

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO**

**ĐỒ ÁN TỐT NGHIỆP**

*Tên đề tài:*

**Xây dựng và phát triển website Bán đồ gia dụng**

*Giảng viên hướng dẫn*: Th.s Trần Quang Quý

*Sinh viên thực hiện***:** Nguyễn Minh Anh

*Lớp: CNTT – K17L*

Thái Nguyên 2023

# LỜI CẢM ƠN

Lời đầu tiên em xin chân thành cảm ơn các thầy, cô trong khoa Công nghệ thông tin, trường Đại học Công nghệ truyền thông và thông tin Thái Nguyên đã tạo điều kiện thuận lợi cho em trong quá trình học tập tại trường cũng như trong thời gian thực hiện báo cáo thực tập tốt nghiệp.

Đặc biệt, em muốn gửi lời cảm ơn tới Th.s Trần Quang Quý – giảng viên hướng dẫn, đã chỉ bảo, giúp đỡ em khắc phục những khó khăn, thiếu sót để có thể hoàn thành các phần trong báo cáo.

Dù đã rất cố gắng thực hiện báo cáo thực tập tốt nghiệp, nhưng do thời gian, kinh nghiệm của bản thân em còn hạn chế nên không thể tránh khỏi có thiếu sót. Kính mong nhận được sự đóng góp ý kiến từ phía thầy cô, bạn bè để em có thể nâng cao kiến thức của bản thân, phát triển tốt hơn.

*Em xin chân thành cảm ơn!*

|  |  |
| --- | --- |
|  | *Thái Nguyên, tháng năm 2023*  Sinh viên thực hiện  Anh  Nguyễn Minh Anh |

# LỜI CAM ĐOAN

Em xin cam đoan rằng đồ án này là công trình nghiên cứu của bản thân mình. Những phần có sử dụng tài liệu tham khảo có trong đồ án đã được liệt kê và nêu rõ ra tại phần tài liệu tham khảo. Đồng thời những số liệu hay kết quả trình bày trong đồ án đều mang tính chất trung thực.

Nếu như sai em xin chịu hoàn toàn trách nhiệm và chịu tất cả các kỷ luật của bộ môn cũng như nhà trường đề ra.

**MỤC LỤC**

[**LỜI CẢM ƠN** 1](#_Toc129591543)

[**LỜI CAM ĐOAN** 2](#_Toc129591544)

[**danh mục hình ảnh** 5](#_Toc129591546)

[**Lời nói đầu** 6](#_Toc129591547)

[**Chương I. Cơ sở lý thuyết** 8](#_Toc129591548)

[**1.1. Giới thiệu về ngôn ngữ** 8](#_Toc129591549)

[***1.1.1. HTML (HyperText Markup Language)*** 8](#_Toc129591550)

[***1.1.2. CSS (Cascading Style Sheets)*** 8](#_Toc129591551)

[***1.1.3. JavaScript*** 10](#_Toc129591552)

[***1.1.4. PHP*** 11](#_Toc129591553)

[**1.2. Hệ quản trị cơ sở dữ liệu(MySQL)** 16](#_Toc129591563)

[**1.3. Giới thiệu Visual Studio Code** 17](#_Toc129591565)

[***1.3.1. Tổng quan*** 17](#_Toc129591566)

[***1.3.2. Điểm mạnh của Visual Studio Code*** 18](#_Toc129591567)

[***1.3.3. Các tính năng chính với ưu điểm mạnh mẽ của Visual Studio Code*** 18](#_Toc129591568)

[***1.3.4. So sánh điểm mạnh của Visual Studio Code với Sublime Text*** 20](#_Toc129591569)

[**1.4. Xampp** 20](#_Toc129591570)

[**1.5. Tổng quan mô hình MVC** 21](#_Toc129591571)

[***1.5.1. Các thành phần trong MVC*** 21](#_Toc129591572)

[***1.5.2. Cách vận hành của MVC*** 21](#_Toc129591573)

[***1.5.3. Sơ đồ tổng quan MVC*** 22](#_Toc129591574)

[***1.5.4. Ưu, nhược điểm của MVC*** 23](#_Toc129591575)

[**Chương II: Khảo sát và phân tích thiết kế hệ thống** 24](#_Toc129591576)

[**2.1. Khảo sát hiện trạng** 24](#_Toc129591577)

[**2.2. Giải quyết bài toán** 24](#_Toc129591578)

[**2.3. Giới thiệu website** 24](#_Toc129591579)

[**2.4. Công nghệ sử dụng** 25](#_Toc129591580)

[**2.5. Sơ đồ phân cấp chức năng** 26](#_Toc129591581)

[**2.6. Sơ đồ phân cấp chức năng phía người dùng** 27](#_Toc129591582)

[**2.7. Chức năng phía người quản trị website** 27](#_Toc129591583)

[**a. Chi tiết chức năng** 27](#_Toc129591584)

[**b. Sơ đồ phân cấp chức năng phía người quản trị** 29](#_Toc129591585)

[**2.8. Tóm tắt các đối tượng quản lý** 29](#_Toc129591586)

[***2.8.1. Sản phẩm*** 29](#_Toc129591587)

[***2.8.2. Danh mục*** 29](#_Toc129591588)

[***2.8.3. Admin*** 30](#_Toc129591589)

[***2.8.4. Khách hàng*** 30](#_Toc129591590)

[***2.8.5. Đơn hàng*** 30](#_Toc129591591)

[***2.8.6. Đánh giá*** 30](#_Toc129591592)

[**2.9. Tài liệu đặc tả và các sơ đồ cho chức năng** 31](#_Toc129591593)

[**2.10. Mô hình thực thể liên kết** 63](#_Toc129591656)

[**2.11. Biểu đồ lớp** 64](#_Toc129591657)

[**2.12. Thiết kế các bảng dữ liệu** 64](#_Toc129591658)

[**Chương III. Giao diện demo chương trình** 67](#_Toc129591663)

[**3.1. Một số giao diện trang chủ** 67](#_Toc129591664)

[**3.2. Chức năng thêm sản phẩm mới** 68](#_Toc129591665)

[**3.3. Chức năng quản lý hóa đơn** 69](#_Toc129591667)

[**3.4. Chức năng đăng nhập** 70](#_Toc129591669)

[**3.5. Giao diện giỏ hàng** 70](#_Toc129591670)

[**3.6. Giao diện thanh toán** 71](#_Toc129591671)

[**3.7. Giao diện đánh giá sản phẩm** 71](#_Toc129591673)

[**Kết luận** 72](#_Toc129591675)

[**Tài liệu tham khảo** 74](#_Toc129591676)

[**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN** 75](#_Toc129591677)

[**GIÁO VIÊN HƯỚNG DẪN** 75](#_Toc129591678)

**danh mục hình ảnh**

[Hình 1. 1 Ví dụ code HTML 8](#_Toc129621197)

[Hình 1. 2 Ví dụ viết CSS trong HTML 9](#_Toc129621198)

[Hình 1. 3 Ví dụ về Visual Studio Code 18](#_Toc129621199)

[Hình 1. 4 Sơ đồ tổng quan về MVC 22](#_Toc129621200)

[Hình 2. 1 Sơ đồ phân cấp chức năng 26](#_Toc129621201)

[Hình 2. 2 Sơ đồ phân cấp chức năng phía người dùng 27](#_Toc129621202)

[Hình 2. 3 Sơ đồ phân cấp chức năng phía quản trị viên 29](#_Toc129621203)

[Hình 3. 1 Giao diện trang chủ 67](#_Toc129621204)

[Hình 3. 2 Giao diện trang chủ 67](#_Toc129621205)

[Hình 3. 3 Giao diện trang chủ 68](#_Toc129621206)

[Hình 3. 4 Giao diện quản lý của quản trị viên 68](#_Toc129621207)

[Hình 3. 5 Giao diện thêm sản phẩm 69](#_Toc129621208)

[Hình 3. 6 Giao diện quản lý hóa đơn 69](#_Toc129621209)

[Hình 3. 7 Giao diện chức năng đăng nhập 70](#_Toc129621210)

[Hình 3. 8 Giao diện giỏ hàng 70](#_Toc129621211)

[Hình 3. 9 Giao diện thanh toán 71](#_Toc129621212)

[Hình 3. 10 Giao diện đánh giá sản phẩm 71](#_Toc129621213)

**Lời nói đầu**

Như bạn đã biết, kể từ khi dịch bùng phát đến nay, các trang thương mại điện tử (TMĐT) tăng trưởng ít nhất 20% so với những tháng cuối năm 2019, cá biệt một số trang mua sắm có mức tăng trưởng lên đến 150% so với ngày thường. Dịch COVID-19 đang làm người tiêu dùng thay đổi thói quen mua sắm, việc hạn chế đi đến chỗ đông người đã đem lại cơ hội lớn cho kênh mua sắm online, đặt hàng từ nhà. Nhiều người tiêu dùng chọn mua sắm online để hạn chế số lần phải đến các cửa hàng mua sắm trực tiếp, giảm nguy cơ lây nhiễm dịch bệnh.

Online là một thị trường lớn và đầy tiềm năng đối với kinh doanh bán lẻ. Trong thời đại hiện nay, người tiêu dùng đang có xu hướng “online hóa” việc mua sắm để tiết kiệm thời gian và công sức, do đó, các doanh nghiệp bán lẻ cũng đang dần tiến sâu vào thị trường kinh doanh trực tuyến này với mong muốn tăng hiệu suất và doanh thu bán hàng.

Được biết, khi bán hàng theo phương pháp truyền thống rất bất cập trong vấn đề quảng cáo sản phẩm cũng như thống kê doanh thu. Việc thống kê báo cáo doanh thu hoàn toàn dựa trên phần mềm excel, việc này đẫn đến một số vấn đề khá khó khăn như thời gian xuất một báo cáo rất lâu, tốn nhiều thời gian và nhân lực, nhập liệu nếu có nhầm lẫn sẽ không được cảnh báo và xử lý, chẳng hạn như nếu xuất một sản phẩm mà trong kho số lượng tồn của sản phẩm đó không đủ để xuất hay sản phẩm đã hết hàng nhưng khách hàng không được thông báo về tình trạng này, nhân viên cũng không được nhắc để nhập thêm loại hàng ấy. Ngoài ra, khách hàng không chủ động được trong việc đặt trước và mua hàng của shop, mọi việc bán và mua phải diễn ra đồng thời giữa cả bên mua và bên bán.

Do đó, vấn đề xây dựng một shop bán hàng online rất cần thiết. Việc này giúp shop bán hàng quản lý hàng hóa tốt hơn, thống kê báo cáo doanh thu nhanh chóng, chính xác, giảm nhân lực, giảm chi phí và thời gian. Việc quảng cáo sản phẩm và thương hiệu cũng dễ dàng hơn. Về mặt khách hàng, một shop online sẽ giúp họ tiết kiệm thời gian mua sắm cũng nhưu giúp “đặt chỗ” trước những sản phẩm mà họ yêu thích, không sợ bị mua trước hay hết hàng khi họ không đến shop chủ động trong việc xem và đặt hàng.

Do vậy, sau khi tìm hiểu em đã quyết định chọn đề tài “Xây dựng và phát triển Website bán đồ gia dụng” nhằm giúp các shop cải thiện được vấn đề quản lý, quảng cáo hàng hóa của mình. Đồng thời phù hợp với xu hướng phát triển của thế giới hiện nay.

Với điều kiện thời gian và kiến thức còn hạn chế, chưa có nhiều kinh nghiệm về thực tế nên bài làm của em không tránh những thiếu sót, em rất mong nhận được ý kiến đóng góp của các thầy cô để em hoàn thiện đề tài của mình cũng như nâng cao kỹ năng, kinh nghiệm để phục vụ cho công việc sau này của mình được tốt hơn.

Báo cáo được tổ chức làm 4 phần như sau:

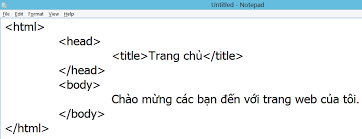
* Chương 1: Cơ sở lý thuyết
* Chương 2: Khảo sát và phân tích thiết kế hệ thống
* Chương 3: Giao diện demo của chương trình
* Chương 4: Kết luận

**Chương I. Cơ sở lý thuyết**

* 1. **Giới thiệu về ngôn ngữ**
     1. ***HTML (HyperText Markup Language)***

Là ngôn ngữ đánh dấu siêu văn bản được sử dụng để tạo các tài liệu có thể truy cập trên mạng. Tài liệu HTML được tạo nhờ dùng các thẻ và các phần tử của HTML. File được lưu trên máy chủ dịch vụ web với phần mở rộng “.htm” hoặc “.html”.

