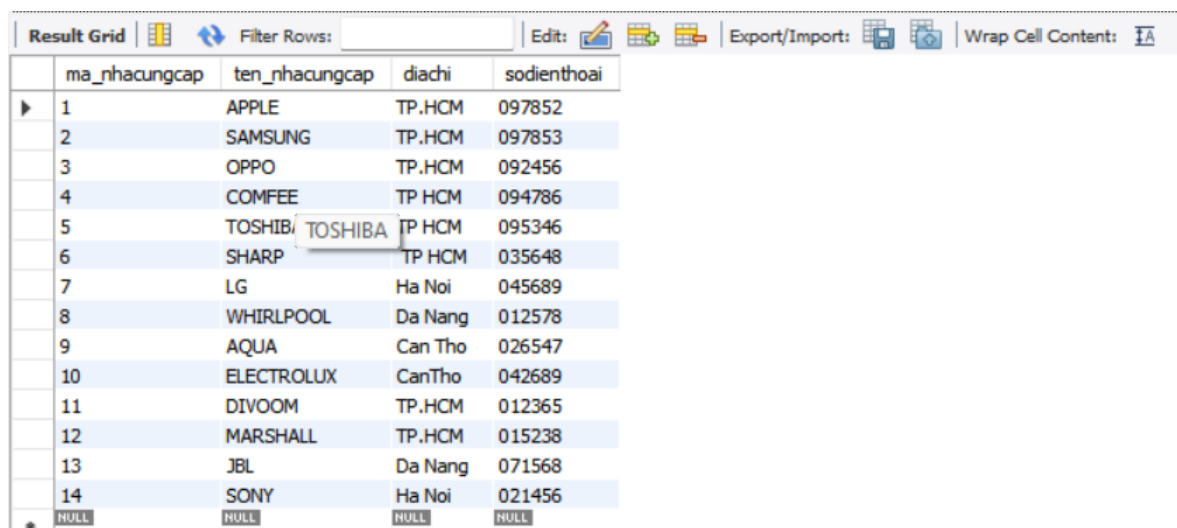
Tên tắt thuộc tính	Diễn giải	Loại giá trị	Kiểu dữ liệu	Miền giá trị	Chiều dài	Ghi chú
manhacungcap	Mã nhà cung cấp	Bắt buộc	Int	Khóa chính		
tennhacungcap	Tên nhà cung cấp		Varchar		255	
diachi	Địa chỉ		Varchar		255	
sodienthoai	Số điện thoại		Varchar		255	

Tên bảng: DANH MUCSANPHAM

Bảng 3: Bảng Danh mục sản phẩm

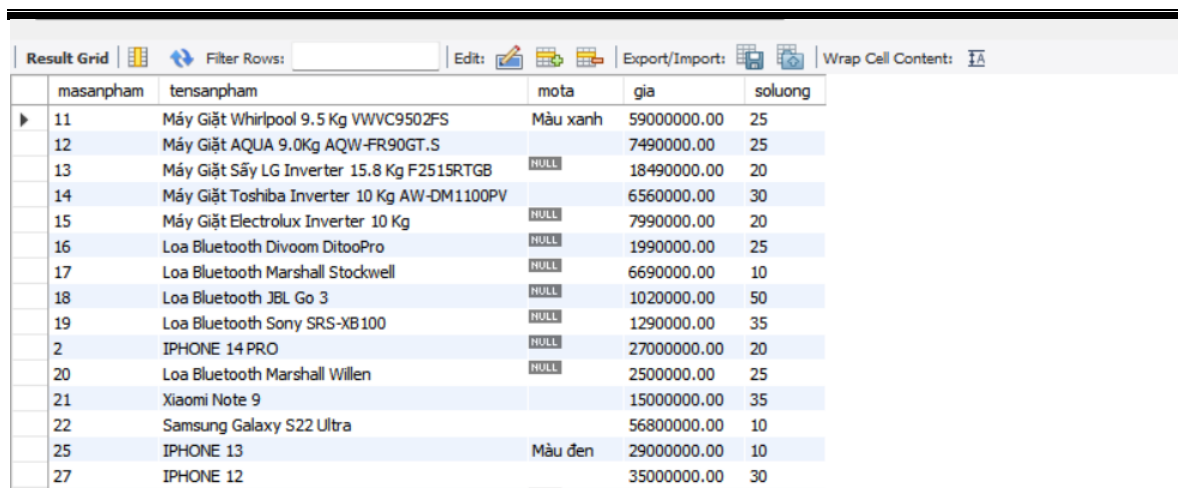
Tên tắt thuộc tính	Diễn giải	Loại giá trị	Kiểu giá trị	Miền giá trị	Chiều dài	Ghi chú
madanhmucsp	Mã danh mục sản phẩm	Bắt buộc	Int	Khóa chính		
tendanhmucsp	Tên danh mục sản phẩm		Varchar		255	
mota	Mô tả		Text			

