BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC GIA ĐỊNH**

**KHOA: CÔNG NGHỆ THÔNG TIN**

Logo, company name

Description automatically generated

**TIỂU LUẬN**

**XÂY DỰNG WEBSITE BÁN QUẦN ÁO NAM VÀ NỮ**

**MÔN: LẬP TRÌNH NODE.JS**

Ngành: **CÔNG NGHỆ THÔNG TIN**

Chuyên ngành: **CÔNG NGHỆ PHẦN MỀM**

Giảng viên hướng dẫn: **LÊ HUỲNH PHƯỚC**

Sinh viên thực hiện: **NHÓM 8**

MSSV:

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: Lập trình Node.js**

1. **Họ và tên sinh viên: NHÓM 8**

**Tên đề tài**: Xây dựng website bán quần áo nam và nữ

1. **Nhận xét**:

***Những kết quả đạt được:***

***Những hạn chế:***

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

Sinh viên:…………………………………………………………………………….

Điểm số: ……….…… Điểm chữ: ……………………………………………….

|  |  |
| --- | --- |
|  | *TP. HCM, ngày … tháng … năm 20……*  **Giảng viên chấm thi**  *(Ký và ghi rõ họ tên)* |

**MỤC LỤC**

[**MỞ ĐẦU 1**](#_Toc164167591)

[**CHƯƠNG 1: GIỚI THIỆU VỀ NODEJS 2**](#_Toc164167592)

[**1.1 Khái niệm 2**](#_Toc164167593)

[**1.2 Lịch sử phát triển của nodejs 2**](#_Toc164167594)

[**1.3 Ưu điểm và nhược điểm của nodejs 3**](#_Toc164167595)

[**1.4 NodeJS hoạt động ra sao? 4**](#_Toc164167596)

[**1.5 Cấu trúc của NodeJS là gì? 4**](#_Toc164167597)

[**1.6 NodeJS được sử dụng để làm gì? 5**](#_Toc164167598)

[**2. MongoDB 6**](#_Toc164167599)

[**CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ CHỨC NĂNG 9**](#_Toc164167600)

[**2.1 Phân tích nghiệp vụ 9**](#_Toc164167601)

[**2.2 Thiết kế chức năng 10**](#_Toc164167602)

[**2.3.Biểu đồ phân cấp chức năng 11**](#_Toc164167603)

[**2.4 Biểu đồ luồng dữ liệu: 12**](#_Toc164167604)

[**2.5 Biểu đồ usecase: 14**](#_Toc164167605)

[**2.6 Biểu đồ tuần tự 15**](#_Toc164167606)

[**2.7.Phân tích thiết kế dữ liệu 23**](#_Toc164167607)

[**2.7.1.Xác định các thực thể và các thuộc tính cho thực thể 23**](#_Toc164167608)

[**2.7.2.Sơ đồ ER: 24**](#_Toc164167609)

[**2.7.3.Sơ Đồ RM: 25**](#_Toc164167610)

[**2.8. Công nghệ sử dụng : 26**](#_Toc164167611)

[**CHƯƠNG 3: XÂY DỰNG VÀ CÀI ĐẶT CHƯƠNG TRÌNH 28**](#_Toc164167612)

[**3.1 Xây dựng giao diện 28**](#_Toc164167613)

[**CHƯƠNG 4: KẾT LUẬN 33**](#_Toc164167614)

[**Tài liệu tham khảo 34**](#_Toc164167615)

**MỞ ĐẦU**

Ngày nay, cùng với sự phát triển của đất nước ngành Công nghệ thông tin đã có những bước phát triển mạnh mẽ không ngừng và tin học đã trở thành chiếc chìa khóa dẫn đến thành công cho nhiều cá nhân trong nhiều lĩnh vực, hoạt động. Với những ứng dụng của mình, ngành Công nghệ thông tin đã góp phần mang lại nhiều lợi ích mà không ai có thể phủ nhận được. Đặc biệt là trong lĩnh vực quản lý kinh tế, tin học đã góp phần tạo ra sự thay đổi nhanh chóng cho bộ mặt xã hội. Nhất là khi việc tin học hóa vào công tác quản lý, bán hàng là một trong những yêu cầu cần thiết đối với các doanh nghiệp hiện nay.

Từ những phần mềm quản lý giúp cho công việc bán hàng của cửa hàng trở nên nhanh chóng và dễ dàng, ngày nay công nghệ thông tin đã phát triển mạnh mẽ hơn nữa, Website ra đời không chỉ đáp ứng cho các khách hàng đến trực tiếp cửa hàng mua sản phẩm mà nó còn phục vụ cho những khách hàng ở xa. Không những thế, việc giới thiệu, quảng bá hình ảnh về cửa hàng, công ty được mở rộng trên quy mô lớn giúp cho việc kinh doanh, buôn bán, trao đổi tin tức thuận lợi hơn rất nhiều, mang lại hiệu hiệu quả kinh tế cao hơn và tạo uy tín trong lòng khách hàng.

Chính vì lợi ích mà Website mang lại mà các cửa hàng, công ty đã áp dụng nó để phát triển công việc kinh doanh và giới thiệu cửa hàng của mình. Trong số đó, thời trang cũng là một ngành sử dụng nhiều đến Website. Do vậy, em lựa chọn đề tài “Thiết kế ứng dụng web thương mại điện bán quần áo AndShop” nhằm đưa đến cho khách hàng một địa điểm lý tưởng trong việc mua sắm.

**CHƯƠNG 1: GIỚI THIỆU VỀ NODEJS**

### **Khái niệm**

Node.js là một nền tảng phát triển ứng dụng được xây dựng trên JavaScript runtime của Chrome V8 Engine. Node.js cho phép thực thi mã JavaScript không chỉ trên trình duyệt web mà còn trên máy chủ, mở ra một loạt các cơ hội mới cho việc phát triển ứng dụng web hiệu quả và linh hoạt.

Node.js được thiết kế để xử lý các yêu cầu đồng thời mà không cần phải chờ đợi các yêu cầu trước đó hoàn thành, nhờ vào mô hình xử lý không đồng bộ (asynchronous). Điều này giúp tăng cường hiệu suất của ứng dụng, đặc biệt là trong các ứng dụng thời gian thực như trò chơi trực tuyến, trò chuyện video, và các ứng dụng cộng tác trực tuyến.

Một trong những điểm mạnh của Node.js là hệ thống mô-đun phong phú, chủ yếu được quản lý thông qua npm (Node Package Manager). Nhà phát triển có thể dễ dàng sử dụng các thư viện và công cụ bên ngoài để tăng cường chức năng và hiệu suất của ứng dụng.

Node.js cũng được ứng dụng rộng rãi trong xây dựng API và kiến trúc microservices, cung cấp một cách tiếp cận linh hoạt và hiệu quả cho việc phát triển các ứng dụng web phức tạp và phân tán.

* 1. **Lịch sử phát triển của nodejs**

Năm 2009: Ryan Dahl, một nhà phát triển phần mềm, giới thiệu Node.js lần đầu tiên tại PyCon. Ông đã sử dụng JavaScript và Chrome V8 engine của Google để tạo ra một nền tảng máy chủ có thể xử lý đa nhiệm một cách hiệu quả.

Năm 2010: Node.js chính thức ra mắt phiên bản đầu tiên với mã nguồn mở. Cộng đồng phát triển Node.js bắt đầu phát triển mạnh mẽ và sự quan tâm từ các nhà phát triển web tăng lên.

Năm 2011: npm (Node Package Manager) được giới thiệu, mang lại một cách tiện lợi để quản lý các mô-đun và gói trong môi trường Node.js. Điều này đã giúp cải thiện quá trình phát triển và chia sẻ mã nguồn một cách dễ dàng hơn.

Năm 2012: Node.js Foundation được thành lập để quản lý và duy trì dự án Node.js. Điều này là một bước quan trọng trong việc tăng cường sự ổn định và phát triển của Node.js như một dự án mã nguồn mở.

Năm 2015: Phiên bản LTS (Long-Term Support) đầu tiên của Node.js được phát hành, đảm bảo tính ổn định và hỗ trợ dài hạn cho các ứng dụng sử dụng Node.js.

Năm 2016: Node.js v6 được phát hành với nhiều cải tiến hiệu suất và tính năng mới, bao gồm hỗ trợ ES6 (ECMAScript 2015) mặc định.

Năm 2018: Hai nhóm Node.js và io.js hợp nhất lại thành một dự án duy nhất dưới sự lãnh đạo của Node.js Foundation, tạo ra một nền tảng mạnh mẽ và đồng nhất cho cộng đồng Node.js.

Năm 2020: Node.js 14 được phát hành, là phiên bản LTS mới nhất tích hợp nhiều tính năng mới và cải tiến, bao gồm hỗ trợ ES modules.

## 1.3 Ưu điểm và nhược điểm của nodejs

\* Ưu điểm của NodeJS

Một số ưu điểm nổi bật có thể kể đến như: IO hướng sự kiện không đồng bộ và cho phép xử lý nhiều yêu cầu cùng một lúc, sử dụng JavaScript, chia sẻ cùng code ở cả hai phía server và client, Node Package Manager và module đang ngày càng phát triển rất mạnh mẽ. Ngoài ra còn cho phép stream file có kích thước lớn và được cộng đồng hỗ trợ tích cực.

\* Nhược điểm của NodeJS

Node JS có những nhược điểm như: việc không có khả năng mở rộng, vì vậy nên không thể tận dụng lợi thế mô hình đa lõi trong các phần cứng cấp server, đòi hỏi phải có kiến thức tốt về JavaScript và rất khó thao tác với cơ sở dữ liệu. Ngoài ra, nó không phù hợp với những tác vụ đòi hỏi nhiều về CPU, mỗi lần callback sẽ đính kèm nhiều callback lồng nhau khác

## 1.4 NodeJS hoạt động ra sao?

Ý tưởng chính của NodeJS đó là sử dụng non-blocking, hướng sự ra vào dữ liệu thông qua những tác vụ thời gian thực một cách nhanh chóng hơn. Bởi vì NodeJS có khả năng xử lý một lượng lớn các kết nối đồng thời bằng thông lượng cao và có khả năng mở rộng một cách nhanh chóng.

Nếu như các trang web truyền thống tạo ra những xử lý yêu cầu mới và chiếm RAM của hệ thống thì tài nguyên của hệ thống được sử dụng sẽ không hiệu quả. Chính vì vậy, giải pháp mà NodeJS đưa ra đó là sử dụng luồng đơn (Single-Threaded), kết hợp cùng với non-blocking I/O để thực thi các yêu cầu, cho phép hỗ trợ hàng trăm ngàn kết nối đồng thời.

## 1.5 Cấu trúc của NodeJS là gì?

Module

Module giống như thư viện JavaScript được sử dụng trong ứng dụng NodeJS Application bao gồm nhiều chức năng.

Console

Bảng điều khiển tương tự như bảng điều khiển của Javascript cung cấp những phương thức gỡ lỗi trên trình duyệt Internet. Console sẽ in các thông báo ra stderr và stdout.

Cluster

NodeJS được xây dựng theo ý tưởng lập trình đơn luồng. Cluster là module cho phép đa luồng bằng cách tạo ra những quy trình con có chung cổng máy chủ, chạy đồng thời.

Global

Biến toàn cục trong NodeJS thường tồn tại trong tất cả các module. Những biến này bao gồm các module, hàm, module, string…

Error Handling

Các lỗi trong NodeJS sẽ được xử lý qua các exception. Ví dụ, chúng ta phải xử lý lỗi xảy ra khi chia cho 0. Lỗi này thường làm crash ứng dụng, do đó chúng ta cần phải xử lý lỗi này để ứng dụng có thể tiếp tục hoạt động bình thường.

## 1.6 NodeJS được sử dụng để làm gì?

Backend for Social Media Networking

Có một sự kết hợp tất yếu giữa các nút để tạo phần phụ trợ cho trang mạng truyền thông xã hội. NodeJS cung cấp khả năng định tuyến cực kỳ nhanh với động cơ V8 cùng với sự xác thực an toàn. Khả năng mở rộng là một trong các yếu tố quan trọng nhất để làm backend phong phú thêm cho mạng truyền thông xã hội.

Chatbots

NodeJS cung cấp những chức năng nâng cao độc quyền cho chatbots và các ứng dụng trò chuyện. Các tính năng chính như: dữ liệu chuyên sâu, ứng dụng nhiều người dùng và lưu lượng truy cập rất lớn trên tất cả các thiết bị đều phổ biến trong chatbots. NodeJS tốt vì nó chạy được trên nhiều thiết bị khác nhau và bao gồm tất cả mô hình trong chatbots và ứng dụng trò chuyện. Nó còn giúp việc thực thi thông báo đẩy và các vòng lặp sự kiện phía máy chủ trở nên dễ dàng hơn.

Single-page Application (SPA) Development

NodeJS cũng giống như những trang web đơn lẻ, nó có thể tự xây dựng một ứng dụng, một trang hay nơi giao diện tương tự như ứng dụng dành cho máy tính. Bởi những tùy chọn linh hoạt nên NodeJS phù hợp để xây dựng các nền tảng mạng xã hội, giải pháp để gửi thư và trang web động. Ngoài ra, chất lượng luồng dữ liệu không đồng bộ trong NodeJS cũng khiến nó trở thành sự lựa chọn tốt nhất để phát triển SPA.

