 **BỘ GIÁO DỤC VÀ ĐÀO TẠO  
TRƯỜNG ĐẠI HỌC SƯ PHẠM KĨ THUẬT TP.HCM  
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**

**ĐỒ ÁN WEBSITE QUẢN LÝ SIÊU THỊ**

**MÃ MÔN HỌC: PROJ215879\_22\_1\_05CLC  
HỌC KỲ I – NĂM HỌC 2022-2023  
Nhóm thực hiện: Nhóm 6 Đồ án CNTT  
Giảng viên hướng dẫn: Trần Công Tú**

***Thành phố Hồ Chí Minh, Tháng 11 năm 2022***

**GIỚI THIỆU**

Thương mại điện tử là một trong những lĩnh vực tạo ra sự đột phá trong thương mại toàn cầu. Hệ thống này xóa bỏ khoảng cách giữa khách hàng với nhà cung cấp. Tiết kiệm chi phí bán hàng và quản lý bán hàng. Đo lường và đánh giá nhu cầu khách hàng một cách nhanh chóng. Đối với hệ thống website siêu thị điện tử cũng nằm trong sự phát triển đó.

Hệ thống siêu thị điện tử giúp cho nhà cung cấp cập nhật thông tin, quảng bá sản phẩm, đo lường nhu cầu khách hàng, v.v. Đây là hình thức truyền tải thông tin nhanh chóng, hiệu quả và chi phí thấp. Những khách hàng có nhu cầu mua cho mình một món hàng có thể trực tiếp mua thông qua website trực tuyến, không mất nhiều thời gian mà khách hàng đã có thể sở hữu món hàng mà mình cần. Do đó, việc xây dựng website siêu thị điện tử là một trong những trường hợp điển hình của một hoạt động thương mại điện tử. Đồ án phân tích thiết kế website siêu thị điện tử dựa trên cách tiếp cận thương mại điện tử và nền tảng lập trình Java của Oracle. Các phần tiếp theo của đồ án được trình bày như sau:

**Phần 1:** Đặc tả. Phần này mô tả rõ sản phẩm, bao gồm các thông tin như đối tượng nghiên cứu, giới hạn và phạm vi nghiên cứu, ngữ cảnh sử dụng phần mềm, dữ liệu, thông tin đầu vào, mục đích, tính năng, giao diện dự kiến v.v.

**Phần 2:** Phân công công việc. Phần này mô tả công việc trong quá trình thực hiện, phần trăm đóng góp của các thành viên tham gia thực hiện đồ án.

**Phần 3:** Thiết kế. Trong phần này mô tả cách thức xây dựng hệ thống, làm rõ các thiết kế bằng các lược đồ, trình bày thuật toán sử dụng trong quá trình xây dựng.

**Phần 4:** Cài đặt và kiểm thử. Phần này mô tả các tình huống kiểm thử.

**Phần 5:** Kết luận.

Cuối cùng là phần **Tài liệu tham khảo.**

Đồ án sử dụng giao diện từ nguồn <https://dribbble.com/>, giao diện này được nhóm chỉnh sửa lại cho phù hợp với nội dung và cấu trúc của đồ án.

1. **Đặc tả**
   1. **Đối tượng nghiên cứu**

**Người sử dụng:** Người sử dụng hệ thống website xác định có hai nhóm chính là người quản trị hệ thống và người dùng. Người quản trị hệ thống có vai trò quản lý sản và người dùng trên cơ sở dữ liệu; Người dùng bao gồm tất cả những người tham gia sử dụng website và có nhu cầu mua hàng trên website. Cụ thể là Khách hàng.

**Công cụ lập trình:** Eclipse IDE, IntelJ IDE, MySQL, Visual Studio Code.

**Loại Project sử dụng:** Maven Project

**Phần mềm yêu cầu:** Google Chrome, Cốc cốc, Microsoft Edge

**Thư viện sử dụng:**

<https://mvnrepository.com/artifact/mysql/mysql-connector-java>

<https://mvnrepository.com/artifact/org.hibernate/hibernate-core>

<https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api>

<https://mvnrepository.com/artifact/javax.servlet.jsp/javax.servlet.jsp-api>

<https://mvnrepository.com/artifact/javax.servlet/jstl>

<https://mvnrepository.com/artifact/com.sun.mail/jakarta.mail>

* 1. **Giới hạn và phạm vi nghiên cứu**
     1. **Mục đích nghiên cứu**

Đây là trang web thương mại điện tử, nhằm cung cấp môi trường để các tổ chức, công ty giới thiệu các mặt hàng đến tay người tiêu dùng bằng phương tiện trực tuyến. Mục đích cụ thể bao gồm thông qua việc nghiên cứu tìm hiểu thông tin trên thị trường, xã hội, thiết kế xây dựng website, cải tiến các kỹ thuật để nâng cao trình độ lập trình website. Bên cạnh đó tìm hiểu cấu trúc và phương pháp viết đồ án nhằm hỗ trợ việc viết đồ án tốt nghiệp.

* + 1. **Phạm vi nghiên cứu**

Phạm vi về nội dung: Đề tài nghiên cứu quá trình ứng dụng website siêu thị điện tử.

Phạm vi về không gian: Đề tài nghiên cứu tại thị trường Việt Nam.

Phạm vi về thời gian: từ 22/09/2022 đến 03/12/2022.

* + 1. **Phương pháp nghiên cứu**

Phương pháp tài liệu: Tìm kiếm các tài liệu có liên quan về website siêu thị điện tử, từ đó phân tích, thống kê số liệu, tổng hợp rồi chỉ ra những phần cũ cần cải tiến hoặc những phần thiếu sót, sai lệch của web siêu thị điện tử, đồng thời tìm ra những tính năng mới phù hợp với nhu cầu khách hàng để từ đó cải tiến những điểm yếu ban đầu và phát triển thành những tính năng mới phù hợp với nhu cầu khách hàng, những tính năng cần thiết cho website siêu thị điện tử.

Phương phát thực nghiệm: thực hiện cái đặt các thuật toán của luận án, chạy thử nghiệm trên các dữ liệu đã được phân tích và cài đặt trong cơ dữ liệu. Kiểm tra đánh giá web trong quá trình chạy thử, đưa ra các phương án mới để kịp thời chỉnh sửa và bổ sung

* + 1. **Giới hạn của hệ thống website**

Hệ thống đã được thiết kế với đầy đủ tính năng của một website bán hàng, tuy nhiên vẫn còn một vài hạn chế như khách hàng chưa thể thanh toán trực tiếp qua website bằng cách liên kết thẻ ngân hàng mà vẫn phải thanh toán tiền mặt khi nhận hàng. Hệ thống chưa có trang ADMIN riêng phục vụ cho người quản lý, người quản lý cần quản lý trực tiếp trên cơ sở dữ liệu, đòi hỏi người quan lý cần có trình độ về lĩnh vực Công nghệ thông tin.

Sản phẩm của website chưa đa dạng về các loại mặt hàng, chưa áp dụng được tính năng giảm giá dựa trên các loại mặt hàng vào các dịp đặc biệt.

* 1. **Tính năng của hệ thống website**

Hệ thống đã được thiết kế với đầy đủ tính năng của một website bán hàng, về phía khách hàng hệ thống website có các chức năng như sau:

* *Hiển thị thông tin về cửa hàng:*

Hầu hết khách hàng hiện nay đều muốn tham khảo các thông tin liên quan đến cửa hàng nhằm kiểm tra và xem xét độ uy tín của cửa hàng, sau đó mới quyết định mua hàng. Chính vì lẽ đó, một website bán hàng cần cung cấp đầy đủ, rõ ràng các thông tin của cửa hàng.

* *Tra cứu sản phẩm với tính năng tìm kiếm:*

Một website có rất nhiều sản phẩm, có thể khiến Khách hàng bối rối và mất thời gian khi tìm sản phẩm mình cần. Với chức năng tìm kiếm ở góc trên bên phải trang chủ website, Khách hàng có thể nhanh chóng tìm ra sản phẩm họ cần.

* *Tính năng lọc:*

Tính năng lọc sản phẩm dựa theo loại sản phẩm, với tính năng này, Khách hàng có thể dễ dàng chọn lựa các nhãn hàng dựa trên sản phẩm họ cần mua.

* *Tính năng thêm sản phẩm vào giỏ hàng, xóa sản phẩm khỏi giỏ hàng, tính năng đặt hàng:*

Khách hàng có thể thêm sản phẩm vào giỏ hàng của họ bằng cách nhấn vào nút *ADD TO CART* bên dưới mỗi mặt hàng, xóa sản phẩm bằng biểu tượng X được hiển thị trong danh sách các sản phẩm ở trang giỏ hàng. Khi cần đặt hàng số mặt hàng đã thêm vào giỏ hàng, Khách hàng chỉ việc nhấn vào biểu tượng giỏ hàng ở góc trên cùng phía bên phải, một trang mới hiển thị các mặt hàng họ đã thêm sẽ hiện ra. Lúc này, Khách hàng chỉ việc nhập đầy đủ thông tin đặt hàng như số điện thoại, địa chỉ v.v. và sau đó xác nhận. Hệ thống sẽ lưu toàn bộ thông tin giao hàng của Khách hàng để tiến hành lên đơn và giao hàng.