3.1.3 Các dữ liệu trong MySQL



	ma_nhacungcap	ten_nhacungcap	diachi	sodienthoai
1		APPLE	TP.HCM	097852
2		SAMSUNG	TP.HCM	097853
3		OPPO	TP.HCM	092456
4		COMFEE	TP.HCM	094786
5		TOSHIBA	TP.HCM	095346
6		SHARP	TP.HCM	035648
7		LG	Ha Noi	045689
8		WHIRLPOOL	Da Nang	012578
9		AQUA	Can Tho	026547
10		ELECTROLUX	Can Tho	042689
11		DIVOOM	TP.HCM	012365
12		MARSHALL	TP.HCM	015238
13		JBL	Da Nang	071568
14		SONY	Ha Noi	021456
*	NULL	NULL	NULL	NULL

Hình 3: Ảnh dữ liệu trong bảng Nhà cung cấp trong MySQL



	masanpham	tensanpham	mota	gia	soluong
11		Máy Giặt Whirlpool 9.5 Kg WVVC9502FS	Màu xanh	59000000.00	25
12		Máy Giặt AQUA 9.0Kg AQW-FR90GT.S		74900000.00	25
13		Máy Giặt Sấy LG Inverter 15.8 Kg F2515RTGB	NULL	184900000.00	20
14		Máy Giặt Toshiba Inverter 10 Kg AW-DM1100PV		65600000.00	30
15		Máy Giặt Electrolux Inverter 10 Kg	NULL	79900000.00	20
16		Loa Bluetooth Divoom DitooPro	NULL	19900000.00	25
17		Loa Bluetooth Marshall Stockwell	NULL	66900000.00	10
18		Loa Bluetooth JBL Go 3	NULL	10200000.00	50
19		Loa Bluetooth Sony SRS-XB100	NULL	12900000.00	35
2		IPHONE 14 PRO	NULL	270000000.00	20
20		Loa Bluetooth Marshall Willen	NULL	25000000.00	25
21		Xiaomi Note 9		150000000.00	35
22		Samsung Galaxy S22 Ultra		568000000.00	10
25		IPHONE 13	Màu đen	290000000.00	10
27		IPHONE 12		350000000.00	30

Hình 4: Ảnh dữ liệu bản sản phẩm trong MySQL

3.1.4. Yêu cầu chức năng:

Chức năng cho admin quản lí:

- Admin có thể thêm sản phẩm mới.
- Xóa sản phẩm khi hết hàng.
- Chỉnh sửa thông tin của sản phẩm.

