|  |  |
| --- | --- |
|  | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------**  ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC  NGÀNH KHOA HỌC MÁY TÍNH  **ĐỀ TÀI: Xây dựng webste bán giày thể thao cho cửa hàng**  **MinhSport**  **CBHD:** ThS. Phạm Thế Anh  **Sinh viên:** Trần Tuấn Minh  **Mã số sinh viên:** 2019605362  Hà Nội – Năm 2024 |

**MỤC LỤC**

# DANH MỤC HÌNH ẢNH

DANH MỤC BẢNG BIỂU

# LỜI CẢM ƠN

# LỜI NÓI ĐẦU

# CHƯƠNG 1: TỔNG QUAN

## Lý do chọn đề tài

Ngày nay nhu cầu mua sắm sản phẩm online đang phát triển mạnh mẽ. Việc xây dựng 1 website để đáp ứng nhu cầu trên là cần thiết vì:

* Sản phẩm có nhu cầu: Giày thể thao là một sản phẩm phổ biến và có nhu cầu cao trong thị trường hiện nay. Việc xây dựng một website bán giày thể thao có thể giúp đáp ứng nhu cầu của khách hàng và tăng doanh số bán hàng cho doanh nghiệp.
* Thị trường tiềm năng: Thị trường giày thể thao đang phát triển mạnh mẽ và tiềm năng với sự tăng trưởng của ngành thể thao và phong cách thời trang đường phố.
* Cạnh tranh: Mặc dù thị trường có tiềm năng nhưng cũng rất cạnh tranh. Việc xây dựng một website bán giày thể thao giúp doanh nghiệp tạo ra sự khác biệt và thu hút được khách hàng với các ưu đãi và dịch vụ tốt.
* Tiết kiệm chi phí: Việc xây dựng một website bán hàng giày thể thao giúp tiết kiệm chi phí so với việc mở cửa hàng bán lẻ truyền thống. Doanh nghiệp có thể giảm chi phí vận hành, quảng cáo, thuê mặt bằng và chi phí nhân sự….

Vì những lý do này, việc xây dựng website bán giày thể thao là một lựa chọn hợp lý cho doanh nghiệp hoạt động trong ngành này.

## Đối tượng, mục tiêu và phạm vi nghiên cứu

### Đối tượng nghiên cứu

- Ngôn ngữ: JavaScript, ReactJS Framework, NodeJS(Express) .

- Công cụ: MongoDBCompass, Postman, Chrome, Visual Studio Code.

- Cơ sở dữ liệu: MongoDB.

### Mục tiêu nghiên cứu

- Phân tích, hiểu bài toán nghiệp vụ thực tế.

- Tìm hiểu công nghệ xây dựng.

- Xây dựng được trang web đáp ứng đầy đủ các yêu cầu đặt ra.

### Phương pháp nghiên cứu

- Phạm vi không gian: công cụ Visual Studio Code, ngôn ngữ JavaScript.

- Phạm vi thời gian: 9 tuần

## Phương pháp nghiên cứu

- Lý thuyết:

Nghiên cứu và tiến hành thu thập các tài liệu có liên quan đến đề tài.

Tổng hợp các tài liệu đã thu thập và tiến hành phân tích.

Chọn lọc để báo cáo.

- Thực nghiệm:

Phân tích các trang web có chức năng tương tự.

Phân tích, tiến hành xây dựng theo phân tích và yêu cầu thực tế.

Tìm hiểu cách thức, tiêu chí kiểm thử giao diện và chức năng.

## 1.4 Công nghệ sử dụng

## 1.4.1. Ngôn ngữ JavaScript

* *Giới thiệu chung về JavaScript*

Đây là ngôn ngữ lập trình kịch bản, chạy trên trình duyệt, chuyên dùng để xây dựng phần front-end của trang web, hiện nay JavaScript đã xuất hiện hầu như tại tất cả các trang web đang hoạt động cùng với những Frameworks của nó.

JavaScript cung cấp DOM API (bộ công cụ dùng để tương tác với thẻ HTML trên web). Với DOM API các lập trình viên có thể thực hiện được rất nhiều thao tác để xử lý logic cho người dùng, hay thực hiện các thao tác để thay đổi chỉnh sửa giao diện của website trong quá trình hoạt động.

Về tổng quan, JavaScript cũng có các thành phần giống như PHP đang có (hướng đối tượng, hướng chức năng, biến, mảng, hàm, vòng lặp, …). Để chạy được JavaScript chỉ cần khai báo thẻ script trong trang HTML và viết các đoạn lệnh của JavaScript là có thể chạy được hoặc với những script lớn, ta có thể thực hiện riêng ra một có đuôi “.js” sau đó thực hiện nhúng tệp vào trang HTML bằng đoạn mã lệnh <script src=” đường dẫn tệp cần nhúng ” /> là có thể thực hiện được.

* *Một vài cập nhật mới cho phiên bản ES6*

Mỗi phiên bản cập nhật, JavaScript lại cung cấp cho người dùng thêm nhiều tính năng mới. Ví dụ: tại các phiên bản gần đây JavaScript cung cấp tính năng module, giúp cho việc tái sử dụng code trở nên dễ dàng hơn, hay là nội suy, với tính năng này việc lấy dữ liệu từ các biến trong JS để nhúng vào một chuỗi cũng trở nên dễ dàng hơn (thay vì phải nối chuỗi).

