|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **KHOA CÔNG NGHỆ THÔNG TIN**  **---------------------------------------**  **E:\dh-cn-hanoi.png**  **ĐỒ ÁN TỐT NGHIỆP**  **XÂY DỰNG WEBSITE BÁN ĐỒ ĂN SUPERFOOD**  **SỬ DỤNG JAVA SPRING BOOT VÀ REACTJS**  NGÀNH: CÔNG NGHỆ THÔNG TIN   |  |  | | --- | --- | | **GVHD:** | **ThS. Nguyễn Thị Cẩm Ngoan** | | **Sinh viên:** | **Nguyễn Đức Quân** | | **Lớp: CNTT5** | **Khóa: K15** |   ***Hà Nội - Năm 2024*** |

# 

# LỜI CẢM ƠN

Thưa thầy cô, khóa luận tốt nghiệp chuyên ngành Công nghệ thông tin với đề tài xây dựng website bán đồ ăn SuperFood sử dụng java Spring Boot và ReactJs mà em vừa trình bày chính là kết quả của cả một quá trình trau dồi và nỗ lực không ngừng của bản thân em. Em rất may mắn khi luôn nhận được sự giúp đỡ, hỗ trợ tận tình từ quý thầy cô, gia đình và bạn bè của mình.

Qua đây, em xin được gửi lời cảm ơn chân thành nhất đến tất cả mọi người đã luôn giúp đỡ, động viên và khích lệ em trong suốt khoảng thời gian vừa qua. Em xin trân trọng cảm ơn cô Nguyễn Thị Cẩm Ngoan . Cô là người đã luôn tận tình chỉ dạy, dẫn dắt và tạo điều kiện hết sức để giúp em có thể hoàn thành tốt nhất bài luận văn của mình.

Bên cạnh đó, em cũng xin được cảm ơn ban lãnh đạo cùng toàn thể các thầy cô trường Đại học Công Nghiệp Hà Nội, khoa Công nghệ thông tin đã luôn giúp đỡ và ở bên cạnh chúng em trong suốt 4 năm đại học vừa qua.

Xin kính chúc thầy cô thật nhiều sức khỏe và luôn thành công trên con đường giảng dạy của mình.

Em xin chân thành cảm ơn mọi người!

# MỤC LỤC

[LỜI CẢM ƠN i](#_Toc167662619)

[MỤC LỤC ii](#_Toc167662620)

[DANH MỤC TỪ VIẾT TẮT iv](#_Toc167662621)

[DANH MỤC HÌNH ẢNH vi](#_Toc167662622)

[DANH MỤC BẢNG BIỂU viii](#_Toc167662623)

[LỜI MỞ ĐẦU 1](#_Toc167662624)

[CHƯƠNG 1: CƠ SỞ LÍ THUYẾT 2](#_Toc167662625)

[1.1. Tổng quan về website bán đồ ăn 2](#_Toc167662626)

[1.2. Công nghệ sử dụng trong xây dựng website 2](#_Toc167662627)

[1.2.1 ReactJs 2](#_Toc167662628)

[1.2.1.1 Lịch sử của ReactJs 2](#_Toc167662629)

[1.2.1.2 Cấu trúc của ReacJs 4](#_Toc167662630)

[1.2.1.3 Ứng dụng của ReactJs 5](#_Toc167662631)

[1.2.2 Java Spring Boot 7](#_Toc167662632)

[1.2.1.1 Giới thiệu về Spring Boot 7](#_Toc167662633)

[1.2.1.2 Ưu điểm của Spring Boot 7](#_Toc167662634)

[1.2.1.3 Các đặc tính cơ bản của Spring Boot 8](#_Toc167662635)

[1.2.3 MySQL 9](#_Toc167662636)

[1.2.3.1 MySQL là gì ? 9](#_Toc167662637)

[1.2.3.2 Ưu điểm của MySQL 11](#_Toc167662638)

[1.2.3.3 MySQL hoạt động như thế nào ? 12](#_Toc167662639)

[1.3. Kiến trúc phần mềm 13](#_Toc167662640)

[CHƯƠNG 2 : PHÂN TÍCH THIẾT KẾ HỆ THỐNG 15](#_Toc167662641)

[2.1. Mô tả nghiệp vụ 15](#_Toc167662642)

[2.2. Đặc tả yêu cầu 15](#_Toc167662643)

[2.2.1. Yêu cầu chức năng 15](#_Toc167662644)

[2.2.2 Yêu cầu phi chức năng 16](#_Toc167662645)

[2.3 Mô tả use case 16](#_Toc167662646)

[2.3.1 Đăng kí tài khoản 17](#_Toc167662647)

[2.3.2 Đăng nhập 17](#_Toc167662648)

[2.3.3. Đặt hàng 18](#_Toc167662649)

[2.3.4. Hủy hàng 19](#_Toc167662650)

[2.3.5. Quản lí người dùng 19](#_Toc167662651)

[2.3.6. Cập nhật thông tin tài khoản 21](#_Toc167662652)

[2.3.7. Xem hàng theo danh mục 22](#_Toc167662653)

[2.3.8. Xem chi tiết sản phẩm 22](#_Toc167662654)

[2.3.9. Tìm kiếm sản phẩm 23](#_Toc167662655)

[2.3.10. Quản lí giỏ hàng 24](#_Toc167662656)

[2.3.11. Xem thống kê 24](#_Toc167662657)

[2.3.12. Quản lí danh mục 25](#_Toc167662658)

[2.3.13. Quản lí sản phẩm 26](#_Toc167662659)

[2.3.14. Quản lí tài khoản 27](#_Toc167662660)

[2.3.15. Quản lí đơn hàng 28](#_Toc167662661)

[2.3.16. Quản lí bài viết 29](#_Toc167662662)

[2.3.17. Quản lí kho mã giảm giá 30](#_Toc167662663)

[2.4. Thiết kế cơ sở dữ liệu 31](#_Toc167662664)

[2.4.1. Mô hình hóa dữ liệu 31](#_Toc167662665)

[2.4.2. Thiết kế bảng 32](#_Toc167662666)

[CHƯƠNG 3 : THIẾT KẾ GIAO DIỆN 37](#_Toc167662667)

[3.1. Giao diện người dùng 37](#_Toc167662668)

[3.2 Giao diện Admin 44](#_Toc167662669)

[CHƯƠNG 4 : KIỂM THỬ PHẦN MỀM 49](#_Toc167662670)

[4.1. Kế hoạch kiểm thử 49](#_Toc167662671)

[4.2. Kiểm thử chức năng phía khách hàng 50](#_Toc167662672)

[4.3. Kiểm thử chức năng phía Admin 52](#_Toc167662673)

[4.4. Kết quả kiểm thử 55](#_Toc167662674)

[CHƯƠNG 5 : KẾT LUẬN 56](#_Toc167662675)

[TÀI LIỆU THAM KHẢO 58](#_Toc167662676)

# DANH MỤC TỪ VIẾT TẮT

|  |  |
| --- | --- |
| Từ viết tắt | Giải thích |
| JRE | Java Runtime Environment |
| API | Application Programming Interface |
| OOP | Object Oriented Programming |
| ANSI | American National Standards Institute |
| POSIX | Portable Operating System Interface |
| HTML | HyperText Markup Language |
| CSS | Cascading Style Sheets |
| JSF | JavaServer Faces |
| IBM | International Business Machines |
| AOP | Aspect Oriented Programming |
| JDBC | Java Database Connectivity |
| POJO | Plain Old Java Object |
| JMS | Java Message Service |
| JTA | Java Transaction API |
| JMX | Java Management Extensions |
| RMI | Remote Method Invocation |
| MVC | Model-View-Controller |
| UI | User Interface |
| GUI | Graphical User Interface |
| AMQP | Advanced Message Queuing Protocol |
| XML | Extensible Markup Language |
| MVCC | Multiversion concurrency control |
| PK | Primary key |
| PFK | Primary Foreign Key |

# DANH MỤC HÌNH ẢNH

[Hình 1.1 : Hình ảnh về ReactJs 3](#_Toc167661608)

[Hình 1.2 : Cấu trúc của ReactJs 4](#_Toc167661609)

[Hình 1.3 : Trang web sử dụng ReactJs 6](#_Toc167661610)

[Hình 1.4 : Tranng Web sử dụng ReatJs 6](#_Toc167661611)

[Hình 1.5 : Hình ảnh về Spring Boot 7](#_Toc167661612)

[Hình 1.6 : Phương thức main trong Spring Boot 8](#_Toc167661613)

[Hình 1.7 : Hình ảnh về MySQL 9](#_Toc167661614)

[Hình 1.8 : Công cụ phpMyAdmin 10](#_Toc167661615)

[Hình 1.9 : Công cụ MySQL Workbench 11](#_Toc167661616)

[Hình 1.10 : Công cụ Sequel pro 11](#_Toc167661617)

[Hình 1.11 : Phương thức hoạt động của MySQL 12](#_Toc167661618)

[Hình 1.12: Phương thức hoạt động của REST API 13](#_Toc167661619)

[Hình 1.13 : Các phương thức trong REST API 14](#_Toc167661620)

[Hình 2.1 : Sơ đồ usecase tổng quát 17](#_Toc167661621)

[Hình 2.2 : Cơ sở dữ liệu 32](#_Toc167661622)

[Hình 3.1 : Giao diện đăng kí 37](#_Toc167661624)

[Hình 3.2 : Giao diện đăng nhập 37](#_Toc167661625)

[Hình 3.3 : Giao diện trang chủ 38](#_Toc167661626)

[Hình 3.4 : Danh sách món ăn 38](#_Toc167661627)

[Hình 3.5 : Chi tiết món ăn 39](#_Toc167661628)

[Hình 3.6 : Danh sách các bài viết 39](#_Toc167661629)

[Hình 3.7 : Chi tiết bài viết 40](#_Toc167661630)

[Hình 3.8 : Danh mục các món liên quan 40](#_Toc167661631)

[Hình 3.9 : Giao diện giỏ hàng 41](#_Toc167661632)

[Hình 3.10 : Giao diện quản lí tài khoản 41](#_Toc167661633)

[Hình 3.11 : Giao diện thanh toán 42](#_Toc167661634)

[Hình 3.12 : Giao diện thanh toán thành công 42](#_Toc167661635)

[Hình 3.13 : Xem chi tiết tài khoản 43](#_Toc167661636)

[Hình 3.14 : tra cứu thông tin đơn hàng 43](#_Toc167661637)

[Hình 3.15 : Giao diện Dashboard 44](#_Toc167661638)

[Hình 3.16 : Quản lí tài khoản khách hàng 44](#_Toc167661639)

[Hình 3.17 : Quản lí danh mục 45](#_Toc167661640)

[Hình 3.18 : Quản lí bài viết trên trang chủ 45](#_Toc167661641)

[Hình 3.19 : Quản lí sản phẩm 46](#_Toc167661642)

[Hình 3.20 : Quản lí đơn hàng 46](#_Toc167661643)

[Hình 3.21: Quản lí nhập hàng về kho 47](#_Toc167661644)

[Hình 3.22 : Quản lí kho voucher 47](#_Toc167661645)

[Hình 3.23 : Quản lí doanh thu theo tháng 48](#_Toc167661646)

[Hình 3.24: Quản lí doanh thu theo ngày 48](#_Toc167661647)

# DANH MỤC BẢNG BIỂU

[Bảng 2.1 : Bảng authority 32](#_Toc167662581)

[Bảng 2.2 : Bảng blog 32](#_Toc167662582)

[Bảng 2.3 : Bảng Cart 33](#_Toc167662583)

[Bảng 2.4 : Bảng Category 33](#_Toc167662584)

[Bảng 2.5: Bảng History pay 33](#_Toc167662585)

[Bảng 2.6: Bảng Import Product 34](#_Toc167662586)

[Bảng 2.7: Bảng invoice 34](#_Toc167662587)

[Bảng 2.8: Bảng invoice\_detail 35](#_Toc167662588)

[Bảng 2.9: Bảng product 35](#_Toc167662589)

[Bảng2.10:BảngProduct\_comment 35](#_Toc167662590)

[Bảng 2.11: Bảng product comment image 36](#_Toc167662591)

[Bảng 2.12 : Bảng Product\_image 36](#_Toc167662592)

[Bảng 2.13 : Bảng users 36](#_Toc167662593)

[Bảng 2.14 : Bảng voucher 36](#_Toc167662594)

# 

# LỜI MỞ ĐẦU

Công nghệ thông tin trong giai đoạn hiện nay đang có những bước phát triển như vũ bão trên mọi lĩnh vực hoạt động ở khắp nơi trên toàn thế giới. Không chỉ trên thế giới mà ở Việt Nam, trong vài năm gần đây, có rất nhiều chuyên gia đang đầu tư vào lĩnh vực công nghệ để có thể hòa nhập với thế giới. Việt Nam cho thấy là một quốc gia có tiềm năng phát triển lĩnh vực công nghệ thông tin lớn, trong đó phát triển các giải pháp về công nghệ thông tin đang là một xu hướng. Việc phát triển các giải pháp về công nghệ giúp cho công việc quản lý trở nên dễ dàng hơn, nâng cao hiệu suất sản xuất cũng như tiết kiệm được thời gian và công sức.

Qua những kiến thức tiếp thu được tại nhà trường và thời gian thực tập tại doanh nghiệp, em thấy việc áp dụng công nghệ thông tin vào các lĩnh vực trong kinh doanh, sản xuất là một điều vô cùng quan trọng và cần thiết không chỉ ở Việt Nam mà còn ở trên thế giới. Ở bài báo cáo này, em xin phép được trình bày những kiến thức đã học hỏi được trong quá trình thực tập thực tế, giống như một lộ trình để trở thành một lập trình viên thiết kế Web.

Em rất mong nhận được những lời nhận xét, đánh giá của thầy cô để bài báo cáo trở nên hoàn thiện và kỹ năng của bản thân được phát triển xa hơn nữa trong tương lai.

Em xin chân thành cảm ơn!!!

# CHƯƠNG 1: CƠ SỞ LÍ THUYẾT

## Tổng quan về website bán đồ ăn

Hiện nay, các website bán đồ ăn đang trở thành một phần không thể thiếu trong cuộc sống hàng ngày của mọi người. Với sự phát triển của công nghệ và mạng internet, việc mua sắm và đặt hàng thực phẩm trực tuyến đã trở nên phổ biến hơn bao giờ hết. Những trang web này cung cấp một nền tảng thuận tiện và linh hoạt cho người tiêu dùng để tìm kiếm, chọn lựa và đặt hàng đồ ăn từ các nhà hàng, cửa hàng địa phương hoặc dịch vụ giao hàng.

Một trang web bán đồ ăn là một nền tảng trực tuyến đa chức năng và tiện ích, tạo ra một môi trường mua bán và giao tiếp hiệu quả cho người mua và người bán. Với sự phát triển của công nghệ, website bán đồ ăn đã trở thành một công cụ mạnh mẽ để đáp ứng nhu cầu mua bán đồ ăn đa dạng của mọi người, cung cấp một loạt các sản phẩm và dịch vụ liên quan đến đồ ăn.

Trang web bán đồ ăn SuperFood cung cấp một giao diện trực quan và dễ sử dụng, mang lại trải nghiệm mua sắm thuận tiện cho người dùng. Với việc phân loại sản phẩm theo các danh mục đa dạng như đồ ăn nhanh, món đặc sản địa phương, đồ ăn chay, đồ uống và nhiều loại khác, người mua có thể dễ dàng tìm kiếm và chọn lựa món ăn phù hợp với khẩu vị và nhu cầu của mình. Thông qua danh sách đồ ăn, người mua có thể xem thông tin chi tiết về từng món, bao gồm mô tả, nguyên liệu, hình ảnh và giá cả. Việc cung cấp thông tin đầy đủ và minh bạch giúp người mua có cái nhìn toàn diện về sản phẩm trước khi quyết định đặt hàng. Đồng thời, tính năng đánh giá và nhận xét từ người dùng trước cũng giúp tăng cường tính minh bạch và tin cậy, giúp người mua đưa ra quyết định mua hàng một cách tự tin và thông minh.

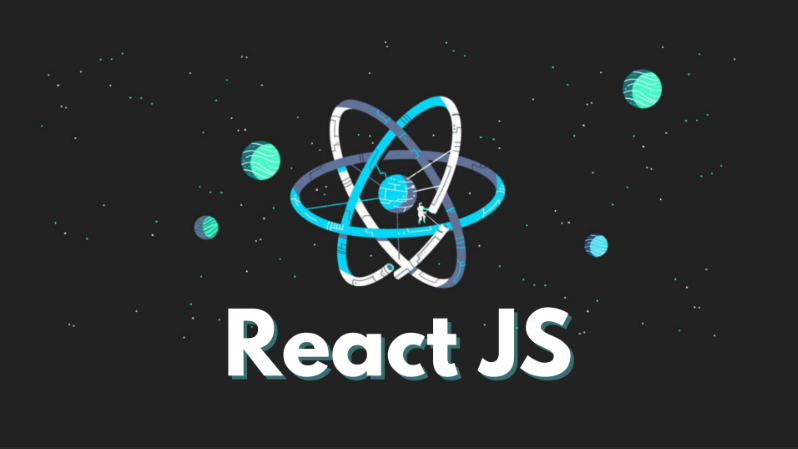
## Công nghệ sử dụng trong xây dựng website

### 1.2.1 ReactJs

#### 1.2.1.1 Lịch sử của ReactJs

ReactJS là một thư viện trong đó có chứa nhiều JavaScript mã nguồn mở và cha đẻ của ReactJS đó chính là một ông lớn với cái tên ai cũng biết đó chính là Facebook. Mục đích của việc tạo ra ReactJS là để tạo ra những ứng dụng website hấp dẫn với tốc độ nhanh và hiệu quả cao với những mã hóa tối thiểu. Và mục đích chủ chốt của ReactJS đó chính là mỗi website khi đã sử dụng ReactJS thì phải chạy thật mượt thật nhanh và có khả năng mở rộng cao và đơn giản thực hiện.

ReactJS được giới thiệu công khai lần đầu tiên bởi Jordan Walke, một kỹ sư phần mềm tại Facebook, trong một bài đăng trên blog công ty vào tháng 5 năm 2013. Ông giới thiệu React như là một thư viện JavaScript mới để xây dựng các giao diện người dùng linh hoạt và dễ bảo trì.



Hình 1.1 : Hình ảnh về ReactJs

Phiên bản đầu tiên của ReactJS được phát hành dưới dạng mã nguồn mở vào tháng 5 năm 2013. Tuy nhiên, nó không được công bố chính thức và được xem như một dự án thử nghiệm ban đầu …

Sau khi giới thiệu ban đầu, ReactJS đã nhanh chóng thu hút sự quan tâm của cộng đồng phát triển web. Facebook đã tiếp tục phát triển và mở mã nguồn của ReactJS, cho phép các nhà phát triển từ khắp nơi trên thế giới đóng góp vào mã nguồn. Facebook bắt đầu sử dụng ReactJS trong các sản phẩm thực tế của mình, bao gồm trang web mạng xã hội Facebook và Instagram. Sự thành công của React trong việc cải thiện hiệu suất và hiệu quả phát triển giao diện người dùng đã giúp nó trở thành một công cụ phổ biến trong ngành công nghiệp công nghệ.