Ngày nay với sự phát triển của công nghệ web việc sử dụng các công cụ HTML càng trở nên quan trọng trong sợ phát triển của nghành công nghệ web hiện đại.

****

Hình 1. 1 Ví dụ code HTML

Một tài liệu HTML gồm 3 phần cơ bản:

* Phần html: Mọi tài liệu html phải bắt đầu bằng thẻ mở <html> và kết thúc bằng thẻ đóng html </html> “<html></html>”. Thẻ html báo cho trình duyệt biết nội dung giữa hai cặp thẻ này là một tài liệu html.
* Phần tiêu đề: phần tiêu đề bắt đầu bởi thẻ <head> và kết thúc bởi thẻ </head>. Phần này chứa tiêu đề mà được hiển thị trên thanh điều hướng của trang web. Tiêu đề nằm trong thẻ <title>. Bắt đầu là thẻ <title> và kết thúc là thẻ </title>. Tiêu đề là khá quan trọng khi người dùng tìm kiếm thông tin, tiêu đề trang Web cung cấp từ khóa chính yếu cho việc tìm kiếm.
* Phần thân: phần này nằm sau phần tiêu đề. Phần thân bao gồm văn bản, hình ảnh… và các liên kết bạn mới hiển thị trên trang Web của mình. Phần thân bắt đầu bởi thẻ <body> và kết thức bởi </body>.
  + 1. ***CSS (Cascading Style Sheets)***

Trong lĩnh vực xây dựng, chúng ta có trang trí nội thất; trong lĩnh vực

thẩm mỹ - làm đẹp, chúng ta có kỹ thuật make-up; còn trong lĩnh vực thiết kế web chúng ta có CSS. Còn CSS (Cascading Style Sheets) là một ngôn ngữ quy định cách trình bày cho các tài liệu viết bằng HTML, XHTML, XML, SVG, hay UML…

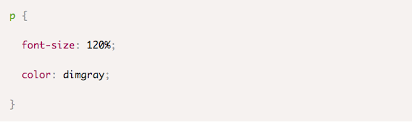
Như chúng ta đã biết HTML hỗ trợ một số thuộc tính định dạng cơ bản cho text, picture, table, … nhưng nó không thật sự phong phú và chính xác như nhau trên mọi hệ thống. CSS cung cấp hàng trăm thuộc tính trình bày dành cho các đối tượng với sự sáng tạo cao trong kết hợp các thuộc tính giúp mang lại hiệu quả. Ngoài ra, hiện tại CSS đã được hỗ trợ bởi tất cả các trình duyệt, nên chúng ta hoàn toàn có thể tự tin trang web của mình có thể hiển thị hầu như “như nhau” dù trên một hệ thống sử dụng Windows, Linux hay trên một máy Mac miễn là chúng đang sử dụng một phiên bản trình duyệt mới nhất.

Sử dụng các mã định dạng trực tiếp trong HTML tốn hao nhiều thời gian thiết kế cũng như dung lượng lưu trữ trên đĩa cứng. Trong khi đó CSS đưa ra phương thức “tờ mẫu ngoại” giúp áp dụng một khuôn mẫu chuẩn từ một file CSS ở ngoài. Nó thật sự có hiệu quả đồng bộ khi tạo một website có hàng trăm trang, hay khi muốn thay đổi một thuộc tính trình bày nào đó. Hãy thử tưởng tượng một website với hàng trăm trang và chúng ta muốn thay đổi font chữ hay màu chữ cho một thành phần nào đó. Đó thật sự sẽ là một công việc buồn chán và tốn nhiều thời gian. Nhưng với việc sử dụng CSS việc đó là hoàn toàn đơn giản cũng như là một trò ma thuật nào đó.

Ngoài ra, CSS còn cho phép áp đặt những kiểu trình bày thích hợp hơn cho các phương tiện khác nhau như màn hình máy tính, máy in, sản

phẩm…

CSS được cập nhật liên tục mang lại các trình bày phức tạp và tinh vi hơn.

****

Hình 1. Ví dụ viết CSS trong HTML

* + 1. ***JavaScript***

JavaScript là một ngôn ngữ kịch bản (scripting language) được dùng để tạo các script ở máy client (client-side script) và máy server (server-sidescript).

Các script ở máy client được thực thi tại trình duyệt, các script ở máy

server được thực hiện trên server.

HTML lúc đầu được phát triển như là một định dạng của tài liệu có thể chuyển dữ liệu trên Internet. Tuy nhiên, không lâu sau đó, trọng tâm của HTML nặng tính hàn lâm và khoa học dần chuyển hướng sang người dùng thường nhật vì ngày nay người dùng xem Internet như là một nguồn thông tin và giải trí. Các trang Web ngày càng mang tính sáng tạo và đẹp mắt hơn nhằm thu hút nhiều người dùng hơn. Nhưng thực chất kiểu dáng và nội dung bên trong vẫn không thay đổi. Và người dùng hầu như không thể điều khiển trên trang Web mỗi khi nó được hiển thị.

Javascript được phát triển như là một giải pháp cho vấn đề nêu trên.

Javascript là một ngôn ngữ kịch bản được Sun Microsystems và Netscape

phát triển. Nó được dùng để tạo các trang Web động và tương tác trên

Internet. Đối với những người phát triển HTML, Javascript rất hữu ích

trong việc xây dựng các hệ thống HTML có thể tương tác với người dùng

JavaScript là một ngôn ngữ lập trình được nhúng được trong các trang

HTML.

JavaScript nâng cao tính động và khả năng tương tác cho website bằng cách sử dụng các hiệu ứng của nó như thực hiện các phép tính, kiểm tra form, viết các trò chơi, bổ sung các hiệu ứng đặc biệt, tuỳ biến các chọn lựa đồ hoạ, tạo ra các mật khẩu bảo mật và hơn thế nữa.

Chúng ta có thể sử dụng JavaScript để:

* Tương tác với người dùng. Chúng ta có thể viết mã để đáp lại các sự kiện. Các sự này sẽ có thể phát sinh bởi người dùng - - nhấp chuột hay được phát sinh từ hệ thống - - định lại kích thước của trang và v.v.
* Thay đổi nội dung động. Mã JavaScript có thể dùng để thay đổi nội dung và vị trí các phần tử một cách động trên một trang nhằm đáp lại sự tương tác với người dùng.
* Kiểm tra tính hợp lệ dữ liệu. Chúng ta có thể viết mã nhằm kiểm tra tính hợp lệ của dữ liệu do người dùng nhập vào trước khi nó được gửi lên Web server để xử lý.
* Giống như các ngôn ngữ khác, JavaScript cũng tuân thủ một số quy tắc ngữ pháp căn bản. Việc nắm vững các quy tắc ngữ pháp này có thể giúp ta đọc được script và tự viết các script không bị lỗi.
* Dùng Caps: JavaScript phân biệt chữ hoa chữ thường
* Dùng Pairs: Trong JavaScript, luôn luôn có cặp ký hiệu mở và đóng. Lỗi sẽ xuất hiện khi bỏ sót hoặc đặt sai một trong hai ký hiệu này.
* Dùng Spaces (các ký tự trắng): Như HTML, JavaScript thường bỏ qua ký tự trắng. Trong JavaScript, ta có thể thêm vào các ký tự trắng hoặc các tab giúp cho ta dễ dàng đọc hay sửa các file script.
* Dùng Chú thích (Comments): Các chú thích giúp ta ghi chú về chức năng của đoạn script, thời gian và người tạo ra đoạn script.
* Mặc dù cả client-side JavaScript và server-side JavaScript đều dựa trên một ngôn ngữ nền tảng như nhau, nhưng mỗi loại còn có thêm những tính năng chuyên biệt phù hợp với môi trường mà nó chạy. Nghĩa là, client-side JavaScript bao gồm các đối tượng được định nghĩa sẵn chỉ có thể sử dụng trên trình duyêt, Server-side JavaScript bao gồm các đối tượng và các hàm được định nghĩa sẵn chỉ có thể sử dùng trong các ứng dụng phía server (server-side applications).
  + 1. ***PHP***
       1. *Giới thiệu PHP*

PHP (Personal Home Page) ra đời từ năm 1994 do phát minh của Rasmus Lerdorf, và nó tiếp tục được phát triển bởi nhiều cá nhân và tập thể khác, do đó PHP được xem như một sản phẩm của mã nguồn mở.

PHP là kịch bản trình chủ (server script) chạy trên phía server (server side) như các server script khác (asp, jsp, cold fusion).

PHP là kịch bản cho phép chúng ta xây dựng ứng dụng web trên mạng internet tương tác với mọi cơ sở dữ liệu như mySQL, Oracle, SQL Server, Access.

Ngoài việc rất thích hợp cho việc viết một trang web, bên cạnh đó chúng cũng có khả năng nhúng được vào trang HTML một cách thật dễ dàng. Hiện nay, PHP đang là một ngôn ngữ lập trình vô cùng phổ biến ở khắp nơi trên thế giới. Lý do là bởi việc tối ưu hóa các ứng dụng của web, việc có tốc độ nhanh và nhỏ gọn. Hơn thế nữa, các cú pháp của PHP cũng có nhiều điểm giống với C và Java nên các lập trình viên có thể học hoặc xây dựng sản phẩm tương đối nhanh so với những ngôn ngữ khác.

Việc xây dựng ngôn ngữ, tài liệu gốc và thư viện trong PHP có tính cộng đồng thì đóng góp của Zend Inc là rất lớn. Nhằm tạo ra một môi trường làm việc chuyên nghiệp, có thể đưa được PHP vào quy mô phát triển của doanh nghiệp thì các công ty do nhà phát triển PHP tính cốt lõi là vô cùng quan trọng.

* + - 1. *Khái niệm về cú pháp PHP*

Cú pháp PHP chính là cú pháp trong ngôn ngữ C, làm quen với C thì có lợi thế trong lập trình trong PHP.

Để lập trình bằng ngôn ngữ PHP cần chú ý những điểm sau:

* Sử dụng PHP phải có cặp thẻ đóng mở <?php ?>
* Cuối câu có dấu ;
* Biến trong PHP có tiền tố là $.
* Mỗi phương thức đều bắt đầu bằng { và đóng bằng }.
* Khi khai báo biến thì không có kiểu dữ liệu.
* Nên có giá trị khởi đầu cho biến khai báo.
* Phải có ghi chú (comment) cho mỗi feature mới.
* Sử dụng // hoặc # để giải thích cho mỗi câu ghi chú.
* Sử dụng /\* và \*/ cho mỗi đoạn ghi chú
* Khai báo biến có phân biệt chữ hoa và chữ thường.
  + - 1. *Khai báo biến*

Khi thực hiện khai báo biến trong C, cần phải biết tuân thủ quy định như: kiểu dữ liệu trước tên biến và có giá trị khởi đầu, tuy nhiên khi làm việc với PHP thì không cần khai báo kiểu dữ liệu nhưng phải sử dụng tiền tố $ trước biến.

VD: $ten = MinhAnh;

* + - 1. *Kiểu dữ liệu*

Bảng kiểu dữ liệu thông thường:

**Boolean** True hay False.

**Integer** Kiểu số nguyên giá trị lớn nhất xấp xỉ 2 tỷ.

**Float** Kiểu số thực.

**String** Kiểu chuỗi kí tự.

**Array** Mảng với nhiều kiểu dữ liệu.

*1.1.4.4.1. Thay đổi kiểu dữ liệu*

Để thay đổi kiểu dữ liệu có thể sử dụng cách ép kiểu như trong ngôn ngữ lập trình C hay Java.

*1.1.4.4.2. Kiểm tra kiểu dữ liệu của biến*

Để kiểm tra kiểu dữ liệu của biến, có thể sử dụng các hàm như:

Is\_int để kiểm tra biến đó có kiểu integer, nếu biến đó có kiểu integer thì hàm trả về giá trị true. Tương tự, có thể sử dụng các hàm kiểm tra tương ứng với kiểu dữ liệu là is\_array, is\_bool, is\_double, is\_float, is\_long, is\_null, is\_string…

VD: $check=true;

Echo is\_bool ($check);

*1.1.4.4.3. Kiểu mảng*

Mảng là danh sách  các phần tử có cùng kiểu dữ liệu và nó là một trong các kiểu dữ liệu trong php có độ phức tạp tính toán cao. Có 2 loại mảng là **mảng một chiều**hoặc **mảng nhiều chiều**. Riêng với PHP thì các phần tử của mảng có thể không cùng kiểu dữ liệu**,**và các phần tử của mảng được truy xuất thông qua các chỉ mục(vị trí) của nó nằm trong mảng.