* *JavaScript trong thực tiễn*

Trên thực tế, hiện nay các website hiện đại đang dần được thay thế và phát triển bởi các JavaScript Framework như là: ReactJS, VueJS, AngularJS với mục đích tạo ra các trang web có hiệu suất xử lý nhanh hơn nhờ tương tác với DOM ảo và nguyên lý ứng dụng đơn trang (chỉ cập nhật phần nào thay đổi).

## 1.4.2. ReactJS Framework

ReactJS là một thư viện JavaScript chuyên giúp các nhà phát triển xây dựng giao diện người dùng hay UI. Trong lập trình ứng dụng front-end, lập trình viên thường sẽ phải làm việc chính trên 2 thành phần sau: UI và xử lý tương tác của người dùng. UI là tập hợp những thành phần mà bạn nhìn thấy được trên bất kỳ một ứng dụng nào, ví dụ có thể kể đến bao gồm: menu, thanh tìm kiếm, những nút nhấn, card,… Giả sử bạn đang lập trình một website thương mại điện tử, sau khi người dùng chọn được sản phẩm ưng ý rồi và nhấn vào nút “Thêm vào giỏ hàng”, thì việc tiếp theo mà bạn phải làm đó là thêm sản phẩm được chọn vào giỏ hàng và hiển thị lại sản phẩm đó khi user vào xem => xử lý tương tác.

Trước khi có ReactJS, lập trình viên thường gặp rất nhiều khó khăn trong việc sử dụng “vanilla JavaScript” (JavaScript thuần) và JQuery để xây dựng UI. Điều đó đồng nghĩa với việc quá trình phát triển ứng dụng sẽ lâu hơn và xuất hiện nhiều bug, rủi ro hơn. Vì vậy vào năm 2011, Jordan Walke – một nhân viên của Facebook đã khởi tạo ReactJS với mục đích chính là cải thiện quá trình phát triển UI.

Hơn nữa, để tăng tốc quá trình phát triển và giảm thiểu những rủi ro có thể xảy ra trong khi coding, React còn cung cấp cho chúng ta khả năng Reusable Code (tái sử dụng code) bằng cách đưa ra 2 khái niệm quan trọng bao gồm:

* JSX.
* Virtual DOM.

## 1.4.3. Nền tảng NodeJS

NodeJs là một môi trường runtime chạy Javascript đa nền tảng và có mã nguồn mở, được sử dụng để chạy các ứng dụng web bên ngoài trình duyệt của client. Nền tảng này được phát triển bởi Ryan Dahl vào năm 2009, được xem là một giải pháp hoàn hảo cho các ứng dụng sử dụng nhiều dữ liệu nhờ vào mô hình hướng sự kiện (event-driven) không đồng bộ.

## 1.4.4. Cơ sở dữ liệu

**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. **MySQL** được **tích hợp apache**, **PHP** và quản lý dữ liệu thông qua các cơ sở dữ liệu. Mỗi cơ sở dữ liệu có thể có nhiều bảng quan hệ chứa dữ liệu. **MySQL** cũng có cùng một cách truy xuất và mã lệnh tương tự với ngôn ngữ SQL.

* Ưu điểm:

- Dễ sử dụng: **MySQL** là cơ sở dữ liệu tốc độ cao, ổn định, dễ sử dụng và hoạt động trên nhiều hệ điều hành, cung cấp một hệ thống lớn các hàm tiện ích rất mạnh.

- Độ bảo mật cao:  **MySQL** rất thích hợp cho các ứng dụng có truy cập CSDL trên Internet khi sở hữu nhiều nhiều tính năng bảo mật thậm chí là ở cấp cao.

- Đa tính năng: **MySQL** hỗ trợ rất nhiều **chức năng SQL** được mong chờ từ một hệ quản trị cơ sở dữ liệu quan hệ cả trực tiếp lẫn gián tiếp.

- Khả năng mở rộng và mạnh mẽ: **MySQL** có thể xử lý rất nhiều dữ liệu và hơn thế nữa nó có thể được mở rộng nếu cần thiết.

- Nhanh chóng: Việc đưa ra một số tiêu chuẩn cho phép MySQL để làm việc rất hiệu quả và tiết kiệm chi phí, do đó nó làm tăng tốc độ thực thi.

* Nhược điểm:

- Giới hạn: Theo thiết kế, **MySQL** không có ý định làm tất cả và nó đi kèm với các hạn chế về chức năng mà một vài ứng dụng có thể cần.

- Độ tin cậy: Cách các chức năng cụ thể được xử lý với **MySQL** (ví dụ tài liệu tham khảo, các giao dịch, kiểm toán,…) làm cho nó kém tin cậy hơn so với một số hệ quản trị cơ sở dữ liệu quan hệ khác.

