

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC GIA ĐỊNH
KHOA: CÔNG NGHỆ THÔNG TIN**



TIỂU LUẬN

**WEBSITE GIỚI THIỆU, BÁN HÀNG THỜI
TRANG GUCCI**

MÔN: LẬP TRÌNH NODEJS

Ngành: CÔNG NGHỆ THÔNG TIN

Chuyên ngành: KỸ THUẬT PHẦN MỀM

Giảng viên hướng dẫn: LÊ HUỲNH PHƯỚC

Sinh viên thực hiện: ĐẶNG LÊ PHƯƠNG LINH

MSSV: 2108110035

Lớp: K15DCPM04

TP. Hồ Chí Minh, tháng 4 năm 2024

Khoa/Viện: Công nghệ thông tin

NHẬN XÉT VÀ CHẤM ĐIỂM CỦA GIẢNG VIÊN
TIỂU LUẬN MÔN: QUẢN LÝ DỰ ÁN PHẦN MỀM

- 1. Họ và tên sinh viên: Đặng Lê Phương Linh**
- 2. Tên đề tài: website giới thiệu, bán hàng thời trang gucci**
- 3. Nhận xét:**

a) Những kết quả đạt được:

.....

.....

.....

.....

.....

b) Những hạn chế:

.....

.....

.....

.....

.....

- 4. Điểm đánh giá (theo thang điểm 10, làm tròn đến 0.5):**

Sinh viên: Đặng Lê Phương Linh

Điểm số: Điểm chữ:

TP. HCM, ngày ... tháng ... năm 20.....

Giảng viên chấm thi
(Ký và ghi rõ họ tên)

MỤC LỤC

CHƯƠNG 1 : GIỚI THIỆU VỀ WEBSITE BÁN HÀNG THỜI TRANG.....	1
1. Giới thiệu về NODE.JS	1
2. Giới thiệu về JAVASCRIPT	3
3. Giới thiệu về CLIENT-SERVER.....	5
4. Giới thiệu về mô hình MVC	7
5. Giới thiệu về một số thư viện chính và Middleware được sử dụng trong website	9
6. Giới thiệu về MONGODB	14
CHƯƠNG 2 : MÔ TẢ CÁC CHỨC NĂNG CỦA WEBSITE	17
1. Người dùng:	17
2. Admin:	20
CHƯƠNG 3 : HÌNH ẢNH DEMO CỦA WEBSITE	24
CHƯƠNG 4 : KẾT QUẢ ĐẠT ĐƯỢC VÀ CHƯA ĐẠT ĐƯỢC	30
TÀI LIỆU THAM KHẢO.....	31

CHƯƠNG 1 : GIỚI THIỆU VỀ WEBSITE BÁN HÀNG THỜI TRANG

Website được xây dựng dựa trên kiến trúc phần mềm client-server kết hợp mô hình MVC ở phía server và dựa trên nền tảng nodejs với ngôn ngữ lập trình chủ yếu là javascript.

Mục tiêu kinh doanh: Đây là một trang web bán hàng thời trang trực tuyến chuyên cung cấp các sản phẩm thời trang từ các thương hiệu nổi tiếng. Mục tiêu là mang lại trải nghiệm mua sắm trực tuyến tiện lợi, đa dạng và phong phú cho các khách hàng yêu thời trang.

Đối tượng khách hàng: Phục vụ cho mọi lứa tuổi và phong cách, từ thanh thiếu niên đến người trưởng thành, từ phong cách thời trang hiện đại đến cổ điển. Tạo ra một không gian mua sắm trực tuyến đa dạng để mọi người có thể tìm thấy những sản phẩm phù hợp với cá nhân hóa phong cách của họ.

Sản phẩm: Cung cấp một loạt các sản phẩm thời trang bao gồm quần áo, giày dép, túi xách, phụ kiện, và trang sức từ các thương hiệu danh tiếng như Gucci, Chanel, Adidas, Nike, và cả những thương hiệu độc lập nổi tiếng khác. Liên tục cập nhật và mở rộng danh mục sản phẩm để đáp ứng nhu cầu mua sắm đa dạng của khách hàng.

Trải nghiệm người dùng: Cam kết cung cấp trải nghiệm mua sắm trực tuyến tuyệt vời cho khách hàng. Trang web được thiết kế một cách thông minh và dễ sử dụng, giúp người dùng dễ dàng tìm kiếm và khám phá các sản phẩm.

1. Giới thiệu về NODE.JS

a. Khái niệm:

- Node.js là một môi trường thực thi mã JavaScript trên máy chủ, cho phép viết mã JavaScript phía máy chủ, không chỉ phía client như trong trình duyệt.

- Sử dụng kiến trúc non-blocking I/O, Node.js có thể xử lý hàng ngàn kết nối cùng một lúc mà không cần tạo ra một luồng mới cho mỗi kết nối.

- Node.js dựa trên Chrome V8 JavaScript Engine của Google, giúp cải thiện hiệu suất của ứng dụng.

b. Khả năng:

- Xử lý I/O không đồng bộ.
- Xây dựng ứng dụng mạng thời gian thực và có khả năng mở rộng cao.
- Phát triển API và các dịch vụ web RESTful.
- Xử lý dữ liệu lớn và phân tích dữ liệu thời gian thực.

c. Ưu điểm:

- Hiệu suất cao:
 - + Non-blocking I/O giúp tối ưu hóa hiệu suất ứng dụng.
 - + Node.js sử dụng mô hình không đồng bộ (asynchronous) cho việc xử lý I/O, điều này có nghĩa là các yêu cầu không chờ đợi (non-blocking), và server có thể tiếp tục xử lý các yêu cầu khác mà không cần chờ đợi kết quả của yêu cầu trước.
- Phát triển đồng nhất: Sử dụng JavaScript cả phía client và server giúp đơn giản hóa quá trình phát triển và duy trì mã nguồn.
- Cộng đồng lớn: Có cộng đồng lớn và phong phú, cung cấp nhiều gói mô-đun và hỗ trợ kỹ thuật.
- Mở rộng tốt:
 - + Node.js phù hợp cho các ứng dụng có nhiều kết nối đồng thời như chat, trò chơi trực tuyến.

+ Node.js sử dụng kiến trúc event-driven, trong đó các hàm callback được gọi khi xảy ra một sự kiện. Điều này giúp tối ưu hóa sử dụng tài nguyên và làm cho ứng dụng có thể mở rộng tốt.

d. Nhược điểm:

- Khó khăn trong xử lý các công việc đa luồng: Do tính chất single-threaded, Node.js không phù hợp cho các ứng dụng có nhu cầu xử lý nhiều luồng đồng thời.

- Cần kiến thức sâu về JavaScript: Việc phát triển Node.js yêu cầu kiến thức vững về JavaScript và các nguyên tắc của nó.

c. Ứng dụng thực tế:

- Phát triển API và dịch vụ web RESTful.

- Ứng dụng chat thời gian thực.

- Trò chơi trực tuyến đa người chơi.

- Xử lý dữ liệu lớn và phân tích dữ liệu thời gian thực.

- Ứng dụng IoT (Internet of Things) và các ứng dụng có khả năng mở rộng cao.

2. Giới thiệu về JAVASCRIPT

a. Khái niệm:

- JavaScript là một ngôn ngữ lập trình dành cho việc xây dựng các ứng dụng web tương tác.