**Khởi tạo mảng**

<?php

$ten\_mang = array(); // khởi tạo một mảng gán vào biến $ten\_mang.

?>

Giả sử, có 2 sinh viên là Nguyễn Văn A và Nguyễn Văn B, khởi tạo một mảng $sinhvien để lưu 2 sinh viên này lại.

**Cách 1:**

<?php

$sinhvien = array('Nguyễn Văn A', 'Nguyễn Văn B');

?>

Khởi tạo một mảng và gán trực tiếp 2 phần từ vào, vì mảng bắt đầu từ 0 nên nó tự hiểu phần tử đầu tiên có chỉ mục =0, và phần tử thứ 2 = 1.

**Cách 2:**

<?php

$sinhvien = array(

0 => 'Nguyễn Văn A',

1 => 'Nguyễn Văn B'

);

?>

Khởi taọ một mảng và gán trực tiếp 2 phần tử vào, nhưng lúc gán có ghi rõ các chỉ mục cho từng phần tử.

**Cách 3:**

<?php

$sinhvien = array();

$sinhvien[0] = 'Nguyễn Văn A';

$sinhvien[1] = 'Nguyễn Văn B';

?>

Khởi tạo một mảng rỗng. sau đó dùng 2 lệnh để gán 2 phần tử vào, mỗi lệnh gán có chỉ rõ chỉ mục.

**Cách 4:**

<?php

$sinhvien = array();

$sinhvien[] = 'Nguyễn văn A';

$sinhvien[] = 'Nguyễn Văn B';

?>

Khởi tạo một mảng rỗng, sau đó dùng 2 lệnh gán 2 phần tử vào nhưng lại không chỉ rõ chỉ mục, lúc này PHP sẽ kiểm tra thấy mảng đang rỗng nên phần tử đầu tiên nó sẽ mặc định gán chỉ mục = 0, và phần tử tiếp theo sẽ bằng phần tử trước nó + 1 tức là sẽ = 1.

**Truy xuất các phần tử của mảng**

Để **truy xuất các phần tử của mảng chỉ mục** ta dùng cú pháp sau: $tenmang[$index]; trong đó $index là chỉ mục muốn lấy.

<?php

$sinhvien = array(

0 => 'Nguyễn Văn A',

1 => 'Nguyễn Văn B'

);

echo $sinhvien[0]; // Xuất ra màn hình phần tử 0 => Nguyễn Văn A

echo $sinhvien[1]; // Xuất ra màn hình phần tử 1 => Nguyễn Văn B

?>

* + - 1. *Ưu, nhược điểm của PHP*

**Ưu điểm**

* Thứ nhất, PHP được sử dụng miễn phí, do đó mà bạn có thể rất nhiều cơ hội học và nắm bắt loại ngôn ngữ này. Các bạn có thể học loại ngôn ngữ này theo dạng online mà không cần phải lo sợ việc bỏ ra một khoản tiền quá lớn để học.
* Thứ hai, cấu trúc của PHP cực kỳ đơn giản, vậy nên bạn sẽ không mất nhiều thời gian để có thể học được, đây chính là một ưu điểm cực kỳ hấp dẫn của PHP.
* Thứ ba nữa là thư viện của PHP vô cùng phong phú, cũng như được cộng đồng hỗ trợ một cách vô cùng mạnh mẽ. Vậy nên bạn có thể dễ dàng tìm nguồn tài liệu mình cần cũng như có thể nhận được sự hỗ trợ một cách đắc lực nhất.
* Thứ tư, khi học về PHP thì các bạn sẽ có nhiều cơ hội việc làm hơn từ đó cũng giúp tăng thêm thu nhập.
* Thứ năm, PHP không chỉ dừng ở những tính năng hiện tại, trong tương lai thì nó sẽ cón có thể phát triển mạnh mẽ hơn để khẳng định sự vượt trội của mình.

**Nhược điểm**

* Thứ nhất, PHP có hạn chế về cấu trúc ngữ pháp, bởi nó không được thiết kế gọn gàng và đẹp mắt như những loại ngôn ngữ khác
* Thứ hai, PHP chỉ có thể hoạt động và sử dụng cho các ứng dụng trên web. Đó chính là hạn chế cần khắc phục nếu muốn cạnh tranh và phát triển rộng rãi hơn nữa so với các ngôn ngữ lập trình khác.
  + - 1. *So sánh PHP và Java*

**Giống:** PHP và Java đều rất giống nhau về những lợi ích được cung cấp, nhưng cả hai đều rất khác nhau.PHP là một ngôn ngữ kịch bản phía máy chủ, trong khi Java là một lựa chọn phía máy khách.

**Khác:** Khi bạn sử dụng mã PHP, bạn sẽ thực thi nó trong máy chủ thực, trong khi Java sẽ được thực thi trên máy khách của bạn. Điều này có nghĩa là, nếu bạn sử dụng Java và máy khách không có chương trình phù hợp, chẳng hạn như Môi trường thời gian chạy Java, trang web của bạn sẽ không hiển thị. Với PHP, đây không phải là một vấn đề.

**Nên chọn PHP hay Java?:** Nhiều người sẽ nói rằng đó là tất cả về sở thích của lập trình viên, điều này đúng, ở một điểm nào đó.Nhìn vào những lợi ích được cung cấp bởi cả Java và PHP, rõ ràng là cả hai đều rất mạnh mẽ.Tuy nhiên, nhiều lập trình viên sử dụng cả hai ngôn ngữ sẽ cho bạn biết PHP là lựa chọn tốt hơn.

**Lí do nên chọn PHP thay vì Java:** Lý do chính mà các lập trình viên chọn PHP thay vì Java là tốc độ. Nó nhanh hơn nhiều để phát triển và PHP có xu hướng gặp ít vấn đề hơn khi sử dụng chia sẻ lưu trữ. Khi sử dụng Java, tốt nhất bạn nên có một máy chủ chuyên dụng cho lưu trữ của mình.

* 1. **Hệ quản trị cơ sở dữ liệu(MySQL)**

MySQL là hệ quản trị cơ sở dữ liệu được sử dụng cho các ứng dụng Web có quy mô vừa và nhỏ. Tuy không phải là một hệ quản trị cơ sở dữ liệu lớn nhưng chúng cũng có trình giao diện trên Windows hay Linux, cho phép người dùng có thể thao tác các hành động liên quan đến cơ sở dữ liệu.

MySQL là hệ quản trị cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh.

Cũng giống như các hệ quản trị cơ sở dữ liệu khác, khi làm việc với MySQL, chúng ta phải đăng ký kết nối, tạo cơ sở dữ liệu, quản lý người dùng, phân quyền sử dụng, thiết kế đối tượng Table của cơ sở dữ liệu và xử lý dữ liệu.

***1.2.1. Ưu, nhược điểm của MySQL***

**Ưu điểm**

* Dễ sử dụng: MySQL là hệ quản trị 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 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.

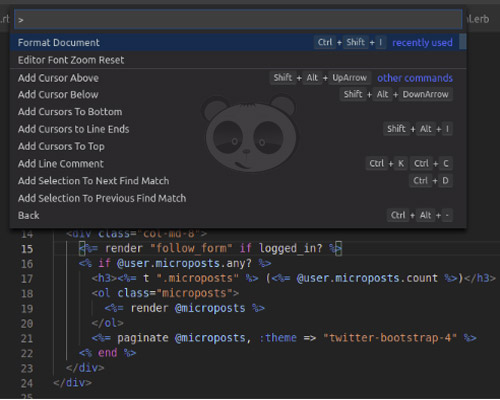
**Nhược điểm**

* 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. **Giới thiệu Visual Studio Code**
     1. ***Tổng quan***

Visual Studio Code là một code editor mạnh mẽ và được đông đảo các web developer, coder, programmer sử dụng và yêu thích. Với các ưu điểm nổi bật là sự nhanh chóng, nhẹ, hỗ trợ đa nền tảng cùng nhiều tính năng và là mã nguồn mở chính

Visual Studio Code hỗ trợ đa dạng các chức năng Debug, đi kèm với Git, có Syntax Highlighting. Đặc biệt là tự hoàn thành mã thông minh, Snippets, và khả năng cải tiến mã nguồn. Nhờ tính năng tùy chỉnh, Visual Studio Code cũng cho phép các lập trình viên thay đổi Theme, phím tắt, và đa dạng các tùy chọn khác. Mặc dù trình soạn thảo Code này tương đối nhẹ, nhưng lại bao gồm các tính năng mạnh mẽ.

Visual Studio Code hỗ trợ nhiều hệ điều hành như Windows, Linux, Mac OS.



Hình 1. Ví dụ về Visual Studio Code

* + 1. ***Điểm mạnh của Visual Studio Code***

+,Chạy quá nhẹ nhàng so với một full IDE mà vẫn thường được dùng trước kia như Eclipse, VS, Dreamware…

+, Các extensions nhiều, phong phú, tiện lợi, cài đặt dễ dàng, và cũng hầu hết miễn phí.

+, Open file nhanh bằng suggestion theo text mình gõ. Không cần nhớ file nằm ở đâu, gõ tên, ST3 sẽ gợi ý file muốn mở.

+, Chế độ tìm kiếm, replace, highlight rất dễ sử dụng.

+, Chế độ Intelligent Suggestion hoạt động smooth, phù hợp nhu cầu sử dụng.

+, Có nhiều themes cả về text và giao diện tools.

+, Sử dụng thêm phím Ctrl, có thể edit cùng lúc nhiều chỗ khác nhau.

+, Tích hợp sẵn terminal vào nó luôn.

+, Có nhiều extensions giúp phát triển nhanh các PHP project dựa trên các OpenSource framework như Laravel, CakePHP…

* + 1. ***Các tính năng chính với ưu điểm mạnh mẽ của Visual Studio Code***

**Người dùng có thể tùy chỉnh nhiều tính năng**

Các ràng buộc chính, menu, đoạn mã, macro và nhiều thứ khác trong Visual Studio Code đều có thể tùy chỉnh với các File JSON đơn giản. Hệ thống này cung cấp cho bạn sự linh hoạt vì các cài đặt có thể được chỉ định trên các loại File và dự án khác nhau.

**High-performance**

Visual Studio Code được xây dựng từ các thành phần tùy chỉnh, cung cấp khả năng đáp ứng chưa từng có. Từ bộ công cụ UI đa nền tảng tùy chỉnh mạnh mẽ, đến các công cụ tô sáng cú pháp. Đây được coi là IDE nhanh nhất và ổn định nhất hiện nay.

**Hỗ trợ đa ngôn ngữ lập trình**

Các ngôn ngữ lập trình như CSS, C + +, C #, HTML, F#… được Visual Studio Code dễ dàng nhận diện và đưa ra cảnh báo nếu chương trình của bạn bị lỗi.

**Hỗ trợ API mạnh mẽ**

Visual Studio Code cho phép các plugin tăng cường chức năng tích hợp. Kiểm soát gói có thể được cài đặt thông qua bảng lệnh, cung cấp quyền truy cập đơn giản vào hàng ngàn gói do cộng đồng xây dựng.

**Hiển thị và tìm kiếm cho cùng một chức năng**

Để kiểm tra số lần một chức năng hoặc một lệnh đã được sử dụng trên Visual Studio Code, sử dụng chuột để tô sáng mã. Tiếp theo, sử dụng phím tắt Ctrl + D để di chuyển và xem vị trí của từng đoạn mã.

**Hỗ trợ tản đa nền tảng**

Visual Studio Code có sẵn cho Mac, Windows và Linux. Giấy phép là tất cả những gì bạn cần để sử dụng Visual Studio Code trên bất kỳ máy tính nào bạn sở hữu, bất kể nó được sử dụng trên hệ điều hành nào. Visual Studio Code sử dụng bộ công cụ UI tùy chỉnh, được tối ưu hóa cho tốc độ, đồng thời tận dụng chức năng gốc trên mỗi nền tảng.

**Kho lưu trữ dữ liệu an toàn**

Tính bảo mật dữ liệu hiện nay được đòi hỏi cao, đặc biệt là đối với các hệ thống lập trình công nghệ thông tin. Người dùng có thể hoàn toàn tin tưởng sử dụng Visual Studio Code vì nó có khả năng kết nối dễ dàng với Git hoặc bất cứ kho lưu trữ dữ liệu hiện có.