- Dung lượng hạn chế: Nếu số bản ghi lớn dần lên thì việc truy xuất dữ liệu khá khó khăn, khi đó chúng ta sẽ phải áp dụng nhiều biện pháp để tăng tốc độ truy xuất dữ liệu như là chia tải database này ra nhiều server, hoặc **tạo cache MySQL.**

## 1.4.5. JSON Web Token (JWT)

JSON Web Token (JWT) là một chuẩn mở (RFC 7519) định nghĩa một cách nhỏ gọn và khép kín để truyền một cách an toàn thông tin giữa các bên dưới dạng đối tượng JSON. Thông tin này có thể được xác minh và đáng tin cậy vì nó có chứa chữ ký số. JWTs có thể được ký bằng một thuật toán bí mật (với thuật toán HMAC) hoặc một public / private key sử dụng mã hoá RSA.

Thành phần chính của JWT:



Hình 2. 1 Thành phần của JWT

*Header*

Gồm hai phần chính: loại token (mặc định là JWT - Thông tin này cho biết đây là một Token JWT) và thuật toán đã dùng để mã hóa (HMAC SHA256 - HS256 hoặc RSA).

Ví dụ:

{

    "typ": "JWT",

    "alg": "HS256"

}

*Payload*

Payload sẽ chứa các thông tin đặt trong chuỗi Token như issuer, issued\_at, expiration\_time, subject, audience, …

Ví dụ:

{

    "iss": "jira:1314039",

    "iat": 1300819370,

    "exp": 1300819380,

    "qsh": "8063ff4ca1e41df7bc90c8ab6d0f6207d491cf6dad7c66ea797b4614b71922e9",

    "sub": "batman",

    "context": {

        "user": {

            "userKey": "batman",

            "username": "bwayne",

            "displayName": "Bruce Wayne"

        }

    }

}

**Public Claims** - Claims được cộng đồng công nhận và sử dụng rộng rãi.

**Private Claims** - Claims tự định nghĩa (không được trùng với Reserved Claims và Public Claims), được tạo ra để chia sẻ thông tin giữa 2 parties đã thỏa thuận và thống nhất trước đó.

*Signature*

Chữ ký Signature trong JWT là một chuỗi được mã hóa bởi header, payload cùng với một chuỗi bí mật theo nguyên tắc sau:

HMACSHA256(

  base64UrlEncode(header) + "." +

  base64UrlEncode(payload),

  secret)

Do bản thân Signature đã bao gồm cả header và payload nên Signature có thể dùng để kiểm tra tính toàn vẹn của dữ liệu khi truyền tải.

# CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## 2.1. Phân tích yêu cầu

### 2.1.1. Yêu cầu chức năng

* Đăng nhập
* Đăng ký
* Tìm kiếm
* Xem giỏ hàng
* Đánh giá sản phẩm
* Xem chi tiết sản phẩm
* Xem sản phẩm theo danh mục
* Liên hệ
* Xem sản phẩm
* Thêm vào giỏ hàng
* Đặt hàng
* Quản lý sản phẩm
* Quản lý danh mục
* Quản lý blog
* Quản lý đơn hàng

### 2.1.2. Yêu cầu phi chức năng

### Hiệu suất:

* + Hệ thống cần đảm bảo khả năng xử lý lớn và đồng thời của dữ liệu để đảm bảo trải nghiệm người dùng mượt mà và không bị gián đoạn.
  + Thời gian phản hồi của hệ thống cần được giữ ở mức thấp, đảm bảo việc truy cập và sử dụng website luôn nhanh chóng và hiệu quả.

### Bảo mật:

* + Hệ thống cần có các biện pháp bảo mật mạnh mẽ để bảo vệ thông tin cá nhân của người dùng, bao gồm mã hóa dữ liệu, kiểm soát truy cập và cơ chế xác thực..
  + Phải có cơ chế sao lưu và khôi phục dữ liệu định kỳ để đảm bảo an toàn và tin cậy cho dữ liệu.

### Tương thích:

* + Website cần phải tương thích với nhiều loại trình duyệt web phổ biến như Chrome, Firefox, Safari và Edge để đảm bảo mọi người dùng có thể truy cập và sử dụng một cách thuận tiện.
  + Giao diện của website cũng cần phải tương thích với các thiết bị di động như điện thoại thông minh và máy tính bảng.

### Dễ dàng điều hướng:

* + Giao diện người dùng cần phải được thiết kế sao cho dễ dàng điều hướng và tìm kiếm thông tin, giúp người dùng tìm được thông tin một cách nhanh chóng và thuận tiện.

### Dễ dàng sử dụng:

* + Giao diện người dùng cần phải được thiết kế đơn giản và dễ sử dụng, giúp người dùng mới dễ dàng tiếp cận và sử dụng hệ thống một cách tự nhiên.

### Bảo Mật Thông Tin:

* + Đảm bảo an toàn và bảo mật cho thông tin cá nhân của người tìm việc và nhà tuyển dụng.
  + Áp dụng các biện pháp bảo mật để ngăn chặn sự truy cập trái phép vào hệ thống và lợi dụng thông tin.

### Giao Diện Thân Thiện:

* + Xây dựng giao diện dễ sử dụng và thân thiện với người dùng, bao gồm cả trên máy tính và thiết bị di động.
  + Đảm bảo trải nghiệm người dùng mượt mà và hiệu quả.