- Ban đầu, JavaScript được phát triển để thêm tính năng tương tác và động cho các trang web, nhưng ngày nay, nó đã trở thành một trong những ngôn ngữ lập trình phổ biến nhất trên thế giới, được sử dụng không chỉ trong trình duyệt web

mà còn trong một loạt các môi trường khác nhau như server-side, mobile apps, desktop apps, game development, và IoT.

b. Khả năng:

- Xây dựng các ứng dụng web tương tác: JavaScript được sử dụng chủ yếu để thêm các tính năng động và tương tác cho các trang web.

- Phát triển ứng dụng web và mobile: JavaScript được sử dụng rộng rãi trong các framework và thư viện như React, Angular, và Vue.js để phát triển ứng dụng web và mobile.

- Phát triển server-side: Node.js cho phép sử dụng JavaScript để phát triển các ứng dụng server-side, mở ra nhiều cơ hội phát triển ứng dụng full-stack với cùng một ngôn ngữ.

- Phát triển game: JavaScript cũng được sử dụng trong phát triển game, đặc biệt là trong việc phát triển trò chơi web và trò chơi mobile.

c. Ưu điểm:

- Dễ học và sử dụng: JavaScript có cú pháp đơn giản và dễ hiểu, phù hợp cho cả người mới bắt đầu và những nhà phát triển có kinh nghiệm.

- Tính linh hoạt: JavaScript là một ngôn ngữ đa mô hình, cho phép sử dụng nhiều phong cách lập trình khác nhau như procedural, object-oriented, và functional programming.

- Sự phát triển nhanh chóng: JavaScript có cộng đồng lớn và phong phú, với nhiều framework và thư viện mạnh mẽ được phát triển liên tục.

d. Nhược điểm:

- Bảo mật: JavaScript có thể gây ra các vấn đề bảo mật như cross-site scripting (XSS) nếu không được sử dụng đúng cách.
- Hiệu suất: Trong một số trường hợp, JavaScript có thể không hiệu quả về mặt hiệu suất so với các ngôn ngữ khác, đặc biệt là trong việc xử lý các tính toán phức tạp.

e. Ứng dụng thực tế:

- Phát triển các tính năng động cho các trang web và ứng dụng web.
- Xây dựng ứng dụng web và mobile với các framework như React, Angular, và Vue.js.
- Phát triển ứng dụng server-side với Node.js.
- Phát triển trò chơi web và trò chơi mobile.

3. Giới thiệu về CLIENT-SERVER

a. Khái niệm:

- Client-server là một mô hình phần mềm mà trong đó các máy tính hoạt động như một client gửi yêu cầu và máy chủ xử lý các yêu cầu này và trả về kết quả.
- Client là một ứng dụng hoặc thiết bị sử dụng dịch vụ từ máy chủ thông qua mạng.
- Server là một máy tính hoặc thiết bị cung cấp dịch vụ hoặc tài nguyên cho các client thông qua mạng.

b. Khả năng:

- Phân phối công việc: Client và server phân chia các nhiệm vụ và trách nhiệm giúp tối ưu hóa hiệu suất hệ thống.
- Mở rộng dễ dàng: Cấu trúc client-server cho phép mở rộng hệ thống bằng cách thêm client hoặc server mới mà không cần thay đổi toàn bộ kiến trúc.

c. Ưu điểm:

- Phân chia trách nhiệm: Client và server có thể chịu trách nhiệm cho các nhiệm vụ cụ thể, giúp tăng tính linh hoạt và dễ dàng bảo trì.
- Bảo mật: Kiến trúc client-server cho phép thực hiện các biện pháp bảo mật tập trung, như xác thực và phân quyền truy cập.
- Hiệu suất: Các tác vụ có thể được phân phối giữa client và server, giúp tối ưu hóa hiệu suất hệ thống.

d. Nhược điểm:

- Độ phức tạp: Việc triển khai và duy trì một hệ thống client-server có thể phức tạp hơn so với các mô hình khác, đặc biệt là khi cần đồng bộ hóa dữ liệu giữa client và server.
- Độ trễ: Sự phụ thuộc vào mạng có thể dẫn đến độ trễ trong việc truy cập và xử lý dữ liệu.

e. Ứng dụng thực tế:

- Web applications: Trong mô hình web, trình duyệt web hoạt động như client và máy chủ web phục vụ các yêu cầu từ client.
- Email: Email clients gửi và nhận email thông qua máy chủ email.

- Game servers: Trong trò chơi trực tuyến, máy chủ game xử lý các yêu cầu từ các máy chơi game (client) và điều khiển trò chơi.

4. Giới thiệu về mô hình MVC

a. Khái niệm:

Mô hình MVC (Model-View-Controller) là một kiến trúc phần mềm được sử dụng để phát triển ứng dụng web và các ứng dụng khác. Dưới đây là một giới thiệu tổng quan về mô hình MVC:

- Model (Mô hình):

+ Mô hình đại diện cho dữ liệu của ứng dụng và quản lý tất cả các logic liên quan đến dữ liệu. Nó không chỉ đơn thuần là dữ liệu mà còn bao gồm các hành động (action) để thao tác với dữ liệu đó.

+ Mô hình thường không biết gì về người dùng hoặc giao diện người dùng, nó chỉ tập trung vào quản lý dữ liệu và logic của ứng dụng.

- View (Giao diện):

+ View là phần giao diện người dùng của ứng dụng, nơi mà dữ liệu từ mô hình được hiển thị và mà người dùng tương tác với ứng dụng.

+ View thường làm nhiệm vụ hiển thị dữ liệu và nhận các sự kiện từ người dùng (ví dụ: nhấp chuột, nhập liệu).

- Controller (Bộ điều khiển):

+ Controller là thành phần trung gian giữa Model và View. Nó nhận các yêu cầu từ người dùng thông qua giao diện và sau đó tương tác với mô hình để lấy dữ liệu cần thiết.

+ Controller cũng có trách nhiệm điều hướng (routing) yêu cầu từ người dùng đến các phần tử xử lý tương ứng và chuyển kết quả trả về từ mô hình đến giao diện người dùng.

b. Ưu điểm của mô hình MVC:

- Tách biệt logic ứng dụng: Mô hình MVC tách biệt rõ ràng giữa dữ liệu, logic và giao diện, giúp dễ dàng bảo trì và mở rộng ứng dụng.

- Tính linh hoạt: Mô hình này cho phép phát triển song song giữa các nhóm công việc khác nhau, giúp tăng tốc quá trình phát triển ứng dụng.

- Tái sử dụng code: Sự phân tách rõ ràng giữa các phần của mô hình MVC làm cho code có thể tái sử dụng được nhiều lần.

c. Nhược điểm của mô hình MVC:

- Khả năng phức tạp: Trong một số trường hợp, việc triển khai mô hình MVC có thể phức tạp hơn so với các mô hình khác, đặc biệt là khi áp dụng cho các ứng dụng nhỏ.

- Tăng cường chi phí: Mô hình MVC có thể yêu cầu thêm công sức và thời gian đào tạo nhằm đảm bảo rằng tất cả các thành phần đều hoạt động hiệu quả với nhau.

d. Ứng dụng thực tế của mô hình MVC:

- Phát triển ứng dụng web: Mô hình này phù hợp cho việc phát triển các ứng dụng web động và phản ứng.

- Phát triển ứng dụng di động: MVC cũng có thể được sử dụng để phát triển các ứng dụng di động, bằng cách sử dụng các framework như React Native hoặc Flutter.

- Phát triển ứng dụng desktop: Mô hình MVC cũng có thể được áp dụng trong việc phát triển các ứng dụng desktop thông qua các framework như Electron.