* *Tính năng cho phép người dùng tạo tài khoản riêng và đăng nhập vào tài khoản đã đăng ký:*

Tính năng này giúp doanh nghiệp, công ty, cửa hàng dễ dàng nắm bắt sở thích Khách hàng, từ đó đưa ra những sản phẩm phù hợp với nhu cầu. Bên cạnh đó, tính năng này giúp bảo mật quyền riêng tư và thông tin cá nhân của Khách hàng, giúp Khách hàng an tâm hơn trong quá trình mua hàng trên website.

Khi người dùng chưa có tài khoản website, hệ thống sẽ cho phép Khách hàng đăng ký tài khoản và lưu thông tin tài khoản sau khi Khách hàng tạo mới bên dưới cơ sở dữ liệu. Với trường hợp Khách hàng đã có tài khoản và cần đăng nhập, hệ thống sẽ kiểm tra mật khẩu đăng nhập mà Khách hàng nhập vào, nếu mật khẩu này khớp với mật khẩu được lưu trữ ở Cơ sở dữ liệu, hệ thống sẽ cho phép Khách hàng đăng nhập thành công. Ngược lại, hệ thống yêu cầu Khách hàng nhập lại thông tin đăng nhập một lần nữa nếu muốn tiếp tục đăng nhập.

* *Tính năng đăng xuất tài khoản hiện tại:*

Tính năng này cho phép người dùng đăng xuất tài khoản ra khỏi phiên làm việc hiện tại.

* *Tính năng gửi phản hồi:*

Khách hàng có thể gửi phản hồi của họ đến người quản lý website bằng cách đi đến trang liên hệ ở danh mục cuối cùng trong thanh chia danh mục, cụ thể là danh mục *CONTACT*. Mọi phản hồi của Khách hàng sẽ ngay lập tức được gửi đến mail của người quản lý.

* *Tính năng trợ giúp:*

Khách hàng có thể gửi yêu cầu giúp đỡ bằng cách nhấn vào mục *Help* ở góc trên cùng bên phải, gần với logo giỏ hàng. Mọi yêu cầu của Khách hàng sẽ được gửi đến mail của người quản lý.

* 1. **Giao diện dự kiến**

Trang chủ website bao gồm phần header hiển thị tên website và chứa thanh danh mục các loại hàng hóa, thanh danh mục các chức năng. Phần footer hiển thị các đường dẫn đến các chức năng và thông tin liên hệ. Phần body chứa các hình ảnh minh họa cho website và logo các nhãn hàng. Các trang thành phần còn lại cũng bao gồm phần header và footer tương tự như trang chủ, nhưng cấu trúc phần body sẽ khác nhau tùy theo tính năng từng trang.

Graphical user interface, website

Description automatically generated

*Hình 1 – Giao diện dự kiến (1)*

Website

Description automatically generated

*Hình 2 - Giao diện dự kiến (2)*

* 1. **Lược đồ Use Case**

**Diagram

Description automatically generated**

*Hình 3 - Lược đồ Use Case*

* 1. **Bảng mô tả các Use Case**

*Bảng 1.1 – Mô tả Use Case Đăng nhập*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Use Case | Đăng nhập | | |
| Mô tả | Cho phép người dùng đăng nhập vào hệ thống website | | |
| Tác nhân | Khách hàng, Người dùng | | |
| Ngữ cảnh sử dụng | STT | Tình huống | Hệ thống |
|  | 1 | Click vào nút *Login* |  |
|  | 2 |  | Mở giao diện trang Login |
|  | 3 | Nhập thông tin đăng nhập và nhấn *Submit* |  |
|  | 4 |  | Cho phép Khách hàng đăng nhập nếu chứng thực thành công, hiển thị tên Khách hàng và trạng thái hoạt động thay vào ba nút *Create Account*, *Login* và *Help* |
|  | 5 |  | Trả về trang *Create Account* nếu User name chưa tồn tại trong Cơ sở dữ liệu |
|  | 6 |  | Không cho phép đăng nhập, trả về trang Login nếu mật khẩu không khớp với mật khẩu lưu trong cơ sở dữ liệu |
| Điều kiện tiên quyết | Server đang hoạt động | | |
| Điều kiện ảnh hưởng dẫn đến chấm dứt tiến trình hoạt động | Server đang hoạt động, kết nối được thiết lập thành công  Server dừng hoạt động, kết nối thất bại | | |

*Bảng 1.2 – Mô tả Use Case Đăng ký tài khoản*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Use Case | Đăng ký tài khoản | | |
| Mô tả | Cho phép người dùng tạo tài khoản nhằm đăng nhập vào hệ thống website và sử dụng | | |
| Tác nhân | Khách hàng, Người dùng | | |
| Ngữ cảnh sử dụng | STT | Tình huống | Hệ thống |
|  | 1 | Click vào nút *Create**Account* |  |
|  | 2 |  | Mở giao diện trang Create Account |
|  | 3 | Nhập đầy đủ thông tin đăng ký, sau đó nhấn *Register* |  |
|  | 4 |  | Cho phép Khách hàng đăng ký nếu địa chỉ email chưa tồn tại trong cơ sở dữ liệu |
|  | 5 |  | Trả về trang Login sau khi đăng ký thành công |
|  | 6 |  | Không cho phép đăng ký, trả về trang Create Account nếu địa chỉ email đã tồn tại trong cơ sở dữ liệu |
| Điều kiện tiên quyết | Server đang hoạt động | | |
| Điều kiện ảnh hưởng dẫn đến chấm dứt tiến trình hoạt động | Server đang hoạt động, kết nối được thiết lập thành công  Server dừng hoạt động, kết nối thất bại | | |

*Bảng 1.3 – Mô tả Use Case Đăng xuất khỏi tài khoản hiện tại*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Use Case | Đăng xuất khỏi tài khoản hiện tại | | |
| Mô tả | Khách hàng đăng xuất tài khoản khỏi hệ thống website | | |
| Tác nhân | Khách hàng, Người dùng | | |
| Ngữ cảnh sử dụng | STT | Tình huống | Hệ thống |
|  | 1 | Click vào nút *Log out* |  |
|  | 2 |  | Đăng xuất tài khoản khỏi hệ thống website |
| Điều kiện tiên quyết | Server đang hoạt động | | |
| Điều kiện ảnh hưởng dẫn đến chấm dứt tiến trình hoạt động | Server đang hoạt động, kết nối được thiết lập thành công  Server dừng hoạt động, kết nối thất bại | | |

*Bảng 1.4 – Mô tả Use Case Tìm kiếm sản phẩm*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Use Case | Tìm kiếm sản phẩm | | |
| Mô tả | Cho phép Khách hàng tìm kiếm sản phẩm theo từ khóa | | |
| Tác nhân | Khách hàng, Người dùng | | |
| Ngữ cảnh sử dụng | STT | Tình huống | Hệ thống |
|  | 1 | Nhập từ khóa cần tìm vào mục tìm kiếm góc phải website |  |
|  | 2 |  | Hiển thị danh sách các sản phẩm với từ khóa Khách hàng đã nhập |
| Điều kiện tiên quyết | Server đang hoạt động | | |
| Điều kiện ảnh hưởng dẫn đến chấm dứt tiến trình hoạt động | Server đang hoạt động, kết nối được thiết lập thành công  Server dừng hoạt động, kết nối thất bại | | |

*Bảng 1.5 – Mô tả Use Case Thêm sản phẩm vào giỏ hàng*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Use Case | Thêm sản phẩm vào giỏ hàng | | |
| Mô tả | Cho phép Khách hàng thêm sản phẩm vào giỏ hàng | | |
| Tác nhân | Khách hàng, Người dùng | | |
| Ngữ cảnh sử dụng | STT | Tình huống | Hệ thống |
|  | 1 | Click vào nút *ADD TO CART* bên dưới mỗi sản phẩm |  |
|  | 2 |  | Thêm sản phẩm vào giỏ hàng có phạm vi là Session, danh sách các sản phẩm này sẽ được hiển thị khi Khách hàng đi đến trang giỏ hàng |
| Điều kiện tiên quyết | Server đang hoạt động | | |
| Điều kiện ảnh hưởng dẫn đến chấm dứt tiến trình hoạt động | Server đang hoạt động, kết nối được thiết lập thành công  Server dừng hoạt động, kết nối thất bại | | |

*Bảng 1.6 – Mô tả Use Case Xóa sản phẩm khỏi giỏ hàng*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Use Case | *Xóa sản phẩm khỏi giỏ hàng* | | |
| Mô tả | Cho phép Khách hàng xóa sản phẩm khỏi giỏ hàng | | |
| Tác nhân | Khách hàng, Người dùng | | |
| Ngữ cảnh sử dụng | STT | Tình huống | Hệ thống |
|  | 1 | Click vào biểu tượng dấu X trong danh sách các sản phẩm ở trang giỏ hàng |  |
|  | 2 |  | Xóa sản phẩm khỏi danh sách các sản phẩm trong giỏ hàng có phạm vi là Session, sau đó quay về trang giỏ hàng |
| Điều kiện tiên quyết | Server đang hoạt động  Khách hàng đã đăng nhập thành công vào hệ thống website | | |
| Điều kiện ảnh hưởng dẫn đến chấm dứt tiến trình hoạt động | Server đang hoạt động, kết nối được thiết lập thành công  Server dừng hoạt động, kết nối thất bại | | |