## 2.3. Thiết kế hệ thống

### 2.3.1. Biểu đồ usecase



Hình 3.1 Biểu đồ usecase hệ thống

### 2.3.2. Mô tả usecase

#### 2.3.2.1. Mô tả usecase Đăng nhập

Mô tả vắn tắt: Ca sử dụng này cho phép người dùng đăng nhập để sử dụng hệ thống.

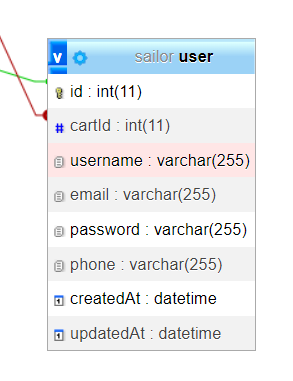
* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. UC bắt đầu khi người dùng truy cập hệ thống.
2. Hệ thống hiển thị màn hình đăng nhập yêu cầu nhập thông tin gồm tên đăng nhập và mật khẩu.
3. Người dùng nhập tên đăng nhập, mật khẩu và kích nút “Đăng nhập”.
4. Hệ thống truy cập bảng USER, kiểm tra thông tin và hiển thị màn hình tương ứng với quyền của người dùng. UC kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bước 3 ở luồng cơ bản, nếu thông tin người dùng nhập không hợp lệ, hệ thống sẽ hiển thị thông báo lỗi. UC kết thúc.
2. Tại bước 3 ở luồng cơ bản, nếu không tìm thấy thông tin người dùng, hệ thống sẽ hiển thị thông báo lỗi. UC kết thúc.
3. Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

* **Các yêu cầu đặc biệt:** Không
* **Tiền điều kiện:** Không
* **Hậu điều kiện:** Không
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**



Hình 3.2 Dữ liệu liên quan usecase Đăng nhập

#### 2.3.2.2. Mô tả usecase Đăng ký

Mô tả vắn tắt: Ca sử dụng này cho người dùng đăng ký tài khoản của shop để đặt hàng.

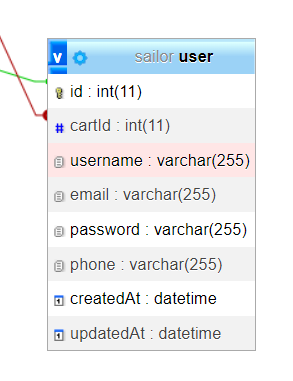
* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. UC bắt đầu khi người dùng chọn “Đăng ký tài khoản”.
2. Màn hình sẽ hiển thị 1 bảng để người dùng điền thông tin bao gồm: tài khoản, mật khẩu, email, số điện thoại.
3. Người dùng điền hết thông tin vào form và ấn nút đăng ký bên dưới. UC kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng, nếu người dùng nhập sai dữ liệu trong bảng thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

* **Các yêu cầu đặc biệt:** Không
* **Tiền điều kiện:** Admin đã đăng nhập thành công.
* **Hậu điều kiện:** Không
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**



Hình 3.3 Dữ liệu liên quan usecase Đăng ký

#### 2.3.2.3. Mô tả usecase Tạo tài khoản admin mới

Mô tả vắn tắt: Ca sử dụng này cho phép admin tạo tài khoản mới.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. UC bắt đầu khi admin chọn “Quản lý tài khoản”.
2. Hệ thống hiển thị màn hình danh sách các tài khoản trong hệ thống.
3. Admin kích nút “Thêm mới” trên màn hình.
4. Hệ thống hiển thị màn hình yêu cầu nhập các thông tin.
5. Admin nhập thông tin và kích nút “Lưu”.
6. Hệ thống truy cập bảng USER, lưu thông tin và hiển thị thông báo. UC kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bước 5 ở luồng cơ bản, nếu thông tin admin nhập không hợp lệ, hệ thống sẽ hiển thị thông báo lỗi. UC kết thúc.
2. Tại bước 5 ở luồng cơ bản, nếu admin kích nút “Hủy”. UC kết thúc.
3. Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

* **Các yêu cầu đặc biệt:** Không
* **Tiền điều kiện:** Admin đã đăng nhập thành công.
* **Hậu điều kiện:** Không
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**

#### 2.3.2.4. Mô tả usecase Tìm kiếm

Mô tả vắn tắt: Ca sử dụng này cho người dùng của hệ thống có thể thực hiện tìm kiếm thông tin của các sản phẩm có trong CSDL.

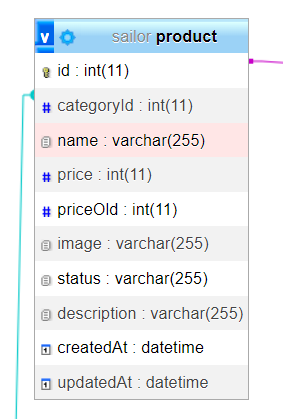
* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Usecase này bắt đầu khi người quản trị nhấn vào biểu tượng “Tìm kiếm” trên thanh header, hệ thống sẽ hiển thị ra giao diện tìm kiếm, người dùng nhập tên của sản phẩm cần tìm kiếm.
2. Sau khi nhập thông tin và ấn tìm kiếm hệ thống sẽ kiểm tra tên sản phẩm có trong bộ dữ liệu hay không. Nếu có thì sẽ hiển thị tất cả thông tin của sản phẩm ra màn hình. UC kết thúc.