5. Giới thiệu về một số thư viện chính và Middleware được sử dụng trong website

a. Thư viện path

- Khái niệm:

+ Thư viện path cung cấp các hàm để thao tác và xử lý các đường dẫn trong môi trường Node.js.

+ Đường dẫn có thể là đường dẫn tương đối hoặc đường dẫn tuyệt đối và thư viện path giúp bạn làm việc với cả hai loại đường dẫn.

- Cách dùng:

+ Sử dụng các phương thức của thư viện path để thực hiện các thao tác như nối đường dẫn, phân tích đường dẫn, kiểm tra tính tuyệt đối, chuẩn hóa đường dẫn, và nhiều hơn nữa.

+ Bắt đầu bằng cách import thư viện path vào ứng dụng của bạn (`const path = require('path');`) và sau đó sử dụng các phương thức cung cấp bởi thư viện.

- Ưu điểm:

+ Đơn giản và dễ sử dụng: Thư viện path cung cấp các phương thức dễ hiểu và dễ sử dụng để thao tác với các đường dẫn.

+ Tính di động: Các phương thức của path là di động giữa các hệ điều hành khác nhau, giúp ứng dụng của bạn hoạt động trên nhiều nền tảng mà không cần phải chỉnh sửa mã.

- Nhược điểm:

+ Giới hạn trong các chức năng: Thư viện path cung cấp các chức năng cơ bản để thao tác với đường dẫn và không hỗ trợ các tính năng phức tạp như thao tác với tệp tin hoặc thư mục thực sự.

+ Phụ thuộc vào môi trường: Mặc dù path cung cấp tính di động giữa các hệ điều hành, nhưng bạn vẫn cần phải lưu ý rằng một số hành vi cụ thể có thể khác nhau giữa các nền tảng.

=> Thư viện path trong Node.js là một công cụ hữu ích và quan trọng cho việc xử lý và quản lý các đường dẫn tệp tin và thư mục trong ứng dụng của bạn.

b. Mongoose

- Khái niệm:

+ Mongoose là một thư viện Node.js được sử dụng để tạo và quản lý các mô hình dữ liệu (data models) cho MongoDB.

+ Nó cung cấp một cách dễ dàng để định nghĩa các mô hình dữ liệu, xác định các quan hệ giữa các bộ sưu tập (collections), và thực hiện các hoạt động truy vấn dữ liệu.

- Cách làm:

+ Bắt đầu bằng cách cài đặt thư viện Mongoose trong ứng dụng Node.js của bạn bằng cách sử dụng npm (npm install mongoose).

+ Định nghĩa các mô hình dữ liệu bằng cách sử dụng Mongoose Schema và Model API.

+ Kết nối ứng dụng của bạn với cơ sở dữ liệu MongoDB sử dụng Mongoose.

+ Sử dụng các phương thức của Mongoose để thực hiện các hoạt động như tìm kiếm, thêm, cập nhật và xóa dữ liệu.

- Ưu điểm:

+ Mô hình hóa dữ liệu: Mongoose cho phép bạn định nghĩa các mô hình dữ liệu linh hoạt, bao gồm các quan hệ giữa các mô hình.

- + Tính di động: Mongoose giúp làm việc với MongoDB trở nên dễ dàng và di động giữa các dự án khác nhau.

- + Kiểm soát dữ liệu: Mongoose cung cấp các công cụ mạnh mẽ để kiểm soát quá trình truy cập và xử lý dữ liệu trong MongoDB.

- Nhược điểm:

- + Độ phức tạp: Mongoose có thể tạo ra một lớp trừu tượng giữa ứng dụng và cơ sở dữ liệu, làm tăng độ phức tạp của ứng dụng trong một số trường hợp.

- + Hiệu suất: Mặc dù Mongoose cung cấp nhiều tiện ích và tính năng, nhưng đôi khi nó có thể tạo ra các truy vấn chậm hoặc không hiệu quả so với việc sử dụng các truy vấn trực tiếp tới MongoDB.

- + Thư viện Mongoose là một công cụ mạnh mẽ cho việc phát triển ứng dụng Node.js sử dụng cơ sở dữ liệu MongoDB. Nó giúp giảm bớt công việc lặp lại, cung cấp một cách tiện lợi để quản lý dữ liệu và tạo ra các ứng dụng dễ bảo trì.

c. Express

- Khái niệm:

- + Express là một framework web Node.js nhẹ và linh hoạt cho việc xây dựng các ứng dụng web và API.

- + Nó cung cấp các tính năng cần thiết để xử lý yêu cầu HTTP, quản lý các tệp tin tĩnh, định tuyến (routing), và phản hồi (response).

- Cách dùng:

- + Bắt đầu bằng cách cài đặt Express trong ứng dụng Node.js của bạn bằng lệnh npm (npm install express).

- + Sử dụng Express để tạo ra một ứng dụng bằng cách tạo một đối tượng ứng dụng (const app = express();).

- + Định nghĩa các tuyến đường (routes) để xử lý các yêu cầu từ client bằng cách sử dụng các phương thức như app.get(), app.post(), và app.use().

- + Khởi động máy chủ bằng cách lắng nghe kết nối tới một cổng nhất định (app.listen()).

- Ưu điểm:

+ Dễ học và sử dụng: Express có cú pháp đơn giản và dễ hiểu, phù hợp cho cả người mới bắt đầu và những người có kinh nghiệm trong Node.js.

+ Linh hoạt: Express không ép buộc một cấu trúc dự án cụ thể nào và cho phép bạn tự do tổ chức mã nguồn của mình theo cách bạn muốn.

+ Cộng đồng lớn: Express có một cộng đồng lớn và phong phú, với nhiều tài liệu, ví dụ và plugin được cung cấp.

- Nhược điểm:

+ Thiếu tính năng tích hợp sẵn: Mặc dù Express cung cấp các tính năng cơ bản để phát triển ứng dụng web, nhưng bạn có thể cần phải sử dụng các middleware hoặc thư viện bên thứ ba để bổ sung các tính năng phức tạp hơn.

+ Khó khăn trong việc quản lý ứng dụng lớn: Trong các ứng dụng lớn, việc quản lý các tuyến đường và logic ứng dụng có thể trở nên phức tạp hơn.

+ Express là một trong những lựa chọn phổ biến nhất cho việc phát triển ứng dụng web và API trong Node.js, nhờ vào tính đơn giản, linh hoạt và hiệu suất của nó.

d. Morgan

- Morgan là một middleware phổ biến trong Node.js, thường được sử dụng để ghi lại các yêu cầu HTTP vào các ứng dụng web. Nó cung cấp một cách dễ dàng để ghi lại thông tin chi tiết về mỗi yêu cầu HTTP, bao gồm địa chỉ IP của người dùng, phương thức yêu cầu, đường dẫn yêu cầu, thời gian phản hồi và nhiều thông tin khác.

- Dưới đây là một số điểm quan trọng về Morgan:

+ Cách sử dụng: Để sử dụng Morgan, bạn cần cài đặt nó thông qua npm: `npm install morgan`. Sau đó, import nó vào ứng dụng của bạn và kết nối nó với ứng dụng Express thông qua một dòng mã: `app.use(morgan('combined'))`. Bạn có thể cấu hình Morgan để ghi lại thông tin theo các định dạng khác nhau như 'combined', 'common', 'dev', 'short', và 'tiny'.

+ Ứng dụng:

Morgan thường được sử dụng trong quá trình phát triển và debugging để kiểm tra các yêu cầu HTTP và phản hồi từ server.