*Bảng 1.7 – Mô tả Use Case Đặt hàng*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Use Case | Đặt hàng | | |
| Mô tả | Tiến hành đặt hàng các món hàng đã thêm vào giỏ hàng trước đó | | |
| Tác nhân | Khách hàng, Người dùng | | |
| Ngữ cảnh sử dụng | STT | Tình huống | Hệ thống |
|  | 1 | Nhập thông tin đặt hàng ở trang giỏ hàng |  |
|  | 2 | Click vào nút thanh toán (*PAY*) ở trang giỏ hàng sau khi nhập thông tin đặt hàng |  |
|  | 3 |  | Lưu thông tin đặt hàng vào cơ sở dữ liệu |
| Điều kiện tiên quyết | Server đang hoạt động  Khách hàng đã đăng nhập thành công vào hệ thống website Giỏ hàng cần có ít nhất một sản phẩm | | |
| Điều kiện ảnh hưởng dẫn đến chấm dứt tiến trình hoạt động | Server đang hoạt động, kết nối được thiết lập thành công  Server dừng hoạt động, kết nối thất bại | | |

*Bảng 1.8 – Mô tả Use Case Gửi phản hồi/Yêu cầu giúp đỡ*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Use Case | Gửi phản hồi | | |
| Mô tả | Khách hàng gửi ý kiến đóng góp, phản hồi đến người quản lý hệ thống website | | |
| Tác nhân | Khách hàng, Người dùng | | |
| Ngữ cảnh sử dụng | STT | Tình huống | Hệ thống |
|  | 1 | Click vào nút *Contact* |  |
|  | 2 | Nhập thông tin cần thiết, nội dung cần phản hồi/yêu cầu giúp đỡ, sau đó gửi phản hồi/yêu cầu |  |
|  | 3 |  | Gửi phản hồi/yêu cầu đến email người quản lý hệ thống website |
| Điều kiện tiên quyết | Server đang hoạt động  Thiết lập kết nối Java mail thành công | | |
| Điều kiện ảnh hưởng dẫn đến chấm dứt tiến trình hoạt động | Server đang hoạt động, kết nối được thiết lập thành công  Thiết lập kết nối Java mail không thành công  Server dừng hoạt động, kết nối thất bại | | |

*Bảng 1.9 – Mô tả Use Case Xem thông tin về hệ thống website*

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Use Case | *Xem thông tin về hệ thống website* | | |
| Mô tả | Khách hàng xem thêm thông tin về hệ thống website để đánh giá mức độ tin cậy | | |
| Tác nhân | Khách hàng, Người dùng | | |
| Ngữ cảnh sử dụng | STT | Tình huống | Hệ thống |
|  | 1 | Click vào nút *About Us, FAQ’s*ở phần Footer của mỗi trang |  |
|  | 2 |  | Hiển thị thông tin về website và những câu hỏi thường gặp của Khách hàng |
| Điều kiện tiên quyết | Server đang hoạt động | | |
| Điều kiện ảnh hưởng dẫn đến chấm dứt tiến trình hoạt động | Server đang hoạt động, kết nối được thiết lập thành công  Server dừng hoạt động, kết nối thất bại | | |

1. **Phân công công việc**
2. **Thiết kế**
   1. **Mô tả thiết kế**

Hệ thống website được viết theo mô hình MVC(Model-View-Controller). Cụ thể, mọi thao tác của người dùng trên hệ thống website là thao tác trên Browser, và hệ thống sẽ gửi request từ Browser đến Web Server, sau đó Web Server sẽ chuyển request đến Controller để xử lý. Controller khi nhận request được gửi sẽ thông qua các model để tương tác và lấy dữ liệu từ cơ sở dữ liệu, sau đó xử lý request và chuyển dữ liệu sau khi xử lý đến View. Tiếp đến Controller sẽ nhận được một View hoàn chỉnh và gửi trả về Browser cho người dùng.

Hệ thống website sử dụng một cách triển khai của JPA là Hibernate để tương tác với Cơ sở dữ liệu. Việc sử dụng Hibernate tương tác với Cơ sở dữ liệu giúp quá trình tương tác đơn giản và hiệu quả hơn, tránh nhập nhằng hơn rất nhiều so với việc sử dụng phương pháp cũ hơn trước đây là JDBC.

Model của hệ thống website bao gồm các lớp Entity tương ứng với các thực thể trong Cơ sở dữ liệu và các lớp Data Transfer Object(DAO) chứa các phương thức thao tác với Cơ sở dữ liệu dựa trên Hibernate.

View của hệ thống website bao gồm các file JSP, các file CSS và các file JavaScripts. Phần View của hệ thống sử dụng Website Template được thiết kế và định dạng từ nguồn <https://dribbble.com/>. Website Template này đã được chỉnh sửa lại cho phù hợp với cấu trúc đề tài đồ án và phù hợp với các xử lý ở Controlller.

Cuối cùng là về lớp Controller, lớp này chứa file Servlet để xử lý các yêu cầu từ phía Khách hàng/Người dùng.

* 1. **Thiết kế Cơ sở dữ liệu**

**Diagram

Description automatically generated**

*Hình 4 – Sơ đồ Cơ sở dữ liệu*

*Bảng 3.1 – Bảng mô tả các Table trong CSDL*

|  |  |  |
| --- | --- | --- |
| TT | Tên bảng | Mục đích |
| 1 | products | Chứa thông tin về các sản phẩm trên trang web |
| 2 | supliers | Chứa thông tin về nhà cung cấp sản phẩm |
| 3 | categories | Chứa loại hàng hóa và mô tả về loại hàng hóa đó |
| 4 | users | Chứa thông tin về Khách hàng/Người dùng |
| 5 | orders | Chứa thông tin về đơn hàng sau khi đặt hàng |
| 6 | orderdetails | Thông tin chi tiết đơn hàng với đơn hàng và sản phẩm tương ứng |

*Bảng 3.2 – Bảng các trường trong products*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | productID | INT | Mã sản phẩm |
| 2 | supplierID | INT | Mã nhà cung cấp ứng với sản phẩm |
| 3 | categoryID | INT | Mã danh mục sản phẩm ứng với sản phẩm |
| 4 | productName | VARCHAR(50) | Tên sản phẩm |
| 5 | price | INT | Giá tiền |
| 6 | rating | INT | Mức độ đánh giá |
| 7 | updateDate | DATE | Ngày cập nhật |
| 8 | isSoldout | BIT(1) | Đã bán hết |
| 9 | productImage | VARCHAR(100) | Đường dẫn đến ảnh sản phẩm |

*Bảng 3.3 – Bảng các trường trong suppliers*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | supplierID | INT | Mã nhà cung cấp |
| 2 | supplierName | VARCHAR(20) | Tên nhà cung cấp |
| 3 | phone | VARCHAR(10) | Số điện thoại |
| 4 | address | CHAR(50) | Địa chỉ |

*Bảng 3.4 – Bảng các trường trong categories*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | categoryID | INT | Mã danh mục hàng hóa |
| 2 | categoryName | VARCHAR(20) | Tên danh mục hàng hóa |
| 3 | descriptions | VARCHAR(100) | Mô tả |

*Bảng 3.5 – Bảng các trường trong orders*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | orderID | INT | Mã hóa đơn |
| 2 | orderUserID | INT | Mã Khách hàng ứng với hóa đơn |
| 3 | orderAddress | VARCHAR(50) | Địa chỉ nhận hàng |
| 4 | createDate | DATE | Ngày tạo hóa đơn |
| 5 | orderDate | DATE | Ngày nhận hàng |
| 6 | orderPhone | INT | Số điện thoạt đặt hàng |

*Bảng 3.6 – Bảng các trường trong orderdetails*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | detailID | INT | Mã chi tiết hóa đơn |
| 2 | detailOrderID | INT | Mã hóa đơn tương ứng |
| 3 | detailProductID | INT | Mã sản phẩm tương ứng |
| 4 | detailQuantity | INT | Số lượng sản phẩm tương ứng |

*Bảng 3.7 – Bảng các trường trong users*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | userID | INT | Mã Khách hàng |
| 2 | userEmail | VARCHAR(30) | Email Khách hàng |
| 3 | userPassword | VARCHAR(30) | Mật khẩu tài khoản |
| 4 | userName | VARCHAR(30) | Tên Khách hàng |
| 5 | userAddress | VARCHAR(50) | Địa chỉ Khách hàng |
| 6 | userPhone | VARCHAR(10) | Số điện thoại Khách hàng |
| 7 | registrationDate | DATE | Ngày đăng kí |