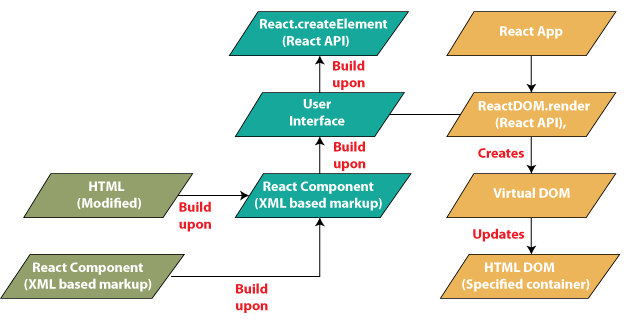
Phiên bản ổn định đầu tiên của ReactJS, phiên bản 0.3.0, đã được phát hành vào tháng 7 năm 2013. Từ đó, React tiếp tục phát triển và cải thiện qua các phiên bản mới, với các tính năng mới và sự cải tiến liên tục.

Ngoài ReactJS, Facebook cũng phát triển và giới thiệu các dự án khác trong hệ sinh thái React như React Native (cho phát triển website di động), Redux (quản lý trạng thái website) và Next.js (một framework React cho phát triển website web).

Hiện nay, ReactJS là một trong những công nghệ phát triển giao diện người dùng phổ biến nhất trên thế giới, được sử dụng bởi hàng ngàn công ty và hàng triệu nhà phát triển trên toàn thế giới để xây dựng các website web hiệu suất cao và dễ bảo trì.

#### 1.2.1.2 Cấu trúc của ReacJs

Mô hình sau sẽ thể hiện một cách tổng quát các thành phần của hệ điều hành Reactjs. Mỗi một phần sẽ được đặc tả một cách chi tiết dưới đây:



Hình 1.2 : Cấu trúc của ReactJs

*React element* : là một đối tượng JavaScript đại diện cho một phần tử giao diện người dùng (UI element). React elements là các đơn vị cơ bản nhất của một ứng dụng React, và chúng được tạo ra để mô tả cách mà một phần tử UI cụ thể sẽ được hiển thị trên giao diện. React.createElement để tạo phần tử React.

*React component* : là một đơn vị độc lập và tái sử dụng của giao diện người dùng. Component trong React được sử dụng để chia nhỏ giao diện thành các phần nhỏ, mỗi phần đó đều có thể được xem và quản lý riêng biệt. Các component có thể chứa mã HTML, CSS và JavaScript để mô tả giao diện người dùng và hành vi của ứng dụng.

Về mặt quan hệ, React components thường sử dụng React elements để mô tả giao diện người dùng mà chúng sẽ hiển thị. Mỗi component thường tạo ra và trả về một hoặc nhiều React elements để xây dựng giao diện người dùng của mình. Điều này tạo ra một cấu trúc tổ chức linh hoạt và dễ bảo trì cho ứng dụng React.

#### 1.2.1.3 Ứng dụng của ReactJs

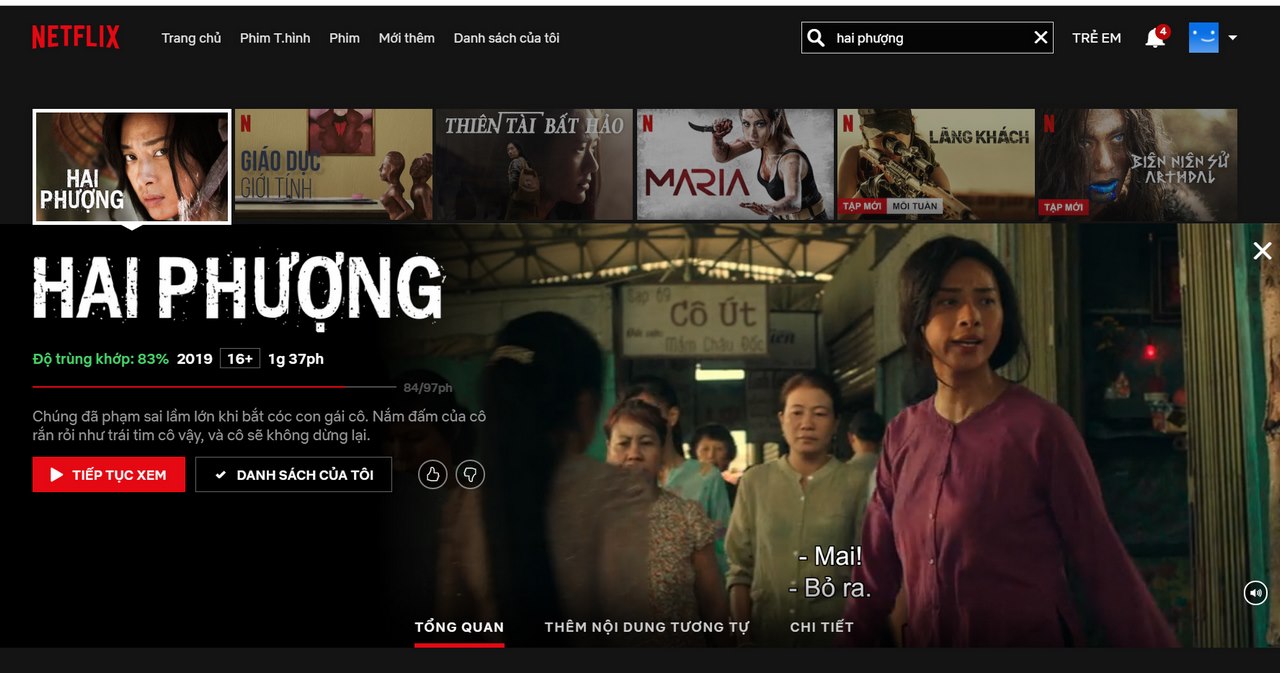
ReactJS là một thư viện JavaScript phổ biến được sử dụng rộng rãi trong việc phát triển các ứng dụng web hiện đại. Với triết lý thiết kế tập trung vào việc tạo ra các giao diện người dùng linh hoạt, dễ bảo trì và hiệu quả, React đã trở thành một trong những công nghệ hàng đầu trong lĩnh vực này.

Một trong những điểm nổi bật của React là sự phân chia giao diện thành các thành phần (components) độc lập và tái sử dụng. Các component trong React có thể được xem như các khối xây dựng của giao diện người dùng, mỗi khối đều có thể chứa mã HTML, CSS và JavaScript cùng với trạng thái và thuộc tính riêng. Điều này giúp tạo ra mã nguồn dễ bảo trì, tổ chức tốt và tái sử dụng cao.

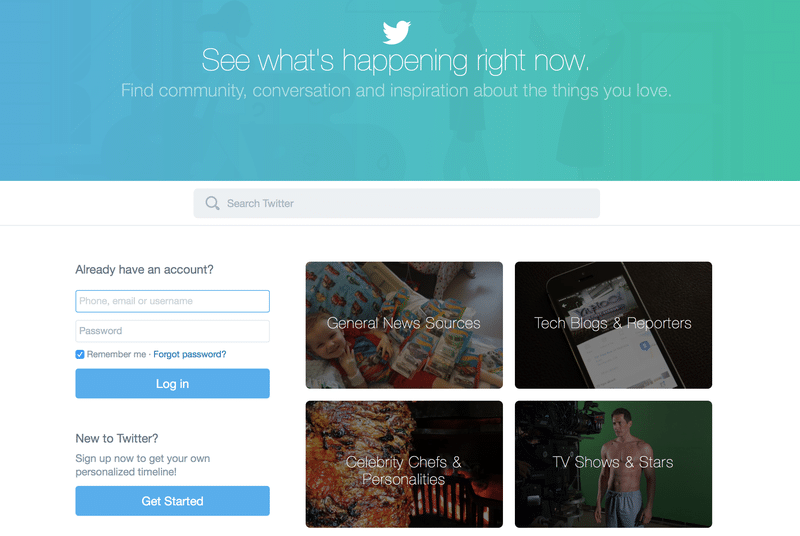
Ngoài ra, React còn sử dụng Virtual DOM (Document Object Model ảo) để tối ưu hóa hiệu suất của ứng dụng. Thay vì cập nhật trực tiếp DOM mỗi khi có thay đổi, React tạo ra một bản sao của DOM trong bộ nhớ, so sánh với DOM thực sự và chỉ cập nhật những phần cần thiết, giúp tăng hiệu suất và độ nhanh nhẹn của ứng dụng.

Với những ưu điểm nổi bật như vậy, ReactJS không chỉ phù hợp cho việc xây dựng các ứng dụng web động và đơn trang mà còn có thể áp dụng trong việc phát triển các ứng dụng di động, desktop, bảng điều khiển quản trị và nhiều loại ứng dụng khác. Điều này đã làm cho ReactJS trở thành một công nghệ phổ biến và mạnh mẽ trong cộng đồng phát triển phần mềm.

Một số trang web lớn nổi tiếng được xây dựng bằng ReactJs như : Facebook, Twitter, NetFlix, DropBox,…



Hình 1.3 : Trang web sử dụng ReactJs



Hình 1.4 : Tranng Web sử dụng ReatJs

### 1.2.2 Java Spring Boot

#### 1.2.1.1 Giới thiệu về Spring Boot

Spring Boot là một module của Spring Framework, cung cấp tính năng RAD (Rapid Application Development) – Phát triển ứng dụng nhanh. Nó được dùng để tạo các ứng dụng độc lập dựa trên Spring một cách nhanh chóng Spring Boot không yêu cầu cấu hình XML và là một chuẩn cho cấu hình thiết kế phần mềm, tăng cao năng suất cho developer.



Hình 1.5 : Hình ảnh về Spring Boot

#### 1.2.1.2 Ưu điểm của Spring Boot

Với mục đích ra đời rất rõ ràng của mình, Spring Boot đã khắc phục được những hạn chế về cấu hình của Spring. Dưới đây là một số ưu điểm của Spring Boot :

+ Hội tụ đầy đủ các tính năng của Spring framework.

+ Đơn giản hóa cấu hình và xây dựng được các ứng dụng độc lập có khả năng chạy bằng “java -jar” nhờ các dependency starter.

+ Dễ dàng deploy vì các ứng dụng server được nhúng trực tiếp vào ứng dụng để tránh những khó khăn khi triển khai lên môi trường production mà không cần thiết phải tải file WAR.

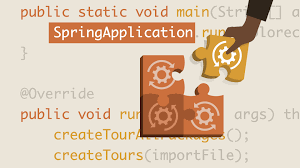
+ Cấu hình ít, tự động hỗ trợ bất cứ lúc nào cho chức năng giống với Sping như tăng năng suất, giảm thời gian viết code và không yêu cầu XML config.

+ Cung cấp nhiều plugin, số liệu, cấu hình ứng dụng từ bên ngoài.

#### 1.2.1.3 Các đặc tính cơ bản của Spring Boot

Vì là một framework RAD nên Spring Boot mang trong nó nhiều đặc tính nổi trội phục vụ cho việc phát triển và cài đặt nhanh một ứng dụng chạy trên Java.

Đặc tính đầu tiên có thể kể đến đó là Spring Boot cung cấp sẵn cho chúng ta một lớp có chứa [hàm main](https://stringee.com/vi/blog/post/phuong-thuc-main-trong-java) và được hoạch định làm điểm mở đầu cho toàn bộ chương trình. Lớp này được đặt tên là Spring Application, nó giúp khởi chạy các ứng dụng từ hàm main, khi chạy nó chúng ta chỉ cần gọi phương thức run.



Hình 1.6 : Phương thức main trong Spring Boot

Spring Boot giúp giảm bớt độ phức tạp trong việc cấu hình ứng dụng trong trường hợp chúng ta sử dụng nhiều môi trường. Với Profiles, Spring Boot cung cấp cho người sử dụng một cách phân chia cấu hình cho từng môi trường. Các bên thực hiện việc cấu hình ứng dụng hoàn toàn có thể được đánh dấu profiles để giới hạn thời điểm hay môi trường mà nó sẽ được tải các cấu hình lên ứng dụng.

Externalized Configurations: Externalized Configuration cho phép bạn có khả năng cấu hình được từ bên ngoài. Vì vậy, một ứng dụng được xây dựng có thể được vận hành và hoạt động trên nhiều môi trường khác nhau. Để thực hiện Externalized Configuration bạn có thể sử dụng các file properties, YAML, các tham số command line hay các biến môi trường.

Đặc tính cuối cùng mà bạn nên biết đó là tất cả các tính năng log nội bộ của Spring Boot đều sử dụng common logging. Chúng được quản lý một cách mặc định, vì vậy bạn không nên sửa các dependency logging nếu không được yêu cầu.

### 1.2.3 MySQL

#### 1.2.3.1 MySQL là gì ?

MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (Relational Database Management System, viết tắt là RDBMS) hoạt động theo mô hình client-server. RDBMS là một phần mềm hay dịch vụ dùng để tạo và quản lý các cơ sở dữ liệu (Database) theo hình thức quản lý các mối liên hệ giữa chúng.



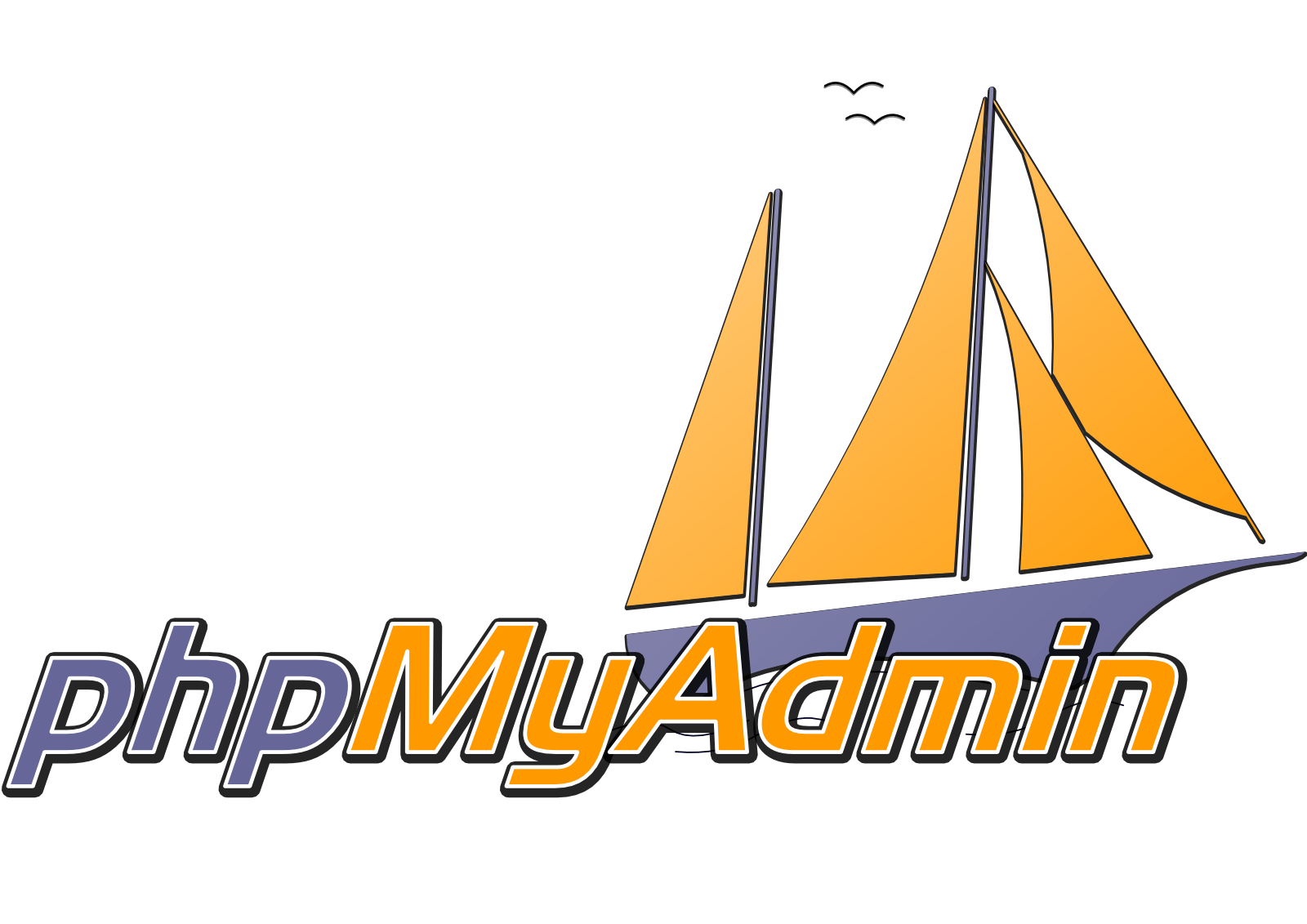
Hình 1.7 : Hình ảnh về MySQL

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. Kể cả khi ban đầu nó chỉ được dùng rất hạn chế nhưng giờ nó đã tương thích với nhiều hạ tầng máy tính quan trọng như Linux, macOS, Microsoft Windows, và Ubuntu.

MySQL Server : là máy tính hay một hệ các máy tính cài đặt phần mềm MySQL dành cho server để giúp bạn lưu trữ dữ liệu trên đó, để máy khách có thể truy cập vào quản lý. Dữ liệu này được đặt trong các bảng, và các bảng có mối liên hệ với nhau. MySQL server nhanh, an toàn, đáng tin cậy. Phần mềm MySQL cũng miễn phí và được phát triển, phân phối và hỗ trợ bởi Oracle Corporation.

MySQL Client : MySQL client không hẵn phải cài phần mềm MySQL của Oracle mà là nói chung của mọi phần mềm có thể thực hiện truy vấn lên một MySQL server và nhận kết quả trả về. MySQL client điển hình là đoạn mã PHP script trên một máy tính hay trên cùng server dùng để kết nối tới cơ sở dữ liệu MySQL database. Phpmyadmin cũng là một MySQL client có giao diện người dùng.