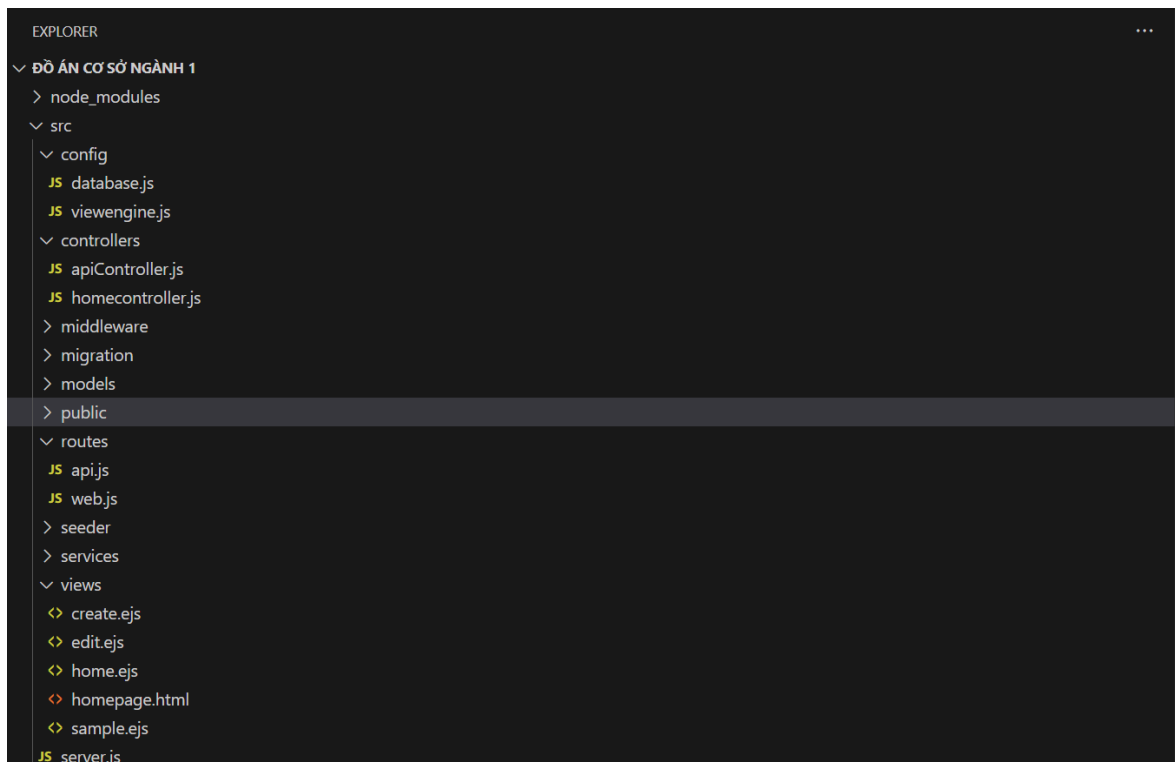
3.2. Tạo và thiết kế tổ chức các thư mục cho trang quản trị

Tải môi trường NodeJS về sau đó vào trong phần terminal của Visual Code Studio sử dụng câu lệnh “npm init” và điền các thông tin cần thiết vào.

```
{
  "name": "hadasoo",
  "version": "1.0.0",
  "description": "Do an co so nganh",
  "main": "server.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1",
    "start": "nodemon ./src/server.js"
  },
}
```

Hình 5: Các thông tin cần điền vào NodeJS

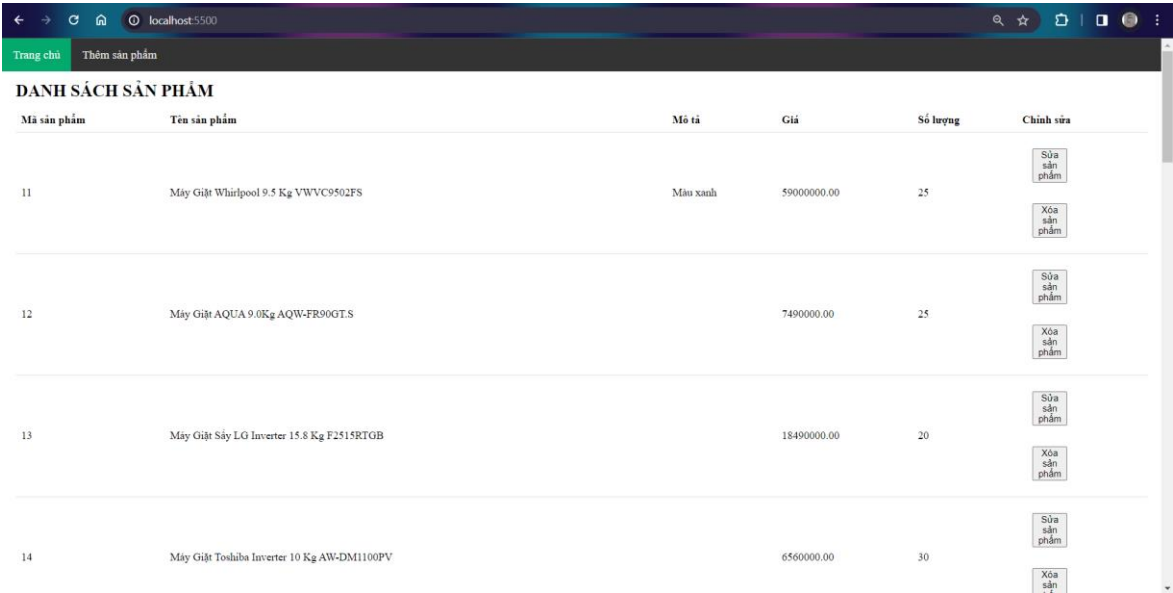
Tiến hành tạo và thiết kế dự án NodeJS theo cây thư mục để dễ quản lý và sửa lỗi, bắt đầu thêm vào những thư viện cần thiết như express, nodemon, mysql2,...sau đó chạy thử trong môi trường NodeJS. Tạo dự án theo mô hình MVC với các điều kiện sau: trang home.ejs sẽ là trang chủ của dự án , trang route sẽ là nơi chứa các đường link và các route, trang homecontroller sẽ là nơi chứa các hàm dùng để thực thi các câu lệnh thêm, sửa, xóa và hiển thị dữ liệu, trang config dùng để định nghĩa các file dữ liệu sql cho thư mục.