* 1. **Thiết kế lớp**

*Bảng 3.8 - Danh sách các lớp được sử dụng trong hệ thống website*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên lớp | Mục đích | Phụ trách |
| 1 | Categories  Implement: Serializable | Tạo entity Categories ánh xạ tương ứng với bảng categories trong cơ sở dữ liệu | Trương Thuận Hòa |
| 2 | OrderDetails  Implement: Serializable | Tạo entity OrderDetails ánh xạ tương ứng với bảng orderdetails trong cơ sở dữ liệu | Trương Thuận Hòa |
| 3 | Orders  Implement: Serializable | Tạo entity Orders ánh xạ tương ứng với bảng orders trong cơ sở dữ liệu | Trương Thuận Hòa |
| 4 | Products  Implement: Serializable | Tạo entity Products ánh xạ tương ứng với bảng products trong cơ sở dữ liệu | Trương Thuận Hòa |
| 5 | Suppliers  Implement: Serializable | Tạo entity Suppliers ánh xạ tương ứng với bảng suppliers trong cơ sở dữ liệu | Trương Thuận Hòa |
| 6 | Users  Implement: Serializable | Tạo entity Users ánh xạ tương ứng với bảng users trong cơ sở dữ liệu | Trương Thuận Hòa |
| 7 | HibernateUtil | Lớp bổ trợ chứa các phương thức bổ trợ cho Hibernate | Trương Thuận Hòa |
| 8 | SendEmailUtil | Lớp bổ trợ chứa các phương thức bổ trợ cho việc kết nối và gửi mail | Trương Thuận Hòa |
| 9 | interface GenericDAO<E> | Interface chứa các phương thức CRUD cần triển khai, được viết dưới dạng Generic giúp cho các Entity khác nhau trong hệ thống website có thể triển khai | Trương Thuận Hòa |
| 10 | HibernateDAO<E> | Lớp dùng để triển khai các phương thức CRUD và đóng kết nối SessionFactory, được viết dưới dạng Generic để kế thừa cho các Entity khác nhau trong hệ thống website | Trương Thuận Hòa |
| 11 | CategoryDAO  Extend: HibernateDAO <Categories>  Implement:GenericDAO <Categories> | Lớp triển khai các phương thức thao tác với dữ liệu trong bảng categories thông qua Entity Categories bằng cách sử dụng Hibernate | Phan Hồng Sơn |
| 12 | OrderDAO  Extend: HibernateDAO <Orders>  Implement:GenericDAO <Orders> | Lớp triển khai các phương thức thao tác với dữ liệu trong bảng orders thông qua Entity Orders bằng cách sử dụng Hibernate | Phan Hồng Sơn |
| 13 | OrderdetailDAO  Extend: HibernateDAO <Orderdetails>  Implement:GenericDAO <Orderdetails> | Lớp triển khai các phương thức thao tác với dữ liệu trong bảng orderdetails thông qua Entity Orderdetails bằng cách sử dụng Hibernate | Phan Hồng Sơn |
| 14 | ProductDAO  Extend: HibernateDAO <Products>  Implement:GenericDAO <Products> | Lớp triển khai các phương thức thao tác với dữ liệu trong bảng products thông qua Entity Products bằng cách sử dụng Hibernate | Phan Hồng Sơn |
| 15 | SupplierDAO  Extend: HibernateDAO <Suppliers>  Implement:GenericDAO <Suppliers> | Lớp triển khai các phương thức thao tác với dữ liệu trong bảng suppliers thông qua Entity Suppliers bằng cách sử dụng Hibernate | Phan Hồng Sơn |
| 16 | UserDAO  Extend: HibernateDAO <Users>  Implement:GenericDAO <Users> | Lớp triển khai các phương thức thao tác với dữ liệu trong bảng users thông qua Entity Users bằng cách sử dụng Hibernate | Phan Hồng Sơn |
| 17 | RegisterUserServlet  Extend: HttpServlet | Servlet xử lý request được gửi đến từ browser khi Khách hàng đăng ký tài khoản mới | Trương Thuận Hòa |
| 18 | LoginUserServlet  Extend: HttpServlet | Servlet xử lý request được gửi đến từ browser khi Khách hàng đăng nhập vào hệ thống website | Trương Thuận Hòa |
| 19 | LogoutUserServlet  Extend: HttpServlet | Servlet xử lý request được gửi đến từ browser khi Khách hàng đăng xuất khỏi hệ thống website | Trương Thuận Hòa |
| 20 | SearchProductServlet  Extend: HttpServlet | Servlet xử lý request được gửi đến từ browser khi Khách hàng tìm kiếm sản phẩm dựa trên từ khóa | Trương Thuận Hòa |
| 21 | AddToCartServlet  Extend: HttpServlet | Servlet xử lý request được gửi đến từ browser khi Khách hàng cần thêm sản phẩm vào giỏ hàng sau khi đăng nhập vào hệ thống website thành công | Phan Hồng Sơn |
| 22 | CheckoutServlet  Extend: HttpServlet | Servlet dùng để kiểm tra Khách hàng đã đăng nhập vào hệ thống website hay chưa trước khi cho phép Khách hàng đi đến trang giỏ hàng | Trương Thuận Hòa |
| 23 | ContactServlet  Extend: HttpServlet | Servlet dùng để gửi yêu cầu và phản hồi của Khách hàng đến email của người quản lý website | Trương Thuận Hòa |
| 24 | OrderServlet  Extend: HttpServlet | Servlet dùng để lưu thông tin đặt hàng và danh sách các sản phẩm đã đặt của Khách hàng xuống Cơ sở dữ liệu | Trương Thuận Hòa |
| 25 | RemoveFromCartServlet  Extend: HttpServlet | Servlet xử lý request được gửi đến từ browser khi Khách hàng xóa sản phẩm khỏi giỏ hàng | Phan Hồng Sơn |
| 26 | RenderProductData Servlet  Extend: HttpServlet | Servlet hiển thị danh sách các sản phẩm lên trang web theo từng danh mục sản phẩm | Phan Hồng Sơn |
| 27 | UpdateQuantityServlet  Extend: HttpServlet | Servlet giúp tăng/giảm số lượng sản phẩm trong giỏ hàng | Phan Hồng Sơn |

*Bảng 3.9 – Danh sách các phương thức trong lớp Categories*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | getCategoryId()  Input: Không có  Output: int | Lấy giá trị thuộc tính categoryId | Categories.java(39) | Trương Thuận Hòa |
| 2 | setCategoryId(int categoryId)  Input: categoryId  Output: Không có | Gán giá trị cho thuộc tính categoryId | Categories.java(43) | Trương Thuận Hòa |
| 3 | getCategoryName()  Input: Không có  Output: String | Lấy giá trị thuộc tính categoryName | Categories.java(48) | Trương Thuận Hòa |
| 4 | setCategoryName(String categoryName)  Input: categoryName  Output: Không có | Gán giá trị cho thuộc tính categoryName | Categories.java(52) | Trương Thuận Hòa |
| 5 | getdescription()  Input: Không có  Output: String | Lấy giá trị thuộc tính description | Categories.java(57) | Trương Thuận Hòa |
| 6 | setdescription(String description)  Input: description  Output: Không có | Gán giá trị cho thuộc tính description | Categories.java(61) | Trương Thuận Hòa |
| 7 | getProductses()  Input: Không có  Output: Set<Products> | Lấy giá trị thuộc tính productses | Categories.java(66) | Trương Thuận Hòa |
| 8 | setProductses(Set<Products> productses)  Input: productses  Output: Không có | Gán giá trị cho thuộc tính productses | Categories.java(70) | Trương Thuận Hòa |

*Bảng 3.10 – Danh sách các phương thức trong lớp Orderdetails*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | getDetailId()  Input: Không có  Output: int | Lấy giá trị thuộc tính detailId | Orderdetails.java (37) | Trương Thuận Hòa |
| 2 | setDetailId(int detailId)  Input: detailId  Output: Không có | Gán giá trị cho thuộc tính detailId | Orderdetails.java (41) | Trương Thuận Hòa |
| 3 | getOrders()  Input: Không có  Output: Orders | Lấy giá trị thuộc tính orders | Orderdetails.java (47) | Trương Thuận Hòa |
| 4 | setOrders(Orders orders)  Input: orders  Output: Không có | Gán giá trị cho thuộc tính orders | Orderdetails.java (51) | Trương Thuận Hòa |
| 5 | getProducts()  Input: Không có  Output: Products | Lấy giá trị thuộc tính products | Orderdetails.java (57) | Trương Thuận Hòa |
| 6 | setProducts(Products products)  Input: products  Output: Không có | Gán giá trị cho thuộc tính products | Orderdetails.java (61) | Trương Thuận Hòa |
| 7 | getDetailQuantity()  Input: Không có  Output: int | Lấy giá trị thuộc tính detailQuantity | Orderdetails.java (66) | Trương Thuận Hòa |
| 8 | setDetailQuantity(int detailQuantity)  Input: detailQuantity  Output: Không có | Gán giá trị cho thuộc tính detailQuantity | Orderdetails.java (70) | Trương Thuận Hòa |

*Bảng 3.11 – Danh sách các phương thức trong lớp Orders*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | getOrderPhone()  Input: Không có  Output: int | Lấy giá trị thuộc tính orderPhone | Orders.java(45) | Trương Thuận Hòa |
| 2 | setOrderPhone(int orderPhone)  Input: orderPhone  Output: Không có | Gán giá trị cho thuộc tính orderPhone | Orders.java(49) | Trương Thuận Hòa |
| 3 | getOrderId()  Input: Không có  Output: int | Lấy giá trị thuộc tính orderId | Orders.java(56) | Trương Thuận Hòa |
| 4 | setOrderId(int orderId)  Input: orderId  Output: Không có | Gán giá trị cho thuộc tính orderId | Orders.java(60) | Trương Thuận Hòa |
| 5 | getUsers()  Input: Không có  Output: Users | Lấy giá trị thuộc tính users | Orders.java(66) | Trương Thuận Hòa |
| 6 | setUsers(Users users)  Input: users  Output: Không có | Gán giá trị cho thuộc tính users | Orders.java(70) | Trương Thuận Hòa |
| 7 | getOrderAddress()  Input: Không có  Output: String | Lấy giá trị thuộc tính orderAddress | Orders.java(77) | Trương Thuận Hòa |
| 8 | setOrderAddress(String orderAddress)  Input: orderAddress  Output: Không có | Gán giá trị cho thuộc tính orderAddress | Orders.java(81) | Trương Thuận Hòa |
| 9 | getCreateDate()  Input: Không có  Output: Date | Lấy giá trị thuộc tính createDate | Orders.java(87) | Trương Thuận Hòa |
| 10 | setCreateDate(Date createDate)  Input: createDate  Output: Không có | Gán giá trị cho thuộc tính createDate | Orders.java(91) | Trương Thuận Hòa |
| 11 | getOrderDate()  Input: Không có  Output: Date | Lấy giá trị thuộc tính orderDate | Orders.java(97) | Trương Thuận Hòa |
| 12 | setOrderDate(Date orderDate)  Input: orderDate  Output: Không có | Gán giá trị cho thuộc tính orderDate | Orders.java(101) | Trương Thuận Hòa |
| 13 | getOrderdetailses()  Input: Không có  Output: Set<Orderdetails> | Lấy giá trị thuộc tính orderdetailses | Orders.java(106) | Trương Thuận Hòa |
| 14 | setOrderdetailses(Set<Orderdetails> orderdetailses)  Input: orderdetailses  Output: Không có | Gán giá trị cho thuộc tính orderdetailses | Orders.java(110) | Trương Thuận Hòa |