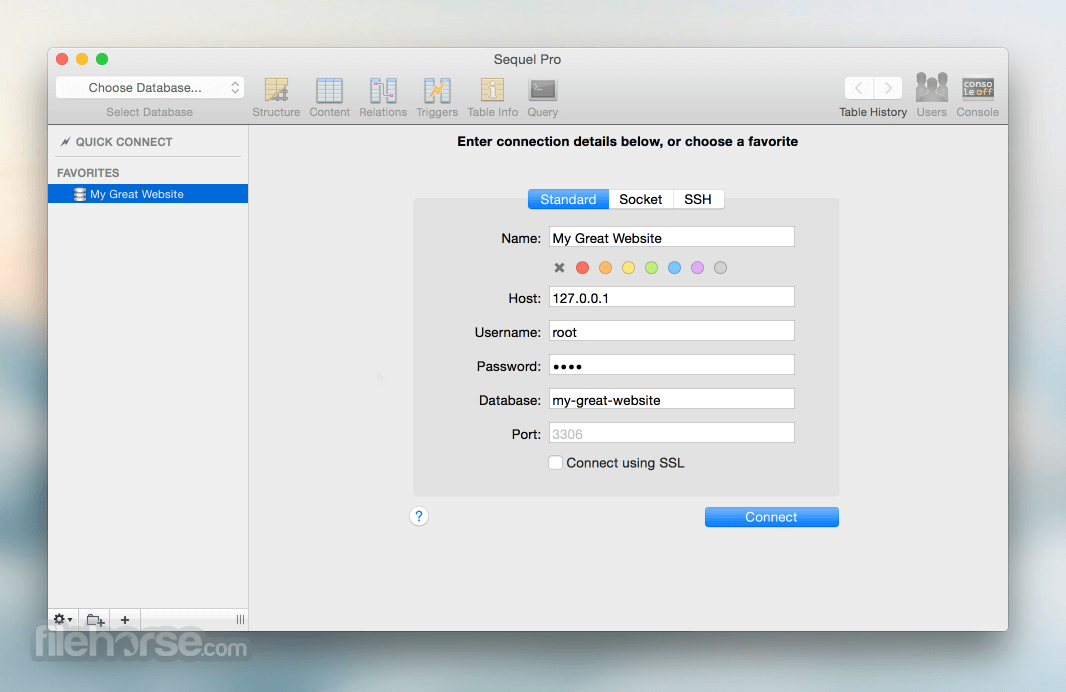
Một số công cụ miễn phí dùng làm MySQL là : MySQL Workbench , Sequel Pro, HeidiSQL, phpMyAdmin, …



Hình 1.8 : Công cụ phpMyAdmin



Hình 1.9 : Công cụ MySQL Workbench



Hình 1.10 : Công cụ Sequel pro

#### 1.2.3.2 Ưu điểm của MySQL

MySQL không phải là hệ quản lý cơ sở dữ liệu (RDBMS) duy nhất trên thị trường, nhưng nó đích thực phổ biến nhất và chỉ xếp sau Oracle Database khi xét đến những thông số chính như số lượng tìm kiếm, profile người dùng trên LinkedIn, và lượng thảo luận trên các diễn đàn internet. Lý do chính là ở những ưu điểm của nó mang lại.

Linh hoạt và dễ dùng : Bạn có thể sửa Source code để đáp ứng nhu cầu của bạn mà không phải thanh toán thêm bất kì chi phí nào. Quá trình cài đặt cũng rất đơn giản và thường không quá 30 phút.

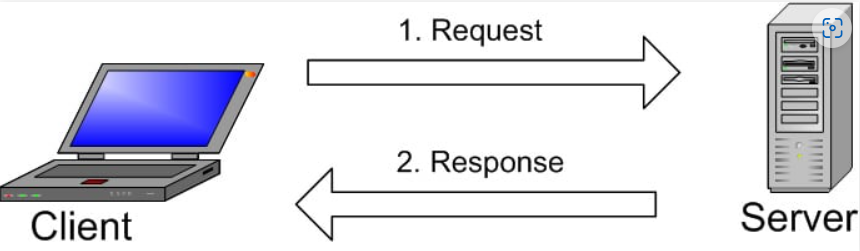
Hiệu năng cao : Nhiều server clusters sử dụng MySQL. Bất kể bạn lưu trữ dữ liệu lớn của các trang thương mại điện tử hoặc những hoạt động kinh doanh nặng nề liên quan đến công nghệ thông tin, MySQL cũng có thể đáp ứng được với tốc độ cao, mượt mà.

Tiêu chuẩn trong ngành : Ngành công nghệ và dữ liệu đã sử dụng MySQL nhiều năm, vì vậy nó là một kỹ năng căn bản một chuyên gia lập trình. Người dùng MySQL cũng có thể triển khai dự án nhanh và thuê các chuyên gia dữ liệu với mức phí nếu họ cần.

An toàn : An toàn dữ liệu luôn là vấn đề quan trọng nhất khi chọn phần mềm RDBMS. Với hệ thống phân quyền truy cập và quản lý tài khoản, MySQL đặt tiêu chuẩn bảo mật rất cao. Mã hóa thông tin đăng nhập và chứng thực từ host đều khả dụng.

#### 1.2.3.3 MySQL hoạt động như thế nào ?

Là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS), MySQL hoạt động theo mô hình quan hệ, nơi dữ liệu được tổ chức thành các bảng có cấu trúc

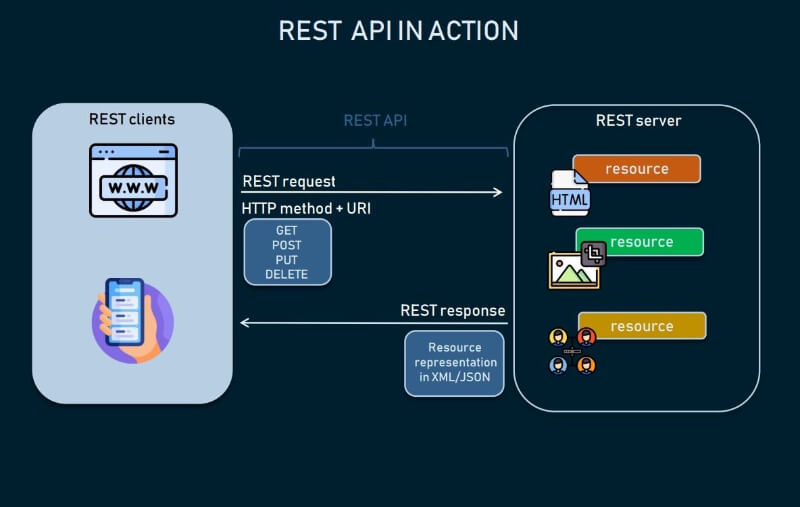


Hình 1.11 : Phương thức hoạt động của MySQL

Một máy client sẽ liên lạc với máy server trong một mạng nhất định. Mỗi client có thể gửi một request từ giao diện người dùng (Graphical user interface – GUI) trên màn hình, và server sẽ trả về kết quả như mong muốn. Miễn là cả hai hiểu nhau. Cách vận hành chính trong môi trường MySQL cũng như vậy, đầu tiên MySQL tạo ra bảng để lưu trữ dữ liệu, định nghĩa sự liên quan giữa các bảng đó. Tiếp theo, Client sẽ gửi yêu cầu SQL bằng một lệnh đặc biệt trên MySQL. Cuối cùng, ứng dụng trên Server sẽ phản hồi thông tin và trả về kết quả trên máy client.

## Kiến trúc phần mềm

Trang web bán đồ ăn SuperFood sử dụng REST API để liên kết và trao đổi dữ liệu giữa phía máy chủ và phía người dùng. REST API (Representational State Transfer Application Programming Interface) là một phương thức giao tiếp phổ biến trong phát triển web, cho phép truy cập vào và quản lý các tài nguyên trên máy chủ từ xa thông qua các yêu cầu HTTP.

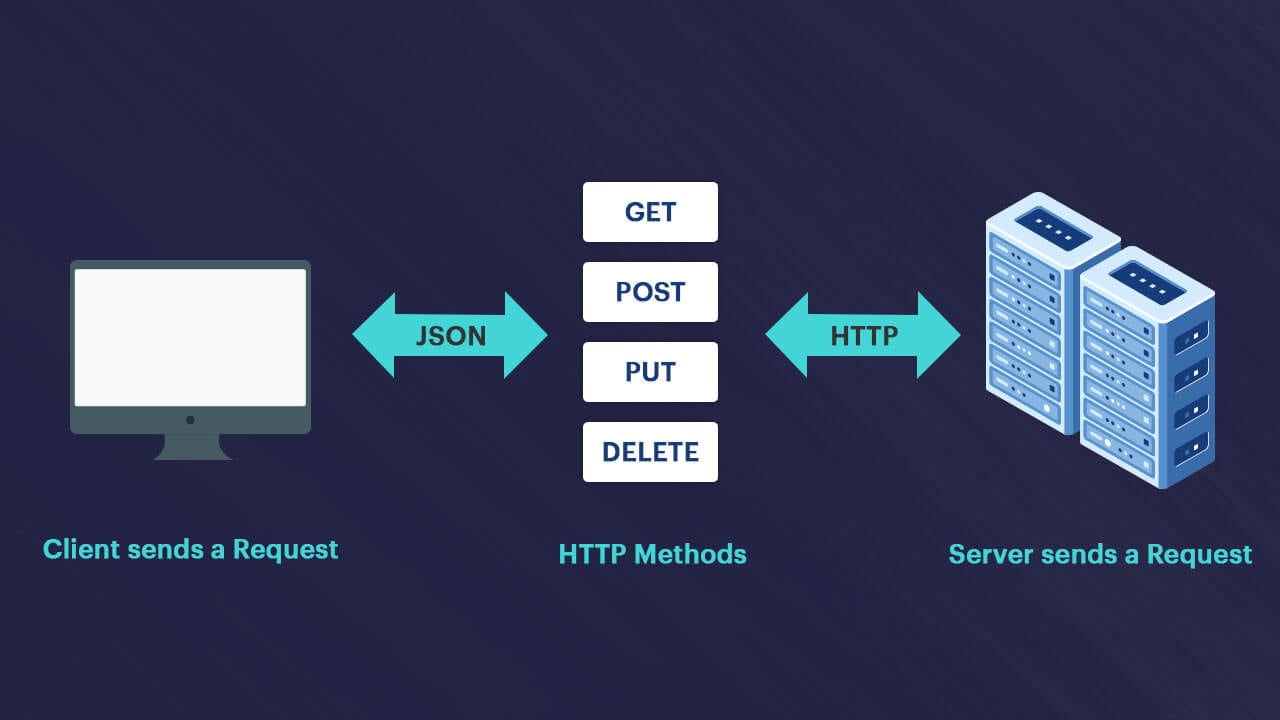


Hình 1.12: Phương thức hoạt động của REST API

REST (REpresentational State Transfer) là một đại diện cho sự chuyển đổi dữ liệu. Trong kiến trúc này client và server hoàn toàn độc lập, chúng không biết gì về nhau. Mỗi một request REST API đều không mang theo trạng thái trước đó (stateless).

API (Application Programming Interface) là giao diện lập trình ứng dụng. Giao diện này không dành cho người dùng cuối mà dành cho các nhà phát triển (developer). Nó là cái “bề mặt”, chỉ thấy được phần khai báo (tên, tham số, kiểu trả về), bộ đồ lòng body thì không biết. “Biết mặt không biết lòng” chính là API.

Các phương thức HTTP như GET, POST, PUT và DELETE được sử dụng để thực hiện các thao tác tương ứng trên các tài nguyên, với mỗi tài nguyên được định danh bằng một URL duy nhất. REST API cung cấp các endpoint để truy cập và tương tác với dữ liệu, và sử dụng các định dạng dữ liệu như JSON hoặc XML để truyền và nhận dữ liệu giữa máy khách và máy chủ. Tính linh hoạt và tiêu chuẩn của REST API giúp tạo ra một phương thức giao tiếp hiệu quả và dễ sử dụng giữa các thành phần của một ứng dụng web.



Hình 1.13 : Các phương thức trong REST API

# CHƯƠNG 2 : PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## 2.1. Mô tả nghiệp vụ

Website được xây dựng với mục tiêu là nơi trưng bày, giới thiệu sản phẩm và các dịch vụ hỗ trợ khách hàng như: liên hệ, phản hồi, tư vấn, tìm kiếm…

Khách hàng bắt đầu duyệt những sản phẩm trên website, các sản phẩm được bố trí theo từng loại đa dạng như sản phẩm mới, sản phẩm bán chạy…Khi đã ưng ý một sản phẩm khách hàng có thể kích vào hình sản phẩm hoặc tên để xem thông tin chi tiết về sản phẩm. Sau đó khách hàng có thể chọn nút “**Mua Ngay**” để đưa sản phẩm vào giỏ hàng của mình, giỏ hàng chỉ đơn giản là danh sách các sản phẩm bao gồm những thông tin như mã sản phẩm, tên sản phẩm, đơn giá, số lượng, thành tiền.

Các giỏ hàng thường cung cấp các tùy chọn để xóa sạch giỏ hàng, xóa một sản phẩm, tiếp tục mua sản phẩm và cập nhật số lượng.

Ngay sau khi khách hàng có tất cả các sản phẩm cần mua, khách hàng có thể chọn nút “**Đặt Mua**”. Trang đơn hàng này gồm có thông tin về khách hàng (họ tên, email, địa chỉ…), thông tin về địa chỉ giao hàng ( địa chỉ, ngày giao…), phương thức thanh toán (tiền mặt, thanh toán tiền trực tuyến), sau cùng là thông tin về giỏ hàng (mã sản phẩm, tên sản phẩm, đơn giá, số lượng, thành tiền).

Sau khi khách hàng đã điền đầy đủ thông tin thì chọn nút “**Đặt Hàng**” để hoàn tất quá trình mua hàng.

## 2.2. Đặc tả yêu cầu

### 2.2.1. Yêu cầu chức năng

Website gồm 2 actor chính : Admin(Quản trị viên) và User (Người dùng)

Admin : Có quyền thêm sửa xóa danh mục sản phẩm, thêm sửa xóa sản phẩm, quản lý tài khoản, quản lý đơn đặt hàng , chặn và bỏ chặn tài khoản người dùng quản lý doanh thu

User : Có quyền tìm kiếm thông tin sản phẩm, đăng ký, đăng nhập, xem bài viết, đặt hàng, xem thông tin đơn hàng đã đặt

### 2.2.2 Yêu cầu phi chức năng

Dung lượng website vừa phải, tốc độ truy xuất nhanh. Hệ thống thông tin phải có chế độ bảo mật, không chấp nhận sai sót. Cơ sở dữ liệu phải được đảm bảo khi hệ thống đang hoạt động. Khi người dùng thực hiện việc đăng nhập vào hệ thống phải thông qua các bước sau: nhập tên đăng nhập và mật khẩu. Nếu khách hàng lần đầu tiên tham gia mua hàng thì yêu cầu đăng ký khách hàng và khách hàng đó được gán một mã số riêng biệt. Khi đăng ký khách hàng cần điền đầy đủ thông tin đăng nhập gồm Email, mật khẩu và nhập lại mật khẩu ngoài ra cần phải cho biết thêm thông tin về địa chỉ và số điện thoại.

Khách hàng chưa đăng nhập sẽ không được xem các thông tin về tài khoản, mua hàng, lịch sử đơn hàng

Thông tin tài khoản sẽ phải được mã hóa, bảo mật, mật khẩu đăng nhập cũng được mã hóa, ngay cả admin cũng không thể biết mật khẩu của tài khoản người dùng.

Thông tin tài khoản sẽ phải được mã hóa, bảo mật, mật khẩu đăng nhập cũng được mã hóa, ngay cả admin cũng không thể biết mật khẩu của tài khoản người dùng.

Thông tin tài khoản sẽ phải được mã hóa, bảo mật, mật khẩu đăng nhập cũng được mã hóa, ngay cả admin cũng không thể biết mật khẩu của tài khoản người dùng.

Tất cả các api của admin và api đặt hàng bắt buộc phải xác thực người dùng

## 2.3 Mô tả use case

****

Hình 2.1 : Sơ đồ usecase tổng quát

### 2.3.1 Đăng kí tài khoản

Tên use case: Đăng ký tài khoản.

Mô tả use case: Use case thực hiện đăng ký tài khoản của người dùng.

Actor chính: Khách hàng viếng thăm.

Actor phụ: không có.

Tiền điều kiện: Khách hàng viếng thăm truy cập vào trang đăng ký tài khoản.

Hậu điều kiện: Thông báo đăng ký thành công và chuyển sang trang đăng nhập.

Luồng hoạt động:

Vào trang đăng ký.

Nhập thông tin đăng ký.

Hệ thống xử lý.

Lưu vào DataBase.

Luồng thay thế: Không có.

Luồng ngoại lệ: Ở bước 3, hệ thống kiểm tra thông tin không hợp lệ quay lại bước 2.

### 2.3.2 Đăng nhập

Tên use case: Đăng nhập.

Mô tả use case: Use case thực hiện đăng nhập.

Actor chính: User.

Actor phụ: không có.

Tiền điều kiện: User truy cập vào trang đăng nhập.

Hậu điều kiện: Thông báo đăng nhập thành công và chuyển sang trang chủ.

Luồng hoạt động:

Vào trang đăng nhập.

Nhập thông tin và nhấn đăng nhập.

Hệ thống xử lý.

Chuyển đến trang chủ.

Luồng thay thế: Không có.

Luồng ngoại lệ: Ở bước 3, hệ thống kiểm tra đăng nhập không hợp lệ quay lại bước 2.

### 2.3.3. Đặt hàng

Tên use case: Đặt hàng

Mô tả use case: Use case thực hiện đặt hàng

Actor chính: User.

Actor phụ: không có.

Tiền điều kiện: User truy cập vào trang đăng nhập.

Hậu điều kiện: Thông báo đặt hàng thành công.

Luồng hoạt động:

Chọn chức năng đặt hàng .

Kiểm tra số lượng sản phẩm trong kho

Nhập thông tin đặt hàng

Kiểm tra trường bắt buộc

Thông báo thành công

Luồng thay thế:

Thông báo lỗi : Số lượng sản phẩm không đủ

Thông báo lỗi : không được bỏ trống thông tin người nhận

Luồng ngoại lệ: Ở bước 3, hệ thống kiểm tra đăng nhập không hợp lệ quay lại bước 2.

### 2.3.4. Hủy hàng

Tên use case: Hủy hàng.

Mô tả use case: Use case thực hiện hủy đơn hàng.

Actor chính: User.

Actor phụ: không có.

Tiền điều kiện: User truy cập vào trang đăng nhập và đã đặt hàng.

Hậu điều kiện: Thông báo hủy đơn hàng thành công .

Luồng hoạt động:

Chọn đơn hàng muốn hủy.

Kiểm tra trạng thái đơn hàng có hợp lệ.

Xác nhận hủy đơn hàng.

Cập nhật trạng thái hủy đơn hàng.

Thông báo hủy đơn thành công.

Luồng thay thế:

Thông báo lỗi : Hủy đơn không thành công.

Luồng ngoại lệ: Ở bước 3, hệ thống kiểm tra đăng nhập không hợp lệ quay lại bước 2.

### 2.3.5. Quản lí người dùng

Mô tả vắn tắt:

Use case này cho phép Admin xem, thêm, sửa và xoá thông tin người dùng trong bảng User, User\_role và Commoncategory.

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi Admin kích vào nút “Quản lí người dùng” trên thanh menu. Hệ thống truy vấn bảng User, User\_role và Commoncategory và hiển thị danh sách các người dùng gồm họ, tên, email, tên tài khoản, số điện thoại, nhóm quyền, phòng ban và trạng thái lên màn hình.

Admin kích đúp vào bất kì thông tin nào của một người dùng. Hệ thống truy vấn bảng User, User\_role và Commoncategory và hiển thị danh sách các thông tin của người dùng đó gồm tên, họ, tên tài khoản, mật khẩu, phòng ban, trạng thái, email, số điện thoại, nhóm quyền, phòng ban lên màn hình. Use case kết thúc.

Thêm người dùng:

Admin kích vào nút “Thêm” trên cửa sổ danh sách người dùng. Hệ thống hiển thị màn hình yêu cầu nhập thông tin chi tiết cho người dùng bao gồm tên, họ, tên tài khoản, mật khẩu, phòng ban, trạng thái, email, số điện thoại, nhóm quyền.