Data Streaming

NodeJS có lợi thế rõ ràng là tệp xử lý ở tốc độ rất nhanh, mã hóa và tải lên nhẹ nhàng. Điều này được nhiều cửa hàng thời trang trực tuyến và các ứng dụng có video truyền tải dữ liệu lớn áp dụng.

IoT Application Development

IoT đã trở nên phổ biến từ những thập kỷ gần đây. Từ đó NodeJS là một trong những giải pháp được ưa chuộng nhất cho các tổ chức muốn xây dựng hệ thống giải pháp IoT riêng tư và công cộng. Lợi thế cốt lõi của NodeJS đó là khả năng xử lý các yêu cầu liền mạch và đồng thời với hàng chục nghìn sự kiện được phát hành bởi hàng tỷ thiết bị trên mạng IoT. Ngoài ra, NodeJS hoạt động trên các luồng, kênh có thể đọc và ghi được. Chính vì vậy, nó là nền tảng phù hợp nhất để phát triển IoT.

**1.7 Một số lý do nên sử dụng NodeJS**

Tốc độ nhanh: NodeJS được xây dựng dựa theo engine JavaScript V8m do đó các thư viện của nó có thể thực thi code rất nhanh chóng.

NPM: Với hơn 60,000 package khác nhau, các nhà phát triển có thể dễ dàng lựa chọn bất cứ tính năng nào để xây dựng cho ứng dụng của mình.

Lập trình không đồng bộ: Tất cả API của NodeJS đều có tính không đồng bộ, vì vậy một server dựa trên NodeJS không phải đợi API trả về dữ liệu.

Không có buffering: NodeJS giúp tiết kiệm được khoản thời gian lớn xử lý file khi cần upload video hay âm thanh. Vì những ứng dụng này không buffer dữ liệu mà chỉ xuất dữ liệu theo từng phần.

Đơn luồng: phần mềm sử dụng mô hình đơn luồng với những vòng lặp sự kiện. Vì vậy, các ứng dụng có thể xử lý lượng yêu cầu lớn hơn rất nhiều so với các server truyền thống.

## 2. MongoDB

MongoDB là gì? MongoDB là một database hướng tài liệu (document), một dạng NoSQL database. Vì thế, MongoDB sẽ tránh cấu trúc table-based của relational database để thích ứng với các tài liệu như JSON có một schema rất linh hoạt gọi là BSON. MongoDB sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ các các kích cỡ và các document khác nhau. Các dữ liệu được lưu trữ trong document kiểu JSON nên truy vấn sẽ rất nhanh.

**2.1 Khi nào sử dụng MongoDB**

Quản lý và truyền tải content – Quản lý đa dạng nhiều product của content chỉ trong một kho lưu trữ data cho phép thay đổi và phản hồi nhanh chóng mà không chịu thêm phức tạp thêm từ hệ thống content.

Cấu trúc Mobile và Social – MongoDB cung cấp một platform có sẵn, phản xạ nhanh, và dễ mở rộng cho phép rất nhiều khả năng đột phá, phân tích real-time, và hỗ trợ toàn cầu.

Quản lý data khách hàng – Tận dụng khả năng query nhanh chóng cho phân tích real-time trên cơ sở dữ liệu người dùng cực lớn vớ các mô hình data phức tạp bằng các schema linh hoạt và tự động sharding cho mở rộng chiều ngang.

**2.2 Ưu nhược điểm của MongoDB**

Ưu điểm của MongoDB

MongoDB mang đến cho người dùng khá nhiều lợi ích:

Linh hoạt trong lưu trữ các kích cỡ dữ liệu khác nhau. Nhờ chúng được lưu dưới dạng JSON nên bạn thoải mái chèn bất kỳ thông tin nào tùy theo nhu cầu sử dụng.

Tiết kiệm thời gian trong việc kiểm tra sự tương thích về cấu trúc khi thêm, xóa hoặc cập nhật dữ liệu. Nhờ MongoDB không có sự ràng buộc trong một khuôn khổ, quy tắc nhất định nào.

Bạn dễ dàng mở rộng hệ thống thông qua việc thêm node vào cluster. Cụm các node này đóng vai trò như thư viện chứa các dữ liệu giao tiếp với nhau.

Tốc độ truy vấn của MongoDB nhanh hơn so với RDBMS do toàn bộ dữ liệu truy vấn đã được ghi đệm lên bộ nhớ RAM. Nhờ thế, những lượt truy vấn sau sẽ được rút ngắn thời gian vì chúng không cần đọc từ ổ cứng.

Trường dữ liệu “\_id” (đại diện cho giá trị duy nhất trong mỗi document) được tự động đánh chỉ mục nên hiệu suất luôn đạt mức cao nhất.

Nhược điểm của MongoDB

Bên cạnh các ưu điểm, MongoDB vẫn còn tồn tại một số điểm hạn chế mà bạn cần chú ý khi cài đặt và sử dụng:

Vì dữ liệu không bị ràng buộc nên trong quá trình sử dụng, bạn cần cẩn thận trong mọi thao tác nhằm tránh xảy ra những điều không mong muốn, làm ảnh hưởng đến dữ liệu.

Chương trình MongoDB tiêu tốn khá nhiều dung lượng bộ nhớ do dữ liệu được lưu dưới dạng key và value. Bên cạnh đó, một số collection chỉ có sự khác biệt về value nên việc lặp lại key là điều khó tránh khỏi. Điều này dẫn đến thừa dữ liệu.

Thông thường, thời gian để dữ liệu chuyển đổi từ RAM xuống ổ cứng khoảng 60s nên nguy cơ bị mất dữ liệu nếu xảy ra mất điện là điều có thể xảy ra.

**2.3 MongoDB hoạt động như thế nào?**

MongoDB hoạt động dưới dạng một hệ thống cơ sở dữ liệu phi quan hệ, lưu trữ dữ liệu dưới dạng tài liệu (document) JSON. Dữ liệu được lưu trữ trong collections và documents. Do đó, database, collection và documents sẽ có liên quan với nhau

Cơ sở dữ liệu MongoDB lưu trữ tài liệu trong các collections, tương tự như bảng trong cơ sở dữ liệu quan hệ. Mỗi collection có thể chứa nhiều tài liệu (documents) có cấu trúc dữ liệu tùy ý.

Bây giờ bên trong collection sẽ có tài liệu (documents). Các tài liệu này sẽ chứa dữ liệu mà bạn muốn lưu trữ trong MongoDB database. Mỗi document có thể chứa nhiều fields dữ liệu, mỗi field được định danh bằng tên và có giá trị tương ứng.

Các tài liệu (documents) được tạo bằng cách sử dụng các field. Các field là các key-value pair trong tài liệu, nó giống như các cột trong cơ sở dữ liệu quan hệ. Giá trị của fields có thể thuộc bất kỳ loại dữ liệu BSON nào như double, string, boolean,...

MongoDB hỗ trợ việc tạo index cho các field dữ liệu trong collection, giúp tăng tốc độ truy vấn. Chúng còn hỗ trợ sao chép dữ liệu giữa các node trong một cluster giúp đảm bảo tính khả dụng và độ tin cậy của hệ thống.

MongoDB phân tán dữ liệu trên nhiều node, giúp tăng khả năng mở rộng của hệ thống, đồng thời chúng còn hỗ trợ tính toán phân tán bằng cách sử dụng MapReduce giúp xử lý dữ liệu lớn một cách hiệu quả.

Khi sử dụng MongoDB, bạn có thể sử dụng API và driver của MongoDB để truy cập và thao tác với dữ liệu trong hệ thống cơ sở dữ liệu này.

**CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ CHỨC NĂNG**

**2.1 Phân tích nghiệp vụ**

Đối tượng khách hàng nhắm tới của cửa hàng sẽ là các bạn trẻ có độ tuổi từ 18-30 tuổi .

Những mặt hàng được bán sẽ các áo thun , áo tank , áo thể thao thời trang đang được rất nhiều các bạn trẻ yêu thích hiện nay

Sản phẩm sẽ được hiển thị và bán online trên Web của của hàng .

**2.2 Thiết kế chức năng**

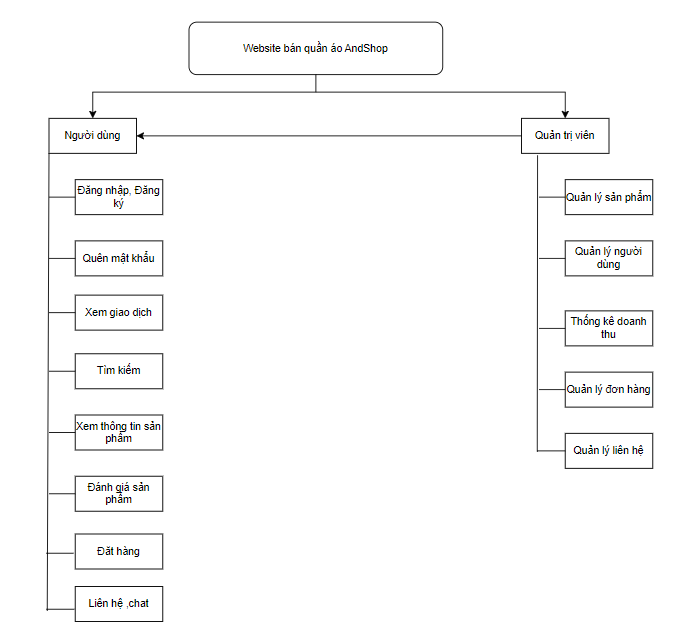
Về phía Người mua hàng:

* Đăng ký tài khoản, đăng nhập ,quên mật khẩu, đổi mật khẩu
* Xem,Sửa thông tin cá nhân ,
* Xem trang chủ, xem sản phẩm.
* Tìm kiếm sản phẩm.
* Thêm sản phẩm vào giỏ hàng, sửa giỏ hàng.
* Mua hàng.
* Đánh giá sản phẩm.
* Xem chi tiêu cá nhân
* Liên hệ , chat

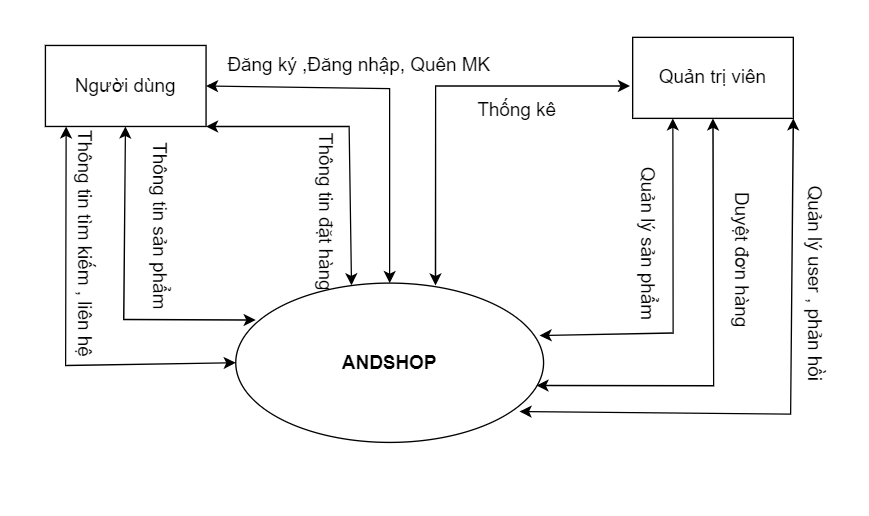
Về phía Quản trị viên:

* Quản lý các sản phẩm.
* Thống kê số lượng đơn hàng và doanh thu bán hàng… .
* Quản lý đơn hàng, chi tiết đơn hàng.
* Có quyền như user