*Bảng 3.12 – Danh sách các phương thức trong lớp Products*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | getProductId()  Input: Không có  Output: int | Lấy giá trị thuộc tính productId | Products.java(49) | Trương Thuận Hòa |
| 2 | setProductId(int productId)  Input: productId  Output: Không có | Gán giá trị cho thuộc tính productId | Products.java(53) | Trương Thuận Hòa |
| 3 | getCategories()  Input: Không có  Output: Categories | Lấy giá trị thuộc tính categories | Products.java(59) | Trương Thuận Hòa |
| 4 | setCategories(Categories categories)  Input: categories  Output: Không có | Gán giá trị cho thuộc tính categories | Products.java(63) | Trương Thuận Hòa |
| 5 | getSuppliers()  Input: Không có  Output: Suppliers | Lấy giá trị thuộc tính suppliers | Products.java(69) | Trương Thuận Hòa |
| 6 | setSuppliers(Suppliers suppliers)  Input: suppliers  Output: Không có | Gán giá trị cho thuộc tính suppliers | Products.java(73) | Trương Thuận Hòa |
| 7 | getProductName()  Input: Không có  Output: String | Lấy giá trị thuộc tính productName | Products.java(78) | Trương Thuận Hòa |
| 8 | setProductName(String productName)  Input: productName  Output: Không có | Gán giá trị cho thuộc tính productName | Products.java(82) | Trương Thuận Hòa |
| 9 | getPrice()  Input: Không có  Output: int | Lấy giá trị thuộc tính price | Products.java(87) | Trương Thuận Hòa |
| 10 | setPrice(int price)  Input: price  Output: Không có | Gán giá trị cho thuộc tính price | Products.java(91) | Trương Thuận Hòa |
| 11 | getRating()  Input: Không có  Output: int | Lấy giá trị thuộc tính rating | Products.java(96) | Trương Thuận Hòa |
| 12 | setRating(int rating)  Input: rating  Output: Không có | Gán giá trị cho thuộc tính rating | Products.java(100) | Trương Thuận Hòa |
| 13 | getUpdateDate()  Input: Không có  Output: Date | Lấy giá trị thuộc tính updateDate | Products.java(106) | Trương Thuận Hòa |
| 14 | setUpdateDate(Date updateDate)  Input: updateDate  Output: Không có | Gán giá trị cho thuộc tính updateDate | Products.java(110) | Trương Thuận Hòa |
| 15 | isIsSoldout()  Input: Không có  Output: boolean | Lấy giá trị thuộc tính isSoldout | Products.java(115) | Trương Thuận Hòa |
| 16 | setIsSoldout(boolean isSoldout)  Input: isSoldout  Output: Không có | Gán giá trị cho thuộc tính isSoldout | Products.java(119) | Trương Thuận Hòa |
| 17 | getProductImage()  Input: Không có  Output: String | Lấy giá trị thuộc tính productImage | Products.java(124) | Trương Thuận Hòa |
| 18 | setProductImage(String productImage)  Input: productImage  Output: Không có | Gán giá trị cho thuộc tính productImage | Products.java(128) | Trương Thuận Hòa |
| 19 | getOrderdetailses()  Input: Không có  Output: Set<Orderdetails> | Lấy giá trị thuộc tính orderdetailses | Products.java(133) | Trương Thuận Hòa |
| 20 | setOrderdetailses(Set<Orderdetails> orderdetailses)  Input: orderdetailses  Output: Không có | Gán giá trị cho thuộc tính orderdetailses | Products.java(137) | Trương Thuận Hòa |

*Bảng 3.13 – Danh sách các phương thức trong lớp Suppliers*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | getSupplierId()  Input: Không có  Output: Integer | Lấy giá trị thuộc tính supplierId | Suppliers.java(40) | Trương Thuận Hòa |
| 2 | setSupplierId(Integer supplierId)  Input: supplierId  Output: Không có | Gán giá trị cho thuộc tính supplierId | Suppliers.java(44) | Trương Thuận Hòa |
| 3 | getsupplierName()  Input: Không có  Output: String | Lấy giá trị thuộc tính supplierName | Suppliers.java(49) | Trương Thuận Hòa |
| 4 | setsupplierName(String supplierName)  Input: supplierName  Output: Không có | Gán giá trị cho thuộc tính supplierName | Suppliers.java(53) | Trương Thuận Hòa |
| 5 | getPhone()  Input: Không có  Output: String | Lấy giá trị thuộc tính phone | Suppliers.java(58) | Trương Thuận Hòa |
| 6 | setPhone(String phone)  Input: phone  Output: Không có | Gán giá trị cho thuộc tính phone | Suppliers.java(62) | Trương Thuận Hòa |
| 7 | getAddress()  Input: Không có  Output: String | Lấy giá trị thuộc tính productName | Suppliers.java(67) | Trương Thuận Hòa |
| 8 | setAddress(String address)  Input: address  Output: Không có | Gán giá trị cho thuộc tính address | Suppliers.java(71) | Trương Thuận Hòa |
| 9 | getProductses()  Input: Không có  Output: Set<Products> | Lấy giá trị thuộc tính productses | Suppliers.java(76) | Trương Thuận Hòa |
| 10 | setProductses(Set<Products> productses)  Input: productses  Output: Không có | Gán giá trị cho thuộc tính productses | Suppliers.java(80) | Trương Thuận Hòa |

*Bảng 3.14 – Danh sách các phương thức trong lớp Users*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | getUserId()  Input: Không có  Output: int | Lấy giá trị thuộc tính userId | Users.java(46) | Trương Thuận Hòa |
| 2 | setUserId(int userId)  Input: userId  Output: Không có | Gán giá trị cho thuộc tính userId | Users.java(50) | Trương Thuận Hòa |
| 3 | getUserEmail()  Input: Không có  Output: String | Lấy giá trị thuộc tính userEmail | Users.java(55) | Trương Thuận Hòa |
| 4 | setUserEmail(String userEmail)  Input: userEmail  Output: Không có | Gán giá trị cho thuộc tính userEmail | Users.java(59) | Trương Thuận Hòa |
| 5 | getUserPassword()  Input: Không có  Output: String | Lấy giá trị thuộc tính userPassword | Users.java(64) | Trương Thuận Hòa |
| 6 | setUserPassword(String userPassword)  Input: userPassword  Output: Không có | Gán giá trị cho thuộc tính userPassword | Users.java(68) | Trương Thuận Hòa |
| 7 | getUserName()  Input: Không có  Output: String | Lấy giá trị thuộc tính userName | Users.java(73) | Trương Thuận Hòa |
| 8 | setUserName(String userName)  Input: userName  Output: Không có | Gán giá trị cho thuộc tính userName | Users.java(77) | Trương Thuận Hòa |
| 9 | getUserAddress()  Input: Không có  Output: String | Lấy giá trị thuộc tính userAddress | Users.java(82) | Trương Thuận Hòa |
| 10 | setUserAddress(String userAddress)  Input: userAddress  Output: Không có | Gán giá trị cho thuộc tính userAddress | Users.java(86) | Trương Thuận Hòa |
| 11 | getUserPhone()  Input: Không có  Output: String | Lấy giá trị thuộc tính userPhone | Users.java(91) | Trương Thuận Hòa |
| 12 | setUserPhone(String userPhone)  Input: userPhone  Output: Không có | Gán giá trị cho thuộc tính userPhone | Users.java(95) | Trương Thuận Hòa |
| 13 | getRegistrationDate()  Input: Không có  Output: Date | Lấy giá trị thuộc tính registrationDate | Users.java(101) | Trương Thuận Hòa |
| 14 | setRegistrationDate(Date registrationDate)  Input: registrationDate  Output: Không có | Gán giá trị cho thuộc tính registrationDate | Users.java(105) | Trương Thuận Hòa |
| 15 | getOrderses()  Input: Không có  Output: Set<Orders> | Lấy giá trị thuộc tính orderses | Users.java(110) | Trương Thuận Hòa |
| 16 | setOrderses(Set<Orders> orderses)  Input: orderses  Output: Không có | Gán giá trị cho thuộc tính orderses | Users.java(114) | Trương Thuận Hòa |

*Bảng 3.15 – Danh sách các phương thức trong lớp CategoryDAO*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | insert(Categories entity)  Input: entity  Output: Categories | Thêm dữ liệu vào database table categories | CategoryDAO.java(8) | Phan Hồng Sơn |
| 2 | update(Categories entity)  Input: entity  Output: Categories | Cập nhật dữ liệu mới vào database table categories | CategoryDAO.java(14) | Phan Hồng Sơn |
| 3 | delete(Categories entity)  Input: entity  Output: Categories | Xóa dữ liệu khỏi database table categories | CategoryDAO.java(20) | Phan Hồng Sơn |
| 4 | select(int id)  Input: id  Output: Categories | Lấy dữ liệu trong database table categories | CategoryDAO.java(26) | Phan Hồng Sơn |
| 5 | close()  Input: Không có  Output: Không có | Đóng kết nối SessionFactory | CategoryDAO.java(32) | Phan Hồng Sơn |