Nó cũng hữu ích trong việc theo dõi log và hoạt động của ứng dụng trong môi trường production.

+ Ưu điểm:

Đễ dàng tích hợp: Morgan tích hợp dễ dàng vào các ứng dụng Express thông qua một dòng mã.

Linh hoạt: Bạn có thể tùy chỉnh Morgan để ghi lại thông tin theo nhiều định dạng khác nhau, tùy thuộc vào nhu cầu cụ thể của ứng dụng của bạn.

+ Nhược điểm:

Quá nhiều thông tin: Trong môi trường production, việc ghi lại quá nhiều thông tin có thể làm tăng kích thước của log và ảnh hưởng đến hiệu suất của ứng dụng.

=> Tóm lại, Morgan là một middleware mạnh mẽ và hữu ích trong quá trình phát triển và debugging các ứng dụng Node.js, giúp ghi lại các hoạt động của yêu cầu HTTP một cách dễ dàng và linh hoạt.

e. Methodoverride

Là một middleware cho phép bạn sử dụng các phương thức HTTP như PUT hoặc DELETE trong các biểu mẫu HTML. Điều này là cần thiết bởi vì các trình duyệt web chỉ hỗ trợ GET và POST cho việc gửi yêu cầu HTTP.

Khi bạn muốn sử dụng các phương thức HTTP khác như PUT hoặc DELETE, bạn có thể sử dụng `methodOverride` để ghi đè phương thức gốc của yêu cầu và thay đổi nó thành phương thức mà bạn muốn sử dụng.

Middleware `methodOverride` cần phải được đặt trước router trong ứng dụng Express để làm việc đúng cách. Bạn cũng cần chú ý rằng các yêu cầu từ trình duyệt cần phải hỗ trợ phương thức POST và có một trường ẩn trong biểu mẫu HTML để gửi phương thức gốc mà bạn muốn sử dụng.

f. Body-parser

Là một middleware cho phép bạn trích xuất các dữ liệu gửi đến server từ phần thân của yêu cầu HTTP (HTTP request body). Điều này là cần thiết khi bạn muốn lấy dữ liệu từ form HTML, AJAX yêu cầu, hoặc các yêu cầu gửi dữ liệu JSON đến server.

g. Cookie-parser

Là một middleware được sử dụng để phân tích và chuyển đổi các cookie được gửi từ trình duyệt của người dùng thành đối tượng JavaScript để sử dụng. Khi người dùng truy cập vào một trang web, trình duyệt sẽ gửi các cookie đến máy chủ để lưu trữ thông tin như thông tin đăng nhập, cài đặt ngôn ngữ, thông tin giỏ hàng, v.v. cookie-parser giúp bạn dễ dàng truy cập và xử lý các cookie này trong ứng dụng Express của mình.

h. Bcrypt

Là một thư viện được sử dụng trong Node.js để băm mật khẩu (password hashing). Băm mật khẩu là quá trình chuyển đổi một mật khẩu thành một chuỗi hash không thể đảo ngược, đảm bảo tính bảo mật khi lưu trữ mật khẩu trong cơ sở dữ liệu. Khi sử dụng bcrypt, mật khẩu được băm thành một chuỗi hash với độ dài cố định và được lưu trữ trong cơ sở dữ liệu. Khi người dùng đăng nhập, mật khẩu nhập vào được băm và so sánh với chuỗi hash đã lưu trữ để xác thực.

6. Giới thiệu về MONGODB

a. Khái niệm:

- MongoDB là một cơ sở dữ liệu không cấu trúc (NoSQL) dựa trên tài liệu, nơi mà dữ liệu được lưu trữ dưới dạng các tài liệu JSON có thể lưu trữ dữ liệu phức tạp và có cấu trúc khác nhau.

- MongoDB sử dụng mô hình dữ liệu không cấu trúc giúp đơn giản hóa việc lưu trữ và truy vấn dữ liệu có cấu trúc thay vì sử dụng các bảng và hàng như trong cơ sở dữ liệu quan hệ.

b. Khả năng:

- Lưu trữ dữ liệu có cấu trúc và không cấu trúc: MongoDB cho phép lưu trữ dữ liệu dưới dạng các tài liệu JSON với cấu trúc linh hoạt.

- Truy vấn mạnh mẽ: MongoDB hỗ trợ các truy vấn phong phú và linh hoạt, bao gồm các truy vấn phức tạp, truy vấn điều kiện, và truy vấn theo vị trí địa lý.

- Tính mở rộng cao: MongoDB có khả năng mở rộng ngang (horizontal scaling), cho phép mở rộng hệ thống bằng cách thêm các node vào cluster mà không làm giảm hiệu suất.

c. Ưu điểm:

- Linh hoạt và dễ dàng mở rộng: MongoDB cho phép lưu trữ dữ liệu có cấu trúc và không cấu trúc, cũng như mở rộng hệ thống dễ dàng bằng cách thêm các node vào cluster.

- Hiệu suất cao: MongoDB được thiết kế để có hiệu suất cao với khả năng xử lý các yêu cầu đồng thời và truy vấn phức tạp.

- Phù hợp cho các ứng dụng web và di động: MongoDB phù hợp cho việc phát triển các ứng dụng web và di động có yêu cầu lưu trữ dữ liệu linh hoạt và có thể mở rộng.

d. Nhược điểm:

- Tính nhất quán thấp: Do MongoDB không hỗ trợ các giao dịch nhất quán trên nhiều tài nguyên, nên có thể gặp phải các vấn đề về nhất quán dữ liệu trong một số trường hợp.

- Tốn nhiều bộ nhớ: MongoDB cần tốn nhiều bộ nhớ hơn so với các hệ thống cơ sở dữ liệu quan hệ để lưu trữ cùng một lượng dữ liệu.

e. Ứng dụng thực tế:

- Lưu trữ dữ liệu cho các ứng dụng web và di động.
- Lưu trữ dữ liệu IoT (Internet of Things).
- Lưu trữ log và dữ liệu dạng văn bản.
- Phát triển các hệ thống Real-time Analytics và Big Data.

CHƯƠNG 2 : MÔ TẢ CÁC CHỨC NĂNG CỦA WEBSITE

1. Người dùng:

a. Xem sản phẩm:

- Khi người dùng truy cập vào trang web, họ có thể sử dụng chức năng xem sản phẩm để tham khảo thông tin về các sản phẩm có sẵn. Dưới đây là mô tả chi tiết về chức năng này:

+ Danh sách sản phẩm: danh sách sản phẩm sẽ được hiển thị dưới dạng danh sách. Mỗi sản phẩm sẽ có hình ảnh, tên, giá và các chi tiết mô tả ngắn.

+ Chi tiết sản phẩm: Khi người dùng nhấp vào một sản phẩm, họ sẽ được chuyển đến trang chi tiết của sản phẩm đó. Trang này thường bao gồm hình ảnh lớn của sản phẩm, các chi tiết kỹ thuật, mô tả đầy đủ, đánh giá từ người dùng khác (nếu có), thông tin về tình trạng hàng hóa và lựa chọn mua hàng.

+ Thêm vào giỏ hàng: Nếu người dùng quan tâm đến sản phẩm, họ có thể thêm sản phẩm đó vào giỏ hàng bằng cách nhấp vào nút "Thêm vào giỏ hàng".

+ Xem sản phẩm liên quan: Trang chi tiết sản phẩm thường cũng hiển thị các sản phẩm liên quan hoặc gợi ý khác mà người dùng có thể quan tâm.