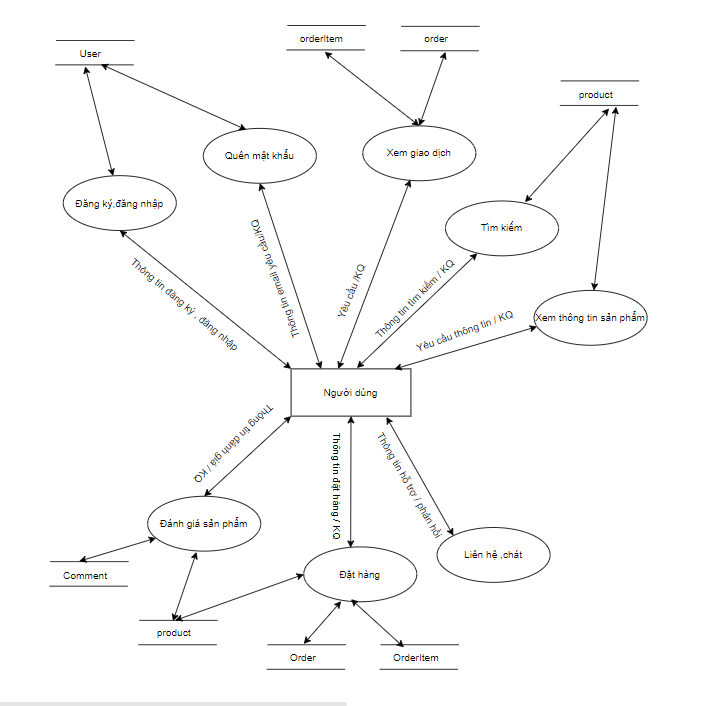
**2.3.Biểu đồ phân cấp chức năng**



**Sơ đồ ngữ cảnh**



**2.4 Biểu đồ luồng dữ liệu:**

****

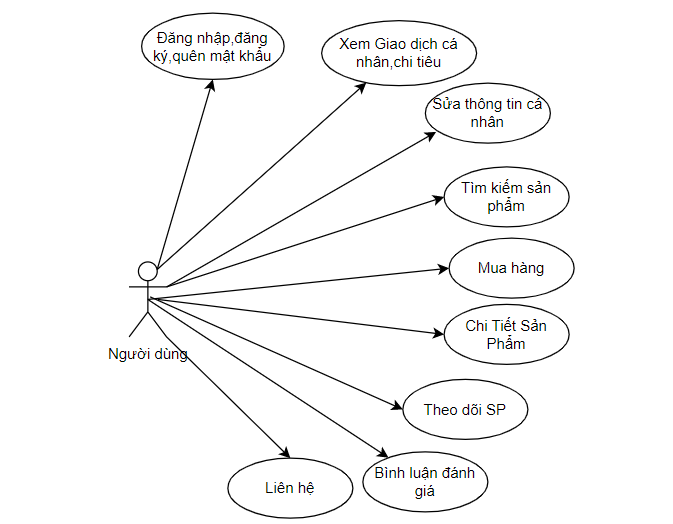
Biểu đồ chức năng của người dùng

****

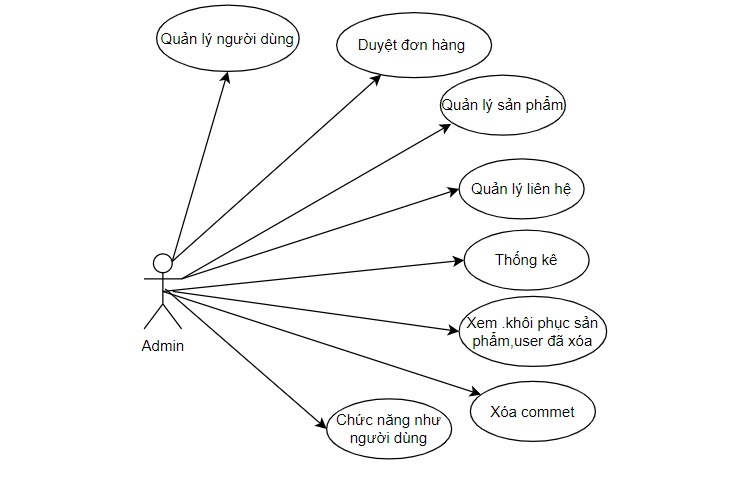
Biểu đồ chức năng quản trị viên

**2.5 Biểu đồ usecase:**

Người dùng:



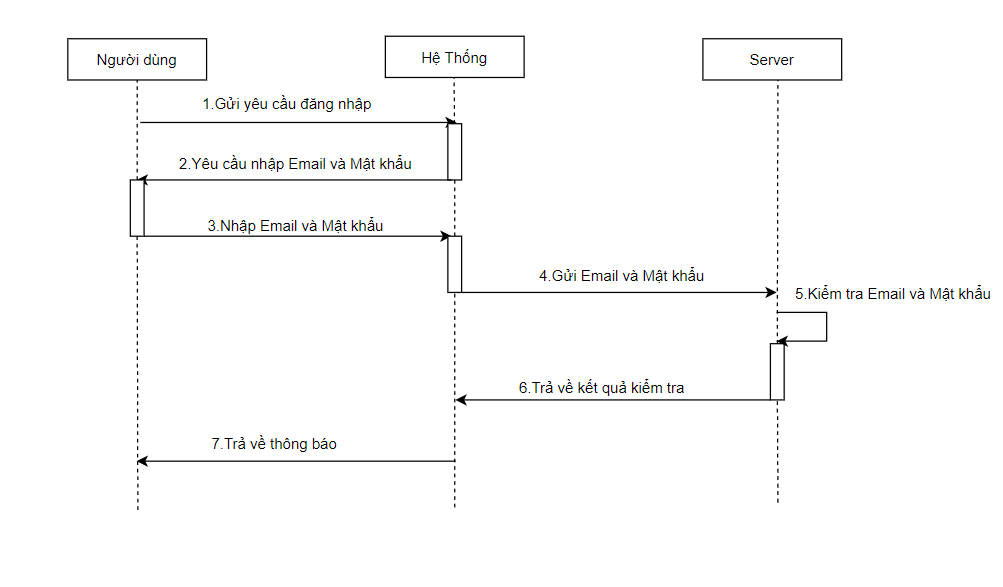
-Admin

****

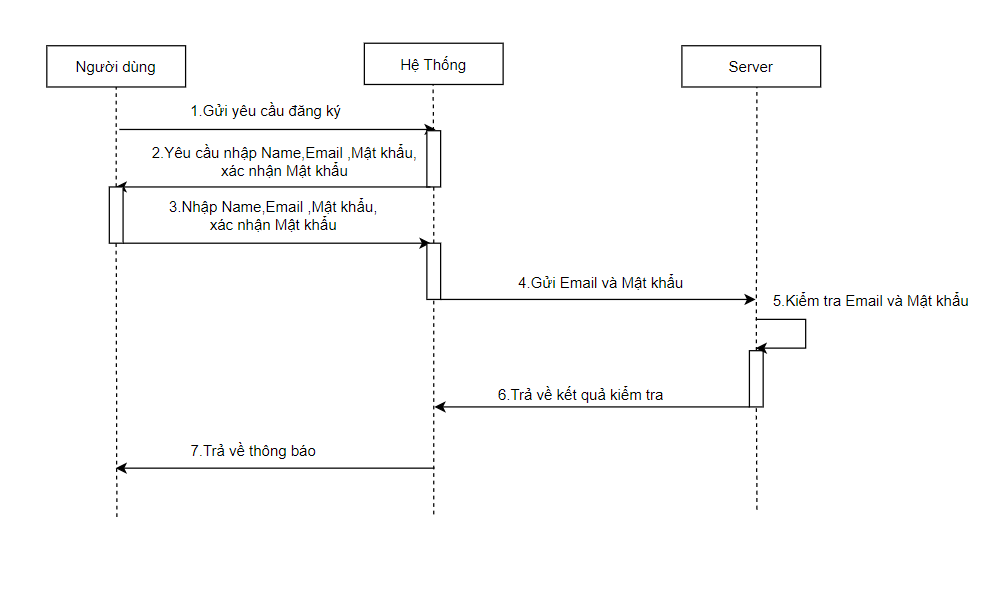
**2.6 Biểu đồ tuần tự**

**Của người dùng**

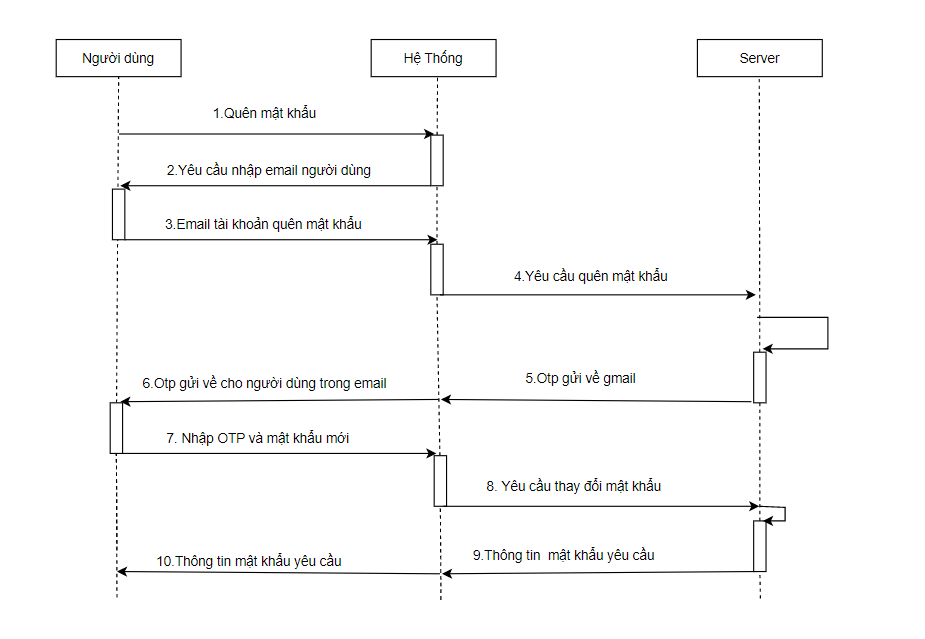
+ Đăng nhập:

****

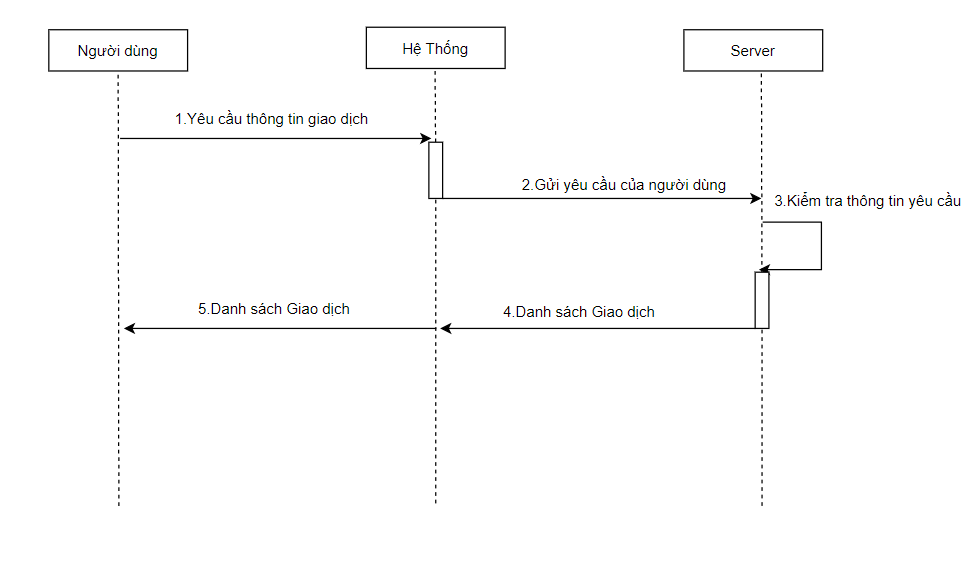
**+**Đăng ký:

****

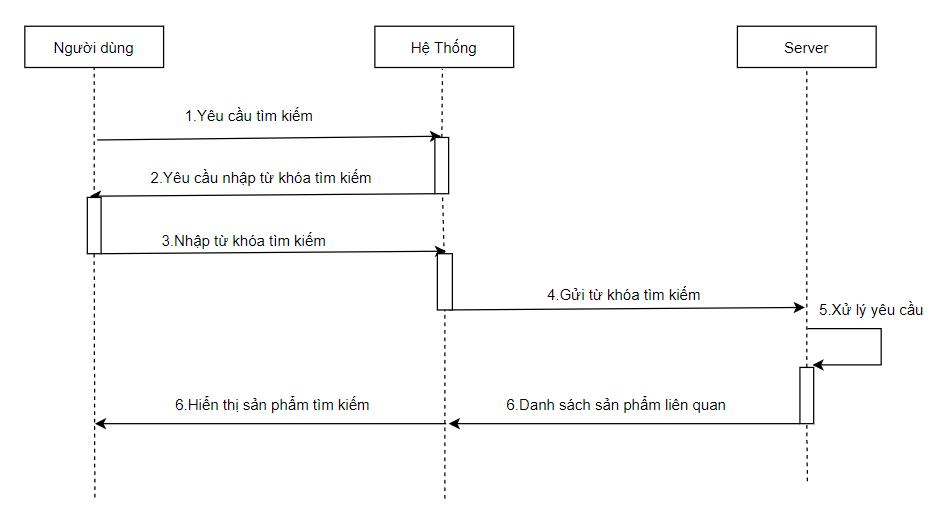
+Quên mật khẩu:



+Xem giao dịch:

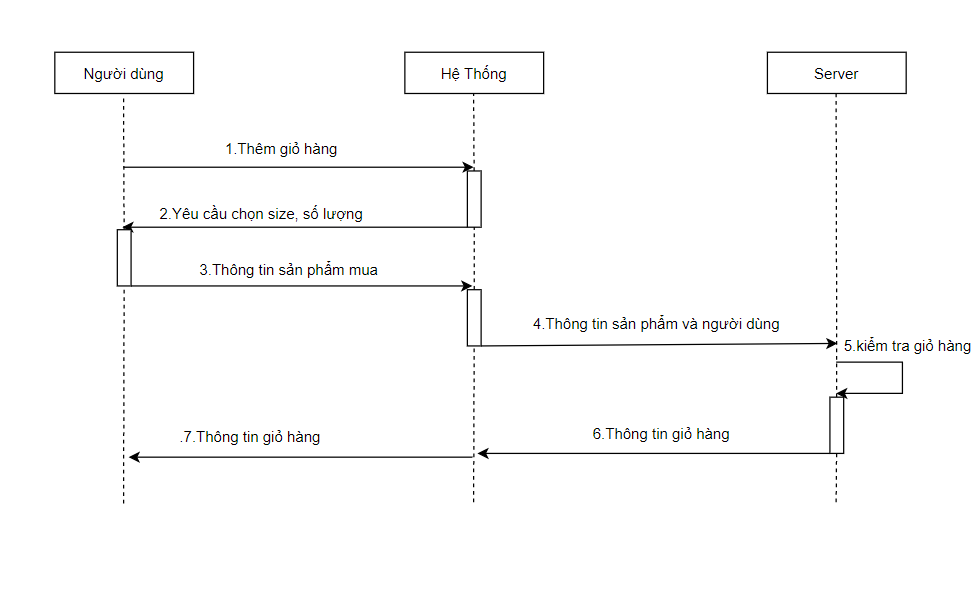
****

+Tìm kiếm:

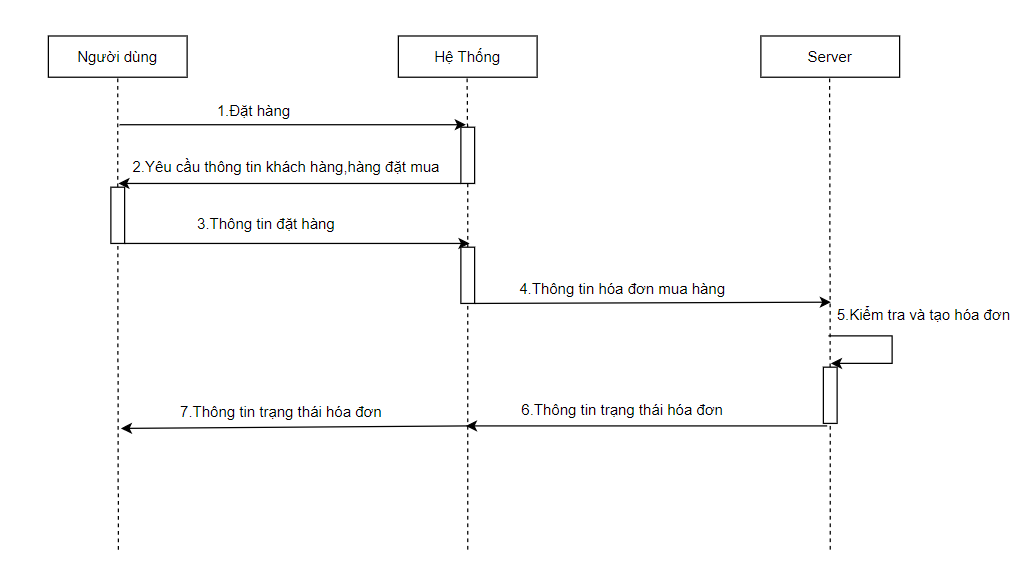
****

-Mua Hàng:

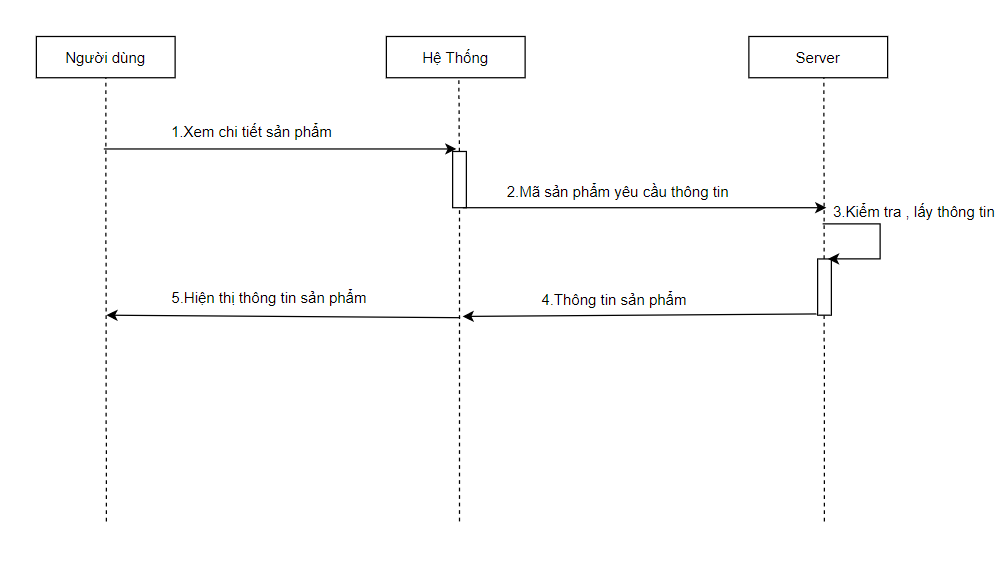
+ thêm giỏ hàng

****

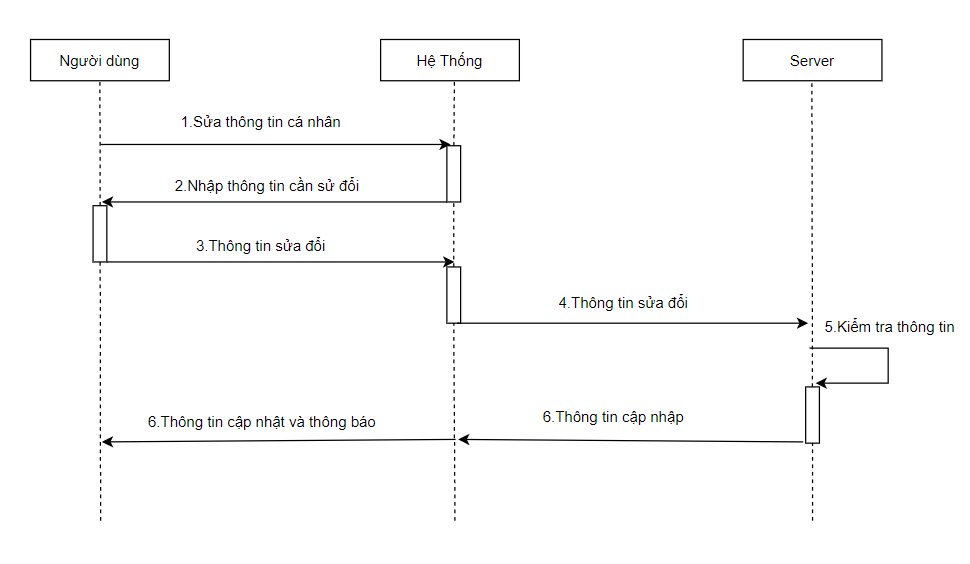
+Đặt hàng:

****

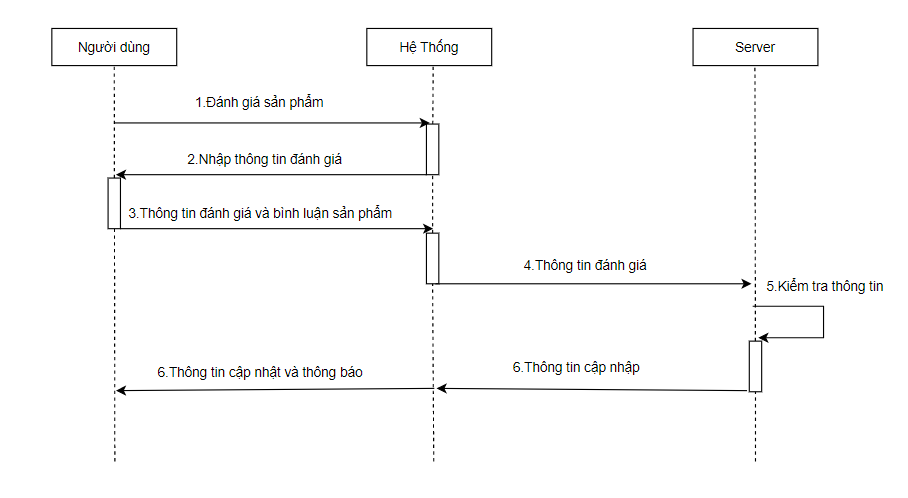
+Xem chi tiết sản phẩm:

****

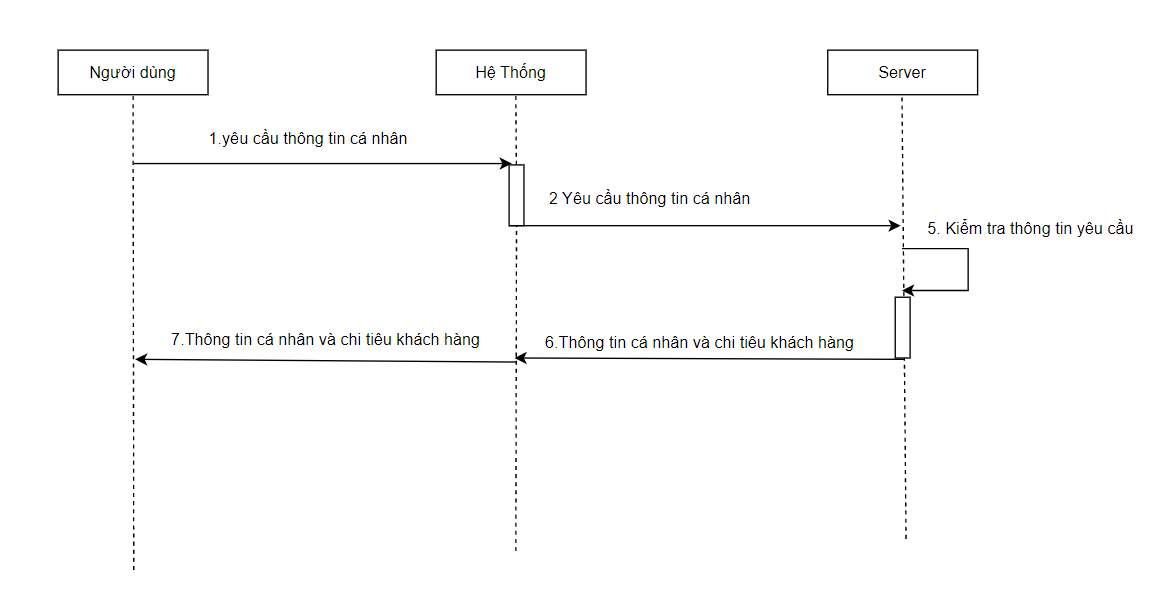
+Sửa đổi thông tin :

****

**+** Đánh giá sản phẩm :

****

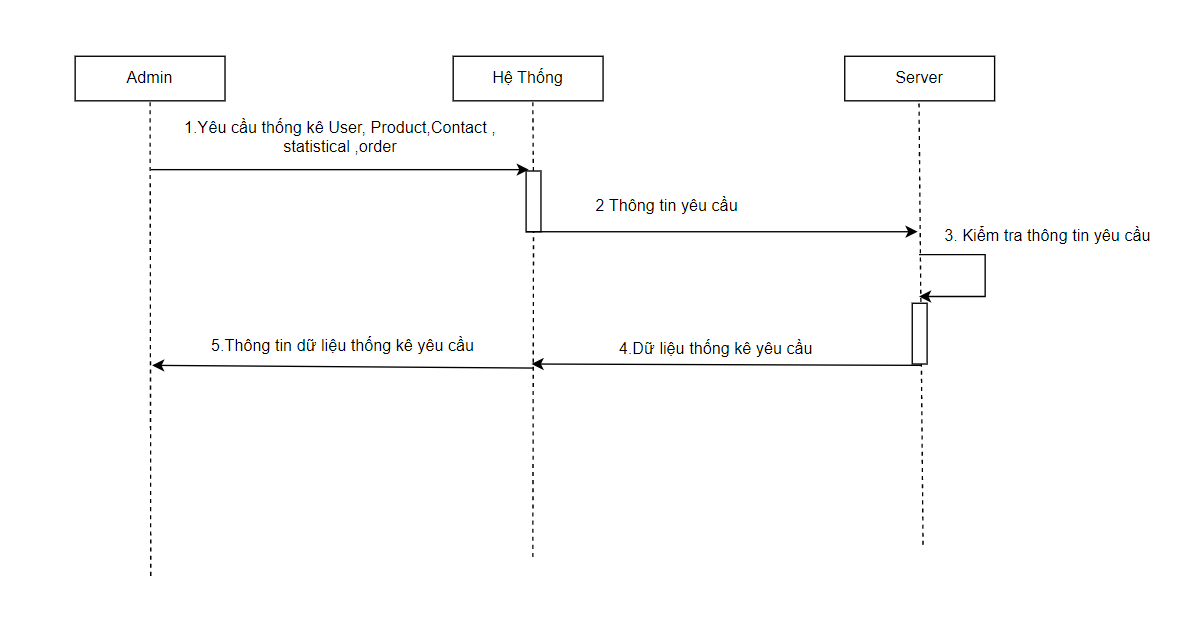
+Xem thông tin cá nhân:

****

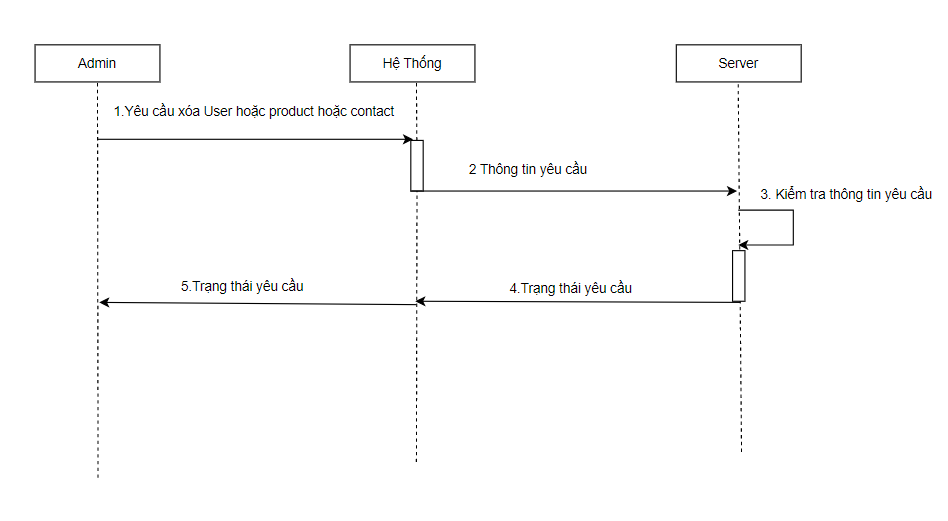
Của Người quản trị :

- Bao gồm tất cả các chức năng như ở trên và ở bên dưới:

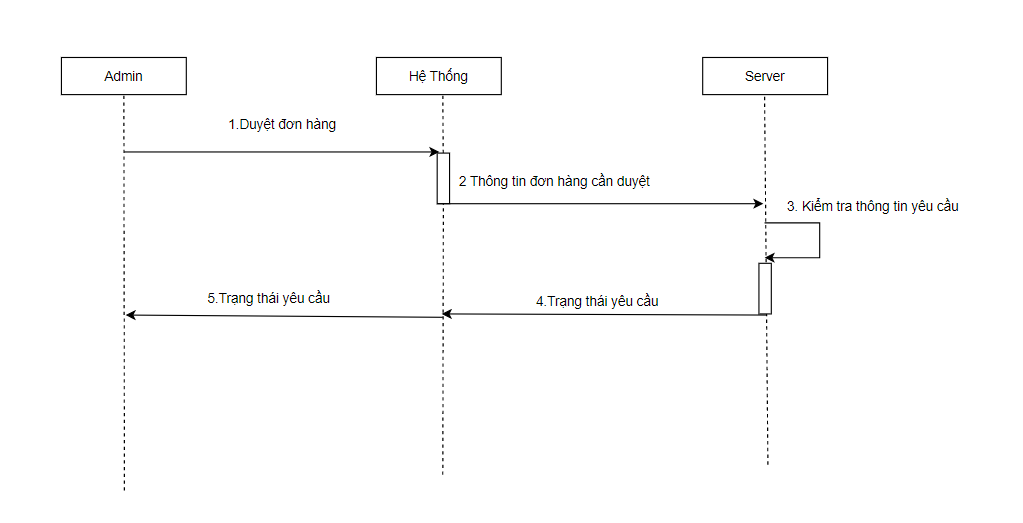
+ Thống kê User , Product , Statistical , order , contact :



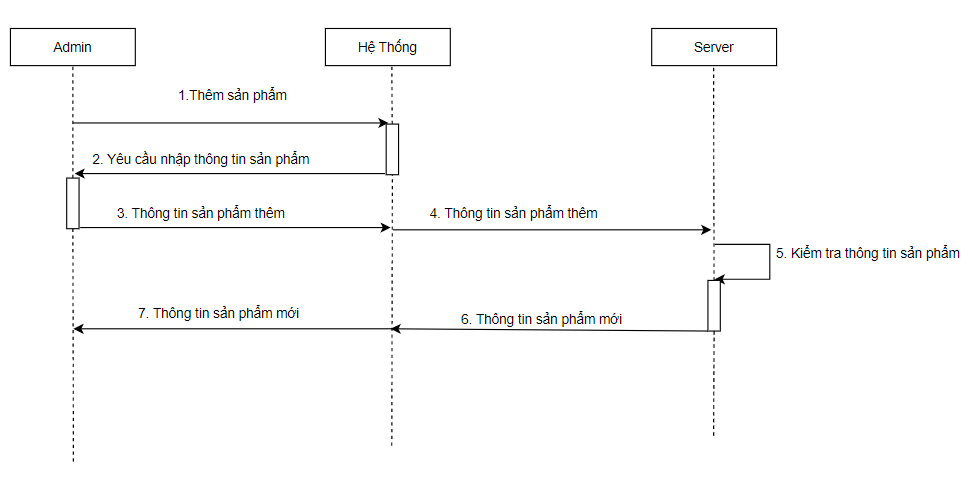
Xóa User , Product , contact :



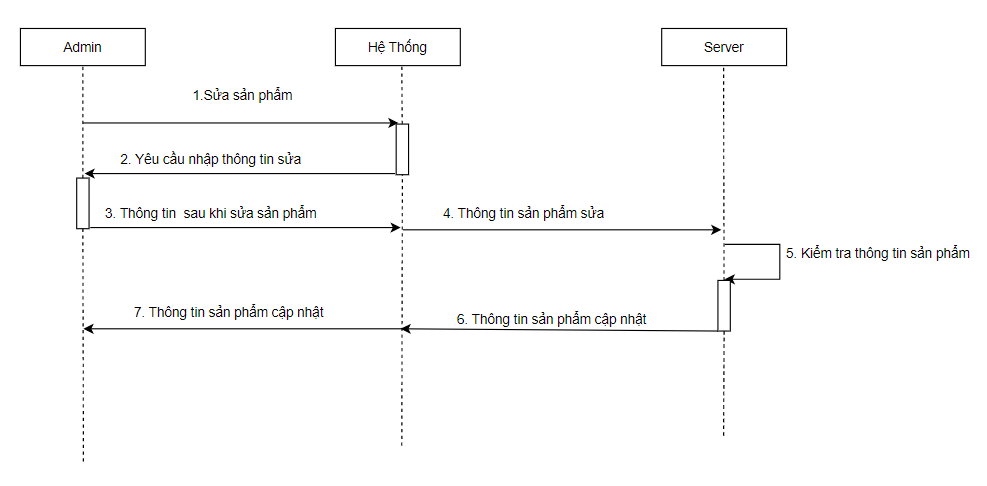
+Duyệt đơn hàng :



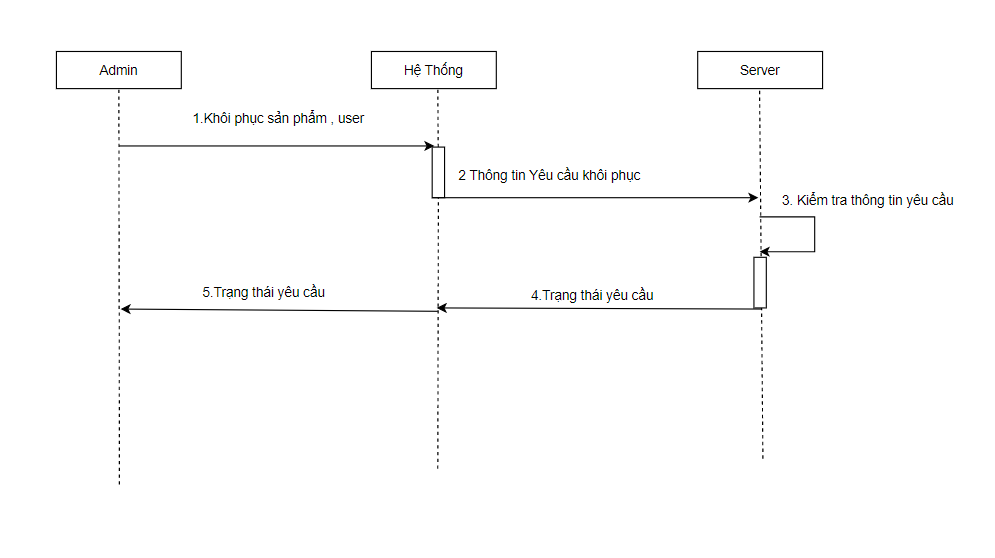
+Thêm mới sản phẩm :



+ Sửa sản phẩm :



+Khôi phục :



**2.7.Phân tích thiết kế dữ liệu**

**2.7.1.Xác định các thực thể và các thuộc tính cho thực thể, loại bỏ các thuộc tính dẫn xuất**

Product(name,view,description, rating ,category ,subject ,images ,price ,originPrice ,sold , createdAt ,updatedAt )

User(userName, email ,admin , password , avatar ,createdAt , updatedAt )

Otp(otp ,user , createdAt ,updatedAt)

Contact(fullName , phoneNumber , message , user, email , createdAt , updatedAt )

Cart(user , product , size , rating , quantity ,createdAt , updatedAt)

Comment(comment , rating , product, user , author, createdAt , updatedat )

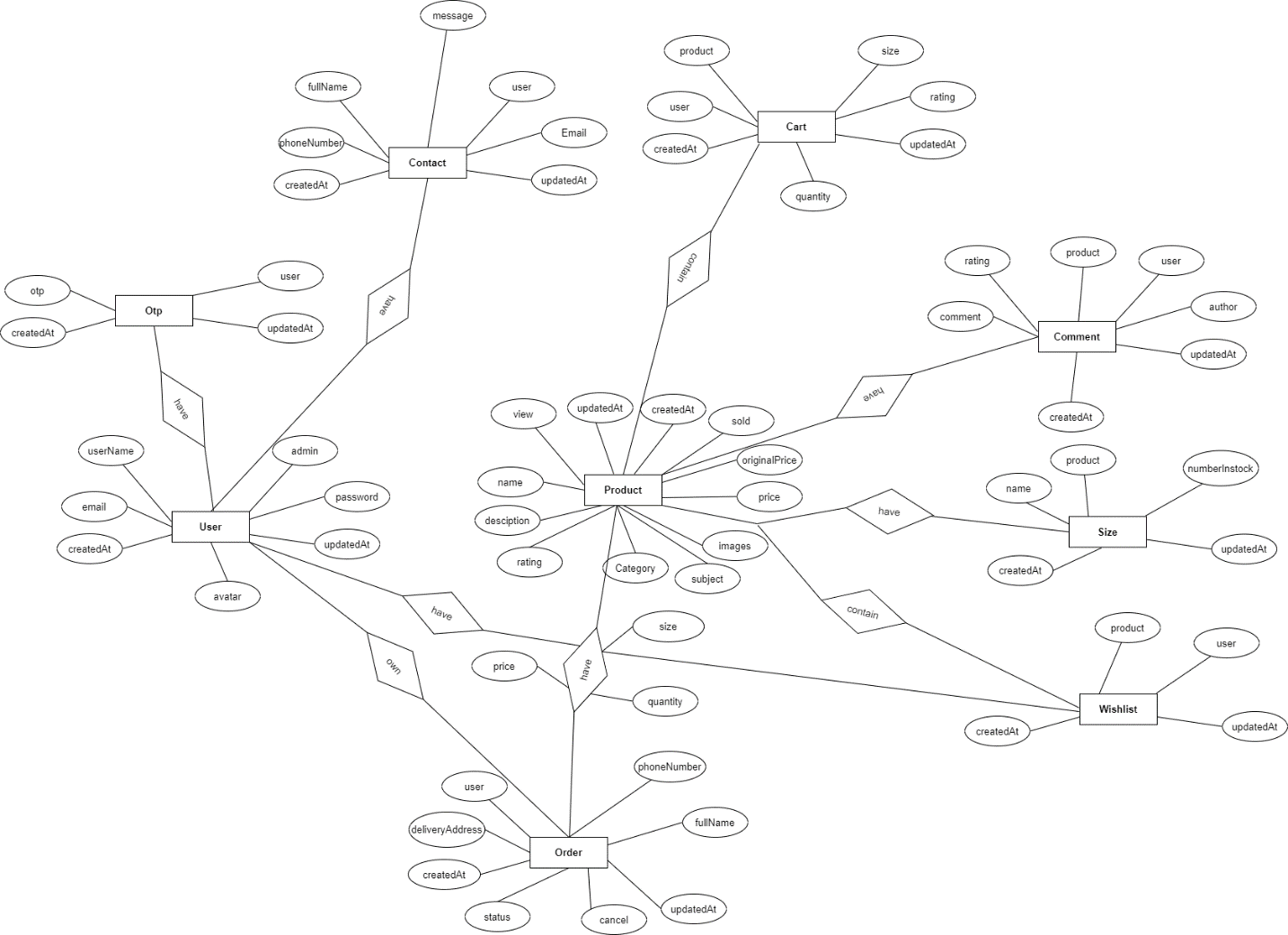
Size(product , name , numberInStock , createdAt , updatedAt )