*Bảng 3.16 – Danh sách các phương thức trong lớp HibernateDAO*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | insert(E entity)  Input: entity  Output: E | Thêm dữ liệu vào database table bất kì bằng cách sử dụng kiểu Generic E | HibernateDAO.java(20) | Phan Hồng Sơn |
| 2 | update(E entity)  Input: entity  Output: E | Cập nhật dữ liệu mới vào database table bất kì bằng cách sử dụng kiểu Generic E | HibernateDAO.java(42) | Trương Thuận Hòa |
| 3 | delete(E entity)  Input: entity  Output: E | Xóa dữ liệu khỏi database table bất kì bằng cách sử dụng kiểu Generic E | HibernateDAO.java(61) | Trương Thuận Hòa |
| 4 | select(Class<E> classE, int id)  Input: classE, id  Output: E | Lấy dữ liệu trong database table bất kì bằng cách sử dụng kiểu Generic E | HibernateDAO.java(78) | Trương Thuận Hòa |
| 5 | close()  Input: Không có  Output: Không có | Đóng kết nối SessionFactory | HibernateDAO.java(101) | Trương Thuận Hòa |

*Bảng 3.17 – Danh sách các phương thức trong lớp OrderDAO*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | insert(Orders entity)  Input: entity  Output: Orders | Thêm dữ liệu vào database table orders | OrderDAO.java(11) | Phan Hồng Sơn |
| 2 | update(Orders entity)  Input: entity  Output: Orders | Cập nhật dữ liệu mới vào database table orders | OrderDAO.java(17) | Phan Hồng Sơn |
| 3 | delete(Orders entity)  Input: entity  Output: Orders | Xóa dữ liệu khỏi database table categories | OrderDAO.java(23) | Phan Hồng Sơn |
| 4 | select(int id)  Input: id  Output: Orders | Lấy dữ liệu trong database table orders | OrderDAO.java(29) | Phan Hồng Sơn |
| 5 | selectByUser(Users users) | Lấy thông tin đơn hàng dựa trên đối tượng Khách hàng được truyền vào | OrderDAO.java(34) | Trương Thuận Hòa |
| 6 | close()  Input: Không có  Output: Không có | Đóng kết nối SessionFactory | HibernateDAO.java(101) | Phan Hồng Sơn |

*Bảng 3.18 – Danh sách các phương thức trong lớp OrderdetailDAO*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | insert(Orderdetails entity)  Input: entity  Output: Orderdetails | Thêm dữ liệu vào database table orderdetails | OrderdetailDAO.java(8) | Phan Hồng Sơn |
| 2 | update(Orderdetails entity)  Input: entity  Output: Orderdetails | Cập nhật dữ liệu mới vào database table orderdetails | OrderdetailDAO.java(14) | Phan Hồng Sơn |
| 3 | delete(Orderdetails entity)  Input: entity  Output: Orderdetails | Xóa dữ liệu khỏi database table orderdetails | OrderdetailDAO.java(20) | Phan Hồng Sơn |
| 4 | select(int id)  Input: id  Output: Orderdetails | Lấy dữ liệu trong database table orderdetails | OrderdetailDAO.java(26) | Phan Hồng Sơn |
| 5 | close()  Input: Không có  Output: Không có | Đóng kết nối SessionFactory | OrderdetailDAO.java(32) | Phan Hồng Sơn |

*Bảng 3.19 – Danh sách các phương thức trong lớp ProductDAO*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | insert(Products entity)  Input: entity  Output: Products | Thêm dữ liệu vào database table products | ProductDAO.java (13) | Phan Hồng Sơn |
| 2 | update(Products entity)  Input: entity  Output: Products | Cập nhật dữ liệu mới vào database table products | ProductDAO.java (18) | Phan Hồng Sơn |
| 3 | delete(Products entity)  Input: entity  Output: Products | Xóa dữ liệu khỏi database table products | ProductDAO.java (23) | Phan Hồng Sơn |
| 4 | select(int id)  Input: id  Output: Products | Lấy dữ liệu trong database table products | ProductDAO.java (28) | Phan Hồng Sơn |
| 5 | selectByCategory(Integer categoryid)  Input: categoryid  Output: List<Products> | Lấy danh sách các sản phẩm dựa theo ID danh mục hàng hóa | ProductDAO.java (32) | Trương Thuận Hòa |
| 6 | selectByCategoryName(String categoryName)  Input: categoryName  Output: List<Products> | Lấy danh sách các sản phẩm dựa theo tên danh mục hàng hóa | ProductDAO.java (44) | Phan Hồng Sơn |
| 7 | searchProduct(String string)  Input: string  Output: List<Products> | Tìm kiếm sản phẩm dựa trên từ khóa được gửi từ browser | ProductDAO.java (57) | Trương Thuận Hòa |

*Bảng 3.20 – Danh sách các phương thức trong lớp SupplierDAO*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | insert(Suppliers entity)  Input: entity  Output: Suppliers | Thêm dữ liệu vào database table suppliers | SupplierDAO.java (8) | Phan Hồng Sơn |
| 2 | update(Suppliers entity)  Input: entity  Output: Suppliers | Cập nhật dữ liệu mới vào database table suppliers | SupplierDAO.java (14) | Phan Hồng Sơn |
| 3 | delete(Suppliers entity)  Input: entity  Output: Suppliers | Xóa dữ liệu khỏi database table suppliers | SupplierDAO.java (20) | Phan Hồng Sơn |
| 4 | select(int id)  Input: id  Output: Suppliers | Lấy dữ liệu trong database table suppliers | SupplierDAO.java (26) | Phan Hồng Sơn |
| 5 | close()  Input: Không có  Output: Không có | Đóng kết nối SessionFactory | SupplierDAO.java (32) | Phan Hồng Sơn |

*Bảng 3.21 – Danh sách các phương thức trong lớp UserDAO*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | insert(Users entity)  Input: entity  Output: Users | Thêm dữ liệu vào database table users | UserDAO.java (11) | Phan Hồng Sơn |
| 2 | update(Users entity)  Input: entity  Output: Users | Cập nhật dữ liệu mới vào database table users | UserDAO.java (16) | Phan Hồng Sơn |
| 3 | delete(Users entity)  Input: entity  Output: Users | Xóa dữ liệu khỏi database table users | UserDAO.java (21) | Phan Hồng Sơn |
| 4 | select(int id)  Input: id  Output: Users | Lấy dữ liệu trong database table users | UserDAO.java (26) | Phan Hồng Sơn |
| 5 | selectByEmail(String email)  Input: email  Output: Users | Lấy thông tin Khách hàng dựa trên email được truyền vào | UserDAO.java (31) | Trương Thuận Hòa |
| 6 | isExist(String userEmail)  Input: userEmail  Output: boolean | Kiểm tra email Khách hàng có tồn tại dưới Cơ sở dữ liệu hay không | UserDAO.java (49) | Trương Thuận Hòa |
| 7 | isValidate(String userEmail, String userPassword)  Input: userEmail, userPassword  Output: boolean | Chứng thực email và password khi Khách hàng đăng nhập vào hệ thống website | UserDAO.java (69) | Trương Thuận Hòa |

*Bảng 3.21 – Danh sách các phương thức trong lớp AddToCartServlet*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | doGet(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Thêm sản phẩm vào danh sách giỏ hàng sau khi Browser gửi request đến Controller theo phương thức get | AddToCartServlet.java (32) | Trương Thuận Hòa |
| 2 | isLogin(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: boolean | Kiểm tra Khách hàng đã đăng nhập vào hệ thống website hay chưa | AddToCartServlet.java (95) | Trương Thuận Hòa |
| 3 | caculateTotalAmount(List<Orderdetails> orderList)  Input: orderList  Output: int | Tính tổng số tiền cần thanh toán cho các sản phẩm hiện tại đang có trong giỏ hàng | AddToCartServlet.java (103) | Phan Hồng Sơn |
| 4 | destroy()  Input: Không có  Output: Không có | Hủy các lớp DAO | AddToCartServlet.java (112) | Trương Thuận Hòa |

*Bảng 3.22 – Danh sách các phương thức trong lớp CheckOutServlet*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | doGet(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Khi gửi request đến Servlet chứa phương thức này theo phương thức get, chuyển Khách hàng đến trang đăng nhập nếu Khách hàng chưa đăng nhập, chuyển Khách hàng đến trang giỏ hàng khi Khách hàng đã đăng nhập vào hệ thống website | CheckOutServlet.java (24) | Trương Thuận Hòa |
| 2 | isLogin(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: boolean | Kiểm tra Khách hàng đã đăng nhập vào hệ thống website hay chưa | CheckOutServlet.java (32) | Trương Thuận Hòa |
| 3 | destroy()  Input: Không có  Output: Không có | Hủy các lớp DAO | CheckOutServlet.java (51) | Trương Thuận Hòa |

*Bảng 3.23 – Danh sách các phương thức trong lớp ContactServlet*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | doGet(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Gửi phản hồi và yêu cầu của Khách hàng đến email người quản lý website khi Controller nhận được request được gửi theo phương thức get từ Browser | ContactServlet.java (15) | Trương Thuận Hòa |