+ Tính năng lọc sản phẩm: Trang web có thể cung cấp tính năng lọc sản phẩm theo các tiêu chí như giá, thương hiệu, kích thước, màu sắc để giúp người dùng thuận tiện hơn trong việc tìm kiếm sản phẩm.

b. Xem tin tức

- Trang chủ: Trang chủ của trang web tin tức sẽ hiển thị một số tin tức nổi bật nhất, có thể là các tin tức mới nhất hoặc các sự kiện quan trọng đang diễn ra. Các tin tức này thường được sắp xếp theo thứ tự thời gian hoặc mức độ quan trọng.

- Danh mục tin tức: Trang web có thể chia các tin tức thành các danh mục khác nhau như Thế giới, Kinh doanh, Giải trí, Thể thao, Công nghệ, Sức khỏe, và nhiều danh mục

khác. Người dùng có thể chọn một danh mục cụ thể để xem tin tức trong lĩnh vực họ quan tâm.

- Xem tin tức chi tiết: Khi người dùng nhấp vào một tin tức, họ sẽ được chuyển đến trang chi tiết của tin tức đó. Trang này sẽ hiển thị nội dung đầy đủ của tin tức, bao gồm tiêu đề, nội dung, hình ảnh minh họa và thông tin về tác giả và nguồn tin.

c. Thêm sửa xóa sản phẩm trong giỏ hàng

- Thêm sản phẩm vào giỏ hàng: Khi người dùng quan tâm đến một sản phẩm, họ có thể nhấp vào nút "Thêm vào giỏ hàng". Hệ thống sẽ thêm sản phẩm đó vào giỏ hàng của họ.

- Xem giỏ hàng: Người dùng có thể xem các sản phẩm đã thêm vào giỏ hàng của mình bằng cách nhấp vào biểu tượng giỏ hàng hoặc các liên kết tương tự. Họ sẽ thấy danh sách các sản phẩm trong giỏ hàng cùng với thông tin về số lượng, giá cả và tổng số tiền.

- Thay đổi số lượng sản phẩm: Trong trang giỏ hàng, người dùng có thể thay đổi số lượng sản phẩm bằng cách chỉnh sửa số lượng trong trường số lượng hoặc sử dụng các nút tăng/giảm số lượng. Hệ thống sẽ tự động cập nhật giá cả và tổng số tiền khi số lượng sản phẩm thay đổi.

- Xóa sản phẩm khỏi giỏ hàng: Nếu người dùng muốn loại bỏ một sản phẩm khỏi giỏ hàng, họ có thể nhấp vào nút "Xóa" hoặc biểu tượng tương ứng với sản phẩm đó trong danh sách giỏ hàng. Hệ thống sẽ loại bỏ sản phẩm đó khỏi giỏ hàng và cập nhật tổng số tiền.

- Tiếp tục mua sắm hoặc thanh toán: Sau khi chỉnh sửa giỏ hàng, người dùng có thể tiếp tục mua sắm bằng cách quay lại trang sản phẩm hoặc tiến hành thanh toán bằng cách chuyển đến trang thanh toán.

d. Đặt hàng

- Nhập thông tin giao hàng: Người dùng cần cung cấp thông tin liên lạc và địa chỉ giao hàng để nhận sản phẩm. Thông tin này có thể bao gồm tên, địa chỉ, số điện thoại, và bất kỳ yêu cầu đặc biệt nào liên quan đến việc giao hàng.

- Chọn phương thức giao hàng: Họ có thể chọn phương thức giao hàng phù hợp với nhu cầu của mình, bao gồm giao hàng tiêu chuẩn, giao hàng nhanh hoặc tự lấy hàng tại cửa hàng.

- Kiểm tra lại đơn hàng: Trước khi xác nhận đơn hàng, người dùng có thể xem lại tất cả các thông tin đã nhập, bao gồm các sản phẩm trong giỏ hàng, địa chỉ giao hàng, phương thức thanh toán và phương thức giao hàng.

- Xác nhận đơn hàng: Khi đã kiểm tra kỹ lưỡng và chắc chắn mọi thông tin đều chính xác, người dùng có thể xác nhận đơn hàng bằng cách nhấn nút "Xác nhận đơn hàng" hoặc "Thanh toán".

- Theo dõi đơn hàng: Sau khi đơn hàng được xác nhận, người dùng có thể theo dõi tình trạng của đơn hàng bằng cách sử dụng số theo dõi hoặc liên kết theo dõi được cung cấp trong email xác nhận.

e. Đăng nhập và đăng ký

- Đăng ký:

+ Tạo tài khoản mới: Người dùng bắt đầu bằng cách nhấp vào nút "Đăng ký" hoặc "Tạo tài khoản". Họ cung cấp thông tin cá nhân cần thiết như tên, địa chỉ email, mật khẩu, và các thông tin khác tùy thuộc vào yêu cầu của trang web.

+ Hoàn tất đăng ký: Sau khi xác nhận, tài khoản của người dùng sẽ được tạo và họ có thể đăng nhập để tiếp tục mua sắm hoặc sử dụng các tính năng khác trên trang web.

- Đăng nhập:

+ Nhập thông tin đăng nhập: Người dùng nhập địa chỉ email và mật khẩu đã đăng ký vào các trường tương ứng trên trang đăng nhập.

+ Xác thực: Hệ thống xác thực thông tin đăng nhập với dữ liệu đã được lưu trữ. Nếu thông tin đúng, người dùng sẽ được chuyển đến trang chính của trang web với tư cách là người dùng đã đăng nhập.

2. Admin:

Có các chức năng của người dùng và những chức năng riêng sau đây

a. Thêm sửa xóa tin tức

- Thêm tin tức:

+ Người quản trị đăng nhập vào hệ thống với tư cách là người quản trị.

+ Trên giao diện quản trị, có một trang hoặc một phần riêng biệt cho việc thêm tin tức mới.

+ Người quản trị điền thông tin về tiêu đề, nội dung, hình ảnh, danh mục và các thông tin khác liên quan đến tin tức.

+ Sau khi điền đầy đủ thông tin, họ có thể lưu tin tức mới và nó sẽ được hiển thị trên trang chính của trang web tin tức.

- Sửa tin tức:

+ Người quản trị truy cập vào trang quản lý tin tức.

+ Họ tìm kiếm hoặc chọn tin tức mà họ muốn chỉnh sửa.

+ Sau đó, họ có thể chỉnh sửa thông tin của tin tức đó, bao gồm cả tiêu đề, nội dung, hình ảnh và các thông tin khác.

- + Sau khi hoàn thành chỉnh sửa, họ lưu lại và các thay đổi sẽ được cập nhật trên trang tin tức.

- Xóa tin tức:

- + Người quản trị truy cập vào trang quản lý tin tức.
- + Họ tìm kiếm hoặc chọn tin tức mà họ muốn xóa.
- + Sau đó, họ chọn tùy chọn xóa tin tức và xác nhận hành động này.
- + Tin tức sẽ được xóa khỏi hệ thống và không còn xuất hiện trên trang web tin tức nữa.

b. Thêm sửa xóa sản phẩm

- Thêm sản phẩm:

- + Người quản trị đăng nhập vào hệ thống với tư cách là người quản trị.
- + Trên giao diện quản trị, có một trang hoặc một phần riêng biệt cho việc thêm sản phẩm mới.
- + Người quản trị điền thông tin chi tiết về sản phẩm, bao gồm tên sản phẩm, mô tả, hình ảnh, giá cả, số lượng có sẵn, và các thuộc tính khác liên quan đến sản phẩm.
- + Sau khi điền đầy đủ thông tin, họ có thể lưu sản phẩm mới và nó sẽ được thêm vào cửa hàng trực tuyến.