**Cung cấp nơi lưu trữ tiện ích mở rộng**

Trường hợp ngôn ngữ lập trình của lập trình viên sử dụng không thuộc danh sách các ngôn ngữ được Visual Studio Code hỗ trợ, họ có thể tải thêm các tiện ích mở rộng. Điều này không làm ảnh hưởng đến hiệu năng của phần mềm bởi phần mở rộng này hoạt động riêng biệt như một chương trình độc lập.

**Intellisense**

Visual Studio Code có khả năng đặc biệt hơn so với các chương trình mã code khác là nó có thể phát hiện bất kỳ đoạn mã nào không đầy đủ. Intellisense sẽ tự động giúp các lập trình viên bổ sung các cú pháp còn thiếu kể cả khi học quên không khai báo biến.

* + 1. ***So sánh điểm mạnh của Visual Studio Code với Sublime Text***

**Phân tích các Symbol chính**

Khi bạn mở một project trong Visual Studio Code nó sẽ tự động mở một trình gọi là trình “phân tích symbol” một từ mỹ miều để nói về việc tìm keywords trong code của bạn. Điều hay về nó là khi type Cmd + Shift + R để mở menu tìm symbol và nhanh chóng tìm ra các class names và methods trong code của mình. Hầu hết làm việc với PHP nên ví dụ, nếu như đã biết class name đang làm là PostController, có thể search và nó trong symbol search và file chứa class PHP sẽ lập tức mở trong editor.

SublimeText, nhưng với codebase to to một chút là sẽ có vấn đề, trong khi Visual Studio Code thì không.

**Tốc độ cực nhanh**

Visual Studio Code thì là text editor nhanh nhất từng sử dụng. Nó mở gần như ngay tức thì và search rất nhanh. Microsoft đã rất cố gắng làm Visual Studio Code vận hành hiệu quả. Nó làm cho editor dễ dàng mở rộng.

**Các Vim Binding tốt hơn**

Vim support của Visual Studio Code chính xác hơn SublimeText, giúp viết code nhanh hơn một chút. Visual Studio Code hỗ trợ Vim bindings bên ngoài, nhưng có thể sử dụng nhiều features hơn nữa nếu bạn dùng Vintageous plugin.

* 1. **Xampp**
* **Khái niệm**

Xampp là chương trình tạo máy chủ Web (Web Server) được tích

hợp sẵn Apache, PHP, MySQL, FTP Server, Mail Server và các

công cụ như phpMyAdmin

* **Đặc điểm**

Xampp là một mã nguồn mở máy chủ web đa nền được phát triển

bởi Apache Friends, bao gồm chủ yếu là Apache HTTP Server,

MariaDB database, và interpreters dành cho những đối tượng sử

dụng ngôn ngữ PHP và Perl.

Xampp là viết tắt của Cross-Platform (đa nền tảng-X), Apache (A),

MariaDB (M), PHP (P) và Perl (P).

Xampp cũng là có thể chạy tốt trên cả Linux, Windows và Mac.

* 1. **Tổng quan mô hình MVC**
     1. ***Các thành phần trong MVC***
* Model (Tầng dữ liệu): là một đối tượng hoặc một tập hợp các đối tượng biểu diễn cho phần dữ liệu chương trình. Nó được giao nhiệm vụ cung cấp dữ liệu cho cơ sở dữ liệu và lưu dữ liệu vào các kho dữ liệu. Tất cả các nghiệp vụ logic được thực thi ở Model. Dữ liệu vào từ người dùng sẽ thông qua View đến Controller và được kiểu tra ở Model trước khi lưu vào cơ sở dữ liệu. Việc truy xuất, xác nhận, và lưu dữ liệu là một phần của Model.
* View (Tầng giao diện): là phần giao diện với người dùng, bao gồm việc hiển thị dữ liệu ra màn hình, cung cấp các menu, nút bấm, hộp đối thoại, chọn lựa…, để người dùng có thể thêm, sửa, xóa, tìm kiếm và làm các thao tác khác đối với dữ liệu trong hệ thống… Thông thường, các thông tin cần hiển thị được lấy từ thành phần Models.
* Controller (Tầng điều khiển): là phần điều khiển của ứng dụng, điều hướng các nhiệm vụ (task) đến đúng phương hướng (method) có chức năng xử lý nhiệm vụ đó. Nó chịu trách nhiệm xử lý các tác động về mặt giao diện, các thao tác đối với models, và cuối cùng là chọn một view thích hợp để hiển thị ra màn hình
  + 1. ***Cách vận hành của MVC***

Hiểu một cách đơn giản, mô hình MVC cho ta thấy cách thức hoạt động của các ứng dụng web.

Hãy tưởng tượng việc bạn phải chuẩn bị một bữa tối với chiếc tủ lạnh đầy thức ăn. Trong trường hợp này, mọi nguyên liệu bạn có chính là Model. Và bạn có nhiều lựa chọn công thức món ăn khác nhau, những lựa chọn này đóng vai trò giống như Controller. Các món ăn sau khi được chế biến và bày biện sẵn sàng, chính là View.

Trong thực tế, MVC là công cụ hữu ích khi vận hành ứng dụng vì nó cho phép bạn hình dung về ứng dụng của mình dễ dàng dàng hơn và sắp xếp các ý tưởng đó thành các đoạn code.

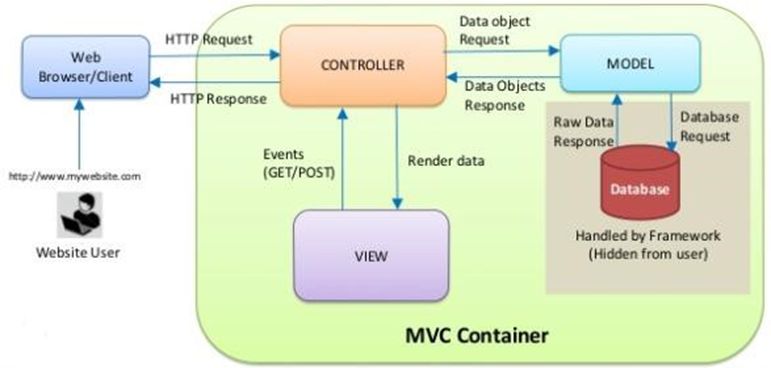
Ví dụ bạn muốn lập trình một ứng dụng giúp lên kế hoạch làm việc. Và ứng dụng này cho phép người sử dụng tạo các nhiệm vụ, và sắp xếp chúng thành các danh sách.

Chức năng Model trong ứng dụng này sẽ giúp định nghĩa “nhiệm vụ” và “danh sách” là tổng hợp các “nhiệm vụ”.

Các đoạn code View sẽ quyết định giao diện của ứng dụng, như phông chữ hay màu sắc.

Và cuối cùng, Controller sẽ chịu trách nhiệm về cách thức người dùng thêm các nhiệm vụ, hay đánh dấu các nhiệm vụ đã được hoàn thành. Controller kết nối nút “thêm” từ View với Model. Như vậy, khi tương tác với nút “thêm” từ View, Model sẽ thêm nhiệm vụ mới.

* + 1. ***Sơ đồ tổng quan MVC***



Hình 1. Sơ đồ tổng quan về MVC

* + 1. ***Ưu, nhược điểm của MVC***

**Ưu điểm**

Mô hình MVC phù hợp với các dự án với các đội ngũ làm việc độc lập với nhau, như vậy các công việc đều có thể được thực hiện một cách linh hoạt.

Như vậy, mô hình này nổi bật nhờ các ưu điểm sau:

* **Hỗ trợ quá trình phát triển nhanh chóng:** Với đặc điểm hoạt động độc lập của từng thành phần, các lập trình viên có thể làm việc đồng thời trên từng bộ phận khác nhau của mô hình này. MVC giúp bạn tiết kiệm rất nhiều thời gian.
* **Khả năng cung cấp đồng thời nhiều khung View:** Với mô hình MVC, bạn có thể tạo ra đồng thời nhiều khung View cho Model.
* **Hỗ trợ các kỹ thuật không đồng bộ:** MVC có thể hoạt động trên nền tảng JavaScript. Điều này có nghĩa là các ứng dụng MVC có thể hoạt động với các file PDF, các trình duyệt web cụ thể, và cả các widget máy tính.
* **Dễ dàng thao tác chỉnh sửa:** Bộ phận Model hoạt động tách biệt với View đồng nghĩa với việc bạn có thể đưa ra các thay đổi, chỉnh sửa hoặc cập nhật dễ dàng ở từng bộ phận.
* **Giữ nguyên trạng thái data:** Mô hình MVC truyền lại dữ liệu nhưng không định dạng lại dữ liệu. Do đó, các dữ liệu này có thể được dùng lại cho các thay đổi sau này.
* **Hỗ trợ các nền tảng phát triển SEO:** Với mô hình MVC, bạn có thể dễ dàng tạo ra các mã SEO URL để thu hút lượng truy cập đối với ứng dụng bất kỳ.

**Nhược điểm**

* Khó khăn trong quá trình điều hướng code: Điều hướng khung có thể phức tạp vì mô hình này bao gồm nhiều lớp và yêu cầu người dùng thích ứng với các tiêu chí phân tách của MVC.
* Không thích hợp việc phát triển các ứng dụng nhỏ vì mô hình này yêu cầu bạn lưu trữ một số lượng lớn các file.
* Nhiều khung hoạt động đồng thời: Việc phân tách một tính năng thành ba bộ phận khác nhau dễ dẫn đến hiện tượng phân tán. Do đó, đòi hỏi các nhà phát triển phải duy trì tính nhất quán của nhiều bộ phận cùng một lúc.

**Chương II: Khảo sát và phân tích thiết kế hệ thống**

**2.1. Khảo sát hiện trạng**

- Được biết, khi bán hàng theo phương pháp truyền thống rất bất cập trong vấn đề quảng cáo sản phẩm cũng như thống kê doanh thu. Việc thống kê báo cáo doanh thu hoàn toàn dựa trên phần mềm excel, việc này đẫn đến một số vấn đề khá khó khăn như thời gian xuất một báo cáo rất lâu, tốn nhiều thời gian và nhân lực, nhập liệu nếu có nhầm lẫn sẽ không được cảnh báo và xử lý, chẳng hạn như nếu xuất một sản phẩm mà trong kho số lượng tồn của sản phẩm đó không đủ để xuất hay sản phẩm đã hết hàng nhưng khách hàng không được thông báo về tình trạng này, nhân viên cũng không được nhắc để nhập thêm loại hàng ấy. Ngoài ra, khách hàng không chủ động được trong việc đặt trước và mua hàng của shop, mọi việc bán và mua phải diễn ra đồng thời giữa cả bên mua và bên bán.

- Sự phát triển mạnh mẽ của ngành công nghệ thông tin nói chung và thương mại điện tử nói riêng. Đồ gia dụng là những vật dụng thiết yếu sử dụng trong gia đình. Nó có thể là những đồ dùng liên quan đến phòng bếp, phòng tắm, phòng khách hoặc bất cứ các vật dụng thiết yếu nào. Khách hàng sẽ được tối ưu hóa công đoạn vệ sinh nhà cửa, nấu nướng nhờ vào các sản phẩm gia dụng tích hợp từ công nghệ hiện đại.

Chính vì cần thiết như vậy nên đồ gia dụng thu hút một lượng đông đảo khách hàng mua sắm liên tục. Nó đóng vai trò quan trọng trong gia đình, hỗ trợ công việc tân trang nhà cửa và chăm sóc gia đình hiệu quả. Đây chính là mặt hàng tiềm năng cho các chị em nội trợ, mẹ bỉm sửa có thời gian rảnh kinh doanh online tại nhà.