* **Luồng rẽ nhánh:**

1. 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.

* **Các yêu cầu đặc biệt:** Không
* **Tiền điều kiện:** Người dùng phải đăng nhập với tài khoản đã đăng kí.
* **Hậu điều kiện:** Không
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**



Hình 3.4 Dữ liệu liên quan usecase Tìm kiếm

#### 2.3.2.5. Mô tả usecase Xem chi tiết sản phẩm

Mô tả vắn tắt: Ca sử dụng này cho phép người dùng của hệ thống xem thông tin của sản phẩm.

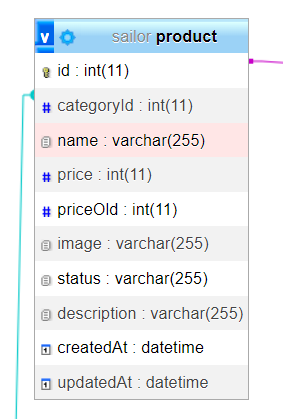
* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. UC này bắt đầu khi người quản trị nhấn vào “Xem thêm” của sản phẩm.
2. Hệ thống sẽ hiển thị ra các thông tin của sản phẩm gồm Tên sản phẩm, danh mục, giá cả, mô tả, tình trạng lên màn hình. UC kết thúc.

* **Luồng rẽ nhánh:**

1. Tại bất kỳ thời điểm nào trong quá trình thực hiện ca sử dụng, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị một thông báo lỗi và ca sử dụng kết thúc.

* **Các yêu cầu đặc biệt:** Không
* **Tiền điều kiện:** Người dùng phải đăng nhập vào hệ thống.
* **Hậu điều kiện:** Không
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**



Hình 3.5 Dữ liệu liên quan usecase Xem chi tiết sản phẩm

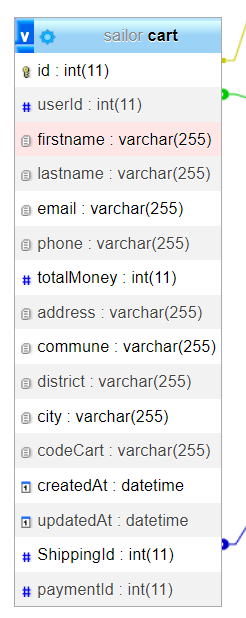
#### 2.3.2.6. Mô tả usecase Xem giỏ hàng

Mô tả vắn tắt: UC này cho phép khách hàng có thể thực hiện xem giỏ hàng xem có bao nhiêu sản phẩm trong giỏ.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. UC bắt đầu khi khách hàng kích vào biểu tượng “xem giỏ hàng”.
2. Hệ thống sẽ hiển thị ra màn hình giỏ hàng của khách hàng gồm thông tin sản phẩm:tên, ảnh, số lượng, giá bán. Khách hàng có thể thêm số lượng sản phẩm mua.
3. Nếu có hàng trong giỏ, khách hàng có thể chọn xóa sản phẩm muốn xóa hoặc xóa hết.
4. Nếu quý khách chưa đăng nhập mà muốn đặt hàng, hệ thống sẽ hiển thị nút “đăng nhập để đặt hàng”. Sau khi đăng nhập thành công tại giỏ hàng sẽ hiển thị nút đặt hàng.

* **Luồng rẽ nhánh:**
* **Các yêu cầu đặc biệt:** Không
* **Tiền điều kiện:** Không.
* **Hậu điều kiện:** Không
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**



Hình 3.6 Dữ liệu liên quan usecase Xem giỏ hàng

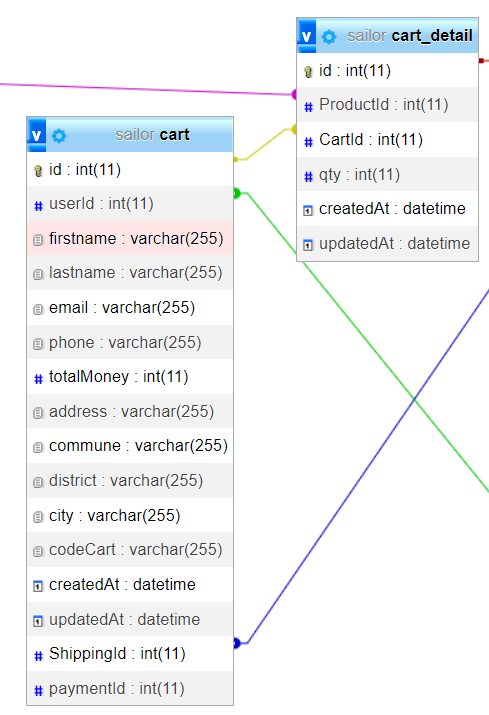
#### 2.3.2.7. Mô tả usecase Thêm sản phẩm (vào giỏ hàng)

Mô tả vắn tắt: UC này cho phép khách hàng có thể thực hiện thêm sản phẩm muốn mua vào giỏ hàng.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. UC bắt đầu khi khách hàng kích vào sản phẩm muốn mua.
2. Hệ thống sẽ hiển thị ra màn hình chi tiết sản phẩm số lượng sản phẩm và nút “thêm giỏ hàng”.
3. Khách hàng có thẻ chọn số lượng muốn mua và kích nút “thêm giỏ hàng”.
4. Hệ thống sẽ tự động thêm sản phẩm vào giỏ hàng và usecase kết thúc.