Admin nhập thông tin người dùng và kích vào nút “Lưu”. Hệ thống sẽ sinh một mã người dùng mới, tạo một người dùng trong bảng User và hiển thị thông báo thành công “Thêm mới thành công!”.

Sửa người dùng:

Khi Admin kích vào bất kì 1 người dùng nào và ấn vào nút “Sửa” trên cửa sổ danh sách người dùng. Hệ thống sẽ lấy thông tin từ mã của người dùng được chọn gồm: tên, họ, tên tài khoản, mật khẩu, phòng ban, trạng thái, email, số điện thoại, nhóm quyền, phòng ban từ bảng User và hiển thị lên màn hình.

Người quản trị nhập thông tin mới cho người dùng đã chọn, kích vào nút “Lưu”. Hệ thống sẽ sửa thông tin của nguời dùng được chọn trong bảng User và sẽ hiển thị thông báo thành công: “Cập nhật người dùng thành công”.

Luồng rẽ nhánh:

Tại bước 1 trong luồng cơ bản, nếu trong bảng User chưa có dữ liệu, thì hệ thống sẽ hiển thị một thông báo: “Không có dữ liệu để hiển thị” và use case kết thúc.

Tại bước 3b trong luồng cơ bản, nếu không nhập đủ thông tin yêu cầu hệ thống sẽ hiển thị lên “Thông tin này là bắt buộc” và use case kết thúc.

Tại bước 4b trong luồng cơ bản, nếu không nhập đúng, đủ thông tin yêu cầu hệ thống sẽ hiển thị lên “Cập nhật người dùng không thành công” và use case kết thúc.

Tại bất kỳ bước nào trong luồng cơ bản, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.

Các yêu cầu đặc biệt: Use case này chỉ cho phép những người có vai trò admin hoặc chủ hệ thống thực hiện.

Tiền điều kiện: Người dùng cần đăng nhập với vai trò người quản trị hệ thống để thực hiện use case.

Hậu điều kiện: Nếu use case được thực hiện thành công thì thông tin về Người dùng sẽ được cập nhật trong hệ thống cơ sở dữ liệu.

Điểm mở rộng: Không có.

### 2.3.6. Cập nhật thông tin tài khoản

Mô tả vắn tắt: use case này cho phép khách cập nhật lại thông tin tài khoản

của mình.

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi người dùng ấn cập nhật thông tin tài khoản

trên giao diện. Hệ thống sẽ hiển thị form cập nhật thông tin tài khoản

để người dùng nhập lại thông tin tài khoản.

Người dùng nhập thông tin cần thiết sau đó ấn cập nhật. Hệ thống

sẽ kiểm tra dữ liệu người dùng nhập. Nếu dữ liệu hợp hệ thì hệ thống

sẽ cập nhật lại thông tin người dùng trong bảng Customer và hiển

thị kết quả lên màn hình. Use case kết thúc.

Luồng rẽ nhánh:

Tại luồng cơ bản 2 nếu người dùng nhập dữ liệu không hợp lệ thì hệ

thống sẽ yêu cầu nhập lại.

Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu

không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một

thông báo lỗi và use case kết thúc.

Tiền điều kiện: Người dùng phải đăng nhập thành công mới có thể cập nhật

thông tin tài khoản.

Hậu điều kiện: Không có.

Các yêu cầu đặc biệt: Không có.

Điểm mở rộng: Không có.

### 2.3.7. Xem hàng theo danh mục

Mô tả vắn tắt: use case này cho phép khách xem hàng theo các danh mục.

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi người dùng ấn vào danh mục bất kì trên

menu. Hệ thống sẽ lấy ra các mặt hàng từ bảng Product nằm trong

danh mục được chọn và hiển thị kết quả lên màn hình. Use case kết

thúc.

Luồng rẽ nhánh:

Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu

không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một

thông báo lỗi và use case kết thúc.

Tiền điều kiện: Không có.

Hậu điều kiện: Không có.

Các yêu cầu đặc biệt: Không có.

Điểm mở rộng: Không có.

### 2.3.8. Xem chi tiết sản phẩm

Mô tả vắn tắt: use case này cho phép khách hàng xem thông tin chi tiết một

mặt hàng.

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi người dùng ấn vào 1 mặt hàng cụ thể trên

giao diện. Hệ thống sẽ lấy thông tin chi tiết của mặt hàng đó từ bảng

Product và hiển thị kết quả lên màn hình.

Luồng rẽ nhánh:

Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu

không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một

thông báo lỗi và use case kết thúc.

Tiền điều kiện: Không có.

Hậu điều kiện: Không có.

Các yêu cầu đặc biệt: Không có.

Điểm mở rộng: Không có.

### 2.3.9. Tìm kiếm sản phẩm

Mô tả vắn tắt: use case này cho phép khách hàng tìm kiếm một sản phẩm

theo tên sản phẩm.

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi người dùng nhập tên sản phẩm vào ô tìm

kiếm và nhấn tìm kiếm. Hệ thống sẽ lấy ra các sản phẩm có tên thỏa

mãn yêu cầu tìm kiếm của khách hàng rồi hiển thị ra màn hình. Use

case kết thúc.

Luồng rẽ nhánh:

Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu

không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một

thông báo lỗi và use case kết thúc.

Tiền điều kiện: Không có.

Hậu điều kiện: Không có.

Các yêu cầu đặc biệt: Không có.

Điểm mở rộng: Không có.

### 2.3.10. Quản lí giỏ hàng

Mô tả vắn tắt: use case này cho phép khách hàng thêm, xóa, cập nhật số

lượng sản phẩm trong giỏ hàng.

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi người dùng ấn vào icon giỏ hàng trên giao

diện. Hệ thống sẽ lấy ra thông tin giỏ hàng của khách hàng và hiển

thị lên màn hình.

Cập nhật số lượng sản phẩm: Người dùng nhấn vào icon tăng giảm

số lượng hàng để cập nhật lại số lượng các mặt hàng trong bảng Cart

và hiển thị kết quả ra giao diện.

Xóa sản phẩm khỏi giỏ hàng: Người dùng ấn xóa để xóa sản phẩm

không mong muốn khỏi giỏ hàng. Hệ thống sẽ xóa sản phẩm khỏi

bảng Cart và hiển thị kết quả ra giao diện.

Use case kết thúc.

Luồng rẽ nhánh:

Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu

không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một

thông báo lỗi và use case kết thúc.

Tiền điều kiện: Người dùng phải đăng nhập thành công mới có thể truy cập

vào giỏ hàng và thực hiện chức năng quản lý giỏ hàng.

Hậu điều kiện: Không có.

Các yêu cầu đặc biệt: Không có.

Điểm mở rộng: Không có.

### 2.3.11. Xem thống kê

Mô tả vắn tắt: use case này cho phép quản trị viên theo dõi doanh số bán

hàng.

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi quản trị viên chọn mục “thống kê” trên

menu. Hệ thống sẽ lấy thông tin từ các bảng Order, OrderDetail, và

hiển thị các kết quả cần thiết lên màn hình.

Luồng rẽ nhánh:

Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu

không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một

thông báo lỗi và use case kết thúc.

Tiền điều kiện: Người dùng phải đăng nhập thành công với tư cách là quản

trị viên thì mới có thể xem thống kê.

Hậu điều kiện: Không có.

Các yêu cầu đặc biệt: Không có.

Điểm mở rộng: Không có

### 2.3.12. Quản lí danh mục

Mô tả vắn tắt: use case này cho phép quản trị viên thêm, xóa, cập nhật thông

tin danh mục.

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi quản trị viên chọn mục “danh mục” trên

menu. Hệ thống sẽ lấy thông tin danh mục trong bảng Category và

hiển thị danh sách và thông tin các danh mục.

Thêm danh mục: quản trị viên ấn vào tạo mới danh mục. Hệ thống

sẽ hiển thị form tạo mới thông tin danh mục. Quản trị viên điền

thông tin danh mục vào form rồi ấn tạo. Hệ thống sẽ lưu danh mục

vào bảng Category và hiển thị thông báo thêm thành công thư mục.

Cập nhật thông tin danh mục: quản trị viên ấn vào cập nhật thông

tin danh mục. Hệ thống sẽ hiển thị form thông tin danh mục. Quản

trị viên sửa thông tin danh mục cần thiết sau đó ấn lưu. Hệ thống sẽ

cập nhật lại thông tin danh mục trong bảng Category và hiển thị

thông báo cập nhật thành công.

Xóa danh mục: quản trị viên ấn xóa danh mục. Hệ thống sẽ hiển thị

form xác nhận xóa. Quản trị viên ấn xác nhận xóa thì hệ thống sẽ

xóa danh mục khỏi bảng Category và hiển thị thông báo xóa thành

công.

Use case kết thúc.

Luồng rẽ nhánh:

Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu

không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một

thông báo lỗi và use case kết thúc.

Tiền điều kiện: Người dùng phải đăng nhập thành công với tư cách là quản

trị viên.

Hậu điều kiện: Không có.

Các yêu cầu đặc biệt: Không có.

Điểm mở rộng: Không có

### 2.3.13. Quản lí sản phẩm

Mô tả vắn tắt: use case này cho phép quản trị thêm, xóa, cập nhật thông tin

sản phẩm.

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi quản trị viên chọn mục “sản phẩm” trên

menu. Hệ thống sẽ lấy thông tin các sản phẩm trong bảng Product

và hiển thị danh sách và thông tin của các sản phẩm có trong hệ

thống.

Thêm mới sản phẩm: quản trị viên ấn “Tạo mới”. Hệ thống sẽ hiển

thị form thêm mới sản phẩm. Quản trị viên nhập thông tin sản phẩm

sau đó ấn thêm. Hệ thống sẽ thêm sản phẩm vào bảng Product và

hiển thị thông báo thêm thành công.

Cập nhật thông tin sản phẩm: quản trị viên ấn “Cập nhật”. Hệ thống

sẽ hiển thị form cập nhật thông tin sản phẩm. Quản trị viên thực hiện

thay đổi các thông tin cần thiết và ấn lưu. Hệ thống sẽ cập nhật lại

thông tin sản phẩm trong bảng Product và hiển thị thông báo cập

nhật thành công.

Xóa sản phẩm: quản trị viên ấn “Xóa”. Hệ thống hiển thị form xác

nhận xóa. Quản trị viên xác nhận xóa thì hệ thống sẽ xóa sản phẩm

khỏi bảng Product và hiển thị thông báo xóa thành công.

Use case kết thúc.

Luồng rẽ nhánh:

Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu

không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một

thông báo lỗi và use case kết thúc.

Tiền điều kiện: Người dùng phải đăng nhập thành công với tư cách là quản

trị viên.

Hậu điều kiện: Không có.

Các yêu cầu đặc biệt: Không có.

Điểm mở rộng: Không có.

### 2.3.14. Quản lí tài khoản

Mô tả vắn tắt: use case này cho phép quản trị viên xem và xóa tài khoản

người dùng.

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi quản trị viên chọn mục “Tài khoản” trên

menu. Hệ thống sẽ lấy thông tin các tài khoản trong bảng Customer

và hiển thị danh sách các tài khoản của người dùng.

Xem tài khoản: quản trị viên ấn “Xem chi tiết”. Hệ thống sẽ lấy ra

thông tin chi tiết của tài khoản trong bảng Customer và hiển thị lên

màn hình

Xóa tài khoản: quản trị viên ấn “Xóa”. Hệ thống sẽ hiển thị form

xác nhận xóa. Quản trị viên xác nhận xóa thì hệ thống sẽ xóa tài

khoản khỏi bảng Customer và hiển thị thông báo xóa thành công.

Use case kết thúc.

Luồng rẽ nhánh:

Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu

không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một

thông báo lỗi và use case kết thúc.

Tiền điều kiện: Người dùng phải đăng nhập thành công với tư cách là quản

trị viên.

Hậu điều kiện: Không có.

Các yêu cầu đặc biệt: Không có.

Điểm mở rộng: Không có.

### 2.3.15. Quản lí đơn hàng

Mô tả vắn tắt: use case này cho phép quản trị viên quản lý những đơn hàng đã được đặt.

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi quản trị viên chọn mục “Đơn hàng” trên

menu. Hệ thống sẽ lấy thông tin các đơn hàng trong bảng Order và

hiển thị danh sách các đơn đặt hàng trong hệ thống.

Xem chi tiết: quản trị viên ấn “Xem chi tiết”. Hệ thống sẽ lấy ra

thông tin chi tiết của đơn hàng trong bảng OrderDetail và hiển thị

kết quả lên màn hình.

Cập nhật trạng thái đơn hàng: quản trị viên ấn “Cập nhật trạng thái”.

Hệ thống sẽ hiển thị form cho phép thay đổi trạng thái đơn hàng. Quản trị viên thực hiện thay đổi trạng thái đơn hàng và ấn lưu. Hệ

thống sẽ cập nhật trạng thái của đơn hàng trong bảng Order và hiển

thị thông báo cập nhật thành công.

Xóa đơn hàng: quản trị viên ấn “Xóa”. Hệ thống sẽ hiển thị form

xác nhận xóa. Quản trị viên xác nhận xóa thì hệ thống sẽ xóa đơn

hàng khỏi bảng Order và OrderDetail và hiển thị thông báo xóa

thành công.

Use case kết thúc.

Luồng rẽ nhánh:

Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu

không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một

thông báo lỗi và use case kết thúc.

Tiền điều kiện: Người dùng phải đăng nhập thành công với tư cách là quản

trị viên.

Hậu điều kiện: Không có.

Các yêu cầu đặc biệt: Không

### 2.3.16. Quản lí bài viết

Mô tả vắn tắt: use case này cho phép quản trị viên thêm, xóa, cập nhật thông

tin bài viết

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi quản trị viên chọn mục “bài viết” trên

menu. Hệ thống sẽ lấy thông tin danh mục trong bảng Category và

hiển thị danh sách và thông tin các danh mục.

Thêm danh mục: quản trị viên ấn vào tạo mới danh mục. Hệ thống

sẽ hiển thị form tạo mới thông tin bài viết . Quản trị viên điền

thông tin danh mục vào form rồi ấn tạo. Hệ thống sẽ lưu bài viết

vào bảng Category và hiển thị thông báo thêm thành công thư mục.

Cập nhật thông tin danh mục: quản trị viên ấn vào cập nhật thông

tin danh mục. Hệ thống sẽ hiển thị form thông tin danh mục. Quản

trị viên sửa thông tin danh mục cần thiết sau đó ấn lưu. Hệ thống sẽ

cập nhật lại thông tin danh mục trong bảng Category và hiển thị

thông báo cập nhật thành công.

Xóa danh mục: quản trị viên ấn xóa bài viết. Hệ thống sẽ hiển thị

form xác nhận xóa. Quản trị viên ấn xác nhận xóa thì hệ thống sẽ

xóa danh mục khỏi bảng Category và hiển thị thông báo xóa thành

công.

Use case kết thúc.

Luồng rẽ nhánh:

Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu

không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một

thông báo lỗi và use case kết thúc.

Tiền điều kiện: Người dùng phải đăng nhập thành công với tư cách là quản

trị viên.

Hậu điều kiện: Không có.

Các yêu cầu đặc biệt: Không có.

Điểm mở rộng: Không có

### 2.3.17. Quản lí kho mã giảm giá

Mô tả vắn tắt: use case này cho phép quản trị viên thêm, xóa, cập nhật thông

tin mã giam giá

Luồng sự kiện:

Luồng cơ bản:

Use case này bắt đầu khi quản trị viên chọn mục “voucher” trên

menu. Hệ thống sẽ lấy thông tin danh mục trong bảng Category và

hiển thị danh sách và thông tin các danh mục.

Thêm danh mục: quản trị viên ấn vào tạo mới voucher. Hệ thống

sẽ hiển thị form tạo mới thông tin voucher. Quản trị viên điền

thông tin voucher vào form rồi ấn tạo. Hệ thống sẽ lưu voucher

vào bảng Category và hiển thị thông báo thêm thành công thư mục.

Cập nhật thông tin danh mục: quản trị viên ấn vào cập nhật thông

tin danh mục. Hệ thống sẽ hiển thị form thông tin voucher. Quản

trị viên sửa thông tin voucher cần thiết sau đó ấn lưu. Hệ thống sẽ

cập nhật lại thông tin voucher trong bảng Category và hiển thị

thông báo cập nhật thành công.

Xóa danh mục: quản trị viên ấn xóa danh mục. Hệ thống sẽ hiển thị

form xác nhận xóa. Quản trị viên ấn xác nhận xóa thì hệ thống sẽ

xóa voucher khỏi bảng Category và hiển thị thông báo xóa thành

công.

Use case kết thúc.

Luồng rẽ nhánh:

Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu

không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một

thông báo lỗi và use case kết thúc.

Tiền điều kiện: Người dùng phải đăng nhập thành công với tư cách là quản

trị viên.

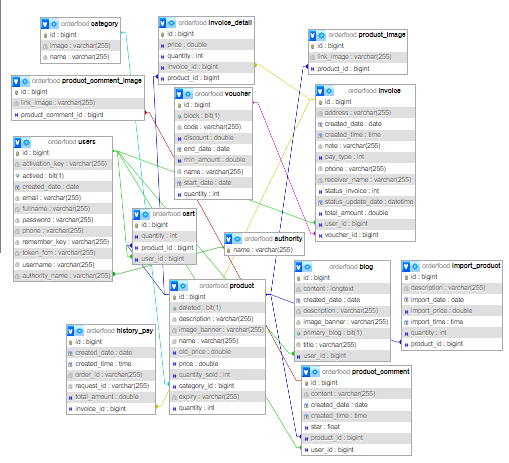
Hậu điều kiện: Không có.

Các yêu cầu đặc biệt: Không có.