**2.2. Giải quyết bài toán**

Từ những nhu cầu và thực trạng khi qua khảo sát thực tế. Em quyết định xây dựng Website bán đồ gia dụng với những yêu cầu:

* Yêu cầu về mặt chức năng: Cho phép phía quản trị có thể quản lí các mặt hàng, quản lí đơn hang…, cho phép phía người dùng có thể đặt hàng…
* Yêu cầu về phi chức năng: chạy tất cả trên các trình duyệt (chrome, coccoc, IE…), thời gian phản hồi nhanh (thời gian phản hồi khoảng dưới 500ms), giao diện đẹp…

**2.3. Giới thiệu website**

Website bán đồ gia dụng cung cấp các chức năng sau:

* Đối với người dùng
* Quản lý tài khoản
* Đăng ký
* Đăng nhập
* Đổi mật khẩu
* Đăng xuất
* Cập nhật thông tin cá nhân
* Tìm kiếm sản phẩm
* Xem chi tiết sản phẩm
* Đánh giá sản phẩm
* Quản lý giỏ hang
* Đặt hàng
* Đối với Admin

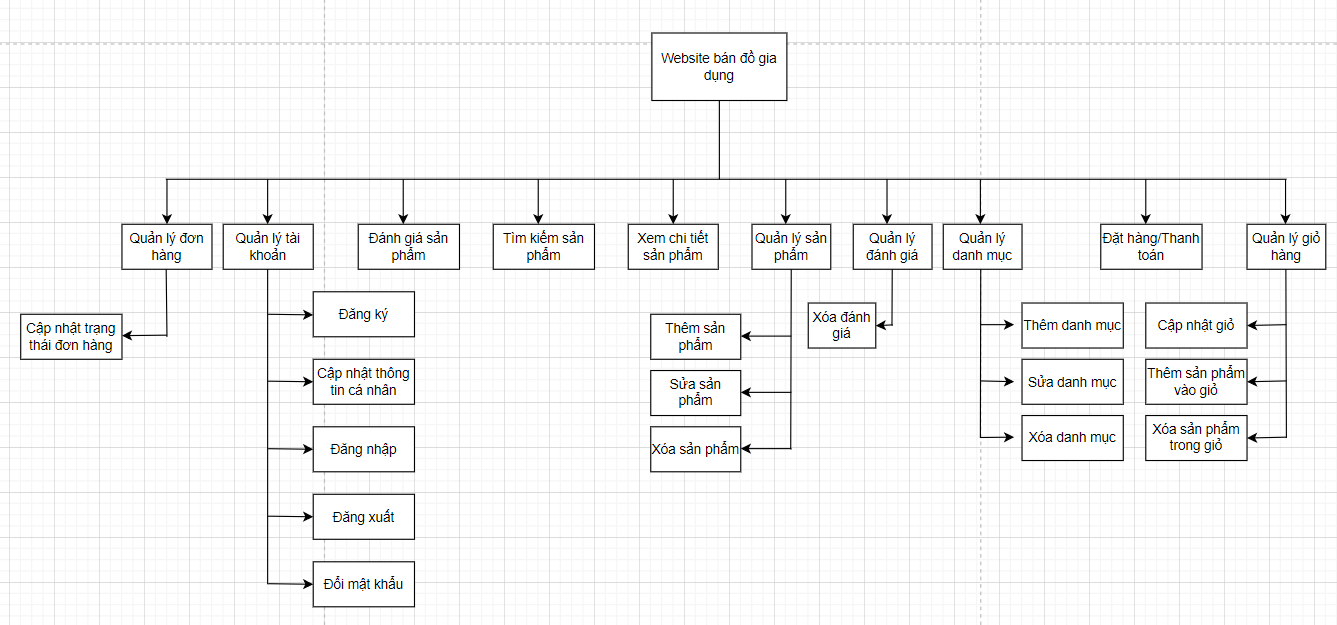
Ngoài các chức năng tương tự người dùng, Admin còn quản lý các chức năng sau:

* Quản lý danh mục sản phẩm
* Quản lý sản phẩm
* Quản lý đơn hàng.
* Quản lý đánh giá.

**2.4. Công nghệ sử dụng**

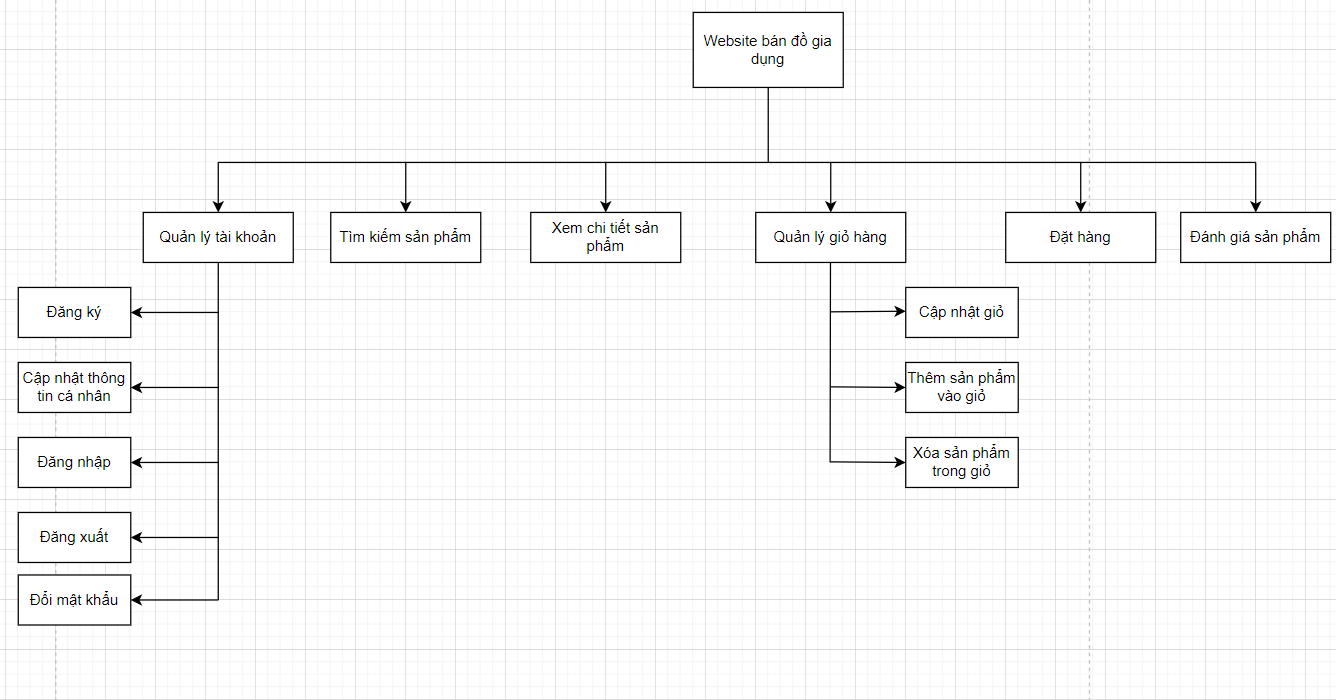
* Ngôn ngữ lập trình: PHP, HTML, CSS, JS
* Cơ sở dữ liệu: mysql
* Công cụ phát triển: Visual Studio Code

**2.5. Sơ đồ phân cấp chức năng**



Hình 2. Sơ đồ phân cấp chức năng

**2.6. Sơ đồ phân cấp chức năng phía người dùng**



Hình 2. Sơ đồ phân cấp chức năng phía người dùng

**2.7. Chức năng phía người quản trị website**

Ngoài chức năng của người dùng, người quản trị website còn có thêm các chức năng khác như:

|  |  |  |
| --- | --- | --- |
| STT | Chức năng | Mô tả |
| 1 | Quản lý sản phẩm | Người quản trị có thể thêm sản phẩm mới, sửa sản phẩm và xóa sản phẩm đã đăng. |
| 2 | Quản lý danh mục | Người quản trị có thể thêm danh mục mới, sửa danh mục và xóa danh mục đã đăng. |
| 3 | Quản lý đơn hàng | Người quản trị có thể cập nhật trạng thái đơn hàng |
| 4 | Quản lý đánh giá | Người quản trị có thể xóa những đánh giá tiêu cực, lăng mạ, spam… |

1. **Chi tiết chức năng**

**Quản lý sản phẩm:** Quản lý các sản phẩm. Có các chức năng như thêm sản phẩm mới, sửa sản phẩm, xóa sản phẩm.

*Đầu vào:* Thông tin sản phẩm.

*Xử lý:* Khi người dùng đăng nhập với tư cách là quản trị và chọn chức năng:

*Thêm sản phẩm*: Hệ thống sẽ chuyển đến trang thêm sản phẩm. Ở trang này người quản trị nhập thông tin thể loại sau đó gửi thông tin đã nhập cho hệ thống. Nếu thông tin hợp lệ, thể loại mới sẽ lưu vào cơ sở dữ liệu.

*Sửa sản phẩm*: Người quản trị chọn sản phẩm cần sửa, hệ thống sẽ chuyển đến trang sửa sản phẩm với sản phẩm đã chọn. Ở trang này người quản trị phải nhập thông tin về thể loại sau đó gửi thông tin đã nhập cho hệ thống. Nếu thông tin hợp lệ hệ thống sẽ lưu vào cơ sở dữ liệu.

*Xóa sản phẩm*: Người quản trị chọn sản phẩm cần xóa, sản phẩm sẽ được xóa ở cơ sở dữ liệu.

**Quản lý danh mục:** Quản lý thông tin các danh mục. Có các chức năng như thêm, sửa, xóa danh mục.

*Đầu vào:* Thông tin danh mục

*Xử lý:* Khi người dùng đăng nhập với tư cách là quản trị và chọn chức năng:

*Thêm danh mục*: Hệ thống sẽ chuyển đến trang thêm danh mục. Ở trang này người quản trị nhập thông tin danh mục sau đó gửi thông tin đã nhập cho hệ thống. Nếu thông tin hợp lệ, danh mục mới sẽ lưu vào cơ sở dữ liệu.

*Sửa danh mục*: Người quản trị chọn danh mục cần sửa, hệ thống sẽ chuyển đến trang sửa danh mục với danh mục đã chọn. Ở trang này người quản trị phải nhập thông tin về danh mục sau đó gửi thông tin đã nhập cho hệ thống. Nếu thông tin hợp lệ hệ thống sẽ lưu vào cơ sở dữ liệu.

*Xóa danh mục*: Người quản trị chọn danh mục cần xóa, danh mục sẽ được xóa ở cơ sở dữ liệu.

**Quản lý đơn hàng:** Quản lý các đơn hàng. Có các chức năng chuyển trạng thái đơn hàng như đơn hàng chờ xử lý, đang xử lý, hoàn thành.

*Đầu vào:* Thông tin đơn hàng từ khách hàng

*Xử lý:* Khi người dùng đăng nhập với tư cách là quản trị và chọn chức năng:

*Chờ xử lý*: Người quản trị chọn đơn hàng đang chờ xử lý, trạng thái đơn hàng sẽ chuyển sang trạng thái chờ xử lý đơn.

*Đang xử lý*: Người quản trị chọn đơn hàng đang xử lý, trạng thái đơn hàng sẽ chuyển từ trạng thái chờ xử lý sang trạng thái đang xử lý đơn hàng.

*Hoàn thành*: Người quản trị chọn đơn hàng đã được giao và chuyển sang trạng thái hoàn thành lưu ở cơ sở dữ liệu.

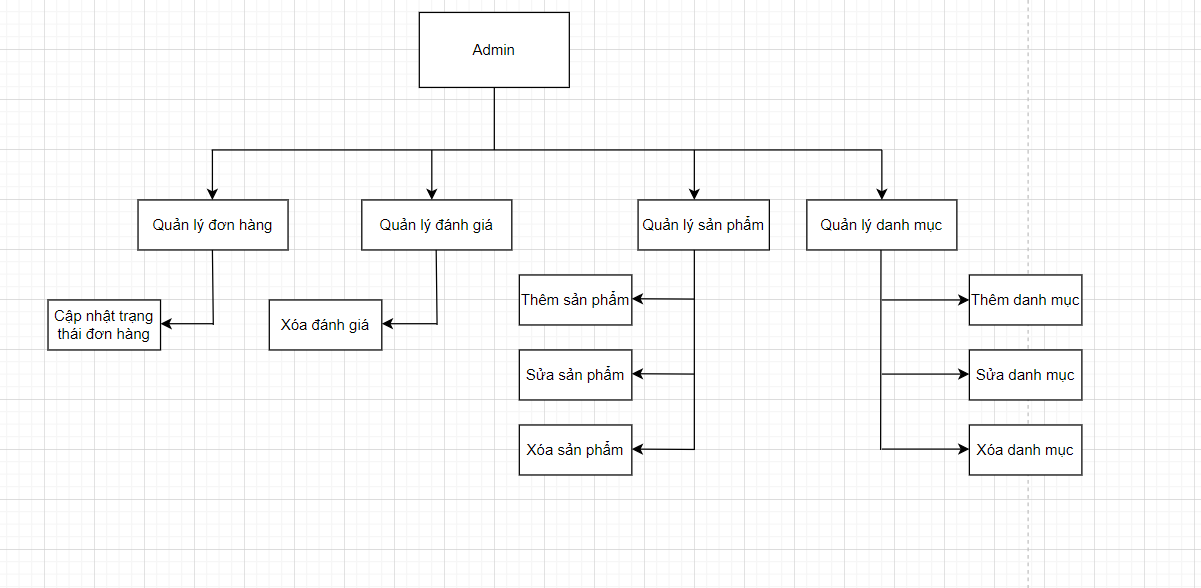
**Quản lý đánh giá:** Quản lý các đánh giá sản phẩm từ phía người dung. Có chức năng xóa những đánh giá tiêu cực, lăng mạ, spam...