Hình 6: Tổ chức cây thư mục cho dự án

3.3. Thiết kế trang quản trị cho website

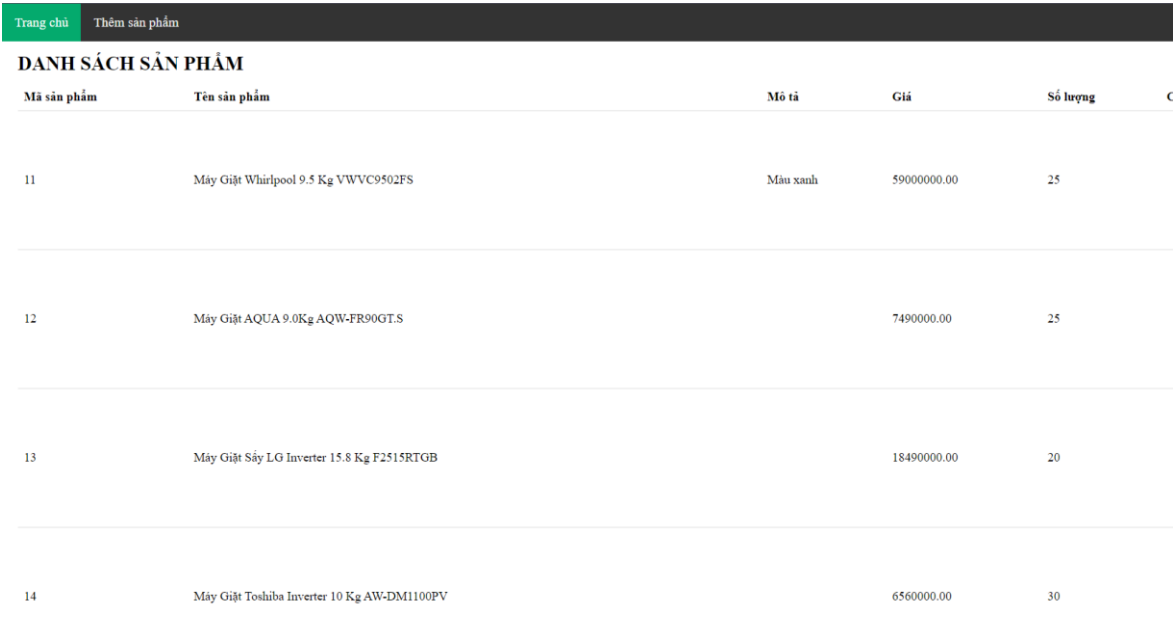
Tạo giao diện cho trang quản trị website sau đó liên kết các cơ sở dữ liệu với trang quản trị thông qua thư viện đã được cài đặt trong môi trường NodeJS là Mysql2 và hiển thị các dữ liệu lên bằng cách thông qua các lệnh được tạo ra từ thư mục homecontroller.js.