- Sửa sản phẩm:

- + Người quản trị truy cập vào trang quản lý sản phẩm.
- + Họ tìm kiếm hoặc chọn sản phẩm mà họ muốn chỉnh sửa.
- + Sau đó, họ có thể chỉnh sửa thông tin của sản phẩm đó, bao gồm cả tên, mô tả, hình ảnh, giá cả, số lượng và các thuộc tính khác.

- + Sau khi hoàn thành chỉnh sửa, họ lưu lại và các thay đổi sẽ được cập nhật trên cửa hàng trực tuyến.

- Xóa sản phẩm:

- + Người quản trị truy cập vào trang quản lý sản phẩm.
- + Họ tìm kiếm hoặc chọn sản phẩm mà họ muốn xóa.
- + Sau đó, họ chọn tùy chọn xóa sản phẩm và xác nhận hành động này.
- + Sản phẩm sẽ được xóa khỏi cửa hàng trực tuyến và không còn xuất hiện trong danh sách sản phẩm nữa.

c. Sửa xóa đơn hàng

- Sửa đơn hàng:

- + Người quản trị đăng nhập vào hệ thống với tư cách là người quản trị.
- + Trên giao diện quản trị, có một trang hoặc một phần riêng biệt cho việc quản lý đơn hàng.
- + Người quản trị tìm kiếm hoặc chọn đơn hàng mà họ muốn chỉnh sửa.
- + Sau đó, Người quản trị có thể thực hiện các thay đổi cần thiết, như cập nhật thông tin liên hệ của khách hàng, điều chỉnh số lượng sản phẩm, hoặc thay đổi phương thức giao hàng.
- + Người quản trị lưu lại các thay đổi và thông tin mới sẽ được cập nhật trong hệ thống.

- Xóa đơn hàng:

- + Người quản trị truy cập vào trang quản lý đơn hàng.
- + Người quản trị tìm kiếm hoặc chọn đơn hàng mà họ muốn xóa.
- + Sau đó, người quản trị chọn tùy chọn xóa đơn hàng và xác nhận hành động này.

- + Đơn hàng sẽ được xóa khỏi hệ thống và không còn xuất hiện trong danh sách đơn hàng nữa.

c. Sửa xóa tài khoản

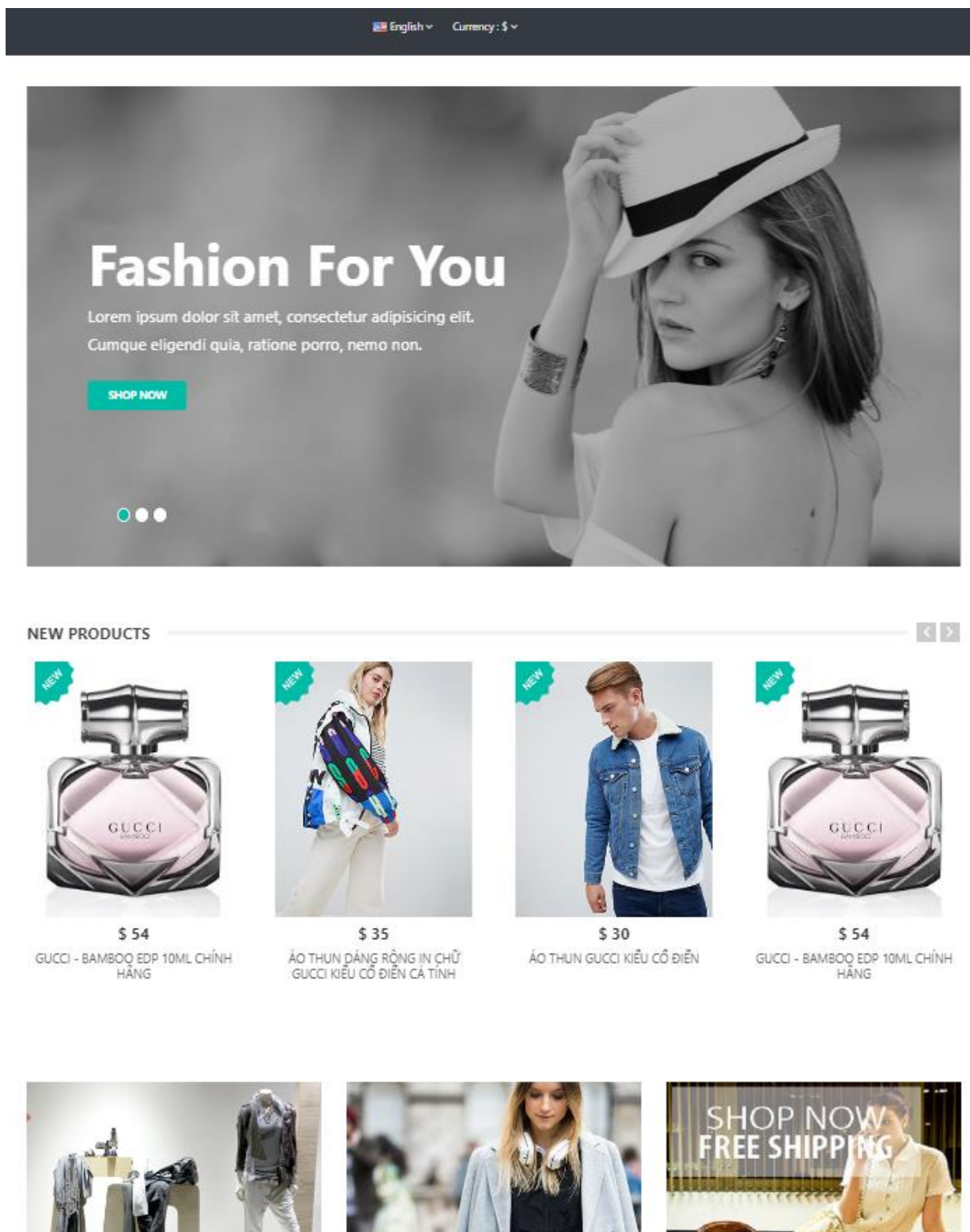
- Sửa tài khoản:

- + Người quản trị đăng nhập vào hệ thống với vai trò của một quản trị viên.
- + Trên giao diện quản trị, có một trang hoặc phần riêng biệt để quản lý người dùng hoặc tài khoản.
- + Người quản trị tìm kiếm hoặc chọn tài khoản mà họ muốn chỉnh sửa.
- + Sau đó, người quản trị có thể thực hiện các thay đổi cần thiết, như cập nhật thông tin cá nhân, đặc quyền truy cập, hoặc thiết lập các cài đặt liên quan đến tài khoản.
- + Họ lưu lại các thay đổi và thông tin mới sẽ được cập nhật trong hệ thống.

- Xóa tài khoản:

- + Người quản trị truy cập vào trang quản lý người dùng hoặc tài khoản.
- + Người quản trị tìm kiếm hoặc chọn tài khoản mà họ muốn xóa.
- + Sau đó, người quản trị chọn tùy chọn xóa tài khoản và xác nhận hành động này.
- + Tài khoản sẽ được xóa khỏi hệ thống và không còn có quyền truy cập vào trang web nữa.