* **Luồng rẽ nhánh:**
* **Các yêu cầu đặc biệt:** Không
* **Tiền điều kiện:** User đã đăng nhập thành công.
* **Hậu điều kiện:** Không
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**



Hình 3.7 Dữ liệu liên quan usecase Thêm sản phẩm(vào giỏ hàng)

#### 2.3.2.8. Mô tả usecase Liên hệ

Mô tả vắn tắt: Use case này cho phép khách hàng có thể thực hiện đăng ký liên hệ thông qua email.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. UC bắt đầu khi khách hàng kích vào nút “liên hệ”.
2. Hệ thống sẽ hiển thị ra màn hình form liên hệ là những thông tin khách hàng cần điền bao gồm: Tên, gmail, chủ đề, thông tin cần liên hệ.
3. Sau khi điền đầy đủ thông tin khách hàng có thể nhấn nút “Send message”.
4. Sau khi điền đầy đủ thông tin khách hàng có thể nhấn nút “Send message.

* **Luồng rẽ nhánh:**

1. Tại bước 3 trong luồng cơ bản, nếu khách hàng không nhập tên, gmail hoặc message thì hệ thống sẽ hiển thị thông báo “Vui lòng điền vào trường này”. Nếu khách hàng nhập thiếu ‘@’ tại gmail thì hệ thống sẽ hiển thị “Vui lòng bao gồm @ trong địa chỉ email…..” và use case kết thúc.

* **Các yêu cầu đặc biệt:** Không
* **Tiền điều kiện:** Không.
* **Hậu điều kiện:** Không
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**

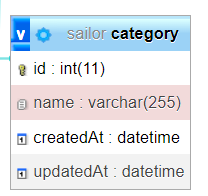
#### 2.3.2.9. Mô tả usecase Xem sản phẩm theo danh mục

Mô tả vắn tắt: Use case này cho phép khách hàng có thể thực hiện xem danh mục các sản phẩm hiện có của cửa hàng.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. UC bắt đầu khi khách hàng chỉ vào nút “Shop Category”.
2. Tại nút “danh mục” hệ thống sẽ hiển thị ra 3 danh mục khác bao gồm: “Giày Nike”, “Giày Adidas”, “Giày Puma”.
3. Khách hàng kích có thể kích vào danh mục mình muốn xem.
4. Hệ thống hiển thị ra màn hình những sản phẩm giày nam đã được chọn ở danh mục và usecase kết thúc.

* **Luồng rẽ nhánh:**
* **Các yêu cầu đặc biệt:** Không
* **Tiền điều kiện:** Admin đã đăng nhập thành công.
* **Hậu điều kiện:** Không
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**



Hình 3.8 Dữ liệu liên quan usecase Xem sản phẩm theo danh mục

#### 2.3.2.10. Mô tả usecase Đặt hàng

Mô tả vắn tắt: UC cho phép người dùng đặt sản phẩm mà người dùng muốn mua, thông qua hệ thống.

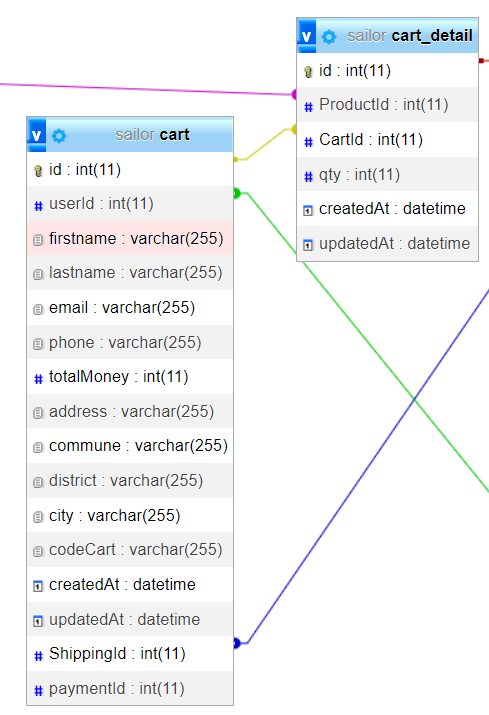
* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. UC bắt đầu khi người dùng chọn “Đặt hàng” ở trong giỏ hàng, khi người dùng nhấn nút “Đặt hàng” hệ thống sẽ trả về một hóa đơn gồm các thông tin liên quan đến sản phẩm như: ID, Tên, SDT, Hình ảnh, Số Lượng, Đơn giá, Thành tiền, Địa chỉ, Postcode.
2. Người dùng sẽ hoàn tất usecase bằng việc nhấn vào hình thức thanh toán chọn một hình thức thanh toán “chuyển khoản” hoặc “tiền mặt” (mặc định) và chọn “Thanh toán ngay” để tiến hành đặt hàng, hệ thống cơ sở dữ liệu sẽ cập nhật lại dữ liệu trong bảng, lúc này usecase kết thúc.