Điểm mở rộng: Không có

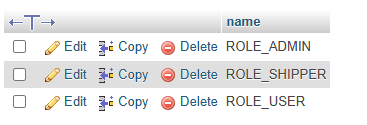
## 2.4. Thiết kế cơ sở dữ liệu

### 2.4.1. Mô hình hóa dữ liệu

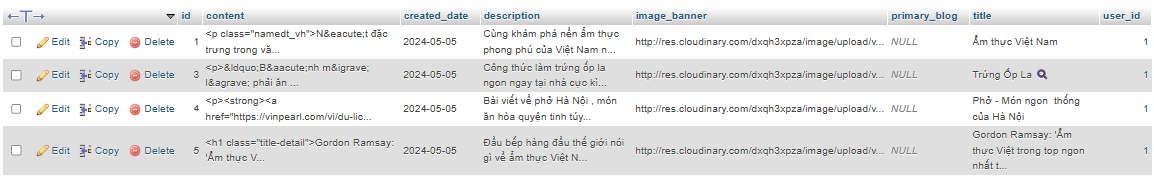


Hình 2.2 : Cơ sở dữ liệu

### 2.4.2. Thiết kế bảng



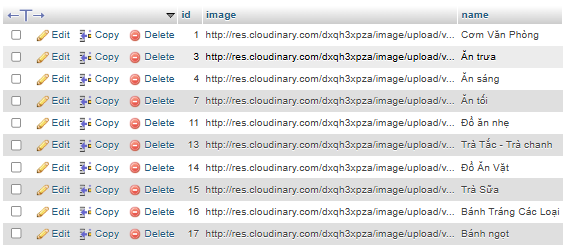
Bảng 2.1 : Bảng authority



Bảng 2.2 : Bảng blog



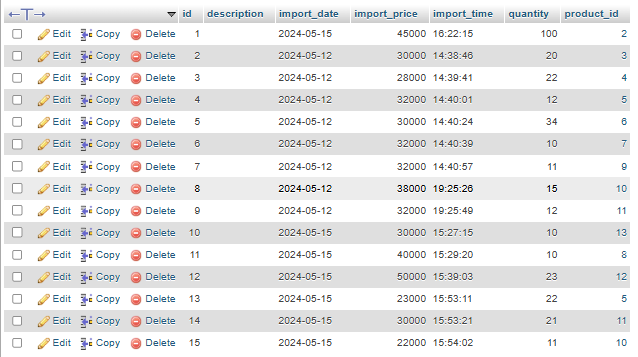
Bảng 2.3 : Bảng Cart



Bảng 2.4 : Bảng Category



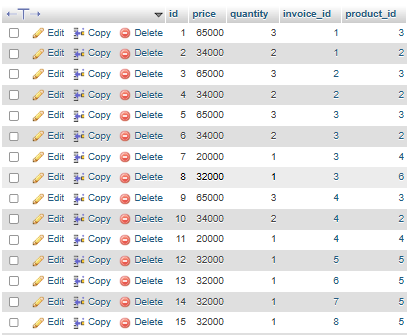
Bảng 2.5: Bảng History pay



Bảng 2.6: Bảng Import Product



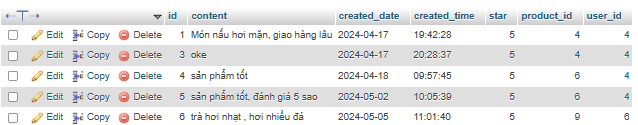
Bảng 2.7: Bảng invoice



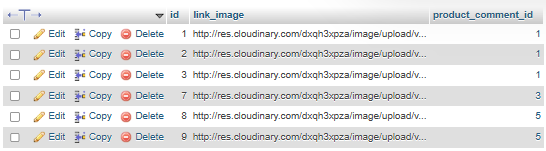
Bảng 2.8: Bảng invoice\_detail



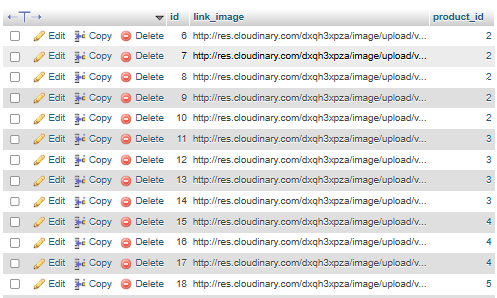
Bảng 2.9: Bảng product



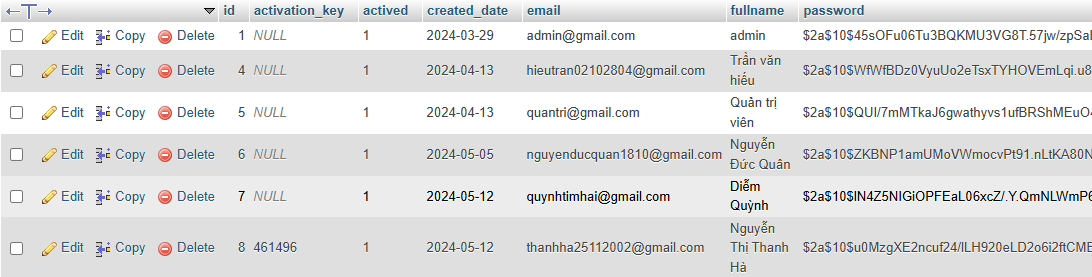
Bảng2.10:BảngProduct\_comment