CHƯƠNG 3 : HÌNH ẢNH DEMO CỦA WEBSITE



Hình 1: trang chủ

Tell Us Your Project

Name *	Email *
Subject *	Phone *
Message *	

SEND MESSAGE

Contact Us

Claritas est etiam processus dynamicus, qui sequitur mutationem consuetudinum lectorum. Mirum est notare quam littera gothica, quam nunc putamus parum claram anteposuerit litterarum formas human.

 Address : 371 Nguyen Kiem, Go Vap, Tp.HCM

 nhannhan0568@gmail.com

 0(1234) 567 890

Working Hours

Monday – Saturday: 08AM – 22PM



Hình 2: liên hệ

[Home](#) > [Login](#)

Login

Account*

Tên tài khoản

Passwords *

Mật khẩu

LOGIN

☐ Remember me

[Lost your password?](#)

Register

Account*

Tên tài khoản

Passwords *

Mật khẩu

Fullname*

Họ và tên

Email*

Email



Phone number*

Số điện thoại

REGISTER

Hình 3: đăng nhập và đăng ký

[Home](#) > [Shopping Cart](#)

DELETE	IMAGE	PRODUCT	PRICE
		Áo Thun Gucci Kiểu Cổ Điển	28

UPDATE CART

USER INFORMATION

Email:

Fullname:

Phone number:

CART TOTALS

Subtotal

\$28.00

Shipping

\$25.00

Total

\$53.00

Login

Hình 4: giỏ hàng

Home > Products

CATEGORIES

☐ Accessories(1)
 ☐ Dresses (2)
 ☐ Tops(3)
 ☐ HandBags(4)

COLOR

Black (10)

Orange (12)

Blue (14)

Yellow (15)

Pink (16)

Green (11)