Wishlist(product , user, createdAt, updatedAt)

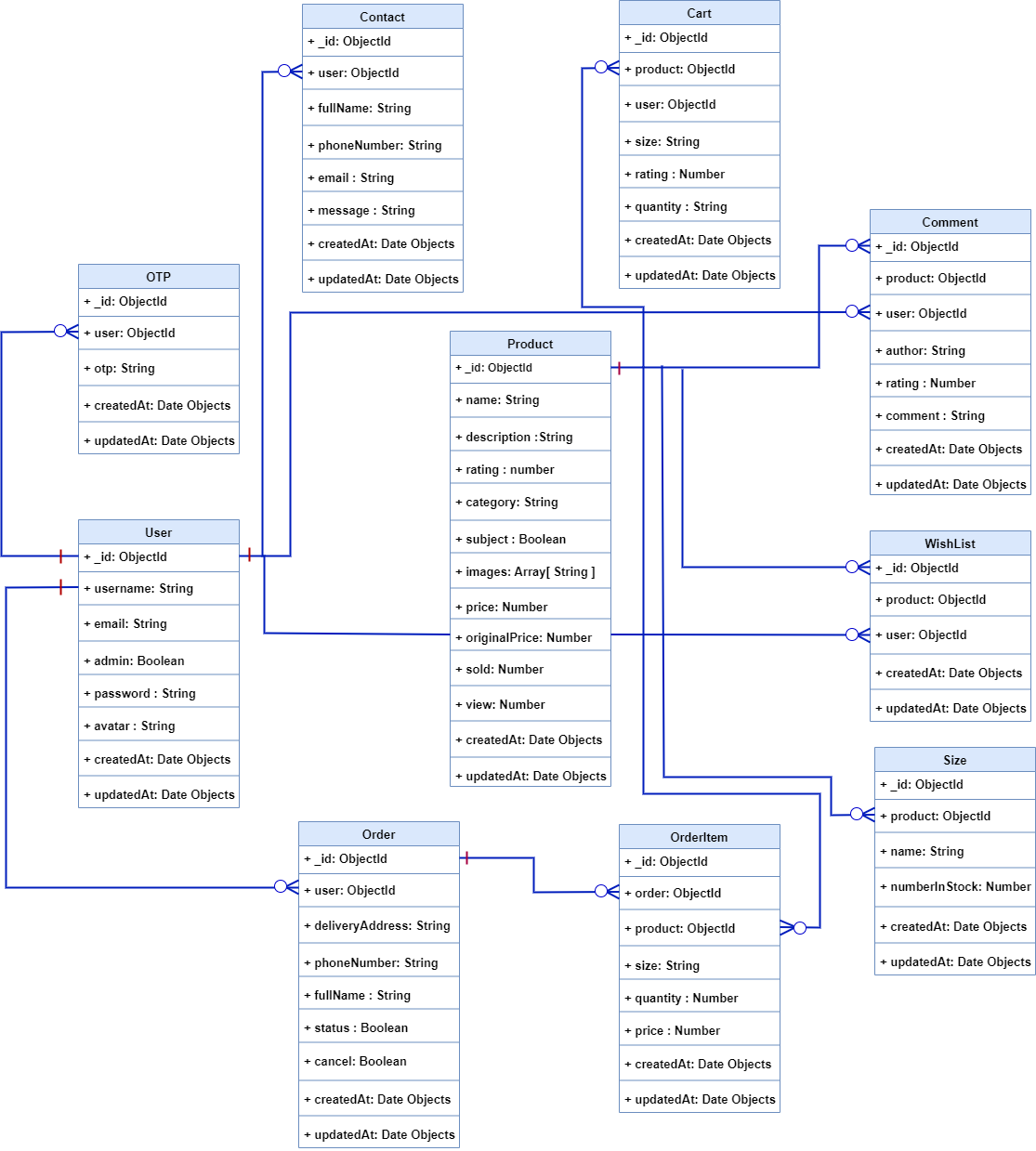
Order(user, deliveryAddress, phoneNumber ,fullName,status , cancel ,createdAt , updatedAt)

OrderItem(order , product ,size ,quantity ,price , createdAt, updatedAt)

**2.7.2.Sơ đồ ER:**

****

**2.7.3.Sơ Đồ RM:**

****

**2.8. Công nghệ sử dụng :**

**2.8.1.Backend**

a.Node.js ( ExpressJS) :

-NodeJS là một môi trường lập trình phổ biến, được dùng để xây dựng các ứng dụng quy mô lớn cần xử lý lượng request đồng thời lớn. Thuộc tính I/O non-blocking đơn luồng giúp nền tảng này trở thành một giải pháp lý tưởng cho việc xây dựng các ứng dụng truyền dữ liệu trong thời gian thực.

-Nodejs sử dụng engine V8 của trình duyệt Google Chrome chính vì vậy bạn biết đấy nó rất mạnh có hiệu suất tốt và cung cấp nhiều API nhất là về lập trình mạng.

- **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.

b.Cơ sở dữ liệu NoSQL(mongoDB)

- Cơ sở dữ liệu NoSQL là Cơ sở dữ liệu được xây dựng dành riêng cho mô hình dữ liệu và có sơ đồ linh hoạt để xây dựng các ứng dụng hiện đại. Cơ sở dữ liệu NoSQL được công nhận rộng rãi vì khả năng dễ phát triển, chức năng cũng như hiệu năng ở quy mô lớn.  
-Vì sao nên sử dụng:

* Linh hoạt: Cơ sở dữ liệu NoSQL thường cung cấp các sơ đồ linh hoạt giúp công đoạn phát triển nhanh hơn và có khả năng lặp lại cao hơn. Mô hình dữ liệu linh hoạt biến cơ sở dữ liệu NoSQL thành lựa chọn lý tưởng cho dữ liệu không được tổ chức thành cấu trúc hoặc có cấu trúc chưa hoàn chỉnh.
* Khả năng thay đổi quy mô: Cơ sở dữ liệu NoSQL thường được thiết kế để tăng quy mô bằng cách sử dụng các cụm phần cứng được phân phối thay vì tăng quy mô bằng cách bổ sung máy chủ mạnh và tốn kém. Một số nhà cung cấp dịch vụ đám mây xử lý các hoạt động này một cách không công khai dưới dạng dịch vụ được quản lý đầy đủ.
* Hiệu năng cao: Cơ sở dữ liệu NoSQL được tối ưu hóa theo các mô hình dữ liệu cụ thể và các mẫu truy cập giúp tăng hiệu năng cao hơn so với việc cố gắng đạt được mức độ chức năng tương tự bằng cơ sở dữ liệu quan hệ.
* Cực kỳ thiết thực: Cơ sở dữ liệu NoSQL cung cấp các API và kiểu dữ liệu cực kỳ thiết thực được xây dựng riêng cho từng mô hình dữ liệu tương ứng.

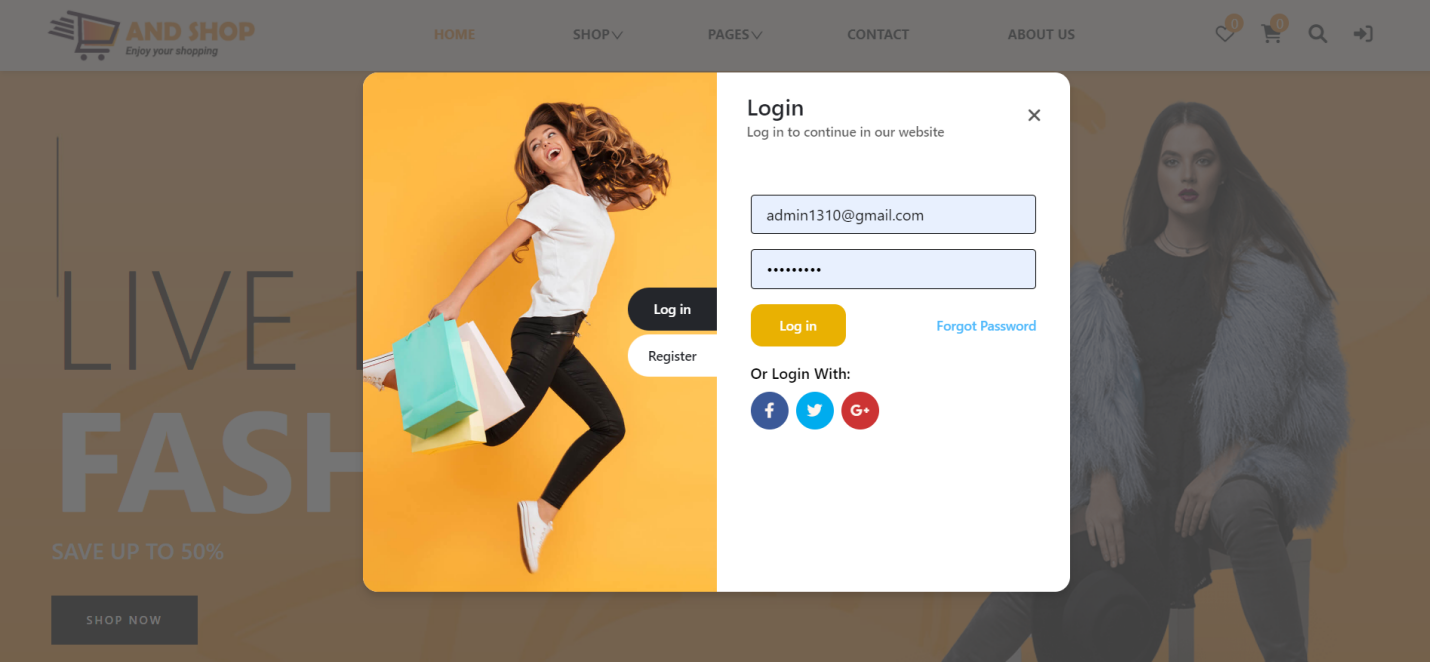
**2.8.2 Front end (VueJs)**

Vue.js là một framework linh động (nguyên bản tiếng Anh: progressive – tiệm tiến) dùng để xây dựng giao diện người dùng (user interfaces). Khác với các framework nguyên khối (monolithic), Vue được thiết kế từ đầu theo hướng cho phép và khuyến khích việc phát triển ứng dụng theo từng bước. Khi phát triển lớp giao diện (view layer), người dùng chỉ cần dùng thư viện lõi (core library) của Vue, vốn rất dễ học và tích hợp với các thư viện hoặc dự án có sẵn. Cùng lúc đó, nếu kết hợp với những kĩ thuật hiện đại như SFC (single file components) và các thư viện hỗ trợ, Vue cũng đáp ứng được dễ dàng nhu cầu xây dựng những ứng dụng một trang (SPA - Single-Page Applications) với độ phức tạp cao hơn nhiều.

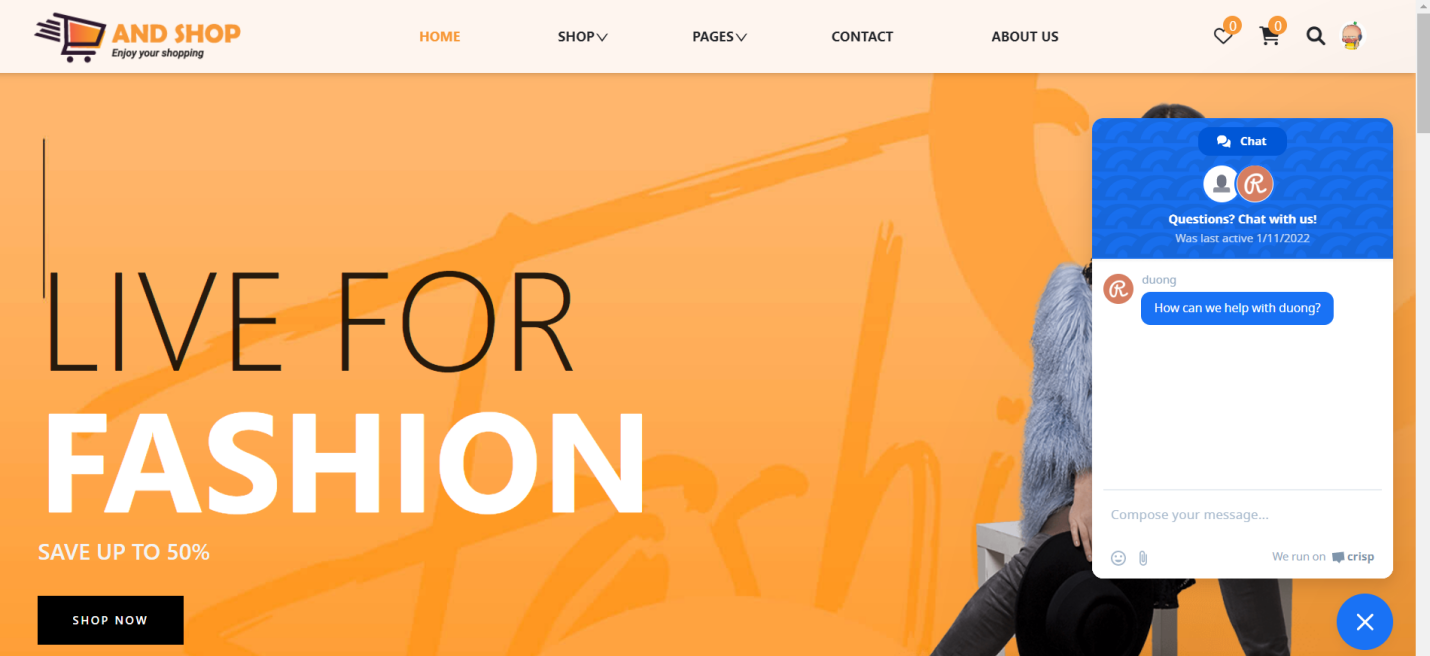
**CHƯƠNG 3: XÂY DỰNG VÀ CÀI ĐẶT CHƯƠNG TRÌNH**