*Bảng 3.24 – Danh sách các phương thức trong lớp LoginUserServlet*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | doGet(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Trả về trang chủ khi Khách hàng cố tình gửi dữ liệu bằng cách nhập vào thanh URL trên Browser | LoginUserServlet. java (25) | Trương Thuận Hòa |
| 2 | init()  Input: Không có  Output: Không có | Khởi tạo đối tượng của lớp UserDAO ngay khi lớp chứa phương thức này được gọi thực thi | LoginUserServlet. java (20) | Trương Thuận Hòa |
| 3 | doPost(HttpServletRequest request, HttpServletResponse response  Input: request, response  Output: Không có | Xử lý request được gửi đến từ Browser theo phương thước post sau khi Khách hàng tiến hành đăng nhập vào hệ thống website | LoginUserServlet. java (34) | Trương Thuận Hòa |
| 4 | checkAccount(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Chứng thực thông tin đăng nhập của người dùng, chuyển Khách hàng đến Trang chủ sau khi đăng nhập thành công, chuyển Khách hàng đến trang đăng nhập để tiến hành đăng nhập lại khi việc chứng thực thông tin đăng nhập thất bại | LoginUserServlet. java (46) | Trương Thuận Hòa |
| 5 | destroy()  Input: Không có  Output: Không có | Hủy các lớp DAO | LoginUserServlet. java (65) | Trương Thuận Hòa |

*Bảng 3.25 – Danh sách các phương thức trong lớp LogoutUserServlet*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | doGet(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Hủy Session đăng nhập của Khách hàng khi Browser gửi request đến Controller theo phương thức get. Phương thức này giúp Khách hàng đăng xuất khỏi tài khoản hiện tại trên hệ thống website | LogoutUserServlet. java (22) | Trương Thuận Hòa |

*Bảng 3.26 – Danh sách các phương thức trong lớp OrderServlet*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | doGet(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Gọi đến phương thức doPost để phương thức doPost là phương thức xử lý request được gửi ở cả 2 dạng là get và post | OrderServlet. java (40) | Trương Thuận Hòa |
| 2 | init()  Input: Không có  Output: Không có | Khởi tạo các đối tượng DAO ngay khi lớp chứa phương thức này được gọi thực thi | OrderServlet. java (32) | Trương Thuận Hòa |
| 3 | doPost(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Lưu thông tin đặt hàng của Khách hàng khi Browser phía Khách hàng gửi request đặt hàng đến Controller theo phương thức post. | OrderServlet. java (45) | Trương Thuận Hòa |
| 4 | sendEmail(Users user, Orders orders, List<Orderdetails> orderlist)  Input: user, orders, orderlist  Output: boolean | Gửi email thông báo xác nhận thông tin về đơn hàng đến Khách hàng sau khi Khách hàng đặt hàng thành công | OrderServlet. java (86) | Trương Thuận Hòa |
| 5 | isLogin(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: boolean | Kiểm tra trạng thái đăng nhập của Khách hàng | OrderServlet. java (107) | Trương Thuận Hòa |
| 6 | destroy()  Input: Không có  Output: Không có | Hủy các lớp DAO | OrderServlet. java (113) | Trương Thuận Hòa |

*Bảng 3.27 – Danh sách các phương thức trong lớp RegisterUserServlet*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | doGet(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Trả về trang chủ khi Khách hàng cố tình gửi dữ liệu bằng cách nhập vào thanh URL trên Browser | RegisterUserServlet. java (30) | Trương Thuận Hòa |
| 2 | init()  Input: Không có  Output: Không có | Khởi tạo đối tượng của lớp UserDAO ngay khi lớp chứa phương thức này được gọi thực thi | RegisterUserServlet. java (25) | Trương Thuận Hòa |
| 3 | doPost(HttpServletRequest request, HttpServletResponse response  Input: request, response  Output: Không có | Xử lý request được gửi đến từ Browser theo phương thước post sau khi Khách hàng tiến hành đăng ký tài khoản trên hệ thống website | RegisterUserServlet. java (36) | Trương Thuận Hòa |
| 4 | createAccount(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Tạo tài khoản mới cho Khách hàng sau khi kiểm tra email đăng ký của Khách hàng chưa tồn tại trong Cơ sở dữ liệu, nếu email đã tồn tại, tạo tài khoản thất bại, chuyển Khách hàng về trang Login | RegisterUserServlet. java (45) | Trương Thuận Hòa |
| 5 | destroy()  Input: Không có  Output: Không có | Hủy các lớp DAO | RegisterUserServlet. java (71) | Trương Thuận Hòa |

*Bảng 3.28 – Danh sách các phương thức trong lớp RemoveFromCartServlet*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | doGet(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Xóa sản phẩm khỏi danh sách giỏ hàng trên hệ thống website đang có phạm vi là Session sau khi Browser phía Khách hàng gửi request xóa phẩm đến theo phương thức get. Nếu Khách hàng chưa đăng nhập, xóa sản phẩm thất bại, chuyển Khách hàng về trang Login | RemoveFromCartServlet. java (23) | Phan Hồng Sơn |
| 2 | isLogin(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: boolean | Kiểm tra trạng thái đăng nhập của Khách hàng | RemoveFromCartServlet. java (50) | Phan Hồng Sơn |
| 3 | caculateTotalAmount(List<Orderdetails> orderList)  Input: orderList  Output: int | Tính tổng số tiền cần thanh toán cho các sản phẩm hiện tại đang có trong giỏ hàng | RemoveFromCartServlet. java (57) | Phan Hồng Sơn |

*Bảng 3.29 – Danh sách các phương thức trong lớp RenderProductDataServlet*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | doGet(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Hiển thị danh sách các sản phẩm theo danh mục lên website khi Khách hàng gửi request theo phương thức get đến Controller sau thao tác click vào danh mục hàng hóa | RenderProductDataServlet. java (25) | Phan Hồng Sơn |
| 2 | init()  Input: Không có  Output: Không có | Khởi tạo đối tượng của lớp ProductDAO ngay khi lớp chứa phương thức này được gọi thực thi | RenderProductDataServlet. java (21) | Phan Hồng Sơn |

*Bảng 3.30 – Danh sách các phương thức trong lớp SearchProductServet*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | doGet(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Hiển thị danh sách các sản phẩm dựa trên từ khóa Khách hàng nhập vào trong mục tìm kiếm sau khi request được gửi từ Browser Khách hàng đến theo phươn thức get | SearchProductServet. java (20) | Trương Thuận Hòa |

*Bảng 3.30 – Danh sách các phương thức trong lớp UpdateQuantityServlet*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, số thứ tự dòng chứa khai báo | Phụ trách |
| 1 | doGet(HttpServletRequest request, HttpServletResponse response)  Input: request, response  Output: Không có | Tăng giảm số lượng loại sản phẩm đã có trong giỏ hàng khi Browser gửi request theo phương thức get đến Controller, khi số lượng một sản phẩm nào đó về 0, loại sản phẩm đó sẽ được xóa khỏi giỏ hàng | UpdateQuantityServlet. java (22) | Phan Hồng Sơn |
| 2 | caculateTotalAmount(List<Orderdetails> orderList)  Input: orderList  Output: int | Tính tổng số tiền cần thanh toán cho các sản phẩm hiện tại đang có trong giỏ hàng | UpdateQuantityServlet. java (59) | Phan Hồng Sơn |