PRICE

\$0 - \$500

WISHLIST

2 products

» My Wishlists

POPULAR TAGS

ipod

sam sung

apple

iphone 5s

superdrive

shuffle

nano

iphone 4s

canon

LADY FASHION

LOREM IPSUM IS SIMPLY DUMMY TEXT OF THE PRINTING

Showing 1–9 of 21 results

Sort By

Position

NEW

\$35

ÁO THUN DÁNG RỘNG IN CHỮ
GUCCI KIỂU CỔ ĐIỂN CÁ TÍNH

NEW

\$30

ÁO THUN GUCCI KIỂU CỔ ĐIỂN

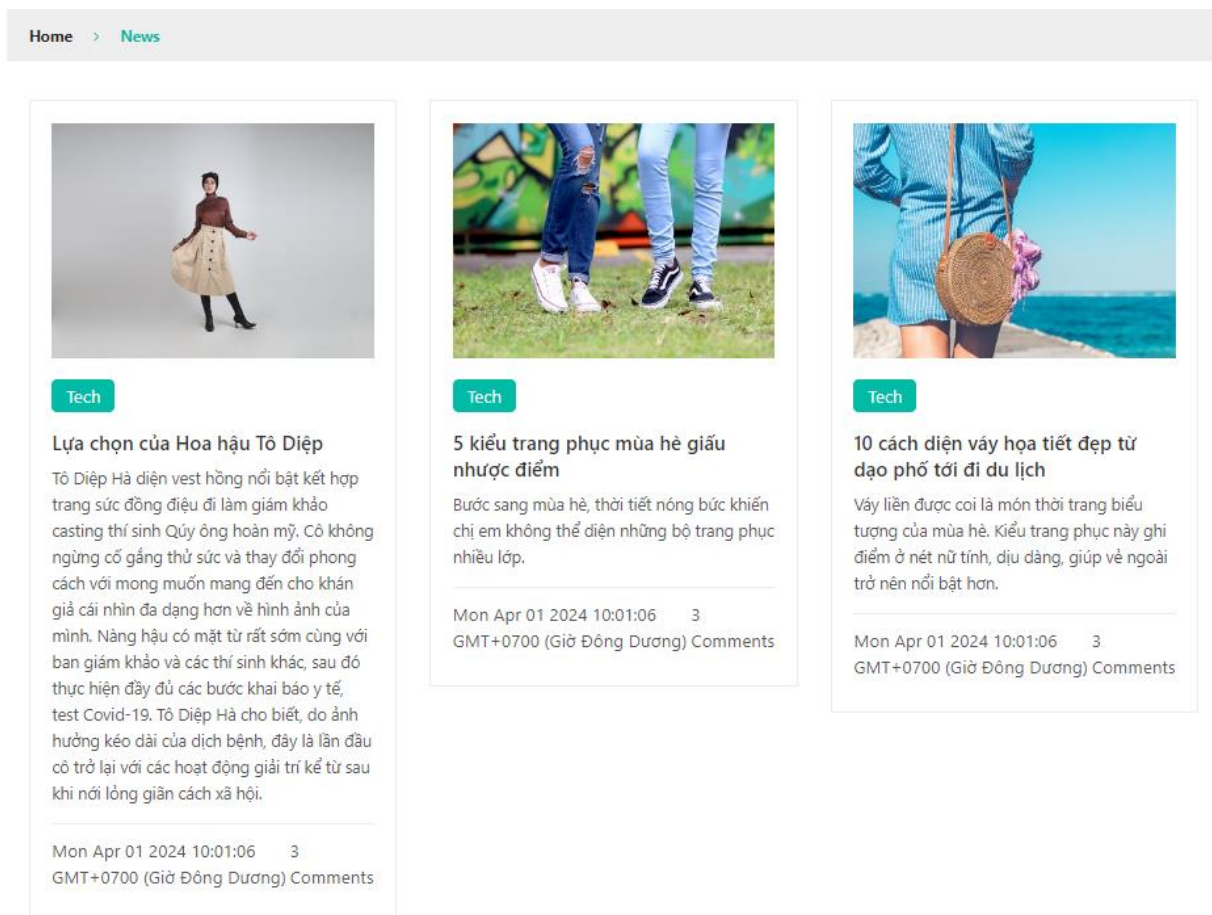
NEW

\$54

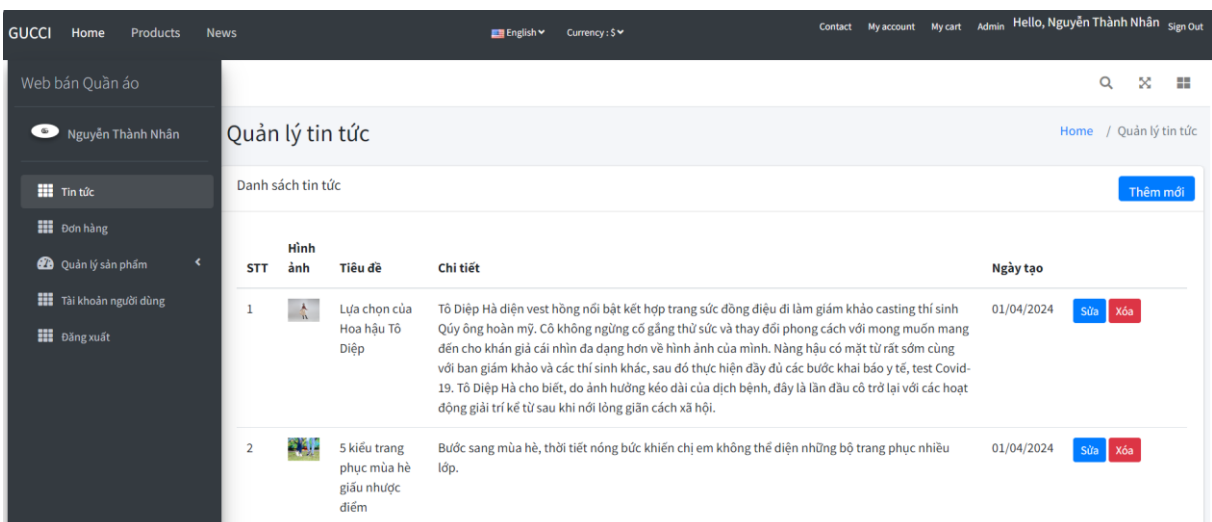
GUCCI - BAMBOO EDP 10ML CHÍNH
HÃNG

Hình 5: danh sách sản phẩm

27



Hình 6: danh sách tin tức



Hình 7: trang quản lý tin tức

Web bán Quần áo

Nguyễn Thành Nhân

Tin tức

Đơn hàng

Quản lý sản phẩm

Tài khoản người dùng

Đăng xuất

Quản lý đơn đặt hàng

Home / Quản lý đơn đặt hàng

Danh sách đơn đặt hàng

STT	Tên người dùng	Email	Số điện thoại	Tên sản phẩm	Số lượng	Tổng	Ngày tạo	
1	Nguyễn Thành Nhân	nhannhan0568@gmail.com	0123456789	Áo Thun Dáng Rộng In Chữ Gucci Kiểu Cổ Điển Cá Tính, Áo Thun Dáng Rộng In Chữ Gucci Kiểu Cổ Điển Cá Tính, Áo Thun Dáng Rộng In Chữ Gucci Kiểu Cổ Điển Cá Tính	3	\$115	Sat Apr 13 2024 23:11:34 GMT+0700 (Giờ Đồng Dương)	<div>SửaXóa</div>
2	Nguyễn Thành Nhân	nhannhan0568@gmail.com	0123456789	Áo Thun Dáng Rộng In Chữ Gucci Kiểu Cổ Điển Cá Tính, Áo Thun Dáng Rộng In Chữ Gucci Kiểu Cổ Điển Cá Tính, Áo Thun Dáng Rộng In Chữ Gucci Kiểu Cổ Điển Cá Tính	3	\$115	Sat Apr 13 2024 23:27:16 GMT+0700 (Giờ Đồng Dương)	<div>SửaXóa</div>
3	Nguyễn Thành Nhân	nhannhan0568@gmail.com	0123456789	Áo Thun Dáng Rộng In Chữ Gucci Kiểu Cổ Điển Cá Tính, Áo Thun Dáng Rộng In Chữ Gucci Kiểu Cổ Điển Cá Tính, Áo Thun Dáng Rộng In Chữ Gucci Kiểu Cổ Điển Cá Tính	3	\$115	Sat Apr 13 2024 23:27:29 GMT+0700 (Giờ	<div>SửaXóa</div>

Hình 8: quản lý đơn hàng

Web bán Quần áo

Nguyễn Thành Nhân

Tin tức

Đơn hàng

Quản lý sản phẩm

Tài khoản người dùng

Đăng xuất

Quản lý sản phẩm

Home / Quản lý sản phẩm

Danh sách sản phẩm

Thêm mới

STT	Hình ảnh	Tên sản phẩm	Danh mục	Số lượng	Giá	Sale	Ngày tạo	
1		Áo Thun Dáng Rộng In Chữ Gucci Kiểu Cổ Điển Cá Tính	Quần áo	7	\$35	\$30	01/04/2024	<div>SửaXóa</div>
2		Áo Thun Gucci Kiểu Cổ Điển	Quần áo	5	\$30	\$28	01/04/2024	<div>SửaXóa</div>
3		Gucci - Bamboo EDP 10ml Chính Hãng	Nước hoa	23	\$54	\$45	08/04/2024	<div>SửaXóa</div>
4		Gucci - Bamboo EDP 10ml Chính Hãng	Nước hoa	23	\$54	\$30	13/04/2024	<div>SửaXóa</div>

Hình 9: quản lý sản phẩm

Web bán Quần áo							
Nguyễn Thành Nhân		Quản lý tài khoản người dùng					
Tin tức		Danh sách thông tin tài khoản người dùng					
Đơn hàng							
Quản lý sản phẩm							
Tài khoản người dùng							
Đăng xuất							
STT	Họ tên	Số điện thoại	Email	Tài khoản	Mật khẩu		
1	Nguyễn Thành Nhân	0123456789	nhannhan0568@gmail.com	Admin123	\$2b\$10\$/jpxt9sA1jx25DgB/aYz0.78xog0ghk7sN4Hd2OaS.CKe0HDw0h.i	Sửa	Xóa
2	Nguyễn Văn A	012345679	Example132@gmail.com	User123	\$2b\$10\$bjmzJxmeVJc4vvBcdW2bguuB6yJFix0gvMPKTegQ3EpcJBdlcDv3m	Sửa	Xóa
3	Phạm Thị B	0123456789	nguyennhan1498@gmail.com	User234	\$2b\$10\$mxRN6bcwclgo9abMaDt5F.lxC7CjKZxm/fN5RgHSf7ZK0h9mL0jX.	Sửa	Xóa

Hình 10: quản lý tài khoản

CHƯƠNG 4 : KẾT QUẢ ĐẠT ĐƯỢC VÀ CHƯA ĐẠT ĐƯỢC

1. Kết quả đạt được:

Giao diện người dùng hấp dẫn: Giao diện trang web bán hàng thời trang được thiết kế hấp dẫn, dễ sử dụng và tương thích với nhiều loại thiết bị.

Tính linh hoạt và mở rộng: Mã nguồn của trang web được tổ chức cấu trúc, dễ bảo trì và mở rộng để hỗ trợ việc thêm mới chức năng hoặc cập nhật.

Tích hợp tính năng quan trọng: Các tính năng quan trọng như tìm kiếm sản phẩm, giỏ hàng, thanh toán an toàn, quản lý đơn hàng và đánh giá sản phẩm đã được triển khai thành công.

Tối ưu hóa trang web: Trang web đã được tối ưu hóa về hiệu suất và tốc độ tải trang để cải thiện trải nghiệm người dùng.

2. Kết quả chưa đạt được:

Tính bảo mật không đủ: Thiếu các biện pháp bảo mật đủ mạnh như xác thực hai yếu tố, mã hóa dữ liệu và bảo vệ chống lại tấn công.

Tương thích trên các trình duyệt: Trang web có thể không hoạt động đúng trên tất cả các trình duyệt hoặc các phiên bản cũ hơn của chúng.

Trải nghiệm người dùng còn thiếu sót: Có thể cần cải thiện trải nghiệm người dùng bằng cách tối ưu hóa giao diện, thêm các tính năng tiện ích và giảm thiểu thời gian đáp ứng.

Tối ưu hóa SEO: Thiếu tối ưu hóa SEO có thể làm giảm khả năng tìm thấy của trang web trên các công cụ tìm kiếm.

Kiểm thử chưa đủ: Cần tiến hành kiểm thử đầy đủ để phát hiện và khắc phục các lỗi hoặc vấn đề không mong muốn trước khi triển khai trang web.

TÀI LIỆU THAM KHẢO

1. Bài giảng của giảng viên Lê Huỳnh Phước, giảng viên trường đại học Gia Định.
2. Các thông tin chi tiết về node.js trên trang topdev.vn
3. Mô tả trên trang chủ của nodejs.org
4. Các tài liệu tham khảo về bootstrap trong việc thiết kế giao diện trên trang getbootstrap.com
5. Tài liệu tham khảo truy vấn cơ sở dữ liệu trên mongoosejs.com