Hình 7: Giao diện Admin của website

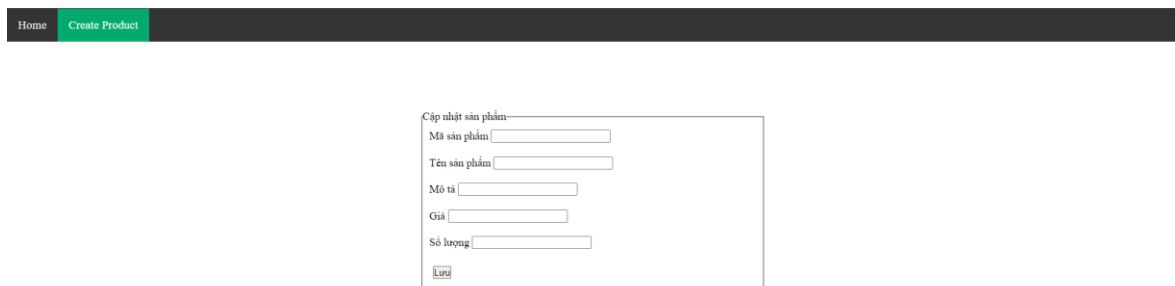
Tạo thêm các chức năng hiển thị, thêm, xóa, sửa sản phẩm

Chức năng hiển thị thông tin sản phẩm:



Hình 8: Giao diện hiển thị thông tin sản phẩm

Chức năng thêm thông tin những sản phẩm mới:



Home Create Product

Cập nhật sản phẩm

Mã sản phẩm

Tên sản phẩm

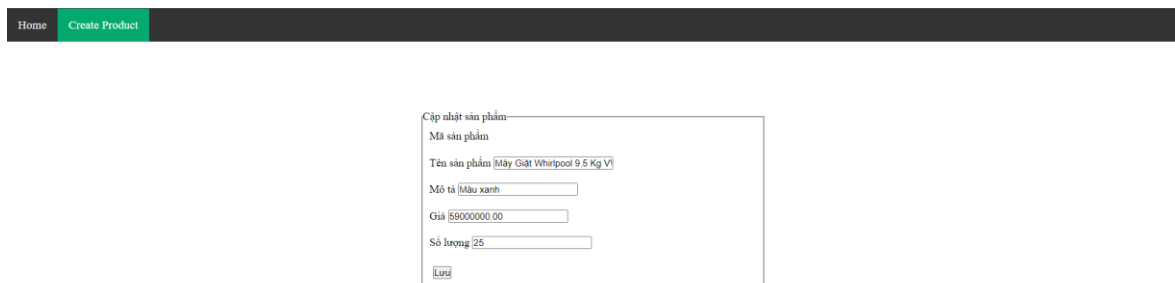
Mô tả

Giá

Số lượng

Hình 9: Giao diện thêm sản phẩm

Chức năng chỉnh sửa thông tin sản phẩm



Home Create Product

Cập nhật sản phẩm

Mã sản phẩm

Tên sản phẩm

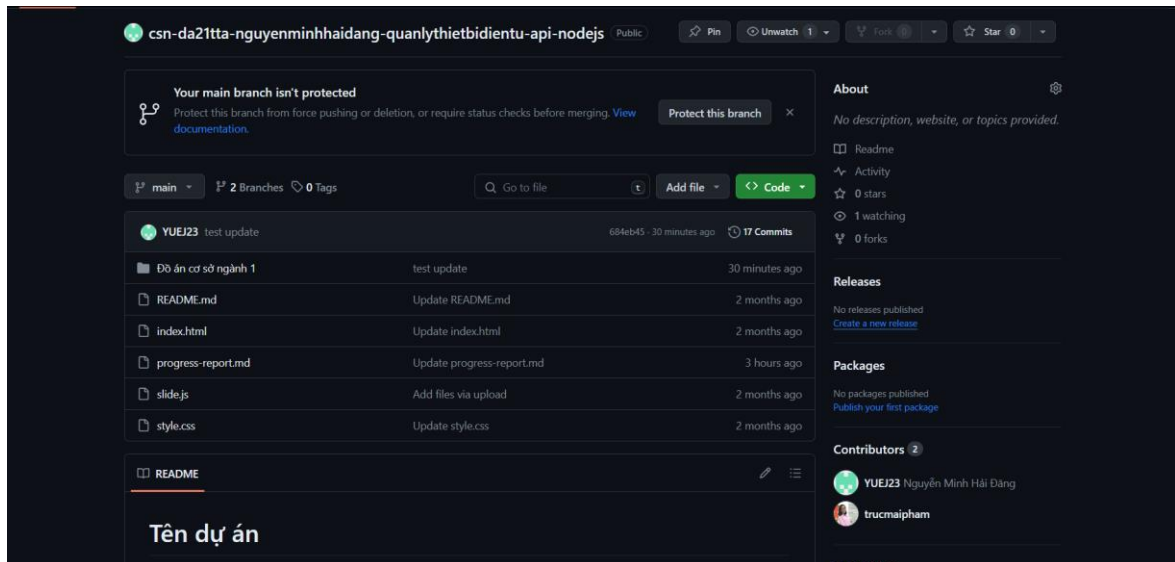
Mô tả

Giá

Số lượng

Hình 10: Giao diện chỉnh sửa sản phẩm

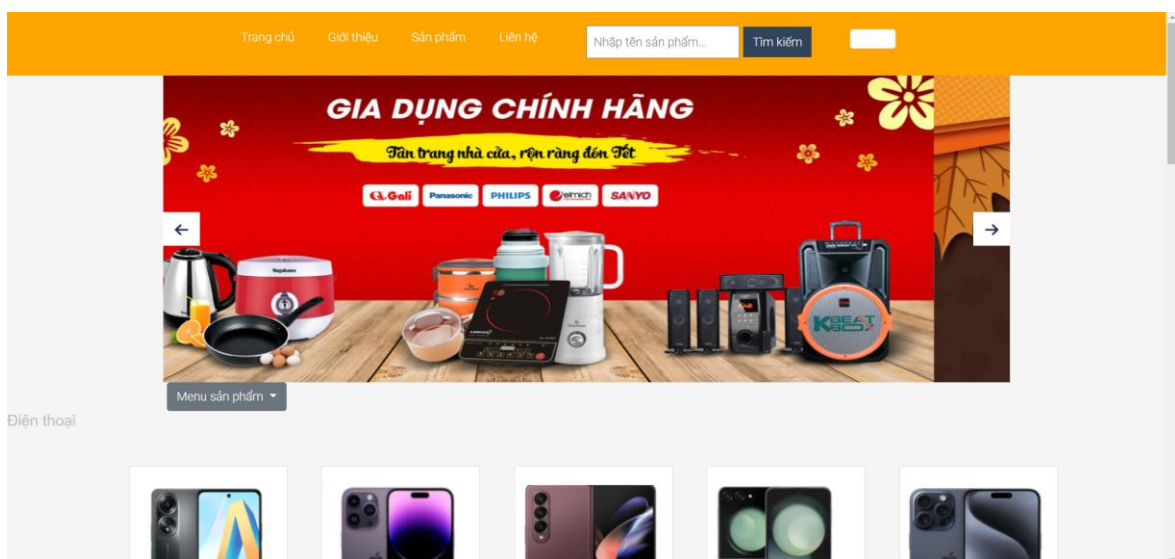
Chạy và kiểm tra thử các chức năng xem đã ổn định chưa sau đó tiến hành đưa dự án lên Github.