* 1. **Thiết kế giao diện**

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Màn hình giao diện | Mục đích | Mô tả |
| 1 | *Hình 5 - Giao diện trang chủ (1)*  Graphical user interface, website  Description automatically generated  *Hình 3.3 - Giao diện trang chủ (2)*  Website  Description automatically generated | Tạo trang chủ giới thiệu cho hệ thống website (trang *index.jsp*) | Phan Hồng Sơn  Mô tả: Bao gồm phần header hiển thị tên website và chứa thanh danh mục các loại hàng hóa, thanh danh mục các chức năng. Phần footer hiển thị các đường dẫn đến các chức năng và thông tin liên hệ. Phần body chứa các hình ảnh minh họa cho website và logo các nhãn hàng |
| 2 | *Hình 6 - Giao diện trang chủ sau khi đăng nhập thành công*  A screenshot of a video game  Description automatically generated | Hiển thị tên và trạng thái hoạt động của Khách hàng sau khi đăng nhập thành công | Phan Hồng Sơn  Mô tả: Sau khi đăng nhập thành công, hiển thị tên và trạng thái hoạt động thay cho 3 nút *Create Account, Login, Help* |
| 3 | *Hình 7 - Giao diện trang sản phẩm*  Graphical user interface, website  Description automatically generated | Hiển thi danh sách các sản phẩm trong danh mục hàng hóa (trang groceries.jsp, household.jsp,  personalcare .jsp, packagedfood .jsp, beverages.jsp, gourmet.jsp) | Phan Hồng Sơn  Cấu trúc trang bao gồm Header và Footer tương tự trang chủ, phần body chứa thông tin các sản phẩm và một Container chứa các đường dẫn đến các trang sản phẩm khác phía bên trái. |
| 4 | *Hình 8 - Giao diện trang sản phẩm Special*  Graphical user interface, application  Description automatically generated | Hiển thị danh sách các sản phẩm đặc biệt | Phan Hồng Sơn  Bao gồm hai chế độ hiển thị sản phẩm ở phần body là chế độ hiển thị tất cả sản phẩm và chế độ hiển thị các sản phẩm đặc biệt trong ngày, trang này cũng chứa Header và Footer tương tự như ở các trang trước đó. |
| 5 | *Hình 9 - Giao diện trang đăng ký*  Graphical user interface, website  Description automatically generated | Tạo giao diện cho Khách hàng đăng ký tài khoản sử dụng hệ thống website | Trương Thuận Hòa  Bao gồm Header và Footer chứa các thông tin giới thiệu và chức năng cần thiết như ở các trang trước. Phần body chứa form đăng ký |
| 6 | *Hình 10 - Giao diện trang đăng nhập*  Graphical user interface, application, website, PowerPoint  Description automatically generated | Tạo giao diện cho Khách hàng đăng nhập tài khoản đã đăng ký trước đó | Trương Thuận Hòa  Bao gồm Header và Footer chứa các thông tin giới thiệu và chức năng cần thiết như ở các trang trước. Phần body chứa form đăng nhập và đường dẫn dẫn đến trang Đăng ký |
| 7 | *Hình 11 - Giao diện trang liên hệ và giúp đỡ*  Graphical user interface  Description automatically generated | Tạo giao diện cho phép Khách hàng gửi phản hồi và yêu cầu đến email người quản lý | Bao gồm Header và Footer giống với các trang trước đó. Phần body hiển thị ảnh bản đồ vị trí của doanh nghiệp/cửa hàng và form gửi phản hồi |
| 8 | *Hình 12 - Giao diện trang tìm kiếm sản phẩm*  Graphical user interface, application, website  Description automatically generated | Trang này hiển thị các sản phẩm sau khi Khách hàng nhập từ khóa để tìm kiếm sản phẩm mong muốn | Trương Thuận Hòa  Cấu trúc trang tương tự trang hiển thị sản phẩm |
| 9 | *Hình 13 - Giao diện trang FAQs*  Graphical user interface, text  Description automatically generated | Hiển thị các câu hỏi thường gặp của Khách hàng | Phan Hồng Sơn  Cấu trúc gần tương tự các trang trước, gồm Header và Footer mô tả các thông tin, phần body chứa các câu hỏi |
| 10 | *Hình 14 - Giao diện trang About us (1)*  Graphical user interface, website  Description automatically generated  *Hình 15 - Giao diện trang About us (2)*  Graphical user interface, website  Description automatically generated | Chứa các thông tin về hệ thống website, về cửa hàng /doanh nghiệp và đội ngũ phát triển website | Phan Hồng Sơn  Phần body chứa các hình ảnh và văn bản mô tả, đồng thời chứa ảnh đội ngũ phát triển website |
| 11 | *Hình 16 - Giao diện trang Giỏ hàng*  Graphical user interface, website  Description automatically generated | Hiển thị các sản phẩm Khách hàng đã chọn, cho phép Khách hàng tăng /giảm số lượng sản phẩm, nhập thông tin đơn đặt hàng và tiến hành đặt hàng. | Bao gồm danh sách các sản phẩm Khách hàng đã thêm vào giỏ hàng, form hiển thị tổng số tiền các sản phẩm góc trái và form nhập thông đặt hàng góc phải |

1. **Cài đặt và kiểm thử**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tình huống** | **Mục đích** | **Giải thích** |
| 1 | Tình huống 1: Tạo tài khoản với email chưa được đăng kí. Tình huống 2: Tạo tài khoản với email đã được đăng kí.  Dự kiến:   - Tình huống 1: Tài khoản thành công sẽ chuyển về trang login gửi email xác nhận đã tạo tài khoản thành công.  - Tình huống 2: Tạo tài khoản không thành công, xóa hết thông tin tài khoản đã điền, và điền lại thông tin mới. | Kiểm tra tính năng đăng kí. | Kiểm tra việc lấy email từ cơ sỡ dữ liệu và kiểm tra email đã được đăng kí hay chưa. |
| 2 | Tình huống 1: Đăng nhập với tài khoản đã được đăng kí với mật khẩu đúng. Tình huống 2: Đăng nhập với tài khoản hoặc mật khẩu không đúng hay chưa được đăng kí.  Dự kiến: chỉ cho phép đăng nhập với thông tin tài khoản mật khẩu đúng với với thông tin đã đăng kí. | Kiểm tra tính năng đăng nhập. | Đăng nhập với sản phẩm đã đăng kí. Kiểm tra email đã tồn tại dưới cơ sỡ dữ liệu chưa và kiểm tra mật khẩu đã đúng với thông tin lúc đăng kí chưa. |
| 3 | Tình huống 1: Chưa đăng nhập.  Tình huống 2: Đăng nhập rồi.  Dự kiến: Khi chăng đăng nhập nút logout không hiện ra. | Kiểm tra tính năng logout. | Kiểm tra id khách hàng đã được lưu trong session hay chưa. Nếu đã lưu sẽ hiện nút logout để người dùng có thể đăng xuất. |
| 4 | Tình huống 1: Thêm sản phẩm vào giỏ hàng nhưng chưa đăng nhập. Tình huống 2: Thêm sản phẩm vào giỏ hàng nhưng đã đăng nhập.  Dự kiến: Chỉ cho phép thêm sản phẩm vào giỏ hàng khi đã đăng nhập. Nếu chưa đăng nhập sẽ chuyển đến trang đăng nhập. | Kiểm tra tính năng thêm vào giỏ hàng. | Khi thêm sản phẩm vào giỏ hàng cần phải đăng nhập để có thể xác nhập được giỏ hàng của khách hàng nào. |
| 5 | Tình huống 1: Bấm vào giỏ hàng nhưng chưa đăng nhập. Tình huống 2: Bấm vào giỏ hàng nhưng đã đăng nhập rồi.  Dự kiến: Khi bấm vào giỏ hàng nhưng chưa đăng nhập, website sẽ tự động chuyển hướng qua trang đăng nhập để người dùng đăng nhập tài khoản. Nếu đăng nhập rồi thì khách hàng được phép vào trang giỏ hàng. | Kiểm tra tính năng chuyển trang giỏ hàng. | Vì khi chưa đăng nhập khách hàng chưa có giỏ hàng của riêng mình nên không được phép vào trang giỏ hàng. |
| 6 | Tình huống 1: Bấm nút PAY trong trang giỏ hàng khi chưa thêm sản phẩm nào vào giỏ hàng  Tình huống 2: Bấm nút PAY vào trang giỏi hàng khi đã có sản phẩm trong giỏ hàng.  Dự kiến: Khi chưa có sản phẩm người dùng bấm nút PAY sẽ không có hành động nào xảy ra. Còn nếu đã có sản phẩm một mail tự động được gửi đến mail của người dùng. | Kiểm tra tính năng thanh toán của khách hàng. | Khi chưa có sản phẩm khách hàng không thể order được. |

1. **Kết luận**
   1. **Kết quả đạt được**

Do thời gian thực hiện đề tài có giới hạn và kiến thức thực hiện đề tài được chúng em tích lũy trong song song với thời gian làm đề tài nên một số tính năng chưa được hoàn thiện cũng như không kịp thêm vào một số tính năng mở rộng.

Em đã thiết kế một website mua hàng online với các chức năng cơ bản:

* Xem thông tin sản phẩm.
* Thanh chia sản phẩm theo loại sản phẩm.
* Thêm sản phẩm vào giỏ hàng.
* Đăng kí, đăng nhập, đăng xuất.
* Chức năng giỏ hàng.
* Chức năng gửi email xác nhận.
* Chức năng thanh toán hóa đơn.
* Chức năn gửi đơn kiếu nại.
* Chức năng tìm kiếm sản phẩm.
  1. **Hạn chế**
* Chưa có trang admin để quản lý website.
* Chưa có tính năng riêng cho người dùng tạo cửa hàng trên website.
* Email tự động chưa được thiết kế.
* Chưa có tính năng thanh toán.
* Chưa hiện thông báo rõ ràng đối với mỗi hành động của khách hàng (khi đăng nhập sai tài khoản mật khẩu, order thành công, …)
  1. **Kinh nghiệm**
* Có thêm kinh nghiệm về tạo cơ sở dữ liệu để cho một website ecommerce để có thể lưu trữ và truy xuất thuận tiện hơn.
* Biết cách thêm các API từ bên thứ 3 để phát triển các tính năng mới của website.
* Có kinh nghiệm làm việc trên môi trường github.
* Tăng khả năng làm việc nhóm và phân chia thời gian hợp lý.
  1. **Hướng phát triển đề tài**
* Xây dựng thêm trang admin để dễ dàng quản lý sản phẩm
* Xây dựng thêm tính năng cho phép người dùng mở của hàng và đăng sản phẩm của mình lên website
* Thiết kế lại email tự động (gửi một email dạng html thay vì dạng text cơ bản)
* Thêm các thông báo khi đăng nhập đăng xuất thêm vào giỏ hàng và thanh toán
* Thêm chứ năng thanh toán (thông qua paypal)
* Thêm chức năng trả lời mail tự động khi khách hàng đặt câu hỏi

1. **Tài liệu tham khảo**

Website lấy template: <https://dribbble.com/>

Kiến thức mô hình MVC: <https://www.youtube.com/watch?v=N8GhaR7K3tI>

Nguồn tài liệu HTML/CSS: <https://www.w3schools.com/>

Nguồn học Hibernate: <https://www.youtube.com/watch?v=J36jLRIvLE0>

Hỗ trợ khi gặp vấn đề/lỗi: <https://stackoverflow.com/>

Các trang tham khảo code:

<https://www.tutorialspoint.com/hibernate/hibernate_native_sql.htm>

<https://www.javaguides.net/2019/02/hibernate-crud-example.html>