**3.1 Xây dựng giao diện**

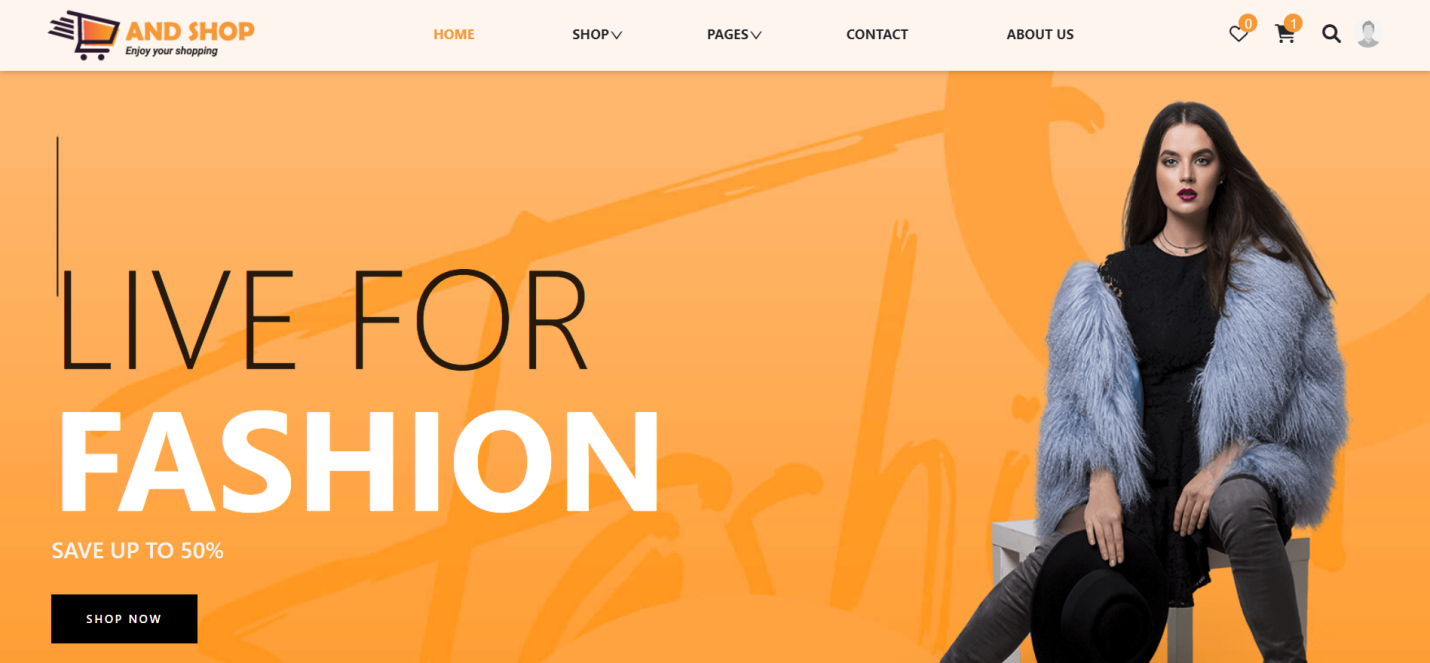
**Giao diện đăng nhập**

****

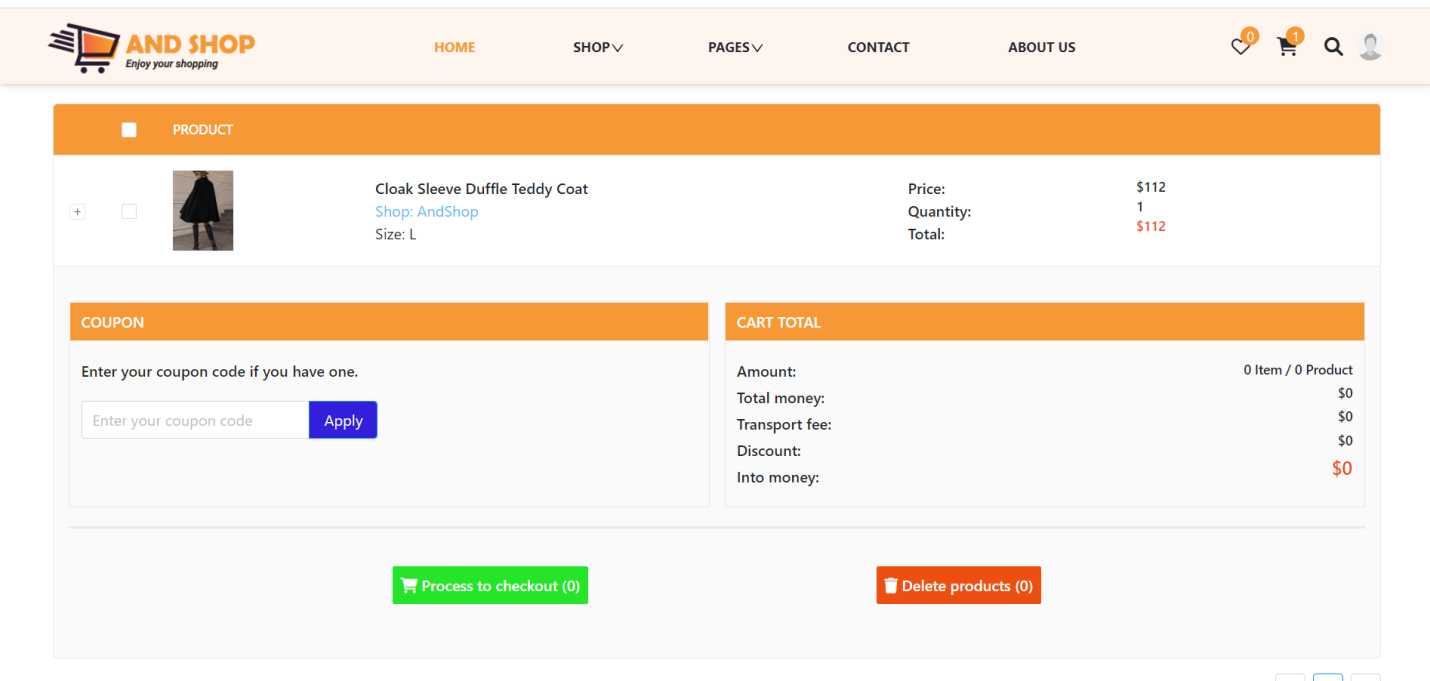
**Giao diện chat**

****

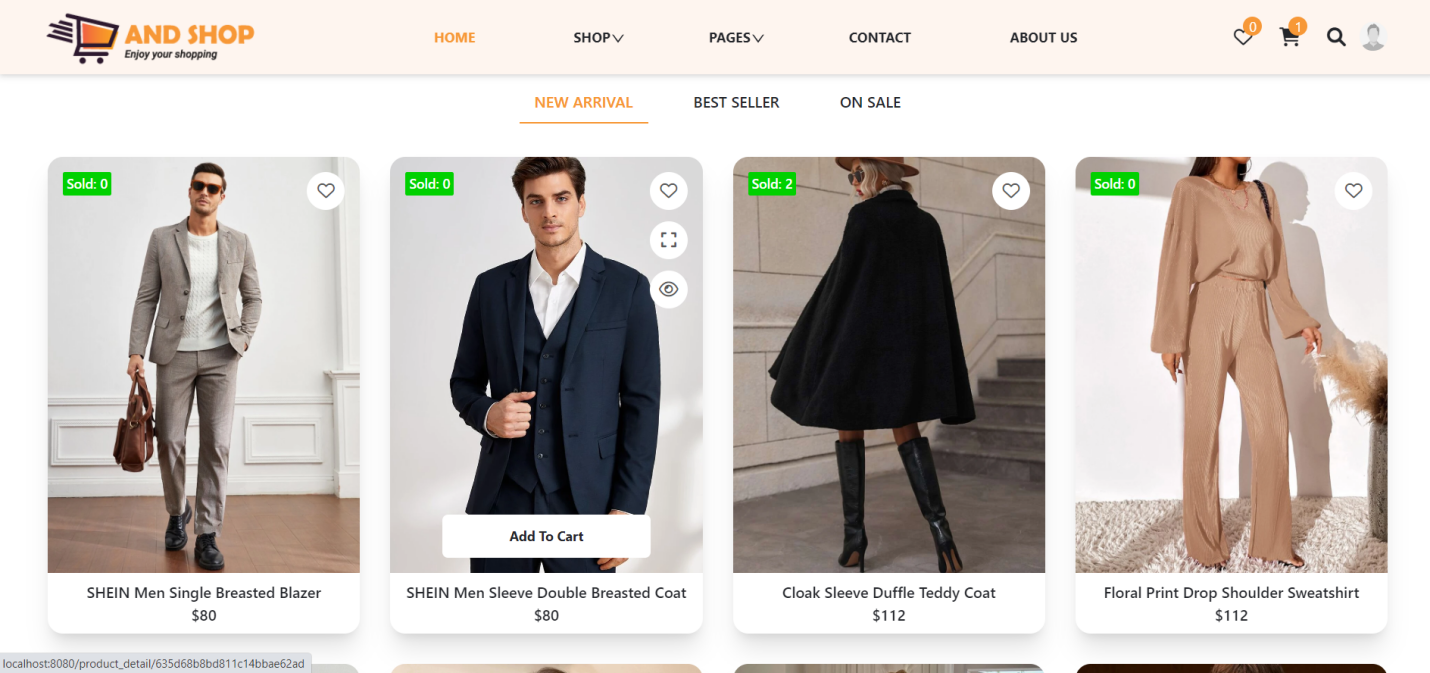
**Giao diện trang chủ**

****

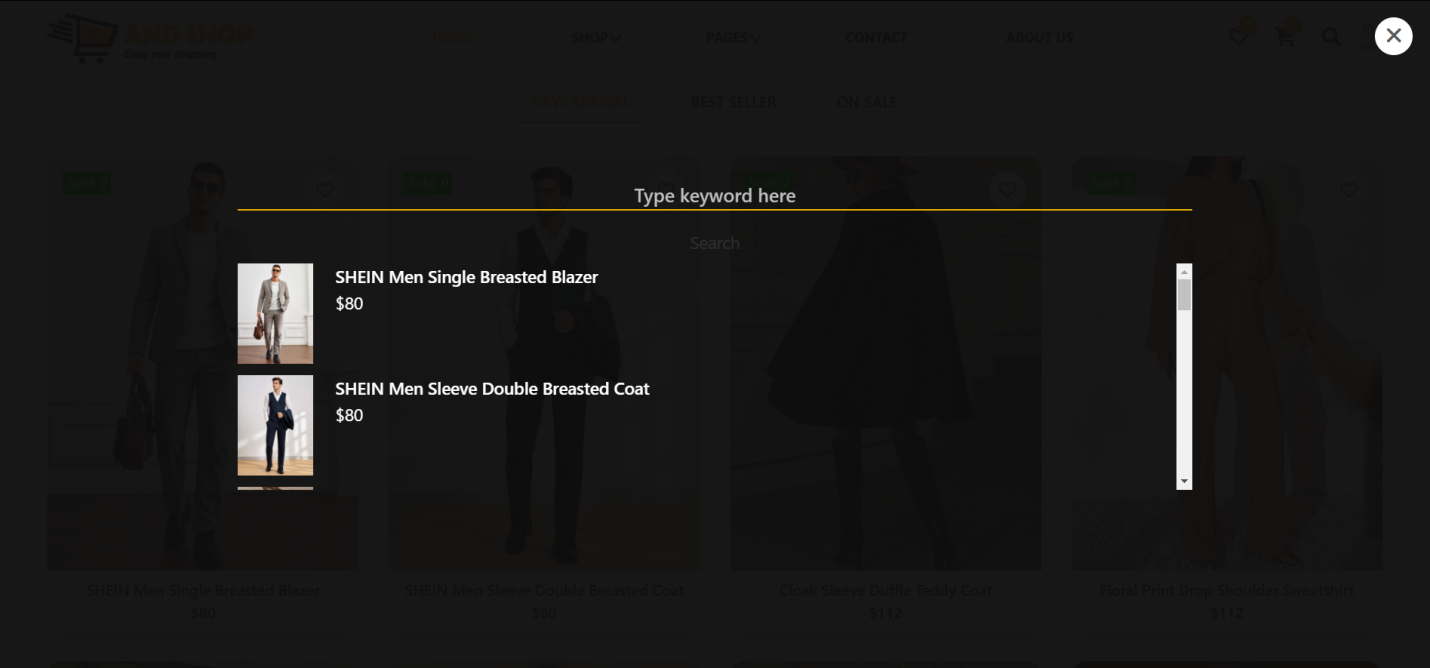
**Giao diện giỏ hang**

****

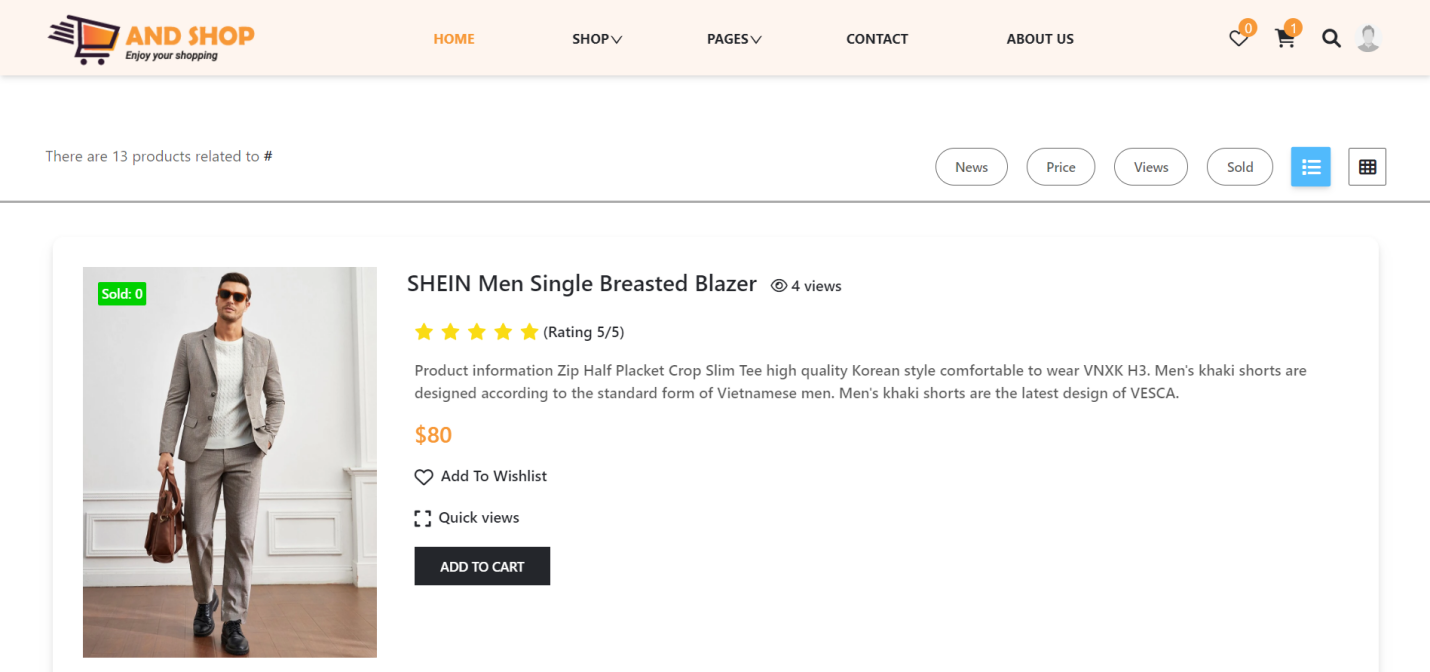
**Giao diện trang chủ sản phẩm**

****

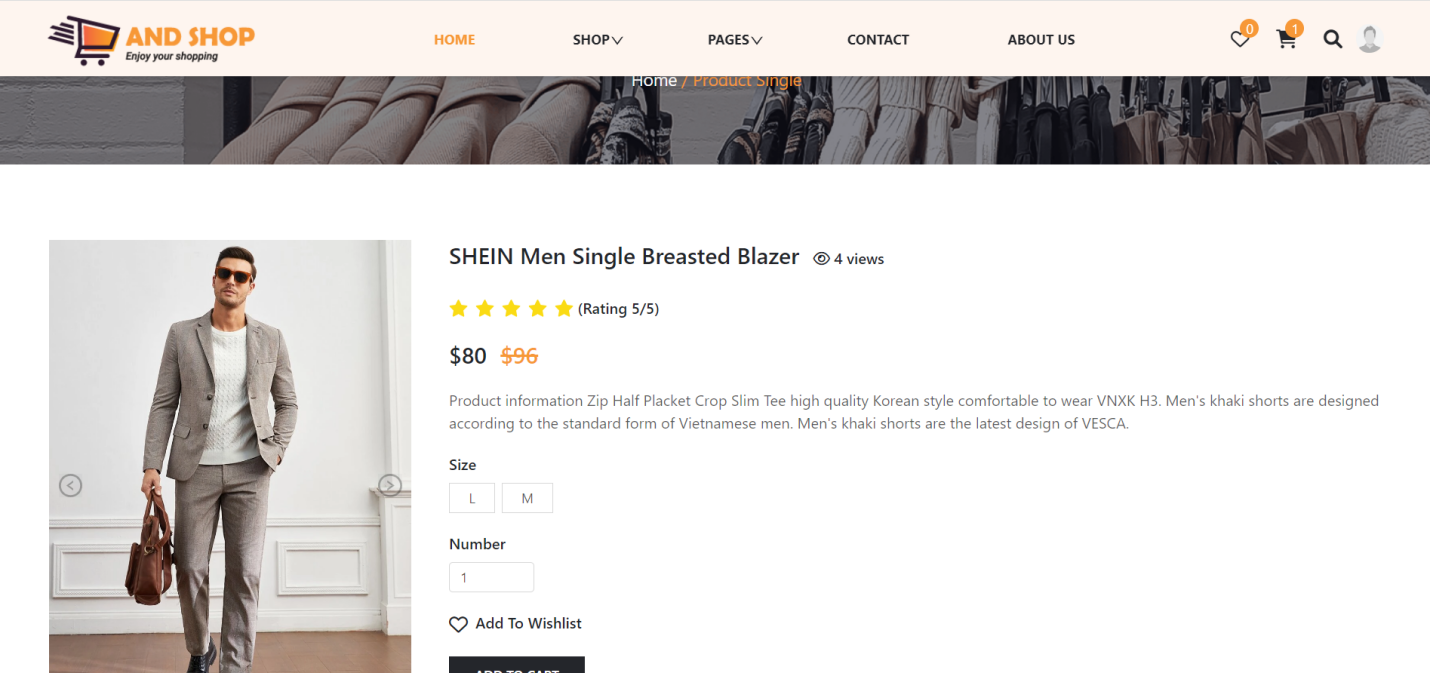
**Giao diện tìm kiếm**

****

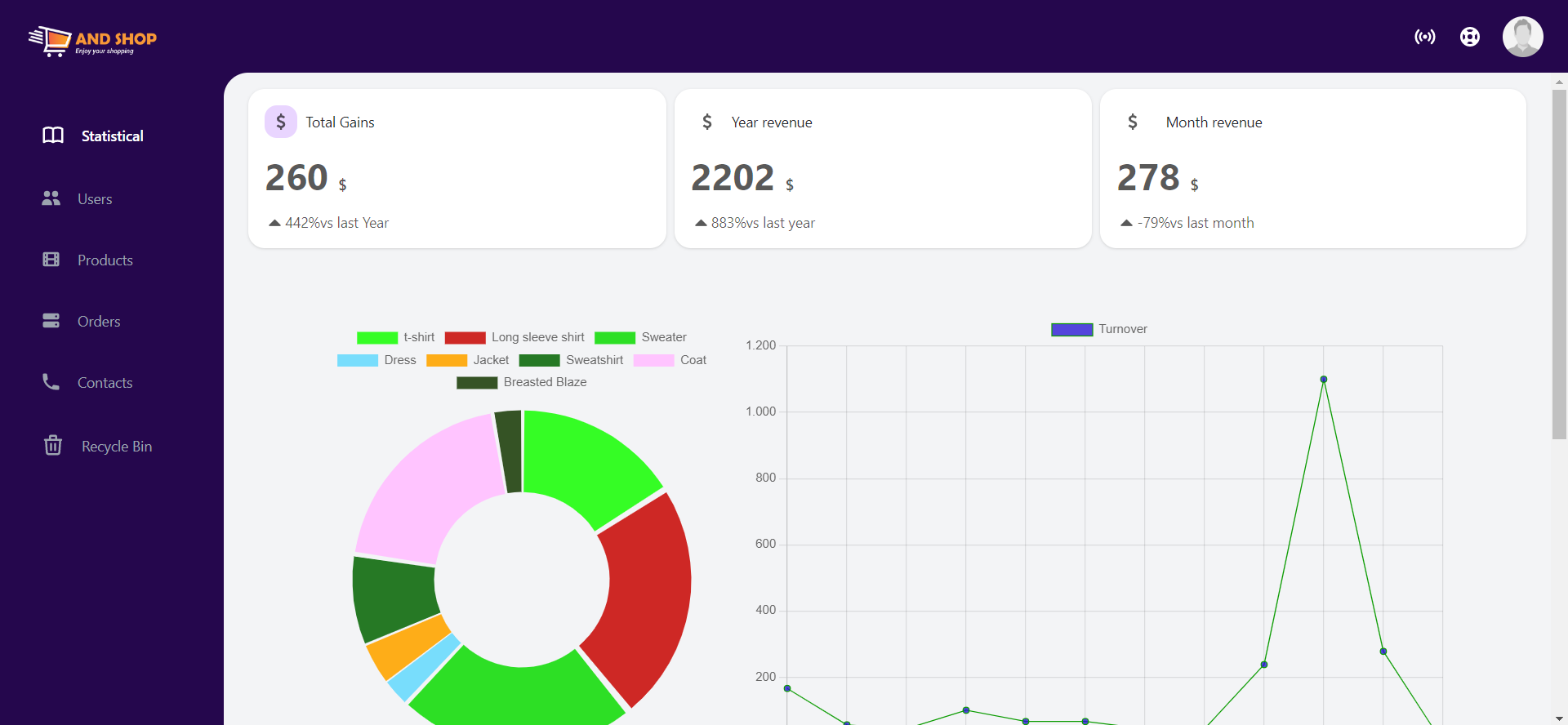
**Giao diện sản phẩm**

****

**Giao diện chi tiết sản phẩm**

****

**Giao diện trang Admin**

****

**CHƯƠNG 4: KẾT LUẬN**

Khi xây dựng một trang web bán quần áo nam và nữ, việc lựa chọn Node.js cho phía backend có thể mang lại nhiều lợi ích:

Hiệu suất và đáp ứng: Node.js được thiết kế để xử lý hàng ngàn kết nối đồng thời mà không gây block, giúp trang web của bạn đáp ứng nhanh chóng đối với các yêu cầu từ người dùng.

Tính linh hoạt và mở rộng: Với Node.js, bạn có thể dễ dàng mở rộng ứng dụng của mình theo nhu cầu, từ việc quản lý số lượng người dùng đồng thời đến việc mở rộng chức năng của trang web

JavaScript đồng nhất: Sử dụng JavaScript trên cả phía frontend và backend giúp giảm thiểu sự phức tạp trong việc phát triển và duy trì mã nguồn