*Đầu vào:* Các đánh giá sản phẩm từ khách hàng

*Xử lý:* Khi người dùng đăng nhập với tư cách là quản trị và chọn chức năng:

*Xóa đánh giá:* Người quản trị đánh giá cần xóa, đánh giá đó sẽ được xóa khỏi cơ sở dữ liệu.

1. **Sơ đồ phân cấp chức năng phía người quản trị**

****

Hình 2. Sơ đồ phân cấp chức năng phía quản trị viên

**2.8. Tóm tắt các đối tượng quản lý**

***2.8.1. Sản phẩm***

Là các mặt hàng được bán ví dụ như ấm siêu tốc, cốc…, những đồ vật liên quan đến đồ gia dụng gia đình

***2.8.2. Danh mục***

Gồm các danh mục của sản phẩm ví dụ như: đồ gia dụng bằng inox, đồ gia dụng điện…

***2.8.3. Admin***

Là người quản trị của website. Phụ trách việc quản lý sản phẩm, danh mục, đơn hàng của người dùng….

***2.8.4. Khách hàng***

Là các cá nhân tới website tìm kiếm sản phẩm cho bản thân để đặt hàng.

***2.8.5. Đơn hàng***

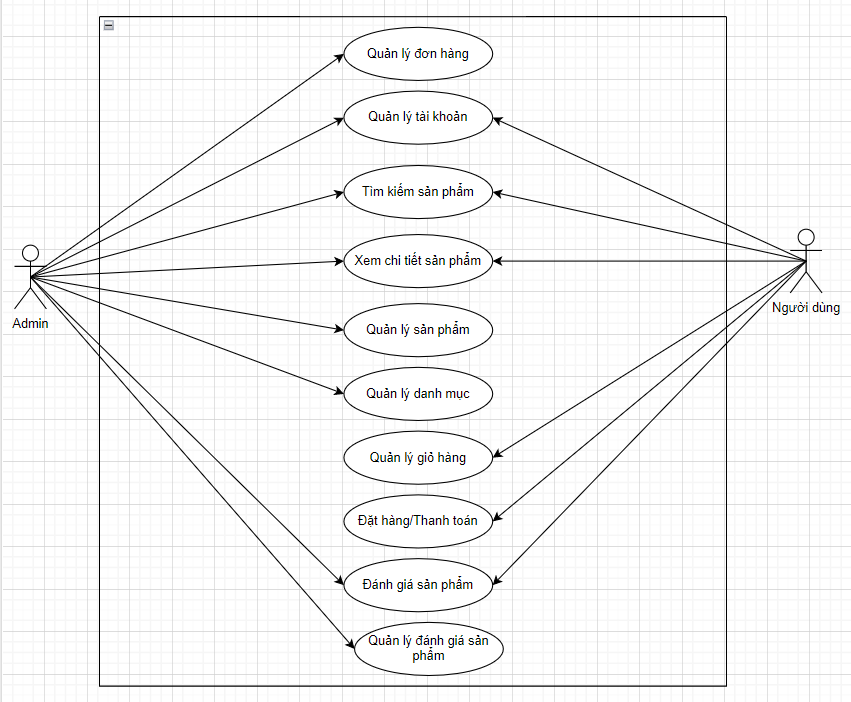
Các đơn hàng từ phía người dùng đã đặt hàng.

***2.8.6. Đánh giá***

Những đánh giá từ phía người dùng về sản phẩm.

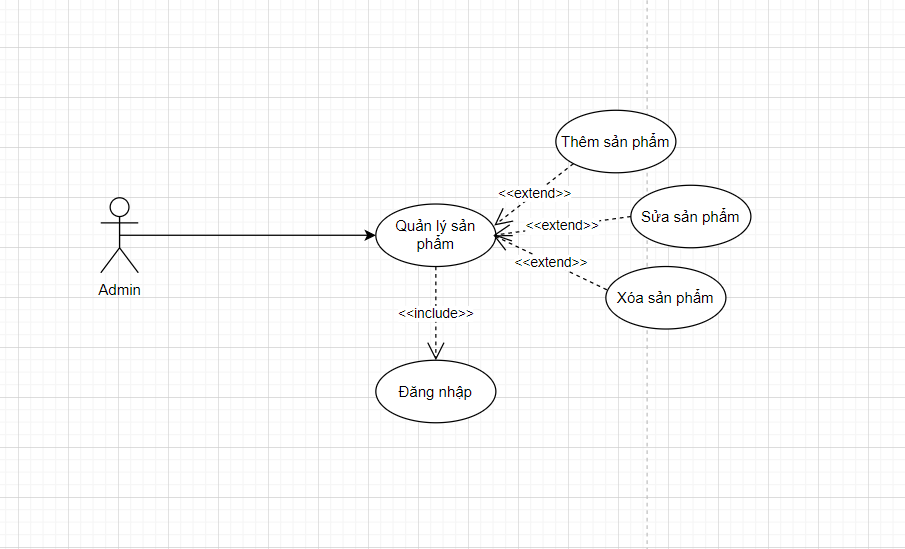
**2.9. Tài liệu đặc tả và các sơ đồ cho chức năng**