Bảng 2.11: Bảng product comment image



Bảng 2.12 : Bảng Product\_image



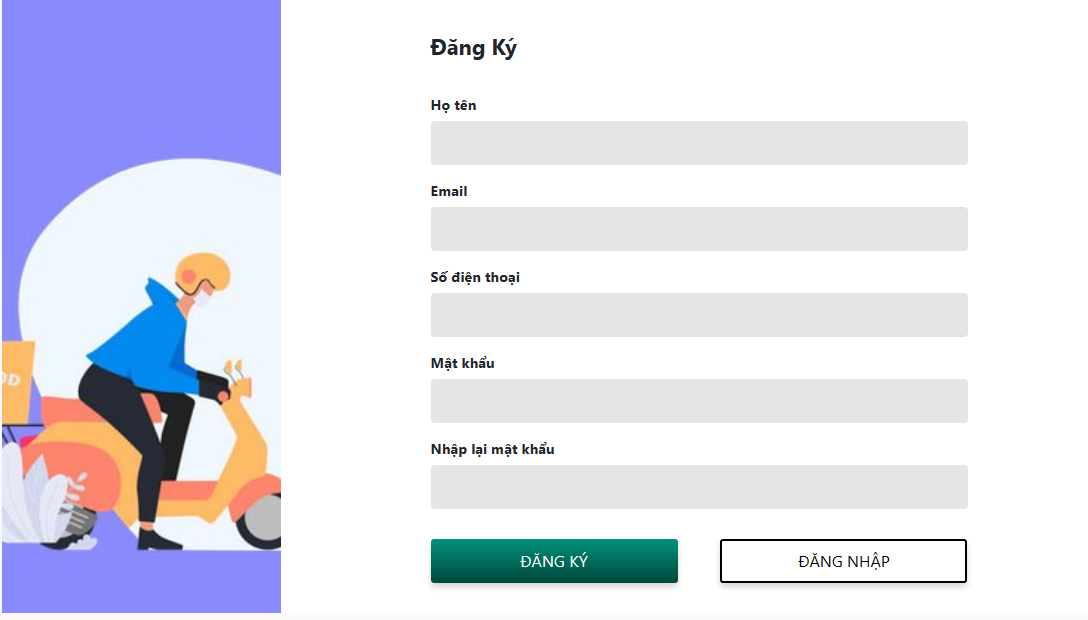
Bảng 2.13 : Bảng users



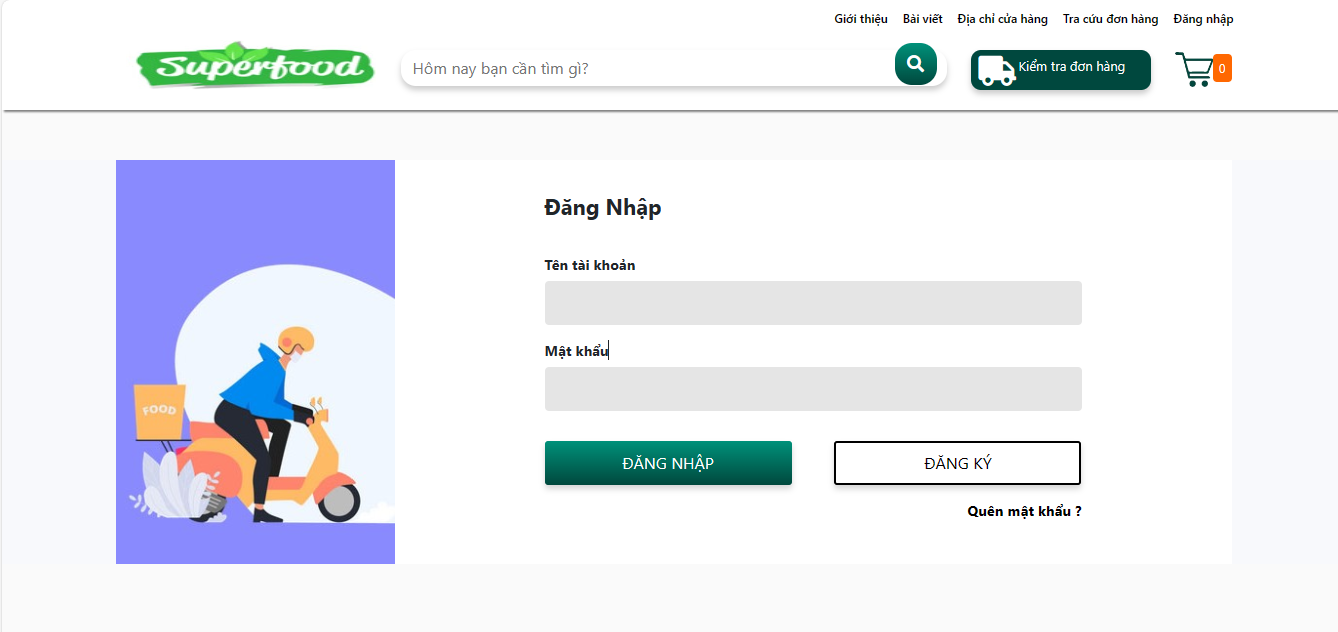
Bảng 2.14 : Bảng voucher

# CHƯƠNG 3 : THIẾT KẾ GIAO DIỆN

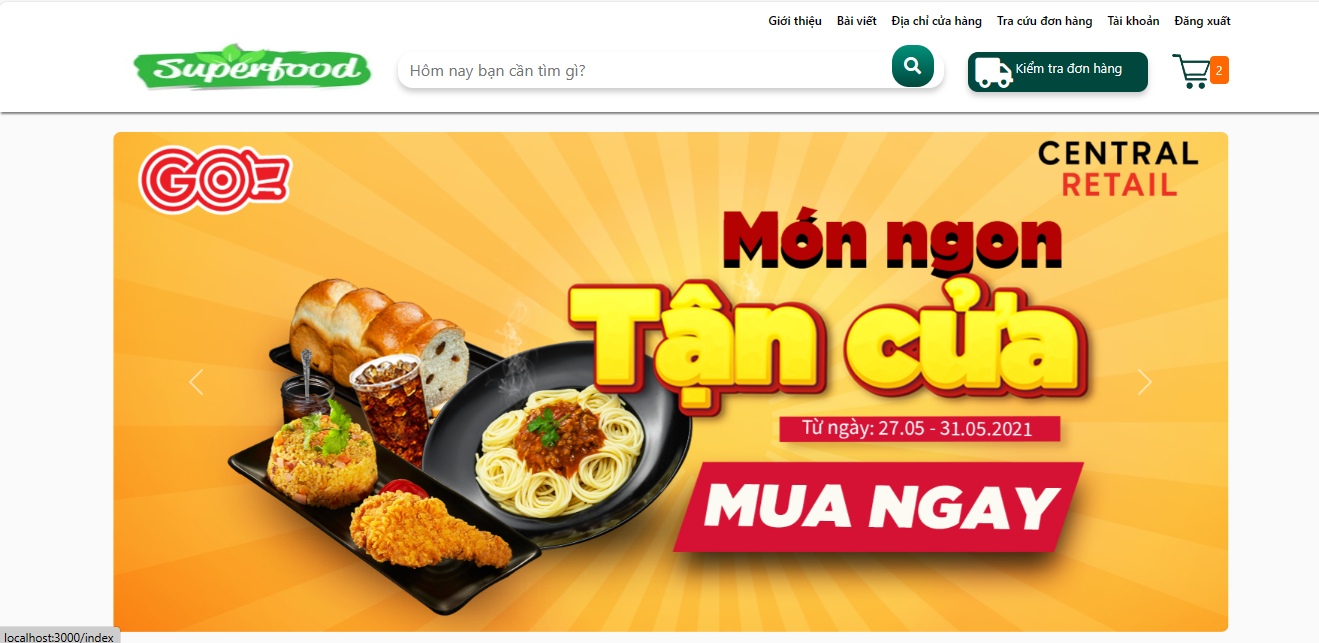
## 3.1. Giao diện người dùng



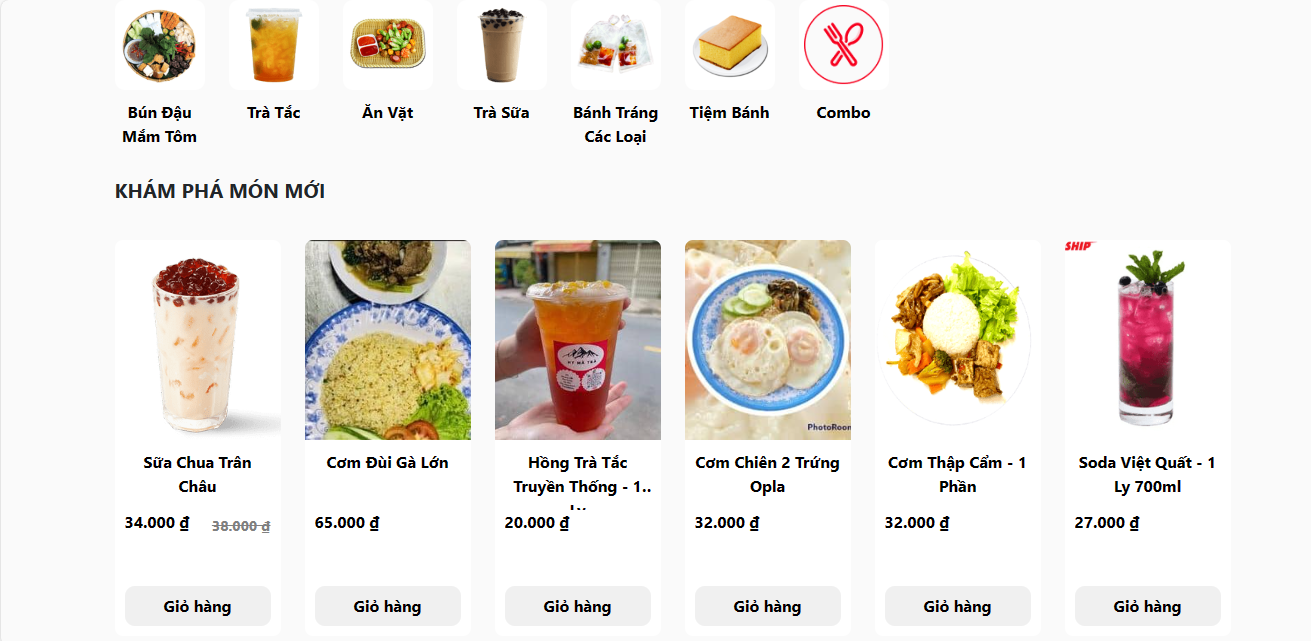
Hình 3.1 : Giao diện đăng kí



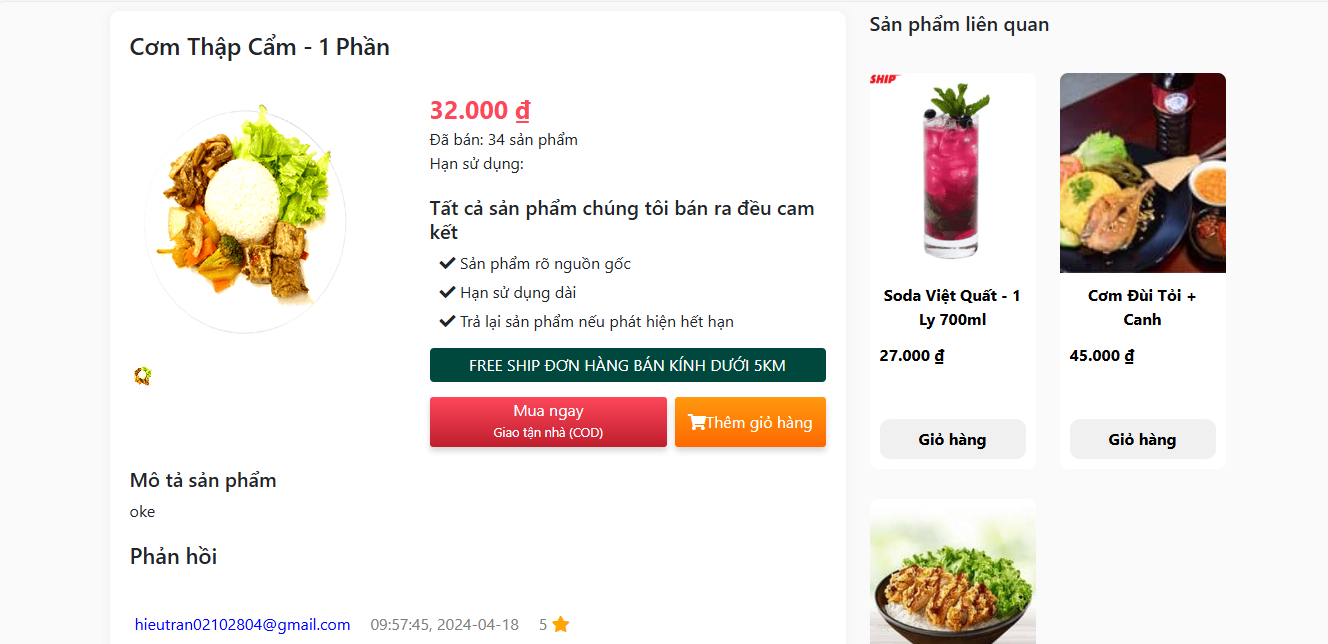
Hình 3.2 : Giao diện đăng nhập



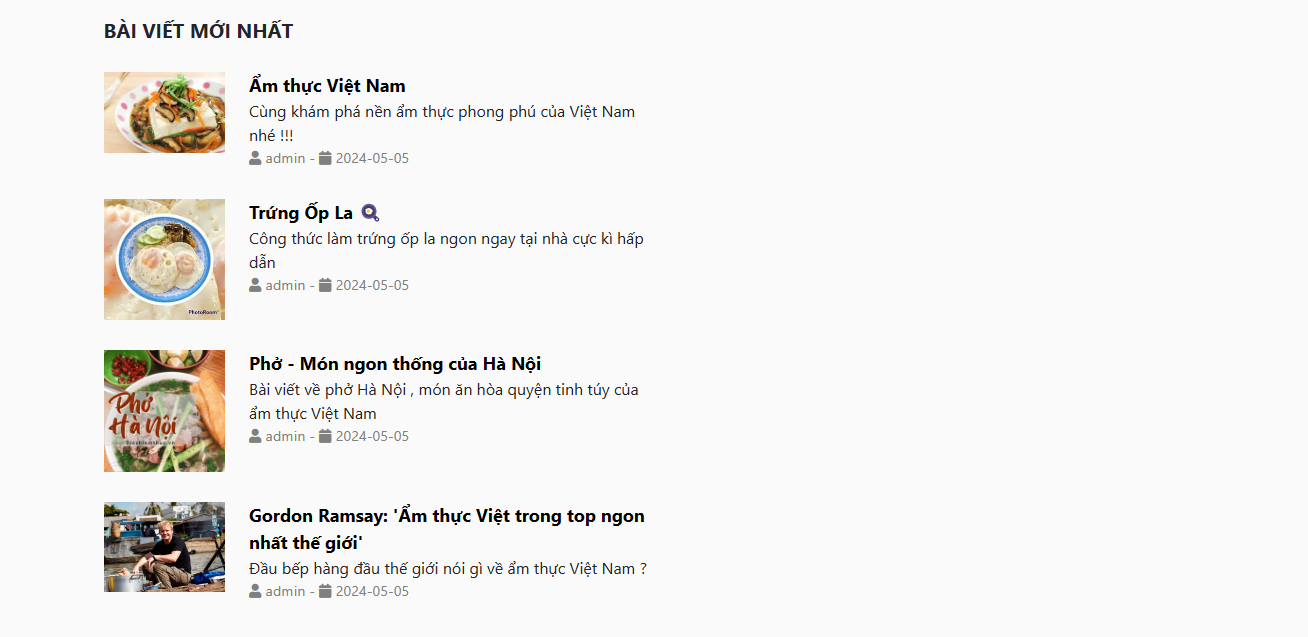
Hình 3.3 : Giao diện trang chủ



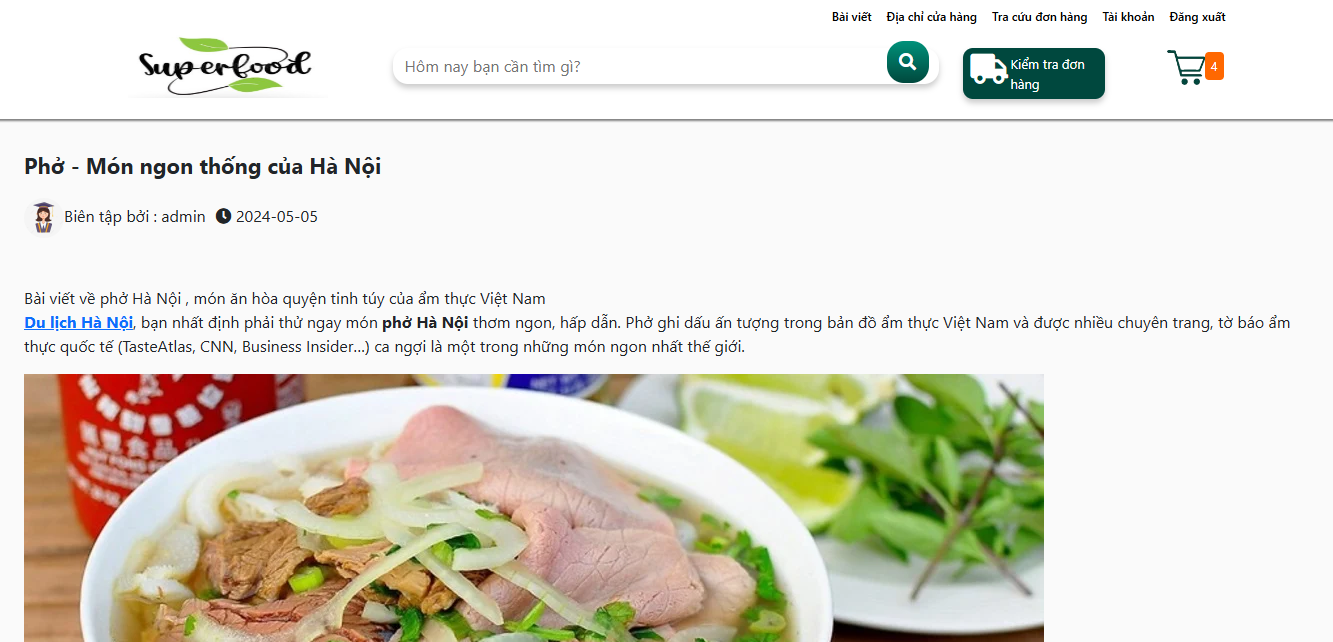
Hình 3.4 : Danh sách món ăn



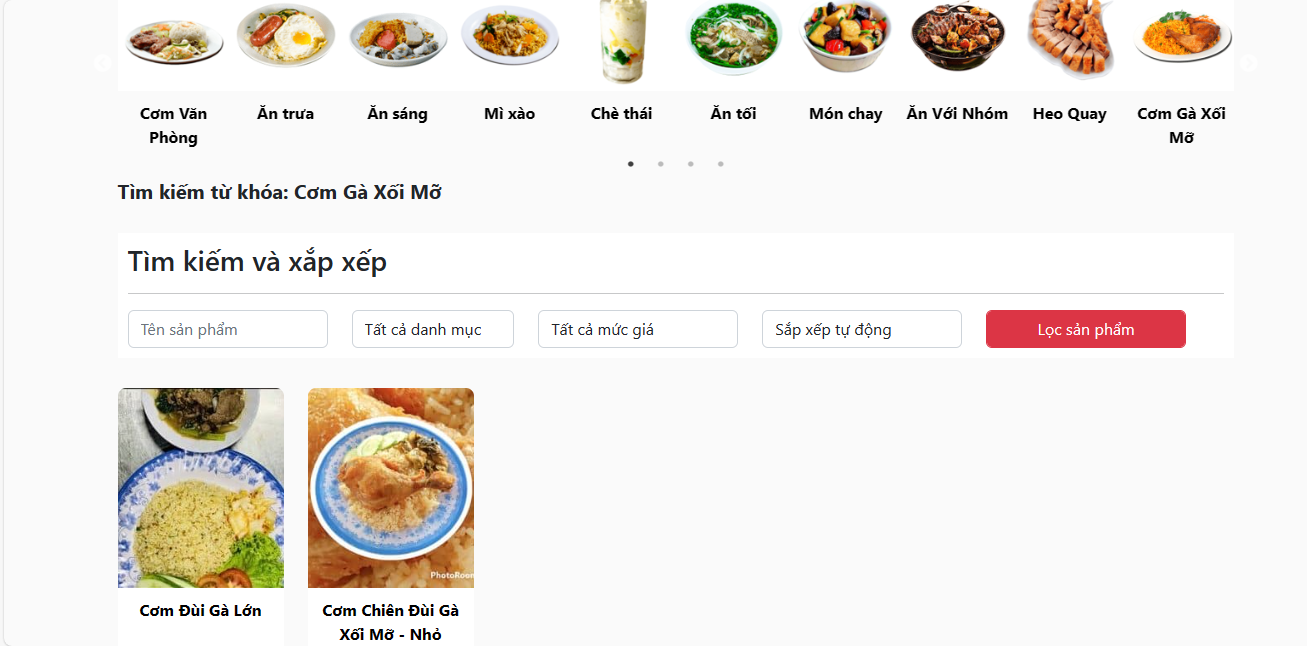
Hình 3.5 : Chi tiết món ăn



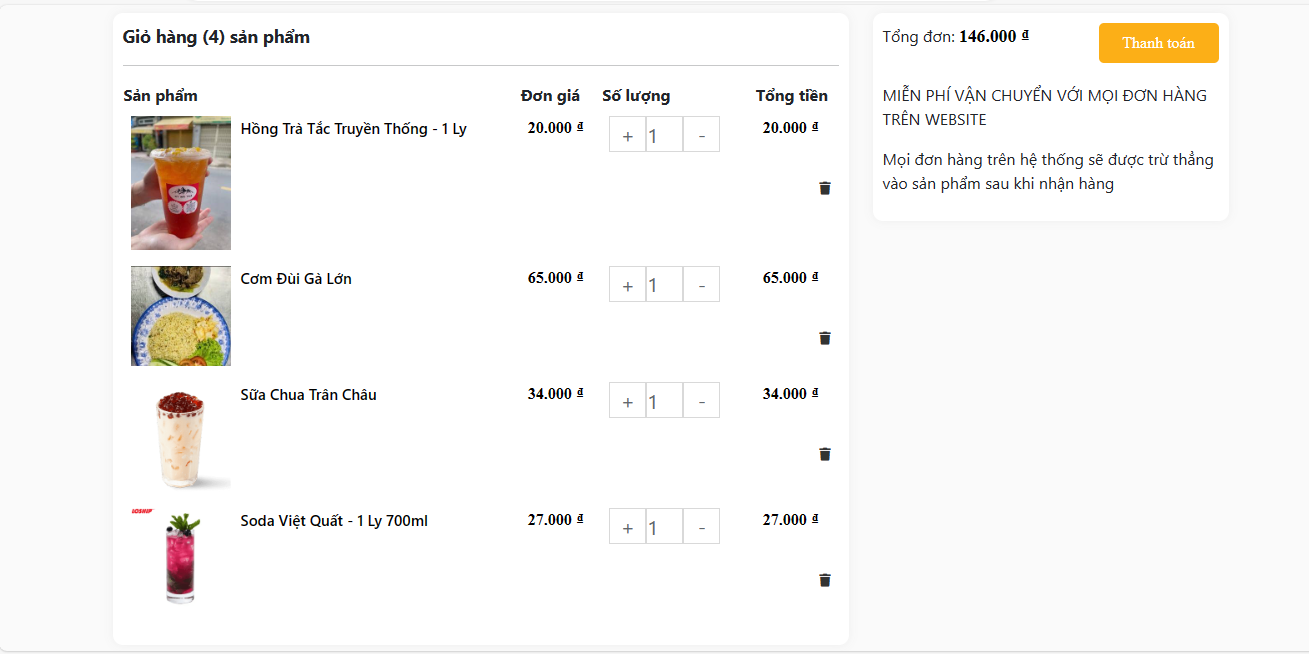
Hình 3.6 : Danh sách các bài viết



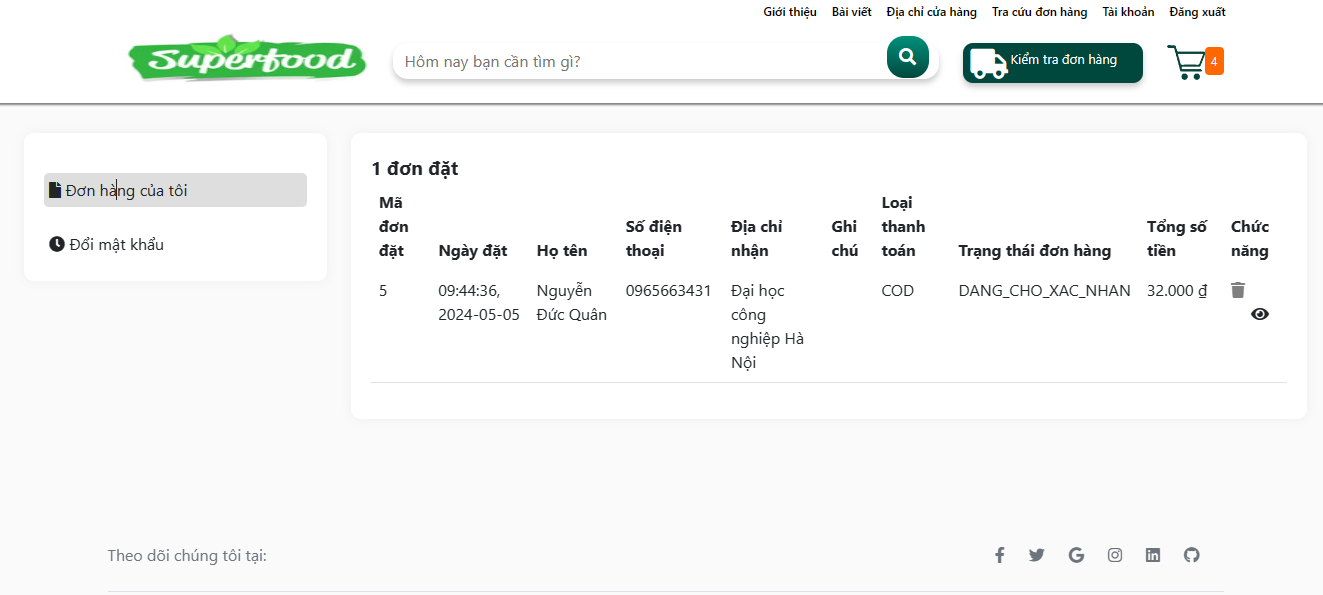