Hình 11: Đưa dự án lên Github

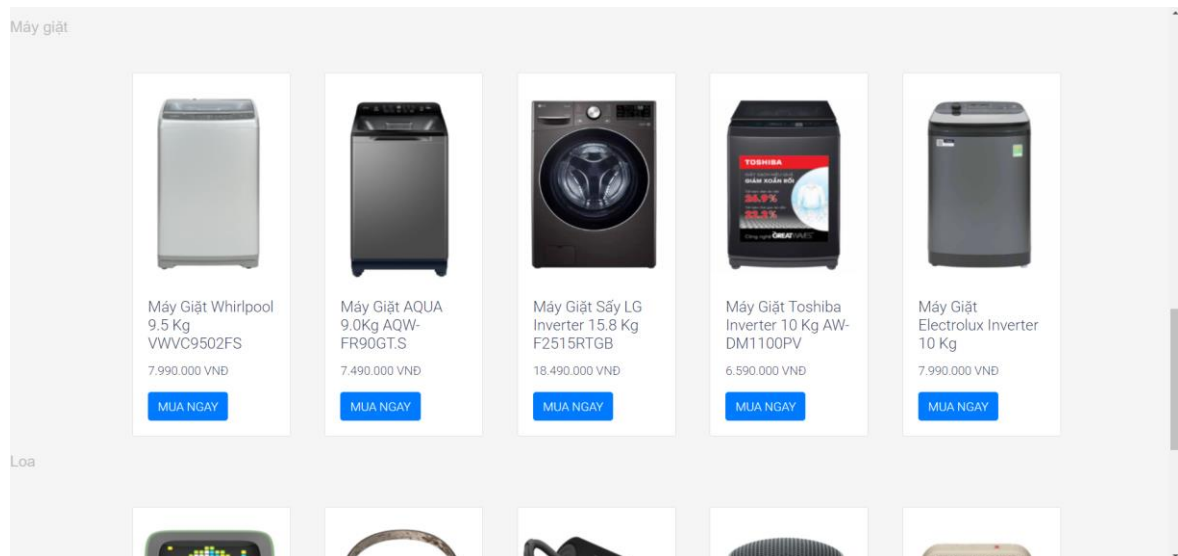
3.4. Thiết kế và tạo giao diện người dùng

Thiết kế giao diện cho người dùng gồm các hiệu ứng chuyển slide banner quảng cáo, thanh điều hướng .



Hình 12: Giao diện người dùng

Các ô chứa thông tin sản phẩm :



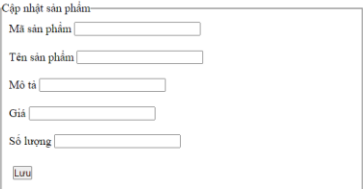
Hình 13: Các sản phẩm được trình bày trên website

CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU

4.1. Kết quả đạt được

Trong quá trình thực hiện đề án tôi đã đạt được những kết quả sau:

Chức năng hiển thị, thêm, sửa và xóa sản phẩm:



Cập nhật sản phẩm

Mã sản phẩm


Tên sản phẩm

Mô tả

Giá

Số lượng

Hình 14: Giao diện hoàn chỉnh của thêm sản phẩm



Cập nhật sản phẩm

Mã sản phẩm

Tên sản phẩm

Mô tả

Giá

Số lượng

Hình 15: Giao diện hoàn chỉnh của chỉnh sửa thông tin sản phẩm

Thiết kế thành công trang quản trị cửa hàng:

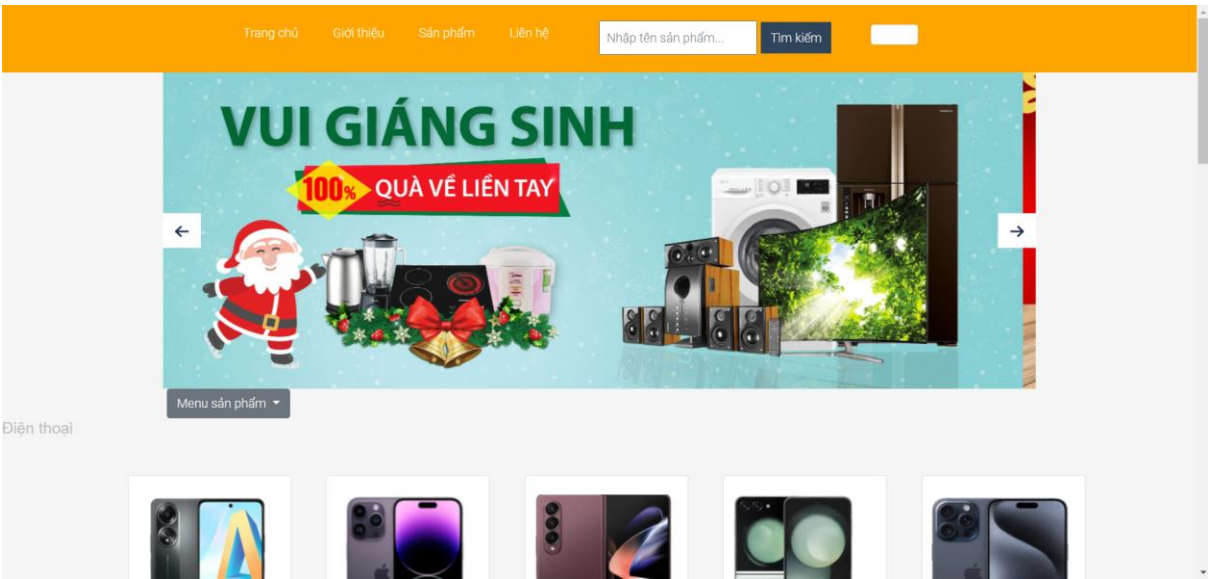
Trang chủThêm sản phẩm

DANH SÁCH SẢN PHẨM

Mã sản phẩm	Tên sản phẩm	Mô tả	Giá	Số lượng	Chỉnh sửa
11	Máy Giặt Whirlpool 9.5 Kg VWC9502FS	Màu xanh	5900000.00	25	<div>Sửa sản phẩmXóa sản phẩm</div>
12	Máy Giặt AQUA 9.0Kg AQW-FR900T.S		7490000.00	25	<div>Sửa sản phẩmXóa sản phẩm</div>
13	Máy Giặt Sấy LG Inverter 15.8 Kg F2515RTOB		18490000.00	20	<div>Sửa sản phẩmXóa sản phẩm</div>
14	Máy Giặt Toshiba Inverter 10 Kg AW-DM1100PV		6560000.00	30	<div>Sửa sản phẩmXóa sản phẩm</div>

Hình 16: Giao diện trang quản trị cửa hàng

Thiết kế thành công trang giao diện của người dùng



Hình 17: Giao diện của người dùng

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Qua thời gian thực hiện đồ án, em đã đạt được những mục tiêu như sau:

Kết quả đạt được:

- Hiểu rõ hơn về những công cụ để sử dụng xây dựng và tạo nên 1 trang website: Visual Studio Code và MySQL.
- Tìm hiểu sâu hơn RESTful API cùng với các ngôn ngữ là HTML, CSS, JS và môi trường NodeJS.
- Giao diện website đơn giản và thân thiện cho người dùng dễ sử dụng.
- Hiểu thêm được về thị trường thương mại điện tử và những nhu cầu thiết yếu của người tiêu dùng.

Hướng phát triển:

Với hạn chế về mặt thời gian nên em có hướng nghiên cứu phát triển và mở rộng đề tài như sau:

- Sẽ cập nhật thêm phân đăng ký và đăng nhập cho người dùng.
- Tạo ra mục thanh toán cho khách hàng
- Kết nối được thêm nhiều cơ sở dữ liệu để trang web được mở rộng hơn.
- Chỉnh sửa giao diện và thêm một số chức năng khác cho website.

DANH MỤC TÀI LIỆU THAM KHẢO

- [1] Đoàn Phước Miên, Phạm Thị Trúc Mai, Tài liệu giảng dạy môn thiết kế và lập trình web (lưu hành nội bộ), trường ĐH Trà Vinh.
- [2] Phạm Minh Dương, Tài liệu giảng dạy môn Phân tích và thiết kế hệ thống thông tin (lưu hành nội bộ), trường ĐH Trà Vinh, 2014.
- [3] Trang W3School : <https://www.w3schools.com/>.
- [4] Oracle Corporation, "MySQL Documentation," 1995. [Online]. Available: <https://www.mysql.com/>.
- [5] Trang Nodejs.org: <https://nodejs.org/en/learn/getting-started/introduction-to-nodej>.
- [6] Trang NpmJs: <https://www.npmjs.com>.

PHỤ LỤC