***2.9.1. Sơ đồ USE CASE tổng quát***

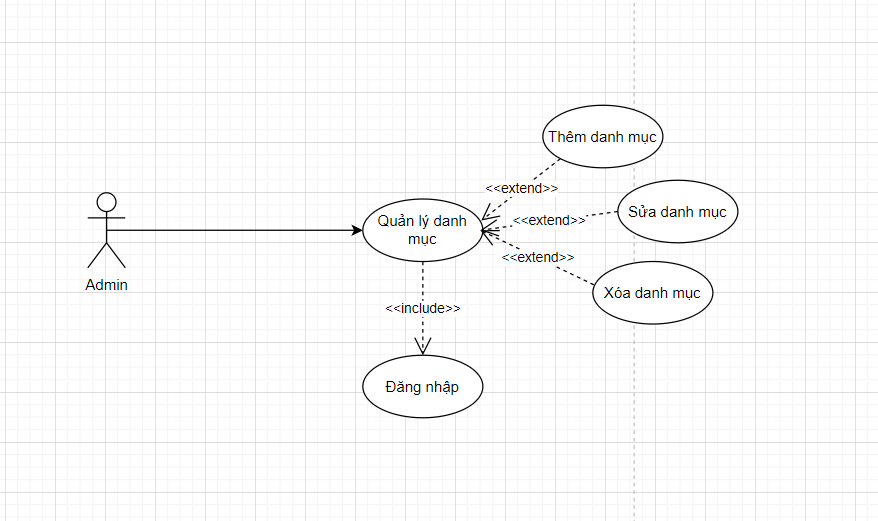
****

***2.9.1.1. Sơ đồ phân rã chức năng theo tác nhân admin***

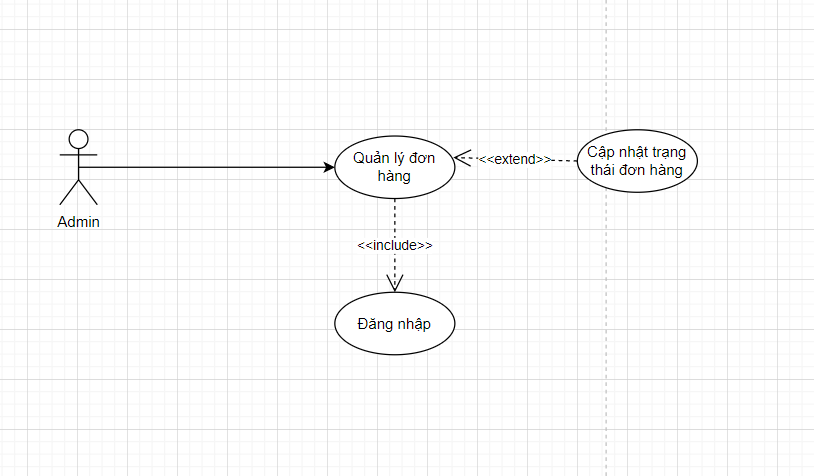
* ***UC quản lý sản phẩm***

******

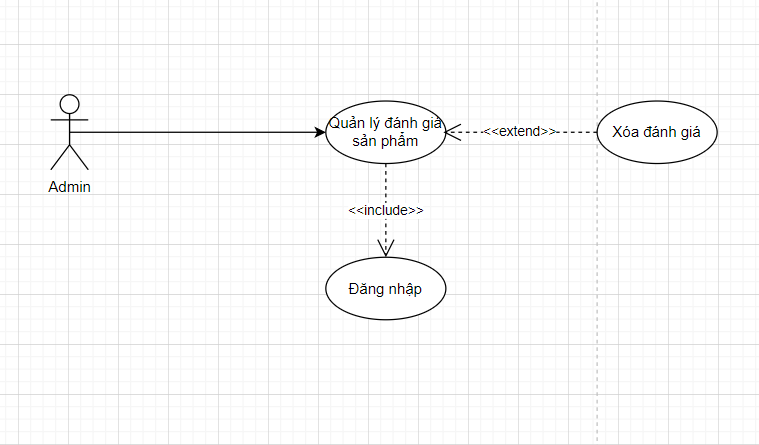
* ***UC quản lý danh mục***

******

* ***UC quản lý đơn hàng***

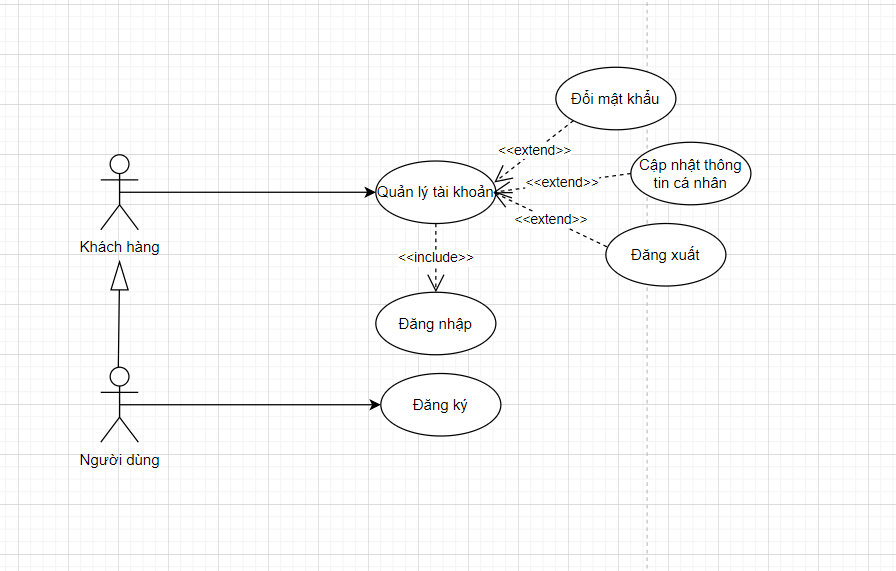
******

* ***UC quản lý đánh giá sản phẩm***

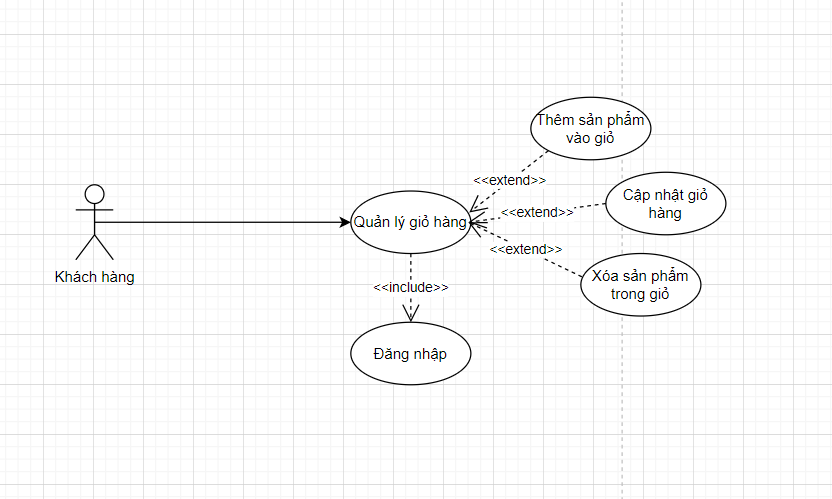
******

***2.9.1.2. Sơ đồ phân rã chức năng theo tác nhân khách hàng***

* ***UC quản lý tài khoản***

******

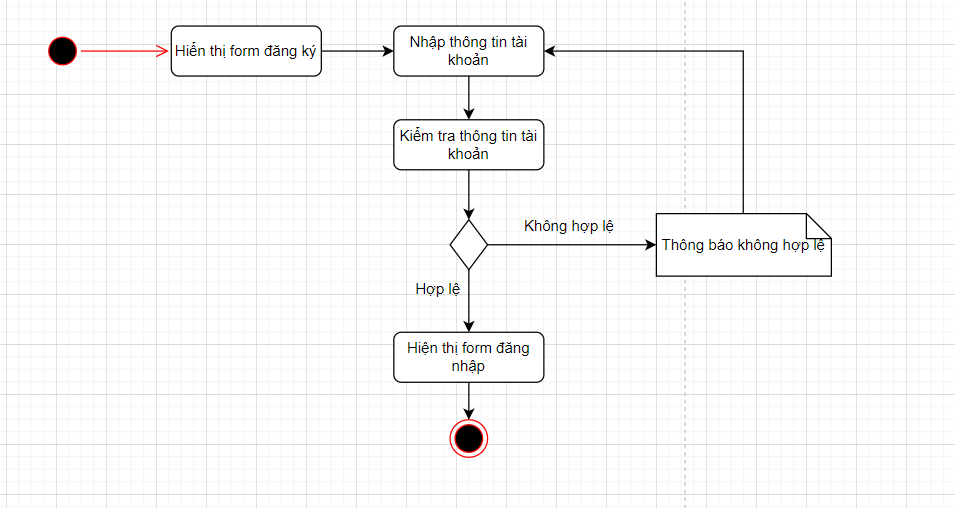
* ***UC quản lý giỏ hàng***

****

### *2.9.2. Chức năng đăng ký*

*2.9.2.1. Tài liệu đặc tả chức năng đăng ký*

|  |  |
| --- | --- |
| **Tên chức năng** | Đăng ký |
| **Tác nhân** | Người dùng |
| **Mô tả** | Chức năng này cho phép các cá nhân đăng ký tài khoản để đăng nhập hệ thống |
| **Đầu vào** | Tên tài khoản và mật khẩu |
| **Đầu ra** | Đăng ký thành công |
| **Điều kiện trước** | * Chưa có tài khoản đăng nhập * Đang ở giao diện trang chủ * Chọn “Đăng ký”. Hệ thống chuyển tới form đăng ký |
| **Điều kiện sau** | * Đăng ký tài khoản thành công: Chuyển tới form đăng nhập * Đăng ký không thành công: Không chuyển sang màn hình khác, yêu cầu kiểm tra lại tên tài khoản và mật khẩu |
| **Luồng sự kiện** | 1. Người dùng truy cập vào hệ thống website 2. Hệ thống hiển thị giao diện trang chủ 3. Chọn “Đăng ký”. Hệ thống chuyển sang form đăng ký, yêu cầu người dùng nhập tên tài khoản và mật khẩu  * Tên tài khoản textbox: là trường bắt buộc, nhập chuỗi ký tự ngẫu nhiên (độ dài lớn hơn hoặc bằng 8 kí tự), đang không tồn tại trong cơ sở dữ liệu * Mật khẩu textbox: là trường bắt buộc, nhập chuỗi ký tự ngẫu nhiên (độ dài lớn hơn hoặc bằng 8 kí tự) * Mật khẩu xác thực textbox: là trường bắt buộc, nhập chuỗi ký tự ngẫu nhiên (độ dài lớn hơn hoặc bằng 8 kí tự) và giống với mật khẩu đã nhập ở trên  1. Nhập tên tài khoản, mật khẩu và xác thực mật khẩu. Nhấn “Đăng ký” 2. Hệ thống kiểm tra thông tin tài khoản và mật khẩu. Xác nhận hợp lệ. 3. Đăng ký thành công. Hệ thống chuyển sang form đăng nhập 4. Kết thúc ca sử dụng |
| **Ngoại lệ** | 1. Người dùng không nhập thông tin tài khoản vào ô textbox và chọn “Đăng kí” 2. Người dùng nhập thông tin tài khoản đã tồn tại trong cơ sở dữ liệu 3. Người dùng nhập thông tin mật khẩu nhỏ hơn 8 kí tự 4. Người dùng nhập thông tin mật khẩu và xác thực mật khẩu không khớp nhau |

*2.9.2.2. Biểu đồ hoạt động cho chức năng đăng ký*

*2.9.2.3. Biểu đồ trình tự cho chức năng đăng ký*

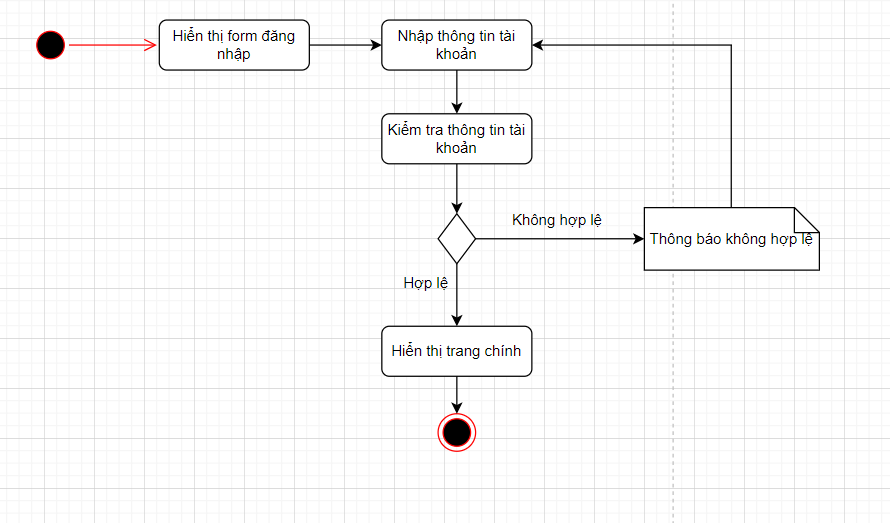
****

### *2.9.3. Chức năng đăng nhập*

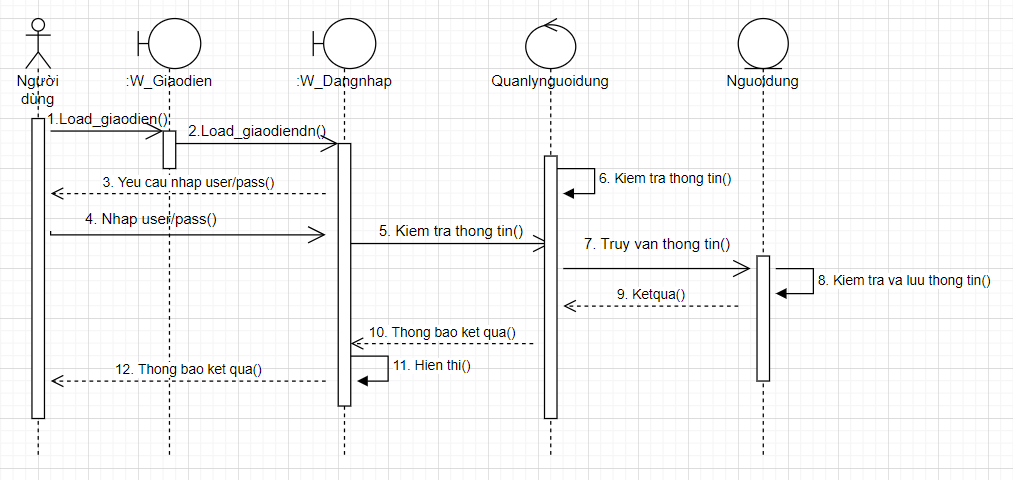
*2.9.3.1. Tài liệu đặc tả chức năng đăng nhập*

|  |  |
| --- | --- |
| **Tên chức năng** | Đăng nhập |
| **Tác nhân** | Admin, khách hàng |
| **Mô tả** | Chức năng này cho phép các cá nhân sử dụng tài khoản đã đăng ký để đăng nhập hệ thống |
| **Đầu vào** | Tên tài khoản và mật khẩu |
| **Đầu ra** | Đăng nhập thành công |
| **Điều kiện trước** | * Đã đăng ký tài khoản thành công * Chọn “Đăng nhập” |
| **Điều kiện sau** | * Đăng nhập thành công: Chuyển tới giao diện chính * Đăng nhập không thành công: Không chuyển sang màn hình khác, yêu cầu kiểm tra lại tên tài khoản và mật khẩu |
| **Luồng sự kiện** | 1. Người dùng truy cập vào hệ thống 2. Hệ thống hiển thị giao diện trang chủ 3. Chọn “Đăng nhập” 4. Hệ thống hiển thị form đăng nhập, yêu cầu người dùng nhập tài khoản và mật khẩu để đăng nhập  * Tên tài khoản textbox: là trường bắt buộc, nhập chuỗi ký tự ngẫu nhiên (độ dài lớn hơn hoặc bằng 8 kí tự), đang tồn tại trong cơ sở dữ liệu * Mật khẩu textbox: là trường bắt buộc, nhập chuỗi ký tự ngẫu nhiên (độ dài lớn hơn hoặc bằng 8 kí tự), đang tồn tại trong cơ sở dữ liệu  1. Nhập tên tài khoản và mật khẩu. Nhấn “Đăng nhập” 2. Hệ thống kiểm tra thông tin tài khoản và mật khẩu. Xác nhận hợp lệ 3. Đăng nhập thành công. Chuyển tới giao diện chính 4. Kết thúc ca sử dụng |
| **Ngoại lệ** | 1. Khách hàng không nhập thông tin tài khoản vào ô textbox và chọn “Đăng nhập” 2. Khách hàng nhập sai thông tin tài khoản trong cơ sở dữ liệu |

*2.9.3.2. Biểu đồ hoạt động cho chức năng đăng nhập*

****

*2.9.3.3. Biểu đồ trình tự cho chức năng đăng nhập*

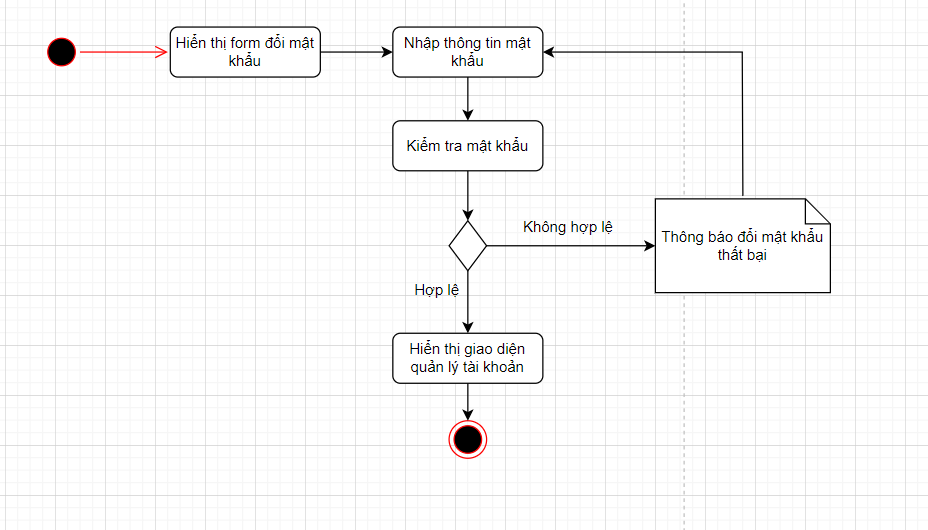


### *2.9.4. Chức năng đổi mật khẩu*

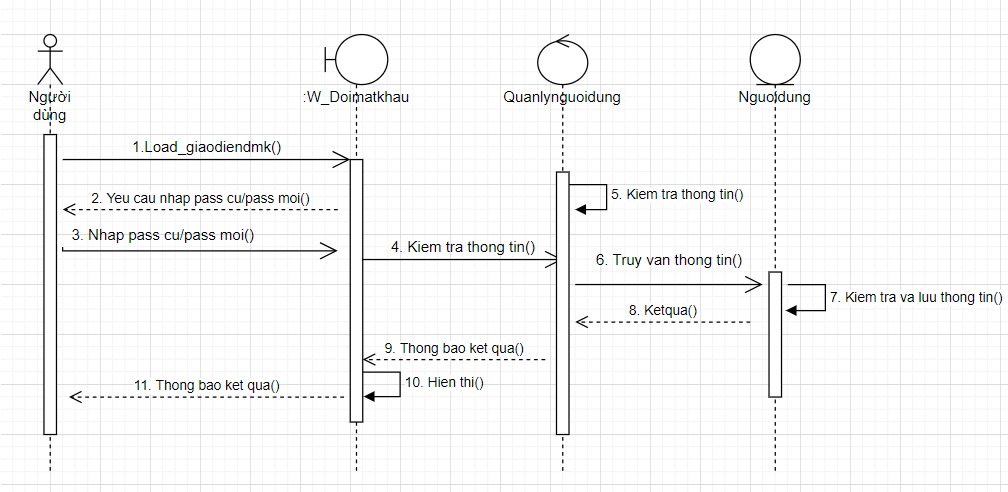
*2.9.4.1. Tài liệu đặc tả chức năng đổi mật khẩu*