Hình 3.7 : Chi tiết bài viết



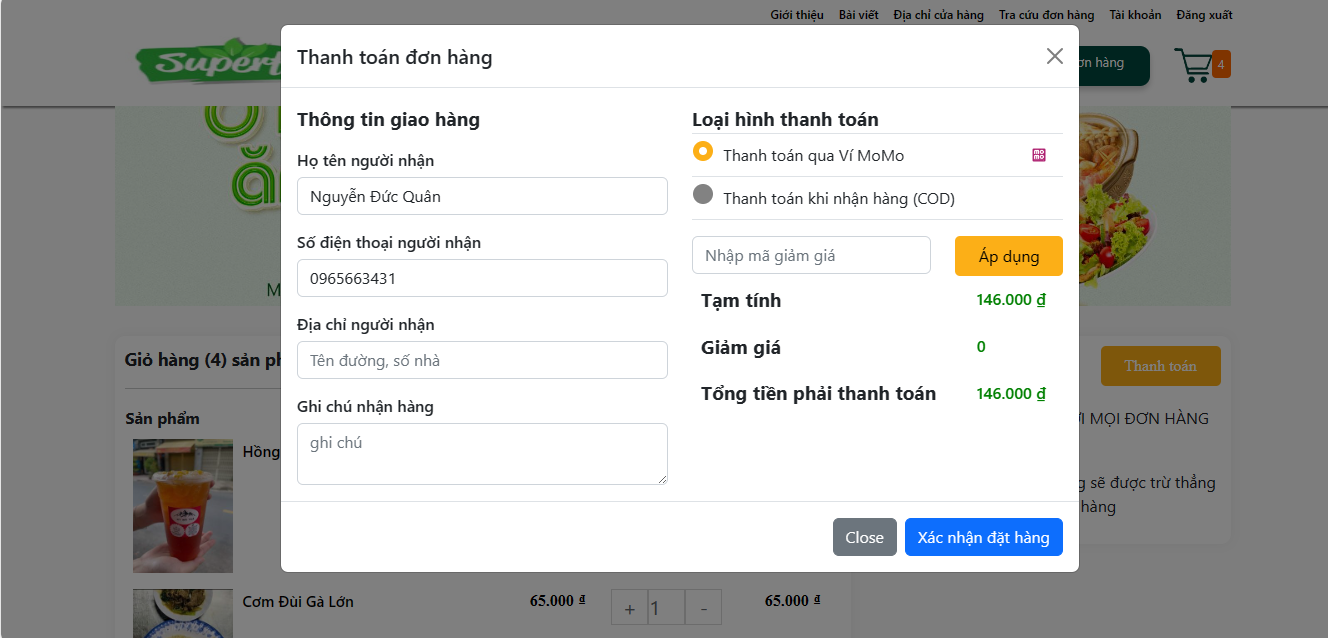
Hình 3.8 : Danh mục các món liên quan



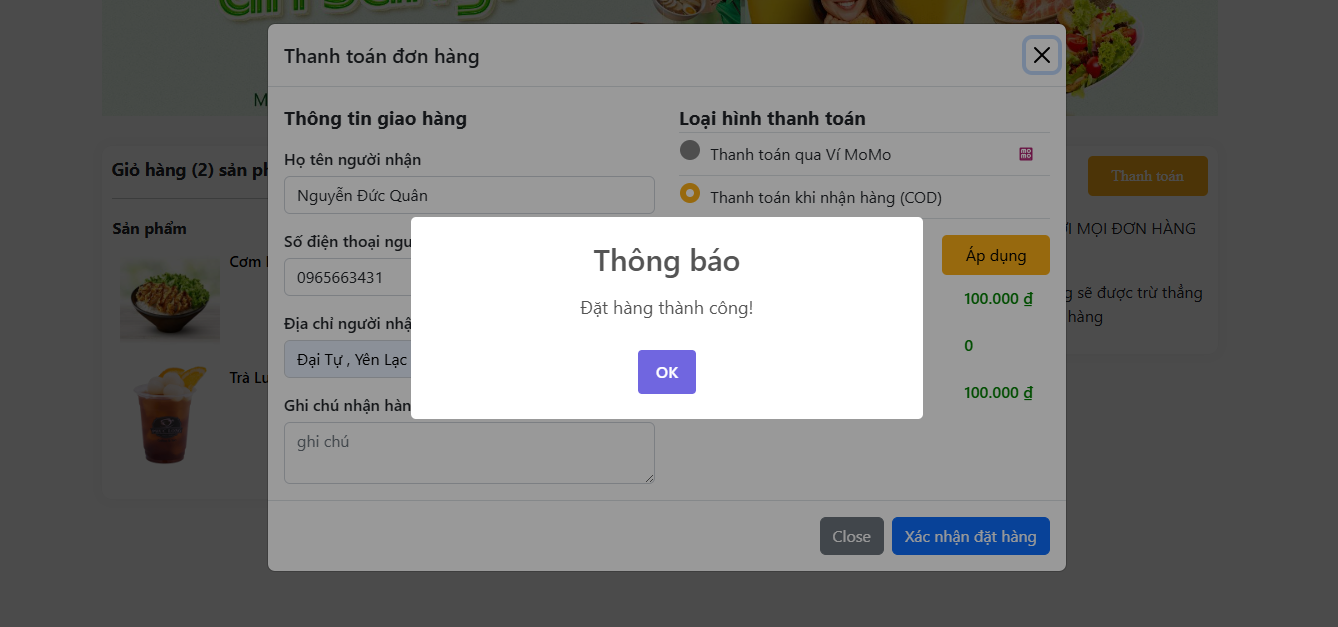
Hình 3.9 : Giao diện giỏ hàng

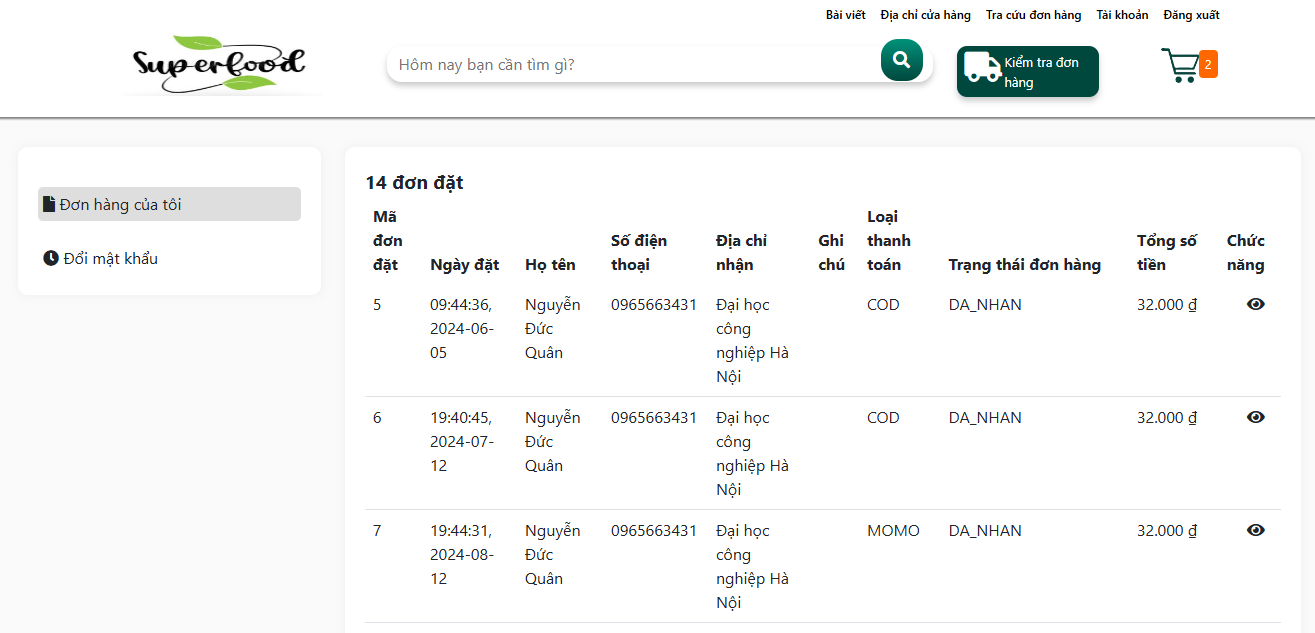


Hình 3.10 : Giao diện quản lí tài khoản

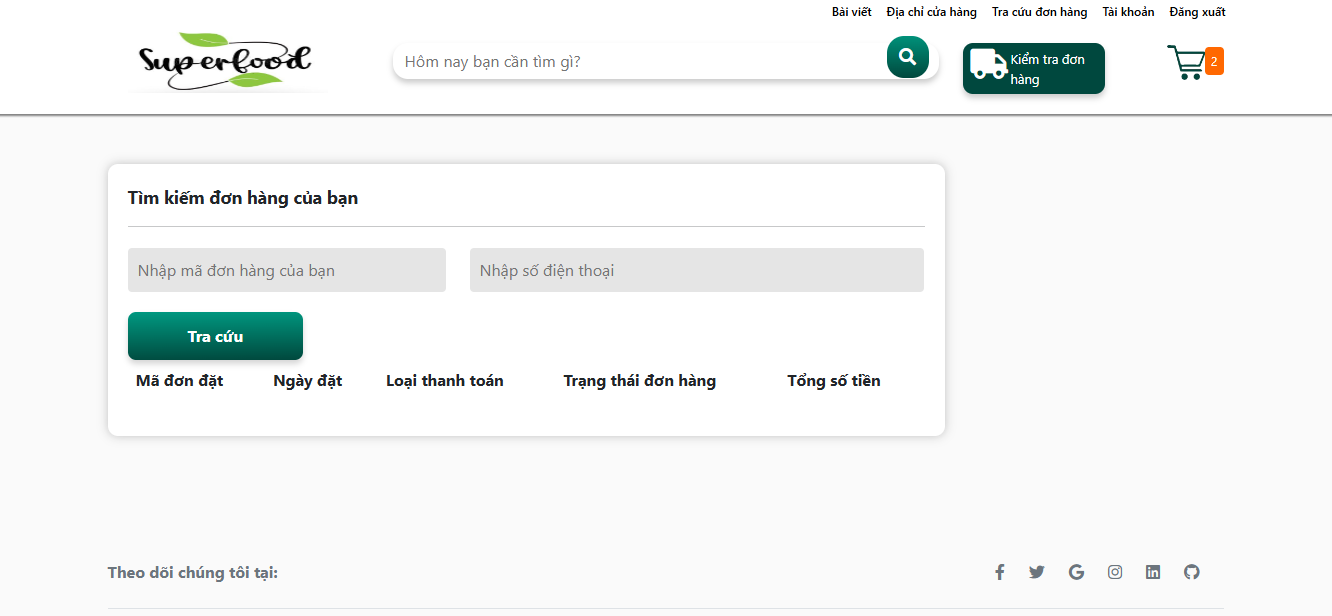


Hình 3.11 : Giao diện thanh toán

  
 Hình 3.12 : Giao diện thanh toán thành công

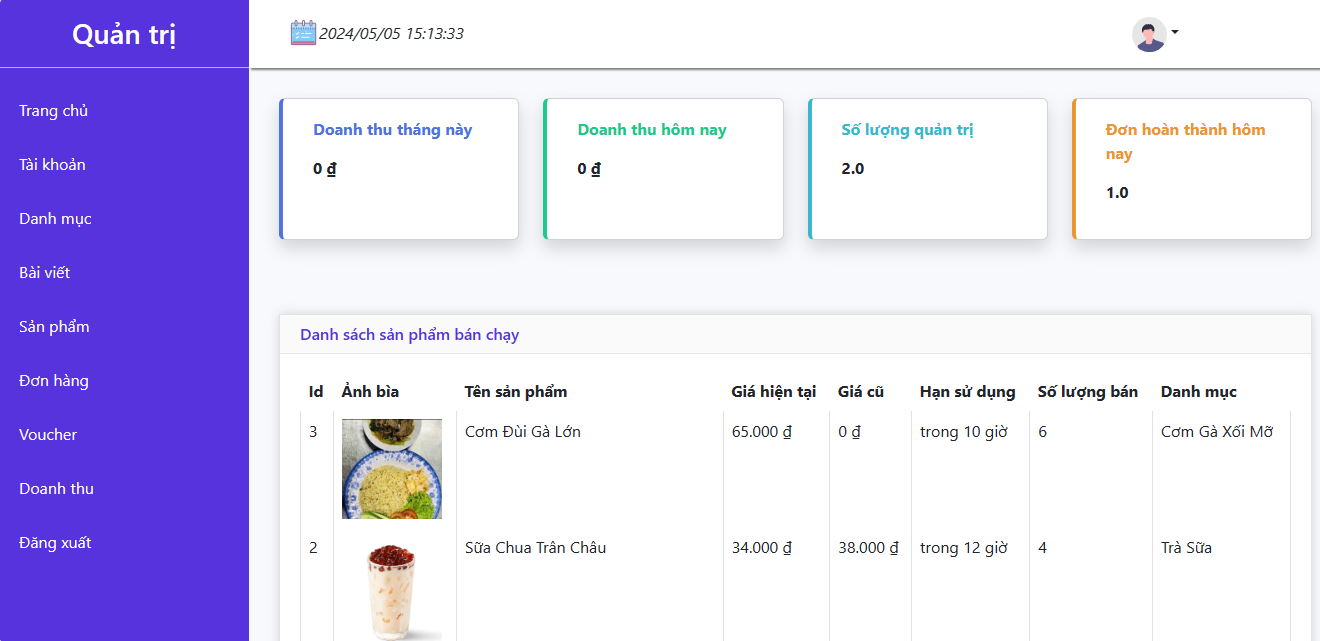


Hình 3.13 : Xem chi tiết tài khoản

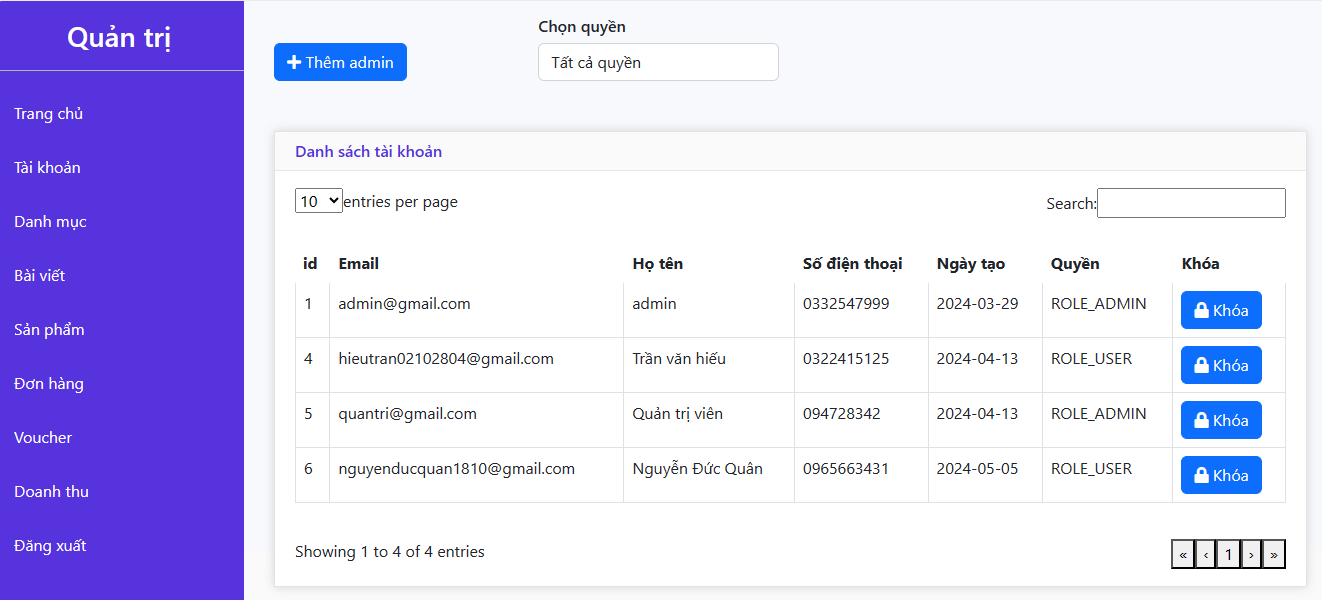


Hình 3.14 : tra cứu thông tin đơn hàng

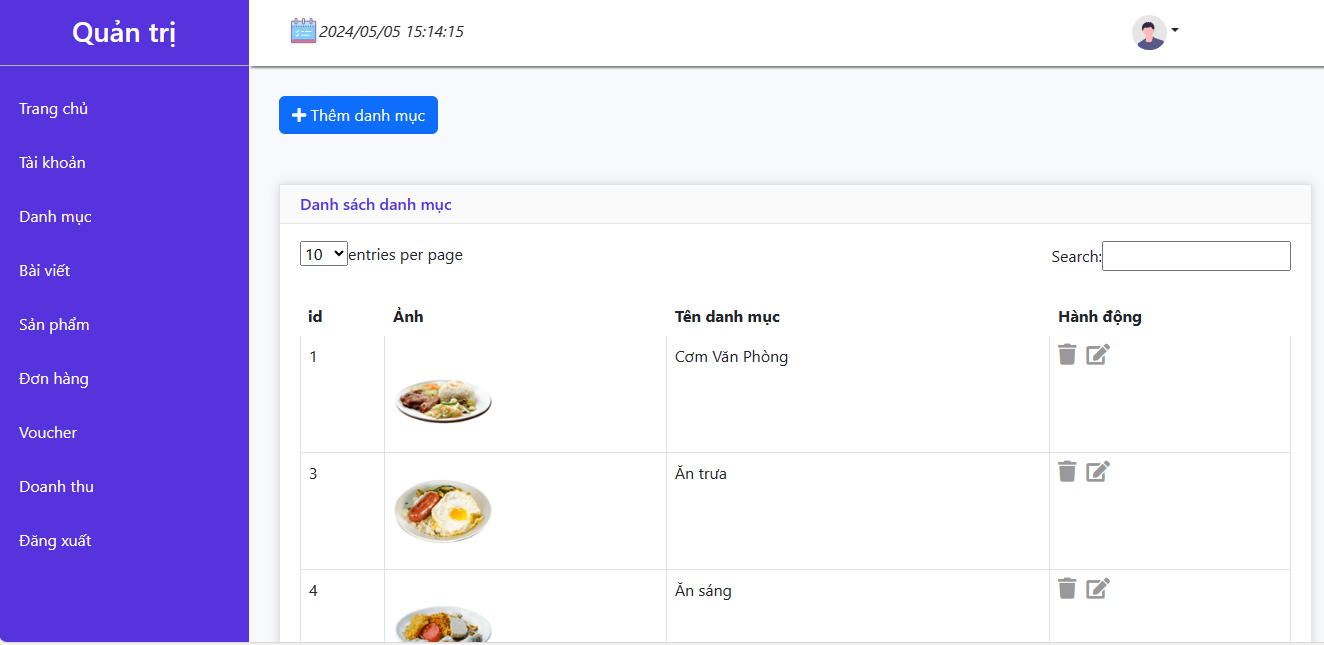
## 3.2 Giao diện Admin



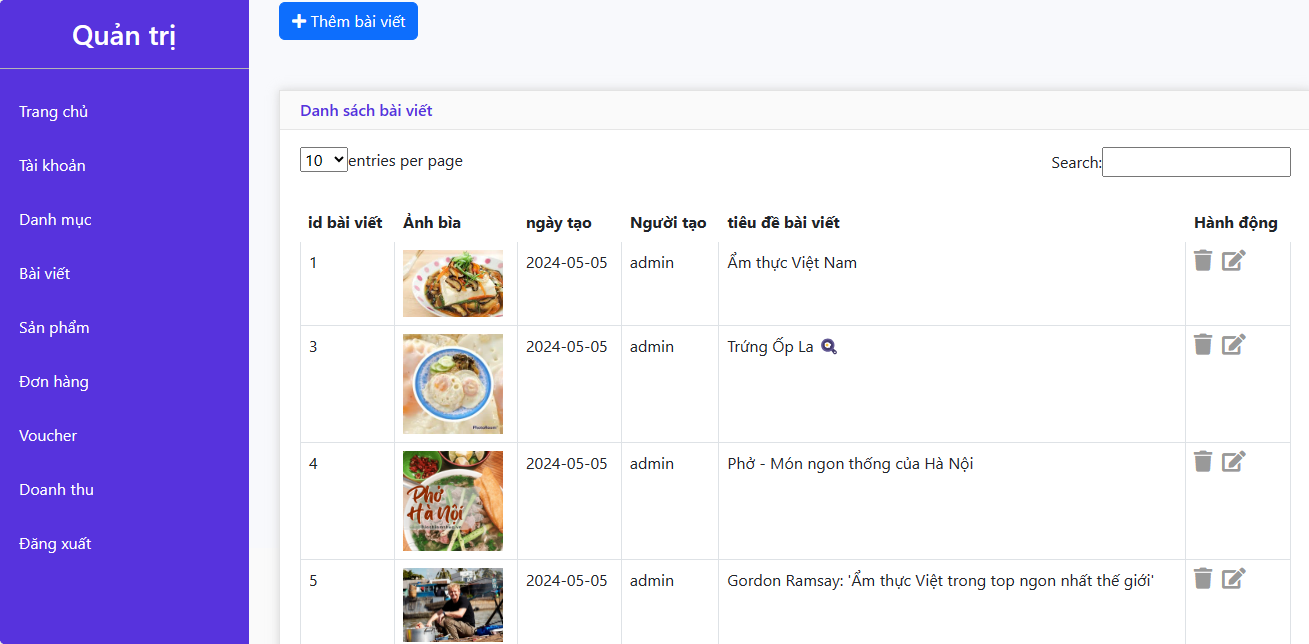
Hình 3.15 : Giao diện Dashboard



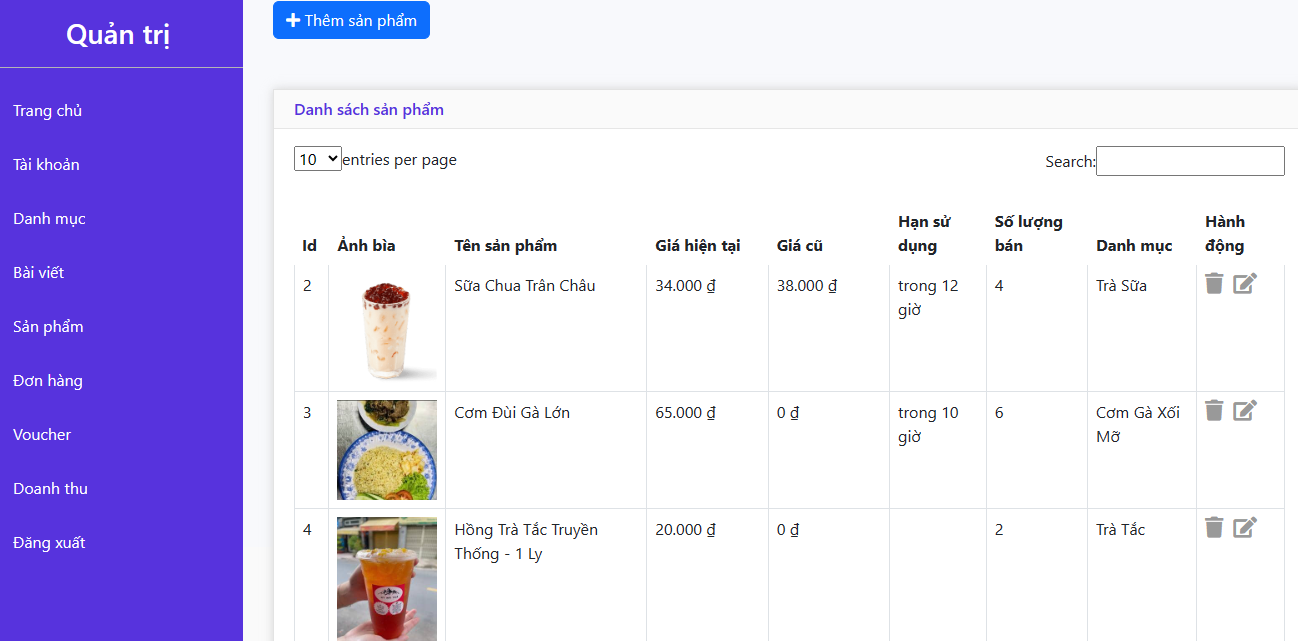
Hình 3.16 : Quản lí tài khoản khách hàng