* **Luồng rẽ nhánh:**

Khi mà người dùng xóa hết sản phẩm trong giỏ hàng, nút đặt hàng sẽ bị ẩn, thông báo về giỏ hàng trống sẽ hiện ra, usecase kết thúc.

* **Các yêu cầu đặc biệt:** Không
* **Tiền điều kiện:** Người dùng phải đăng nhập để đặt hàng.
* **Hậu điều kiện:** Không
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**



Hình 3.9 Dữ liệu liên quan Đặt hàng

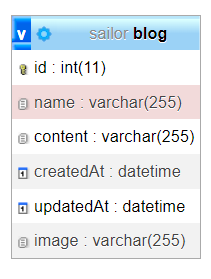
#### 2.3.2.11. Mô tả usecase Xem blog

Mô tả vắn tắt: Use case này cho phép khách hàng có thể thực hiện xem blog các sản phẩm hiện có của cửa hàng.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. UC bắt đầu khi khách hàng chỉ vào nút “Blog” trên thanh header.
2. Hệ thống hiển thị danh sách các blog.
3. Khách hàng kích có thể kích vào blog mình muốn xem.
4. Hệ thống hiển thị ra màn hình nội dung blog đã được chọn và usecase kết thúc.

* **Luồng rẽ nhánh:**
* **Các yêu cầu đặc biệt:** Không
* **Tiền điều kiện:** Không.
* **Hậu điều kiện:** Không
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**



Hình 3.4 Dữ liệu liên quan Xem Blog

#### 2.3.2.12. Mô tả usecase Bảo trì danh mục sản phẩm

Mô tả vắn tắt: Usecase này giúp cho người quản trị hệ thống có thể thực hiện các thao tác liên quan đến việc thêm, sửa, xóa thông tin của các danh mục sản phẩm.

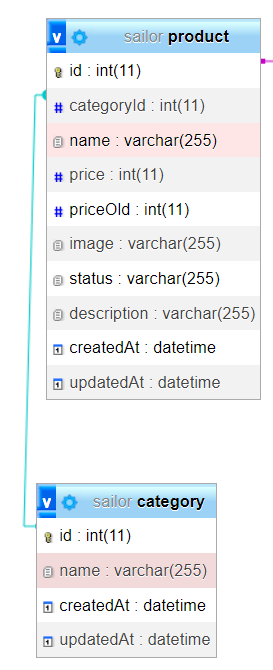
* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Usecase này bắt đầu khi người quản trị nhấn vào “Quản lý danh mục sản phẩm” ở trang Admin, hệ thống sẽ hiển thị ra một trang gồm bảng thêm danh mục sản phẩm mới, và bảng dữ liệu chứa toàn bộ danh mục sản phẩm hiện có.
2. Tại bảng thêm sản phẩm người quản trị có thể thêm các thông tin: Tên danh mục, thứ tự danh mục, cho một danh mục mới. Sau khi điền đầy đủ thông tin, người quản trị nhấn vào nút “Thêm danh mục sản phẩm” để thêm danh mục đấy vào bảng danh mục, hệ thống sẽ cập nhật lại bảng danh mục với danh mục sản phẩm mới và những thông tin vừa được nhập ở bảng thêm.
3. Tại bảng dữ liệu chứa các danh mục sản phẩm, bảng hiển thị các thông tin của sản phẩm gồm: ID, tên danh mục, thứ tự, cột chỉnh sửa gồm 2 chức năng: sửa, xóa.
4. Khi người quản trị nhấn nút sửa, tất cả thông tin của danh mục sản phẩm đó sẽ được điền vào bảng thêm danh mục sản phẩm phía trên và người quản trị có thể thêm, sửa, xóa các mục thông tin trong bảng tùy ý, sau khi sửa xong, nhấn nút “Sửa danh mục sản phẩm”, hệ thống sẽ cập nhật lại thông tin mới vào hệ thống cơ sở dữ liệu.
5. Khi người quản trị nhấn nút “Xóa” hệ thống sẽ xóa danh mục sản phẩm được chọn ra khỏi hệ thống cơ sở dữ liệu và cập nhật lại bảng danh mục sản phẩm.
6. Người dùng nhấn vào “đăng xuất” hay các chức năng bảo trì khác, usecase kết thúc.

* **Luồng rẽ nhánh:**

Hệ thống sẽ trả về thông báo lỗi nếu có lỗi xảy ra trong quá trình xử lý, usecase kết thúc.

* **Các yêu cầu đặc biệt:** Hệ thống cập nhật thông tin thành công
* **Tiền điều kiện:** Người dùng phải đăng nhập với tài khoản Admin.
* **Hậu điều kiện:** Hệ thống cập nhật thông tin thành công
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**



Hình 3.5 Dữ liệu liên quan usecase Bảo trì danh mục sản phẩm

#### 2.3.2.13. Mô tả usecase Bảo trì sản phẩm

Mô tả vắn tắt: Usecase này giúp cho người quản trị hệ thống có thể thực hiện các thao tác liên quan đến việc thêm, sửa, xóa thông tin sản phẩm cho hệ thống.