Thư viện và framework phong phú: Cộng đồng Node.js phát triển mạnh mẽ với nhiều thư viện và framework hữu ích như Express.js, giúp bạn xây dựng và quản lý ứng dụng một cách hiệu quả.

Hỗ trợ realtime (thời gian thực): Nếu bạn muốn cung cấp các tính năng realtime như cập nhật giỏ hàng, thông báo mới, thì Node.js có các thư viện như Socket.io giúp bạn dễ dàng thực hiện điều này.

Tóm lại, việc sử dụng Node.js cho việc phát triển một trang web bán quần áo nam và nữ có thể mang lại sự linh hoạt, hiệu suất và tính mở rộng cho ứng dụng của bạn.

**Tài liệu tham khảo**

1. Node.js Foundation. (n.d.). Node.js Official Website. Truy cập từ <https://nodejs.org/>
2. Mozilla Developer Network. (n.d.). MDN Web Docs - Node.js. Truy cập từ <https://developer.mozilla.org/en-US/docs/Glossary/Node.js>
3. Codecademy. (n.d.). Node.js Basics. Truy cập từ https://www.codecademy.com/learn/learn-node-js
4. Udemy. (n.d.). The Complete Node.js Developer Course. Truy cập từ https://www.udemy.com/course/the-complete-nodejs-developer-course-2/
5. Casciaro, M., & Mammino, L. (2016). Node.js Design Patterns. Packt Publishing.
6. Wilson, J. (2018). Node.js 8 the Right Way. Pragmatic Bookshelf.
7. Stack Overflow. (n.d.). Node.js StackOverflow. Truy cập từ <https://stackoverflow.com/questions/tagged/node.js>
8. Reddit. (n.d.). Node.js Reddit. Truy cập từ <https://www.reddit.com/r/node/>

·