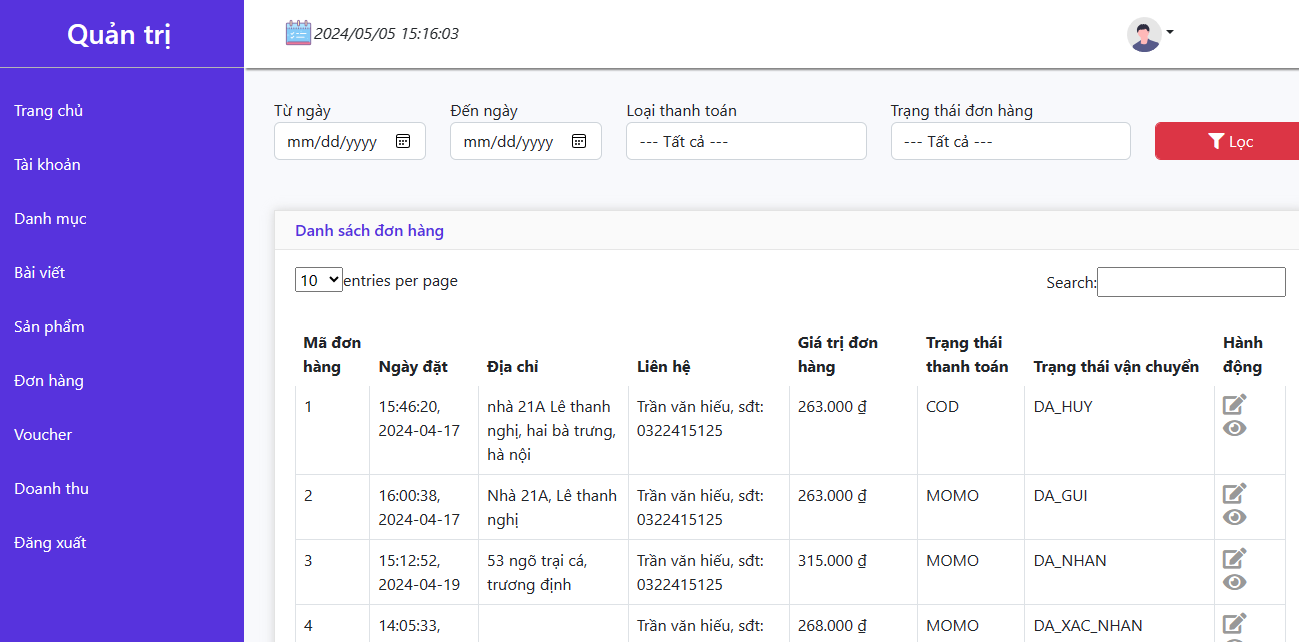
Hình 3.17 : Quản lí danh mục



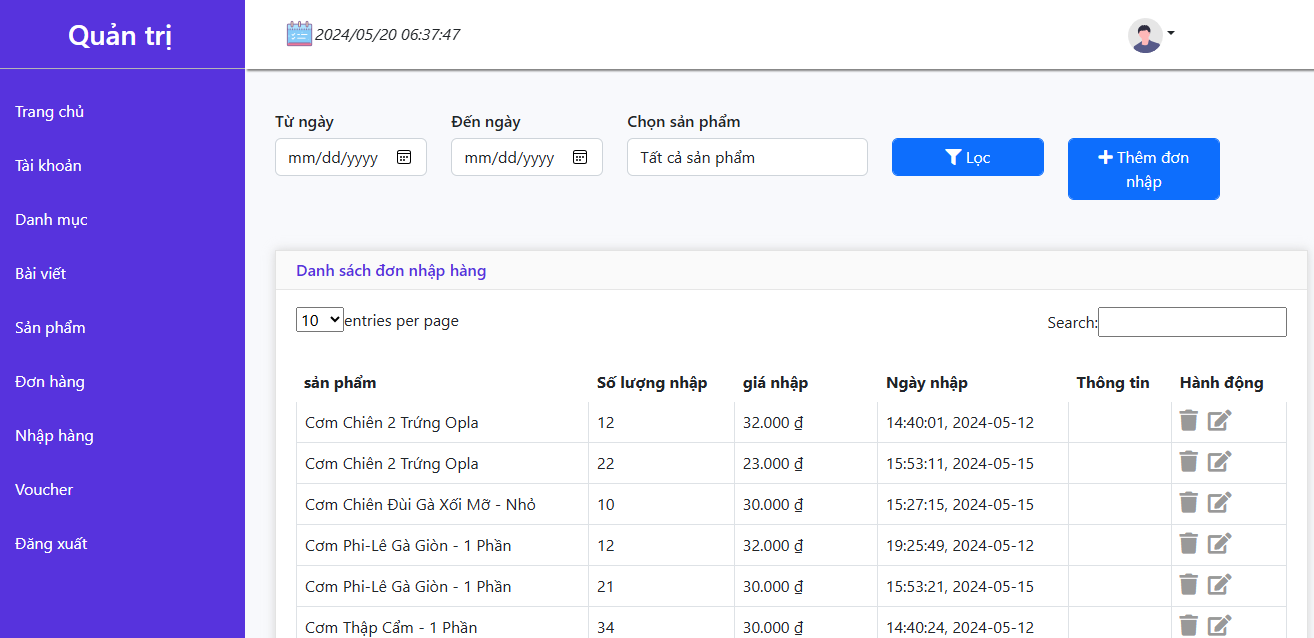
Hình 3.18 : Quản lí bài viết trên trang chủ



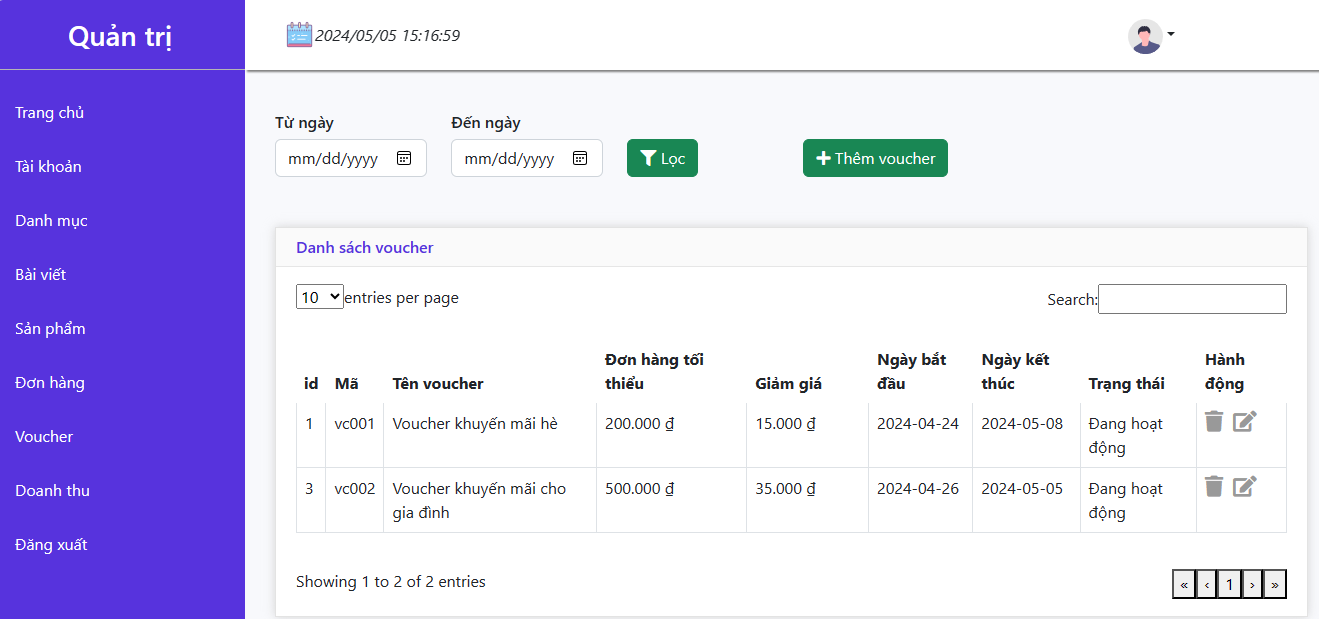
Hình 3.19 : Quản lí sản phẩm



Hình 3.20 : Quản lí đơn hàng



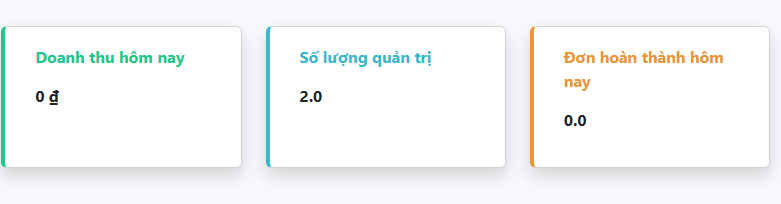
Hình 3.21: Quản lí nhập hàng về kho



Hình 3.22 : Quản lí kho voucher



Hình 3.23 : Quản lí doanh thu theo tháng



Hình 3.24: Quản lí doanh thu theo ngày

# CHƯƠNG 4 : KIỂM THỬ PHẦN MỀM

## 4.1. Kế hoạch kiểm thử

Những chức năng được kiểm thử :

Đăng nhập : Người dùng đăng nhập vào hệ thống.

Đăng xuất: Người dùng đăng xuẩt tài khoản ra khỏi hệ thống

Chức năng của khách hàng :

+ Đăng kí : Người dùng đăng ký tài khoản mới

+ Tìm kiếm : Người dùng tìm kiếm sản phẩm theo tên sản phẩm.

+ Xem chi tiết sản phẩm : người dùng kích vào một sản phẩm để xxem chi tiết

+ Giỏ hàng : Khach hàng quản lí giỏ hàng của mình.

+ Đơn hàng : Khách hàng theo dõi và hủy đơn hàng

+ Quản lí thông tin cá nhân : Khách hàng quản lí thông tin cá nhân.

+ Thanh toán online : Khách hàng thanh toán qua hình thức Momo

Chức năng của admin:

+ Quản lý danh mục: Admin có thể xem, sửa, xóa danh mục

+ Quản lý sản phẩm: Admin có thể xem, sửa, xóa sản phẩm

+ Quản lý đơn hàng: Admin có thể xem và cập nhật trạng thái đơn hàng

+ Quản lý tài khoản khách hàng: Admin có thể xem và xóa tài khoản khách hàng

+ Quản lý bài viết: Admin có thể xem, sửa, xóa bài viết

Chiến lược kiểm thử:

+ Kiểm thử ở mức hệ thống và kiểm thử chấp nhận

+Thiết kế Test Case theo phương pháp phân vùng tương đương

+ Chỉ thực hiện kiểm thử hồi quy, không thực hiện kiểm thử lại.Lịch trình công việc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Mốc công việc | **Sản phẩm** | **Thời gian** | **Bắt đầu** | **Kết thúc** |
| Lập kế hoạch kiểm thử | Test plan | 2 ngày | 01/05/2024 | 02/05/2024 |
| Xem lại các tài liệu | Test plan | 2 ngày | 03/05/2024 | 04/05/2024 |
| Thiết kế các testcase | Test case | 1 ngày | 05/05/2024 | 05/05/2024 |
| Viết các testcase | Test case | 1/2 ngày | 06/05/2024 | 06/05/2024 |
| Xem lại các testcase | Test case | 1/2 ngày | 06/05/2024 | 06/05/2024 |
| Thực thi các testcase | Test case | 1 ngày | 07/05/2024 | 07/05/2024 |
| Ghi nhận và đánh giá kết quả kiểm thử | Test report | 2 ngày | 08/05/2024 | 09/05/2024 |

Các sản phẩm bàn giao:

| STT | Sản phẩm | Ngày bàn giao | Người bàn giao | Người nhận |
| --- | --- | --- | --- | --- |
| 1 | Test plan | 09/05/2024 | Nguyễn Đức Quân | Nguyễn Đức Quân |
| 2 | Test case | 09/05/2024 | Nguyễn Đức Quân | Nguyễn Đức Quân |
| 3 | Test report | 09/05/2024 | Nguyễn Đức Quân | Nguyễn Đức Quân |

## 4.2. Kiểm thử chức năng phía khách hàng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Case** | **Đầu vào** | **Đầu ra mong muốn** | **Kết quả** |
| 1 | Kiểm tra chức năng đăng nhập | Nhập tài khoản & mật khẩu hợp lệ | Đăng nhập thành công và đưa người dùng sang trang chủ. | Pass |
|  |  | Nhập tài khoản & mật khẩu không hợp lệ | Đăng nhập không thành công và thông báo ra màn hình. | Pass |
| 2 | Kiểm tra chức năng đăng ký tài khoản | Đăng ký với thông tin hợp lệ | Đăng ký thành công và chuyển sang trang đăng nhập . | Pass |
|  |  | Đăng ký với thông tin không hợp lệ | Hệ thống thông báo không hợp lệ. | Pass |
| 3 | Kiểm tra chức năng đăng xuất | Đăng xuất tài khoản | Đăng xuất khỏi tài khoản, chuyển sang trang chủ của khách hàng. | Pass |
| 4 | Kiểm tra chức năng tìm kiếm sản phẩm | Nhập từ khóa sản phẩm cần tìm có tồn tại | Hiển thị tất cả các sản phẩm chứa từ khóa vừa nhập. | Pass |
|  |  | Nhập từ khóa sản phẩm cần tìm không tồn tại | Không hiển thị sản phẩm nào. | Pass |
| 5 | Kiểm tra chức năng xem chi tiết sản phẩm | Đã đăng nhập tài khoản | Xem được thông tin chi tiết của sản phẩm | Pass |
|  |  | Chưa đăng nhập tài khoản | Xem được thông tin chi tiết của sản phẩm | Pass |
| 6 | Kiểm tra chức năng thêm vào giỏ hàng | Đã đăng nhập tài khoản | Thêm được sản phẩm vào giỏ hàng | Pass |
|  |  | Chưa đăng nhập tài khoản | Thêm được sản phẩm vào giỏ hàng | Pass |
| 7 | Kiểm tra chức năng xem giỏ hàng | Đã đăng nhập tài khoản | Xem được giỏ hàng | Pass |
|  |  | Chưa đăng nhập tài khoản | Xem được giỏ hàng | Pass |
| 8 | Kiểm tra chức năng đặt hàng | Đã đăng nhập tài khoản | Thông tin người đặt hàng được điền tự động và đặt hàng được | Pass |
|  |  | Chưa đăng nhập tài khoản | Thông tin người dùng phải tự điền và đặt hàng được | Pass |
| 9 | Kiểm tra đơn hàng | Đã đăng nhập tài khoản | Cho phép khách xem được đơn đã đặt và trạng thái của đơn hàng | Pass |
|  |  | Chưa đăng nhập tài khoản | Không hỗ trợ | Pass |
| 10 | Kiểm tra chức năng xem thông tin cá nhân | Đã đăng nhập tài khoản | Xem được thông tin các nhân | Pass |
|  |  | Chưa đăng nhập tài khoản | Không hỗ trợ | Pass |

## 4.3. Kiểm thử chức năng phía Admin

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Case** | **Đầu vào** | **Đầu ra mong muốn** | **Kết quả** |
| 1 | Kiểm tra chức năng thêm, sửa danh mục | Nhập thông tin hợp lệ | Thêm, sửa danh mục thành công và chuyển về màn hình danh sách danh mục | Pass |
| Nhập thông tin không hợp lệ | Báo lỗi tại input có dữ liệu không hợp lệ | Pass |
| 2 | Kiểm tra chức năng xóa danh mục | Đồng ý xóa | Xóa danh mục và nếu danh mục đó có các sản phẩm chứa id danh mục đó thì sẽ được chuyển sang danh mục “Khác” | Pass |
| Không đồng ý xóa | Trở về danh sách danh mục | Pass |
| 3 | Kiểm tra chức năng thêm, sửa sản phẩm | Nhập thông tin hợp lệ | Thêm, sửa sản phẩm thành công và chuyển về màn hình danh sách sản phẩm | Pass |
| Nhập thông tin không hợp lệ | Báo lỗi tại input có dữ liệu không hợp lệ | Pass |
| 4 | Kiểm tra chức năng xóa sản phẩm | Đồng ý xóa | Xóa sản phẩm thành công | Pass |
| Không đồng ý xóa | Trở về danh sách sản phẩm. | Pass |
| 5 | Kiểm tra chức năng tìm kiếm  sản phẩm | Nhập sản phẩm tồn tại trong cơ sở dữ liệu | Hiển thị tất cả sản phẩm có chứa từ khóa theo tên và theo danh mục | Pass |
| Nhập sản phẩm không tồn tại trong cơ sở dữ liệu | Không hiển thị sản phẩm nào | Pass |
| 6 | Kiểm tra chức năng quản lý đơn hàng | Kích vào nút danh sách đơn hàng | Hiển thị danh sách đơn hàng | Pass |
| 7 | Kiểm tra chức năng cập nhật trạng thái đơn hàng | Cập nhật trạng thái đơn hàng(mới, đang xử lý, đã giao hàng,…) | Cập nhật trạng thái đơn hàng thành công | Pass |
| 8 | Kiểm tra chức năng đặt hàng | Đã đăng nhập tài khoản | Thông tin người đặt hàng được điền tự động và đặt hàng được | Pass |
| Chưa đăng nhập tài khoản | Thông tin người dùng phải tự điền và đặt hàng được | Pass |
| 9 | Kiểm tra chức năng thêm, sửa size | Nhập thông tin size | Thêm, sửa size thành công | Pass |
| 10 | Kiểm tra chức năng xóa size | Đồng ý xóa | Xóa size thành công | Pass |
| Không đồng ý xóa | Trở về danh sách size. | Pass |
| 11 | Kiểm tra chức năng xem danh sách khách hàng | Kích vào nút “Danh sách khách hàng” | Hiển thị danh sách khách hàng | Pass |
| 12 | Kiểm tra chức năng xóa khách hàng | Đồng ý xóa | Xóa khách hàng thành công |  |
| Không đồng ý xóa | Trở về danh sách khách hàng. | Pass |

## 4.4. Kết quả kiểm thử

Tỉ lệ test case đạt( passed): 100%

Tỉ lệ test case không đạt (failed): 0%

Hệ thống chạy ổn định trên các trình duyệt web khác nhau (Google Chrome, Microsoft Edge).

# 

# CHƯƠNG 5 : KẾT LUẬN

Qua quá trình làm đồ án tốt nghiệp em đã rút ra cho mình được rất nhiều kinh nghiệm, kiến thức bổ ích phục vụ cho công việc sau khi ra trường. Với đề tài “Xây dựng website bán đồ ăn SupeFood sử dụng Spring boot và ReactJs” em đã học hỏi và tiếp thu được một số kiến thức cũng như kĩ năng như:

Sử dụng được ngôn ngữ Java và nắm được các kiến thức cơ bản về rest api, các kiến thức liên quan đến bắt lỗi, kết nối với cơ sở dữ liệu, kĩ năng debug với Microsoft Visual Studio.

Nắm được các kiến thức về HTML, CSS, Javascript, Jquery, Axios, ReactJs và có thể vận dụng cơ bản vào website để thiết kế giao diện, vẽ biểu đồ, gọi api để lấy dữ liệu từ backend.

Ứng dụng được các câu lệnh trong MySQL để thực hiện truy vấn ra những kết quả cần thiết, để api có thể trả dữ liệu về cho frontend một cách chính xác từ đó hiển thị dữ liệu từ database lên giao diện của website cũng như lưu dữ liệu từ website gửi về cho server.

Nắm bắt được các chức năng cơ bản mà một website cần có và cách thức một website hoạt động.

Do còn thiếu kinh nghiệm nên em chưa hoàn thiện được đầy đủ một ứng

dụng website bán hàng. Trang web cũng đã có những chức năng cơ bản như:

Phía người dùng đã hiển thị được danh sách sản phẩm theo danh mục, có thể xem chi tiết thông tin sản phẩm, có thể thêm sản phẩm vào giỏ hàng, có thể

đăng ký tài khoản, đăng nhập, quản lý tài khoản, có thể đặt hàng, tìm kiếm

sản phẩm, hủy đơn hàng, ... Phía quản trị đã có thể quản lý được sản phẩm, tài khoản, đơn hàng, danh mục, xem báo cáo thống kê, …

Trong thời gian tới em sẽ thêm một số chức năng để trang web được hoàn

thiện hơn như:

Thêm chương trình khuyến mãi, giảm giá cho các sản phẩm, voucher mua sắm khuyến mãi trên website.

Thêm các nội dung cho menu chính như: Liên hệ, câu hỏi thường gặp

Thêm chức năng đánh giá sản phẩm cho người dùng có thể đánh giá chất

lượng sản phẩm.

Nâng cấp chức năng đăng nhập: hiện tại dữ liệu được lưu tạm ở ứng dụng nên khi reload trang web thì sẽ phải đăng nhập lại. Nên em sẽ nâng cấp lên để có thể lưu được thông tin người dùng khi người dùng truy cập mà không phải

đăng nhập lại khi reload.

# TÀI LIỆU THAM KHẢO

[1]. Nguyễn Trung Phú (Chủ biên), Trần Thị Phương Nhung, Đỗ Thị Minh Nguyệt, “Giáo trình thiết kế web”, Nhà xuất bản thống kê - 2019

[2]. Kênh cung cấp tài liệu lập trình trực tuyến: <https://www.w3schools.com/>

[3]. Kênh cung cấp tài liệu lập trình web tiếng Việt: <https://webcoban.vn/>

[4]. Jennifer Niederst Robbins (2014), “Ebook Learning Web Design: A beginner’s guide to HTML, CSS, JavaScript, and Web graphics (Fourth edition)”