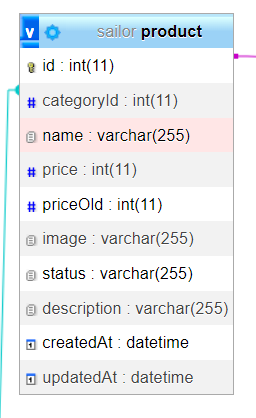
* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Usecase này bắt đầu khi người quản trị nhấn vào “Quản lý sản phẩm” ở trang Admin, hệ thống sẽ hiển thị ra một trang gồm bảng thêm sản phẩm mới, và bảng dữ liệu chứa toàn bộ sản phẩm hiện có.
2. Tại bảng thêm sản phẩm người quản trị có thể thêm các thông tin: Tên sản phẩm, Mã sản phẩm, Giá sản phẩm, Số lượng, Hình ảnh, Tóm tắt, Nội dung, Chọn danh mục sản phẩm, Tình trạng sản phẩm, cho một sản phẩm mới và sau đó, nhấn nút thêm sản phẩm, hệ thống cơ sở dữ liệu sẽ cập nhật sản phẩm mới vào bảng sản phẩm.
3. Tại bảng dữ liệu chứa các sản phẩm, bảng hiển thị các thông tin của sản phẩm gồm: Tên sản phẩm, Mã sản phẩm, Giá sản phẩm, Số lượng, Hình ảnh, Tóm tắt, Nội dung, Danh mục, Trạng thái và Quản lý (gồm 2 nút sửa, xóa).
4. Khi người quản trị nhấn nút sửa, tất cả thông tin của sản phẩm đó sẽ được điền vào bảng thêm sản phẩm phía trên và người quản trị có thể thêm, sửa, xóa các mục thông tin trong bảng tùy ý, sau khi sửa xong, nhấn nút “Sửa sản phẩm”, hệ thống sẽ cập nhật lại thông tin mới vào hệ thống cơ sở dữ liệu.
5. Khi người quản trị nhấn nút “Xóa” hệ thống sẽ xóa sản phẩm được chọn ra khỏi hệ thống cơ sở dữ liệu và cập nhật lại bảng sản phẩm.
6. Người dùng nhấn vào “đăng xuất” hay các chức năng bảo trì khác, usecase kết thúc.

* **Luồng rẽ nhánh:**

Hệ thống sẽ trả về thông báo lỗi nếu có lỗi xảy ra trong quá trình xử lý, usecase kết thúc.

* **Các yêu cầu đặc biệt:** Vai trò của tài khoản phải là vai trò của người quản trị
* **Tiền điều kiện:** Người dùng phải đăng nhập với tài khoản Admin.
* **Hậu điều kiện:** Hệ thống cập nhật thông tin thành công
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**



Hình 3.6 Dữ liệu liên quan usecase Bảo trì sản phẩm

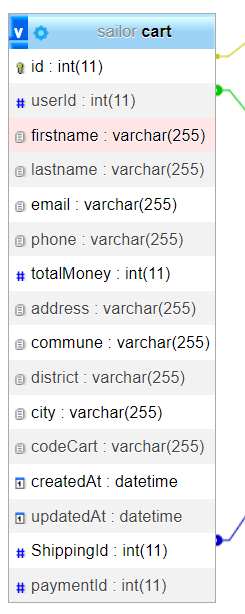
#### 2.3.2.14. Mô tả usecase Quản lý đơn hàng

Mô tả vắn tắt: Usecase này cho phép cho người quản trị hệ thống có thể thực hiện các thao tác liên quan đến việc sửa, đơn hàng trong hệ thống.

* **Luồng sự kiện:**
* **Luồng cơ bản:**

1. Usecase này bắt đầu khi người quản trị nhấn vào “Quản lý đơn hàng” ở trang Admin. Hệ thống sẽ hiển thị lên màn hình thông tin tất cả các ddown hàng mà hệ thống có.
2. Khi người quản trị nhấn nút sửa, tất cả thông tin của đơn hàng đó sẽ được điền vào form sửa và người quản trị có thể sửa các mục thông tin trong bảng tùy ý, sau khi sửa xong, nhấn nút “Sửa”, hệ thống sẽ cập nhật lại thông tin mới vào hệ thống cơ sở dữ liệu.
3. Người dùng nhấn vào “đăng xuất” hay các chức năng bảo trì khác, usecase kết thúc.

* **Luồng rẽ nhánh:**
* **Các yêu cầu đặc biệt:** Không
* **Tiền điều kiện:** Người dùng phải đăng nhập với tài khoản Admin.
* **Hậu điều kiện:** Hệ thống cập nhật thông tin thành công
* **Các điểm mở rộng:** Không
* **Dữ liệu liên quan:**



Hình 3.7 Dữ liệu liên quan usecase Quản lý đơn hàng