|  |  |
| --- | --- |
| **Tên chức năng** | Đổi mật khẩu |
| **Tác nhân** | Admin, khách hàng |
| **Mô tả** | Chức năng này cho phép thay đổi mật khẩu |
| **Đầu vào** | Mật khẩu cũ, mật khẩu mới |
| **Đầu ra** | Đổi mật khẩu thành công |
| **Điều kiện trước** | * Đã đăng nhập hệ thống * Admin/người dùng nhớ mật khẩu * Đang ở màn hình đổi mật khẩu |
| **Điều kiện sau** | * Đổi mật khẩu thành công: Chuyển tới màn hình quản lý tài khoản * Đổi mật khẩu không thành công: Không chuyển sang màn hình khác, yêu cầu kiểm tra lại mật khẩu đã điền vào |
| **Luồng sự kiện** | 1. Người dùng đã đăng nhập vào hệ thống 2. Hệ thống hiển thị giao diện chính 3. Chọn biểu tượng “Tài Khoản” trên thanh menu 4. Hệ thống hiển thị giao diện chính của quản lý tài khoản 5. Chọn “Đổi mật khẩu” 6. Hệ thống hiển thị form đổi mật khẩu, yêu cầu khách hàng nhập mật khẩu cũ, mật khẩu mới và xác nhận lại mật khẩu mới  * Mật khẩu cũ textbox: là trường bắt buộc, nhập mật khẩu hiện tại * Mật khẩu mới textbox: là trường bắt buộc, nhập chuỗi ký tự ngẫu nhiên (độ dài lớn hơn hoặc bằng 8 kí tự) * Nhập lại mật khẩu mới textbox: là trường bắt buộc, trùng mật khẩu mới  1. Nhập vào ba textbox mật khẩu cũ, mật khẩu mới, nhập lại mật khẩu mới. Nhấn “OK” 2. Hệ thống kiểm tra mật khẩu. Xác nhận hợp lệ 3. Hệ thống cập nhật lại mật khẩu 4. Đổi mật khẩu thành công. Quay lại giao diện chính của quản lý tài khoản 5. Kết thúc ca sử dụng |
| **Ngoại lệ** | 1. Khách hàng không nhập thông tin mật khẩu vào ô textbox và chọn “OK”. 2. Khách hàng nhập sai thông tin mật khẩu cũ 3. Khách hàng nhập thông tin mật khẩu mới và xác nhận mật khẩu mới không khớp nhau |

*2.9.4.2. Biểu đồ hoạt động chức năng đổi mật khẩu*

****

*2.9.4.3. Biểu đồ trình tự cho chức năng đổi mật khẩu*

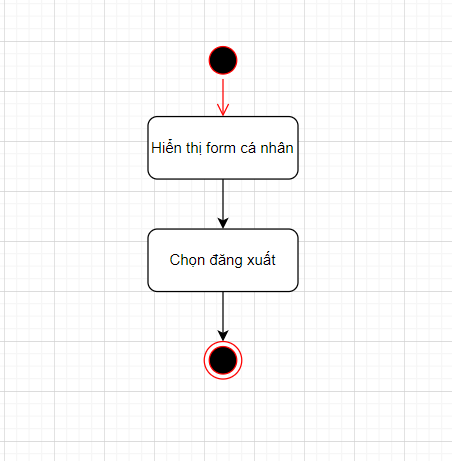
****

### *2.9.5. Chức năng đăng xuất*

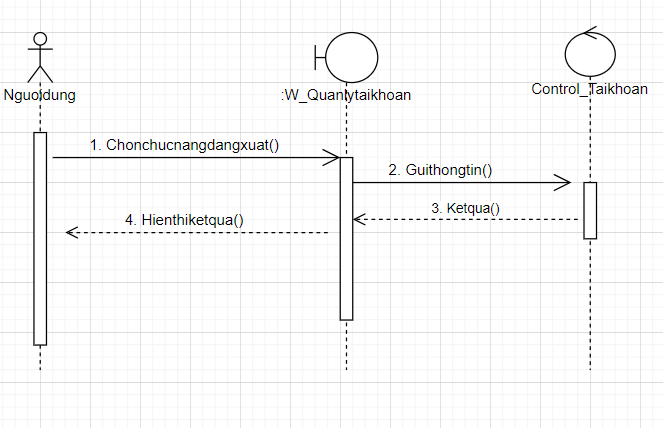
*2.9.5.1. Tài liệu đặc tả chức năng đăng xuất*

|  |  |
| --- | --- |
| **Tên chức năng** | Đăng xuất |
| **Tác nhân** | Admin, người dùng |
| **Mô tả** | Chức năng này cho phép admin/người dùng kết thúc phiên làm việc |
| **Đầu vào** |  |
| **Đầu ra** |  |
| **Điều kiện trước** | Đã đăng nhập hệ thống |
| **Điều kiện sau** | Đăng xuất khỏi hệ thống. Hiển thị giao diện trang chủ |
| **Luồng sự kiện** | 1. Chọn biểu tượng “Tài khoản” trên thanh menu 2. Hệ thống hiển thị màn hình thông tin quản lý tài khoản 3. Chọn “Đăng xuất” 4. Hệ thống chuyển sang giao diện trang chủ 5. Kết thúc ca sử dụng |
| **Ngoại lệ** |  |

*2.9.5.2. Biểu đồ hoạt động cho chức năng đăng xuất*

****

*2.9.5.3. Biểu đồ trình tự cho chức năng đăng xuất*

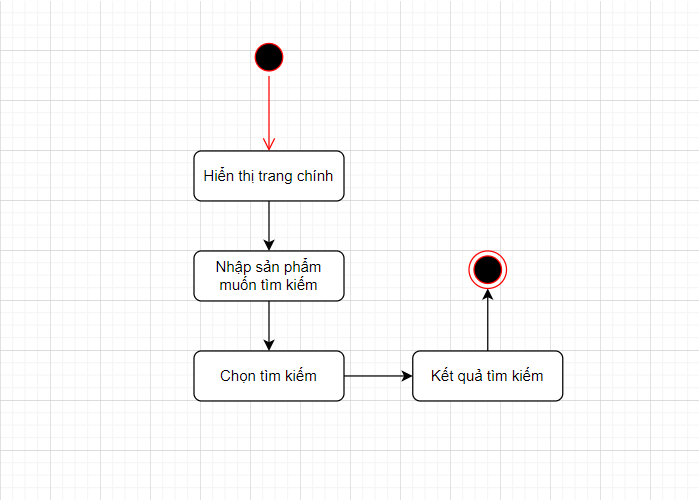
**

### *2.9.6. Chức năng tìm kiếm*

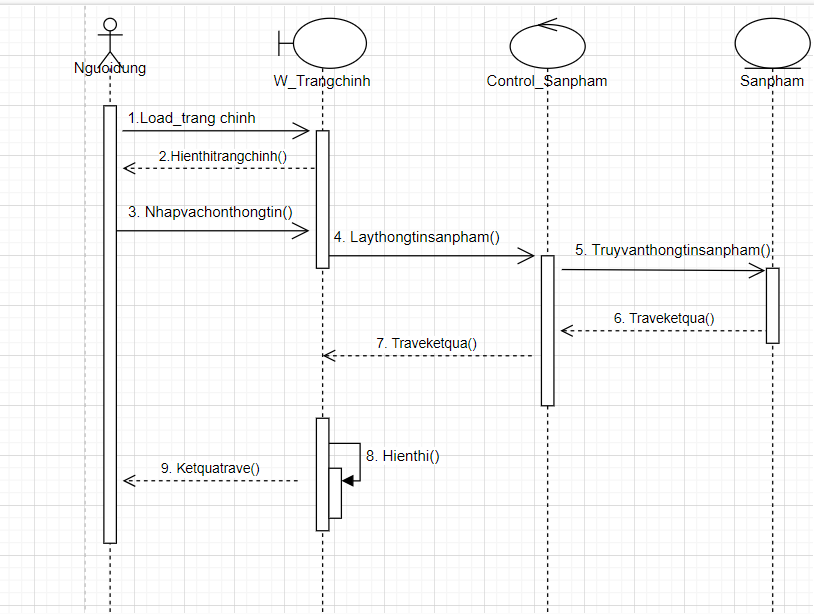
### *2.9.6.1. Tài liệu đặc tả chức năng tìm kiếm*

|  |  |
| --- | --- |
| **Tên chức năng** | Tìm kiếm |
| **Tác nhân** | Admin, người dùng |
| **Mô tả** | Cho phép admin/người dùng tìm kiếm sản phẩm muốn tham khảo |
| **Đầu vào** | Tên sản phẩm |
| **Đầu ra** | Sản phẩm muốn tìm kiếm |
| **Điều kiện trước** |  |
| **Điều kiện sau** | Chọn chức năng tìm kiếm |
| **Luồng sự kiện** | 1. Tại hệ thống giao diện chính. 2. Nhập công việc muốn tìm vào ô textbox tìm kiếm 3. Chọn “Tìm kiếm”. 4. Hệ thống hiển thị kết quả tìm kiếm. 5. Kết thúc ca sử dụng |
| **Ngoại lệ** | 1. Khách hàng bỏ trống ô textbox tìm kiếm và chọn “Tìm kiếm” |

### *2.9.6.2. Biểu đồ hoạt động cho chức năng tìm kiếm*



### *2.9.6.3. Biểu đồ trình tự cho chức năng tìm kiếm*

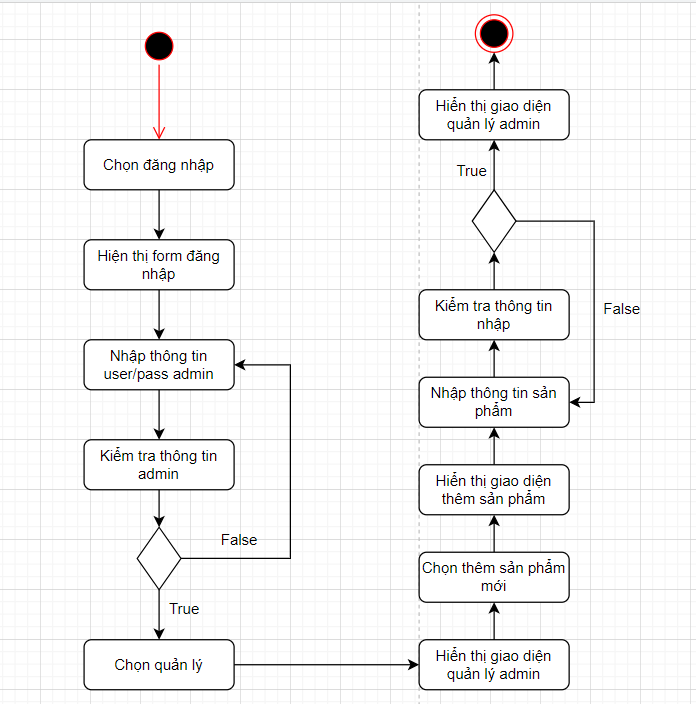


### *2.9.7. Chức năng thêm sản phẩm mới*

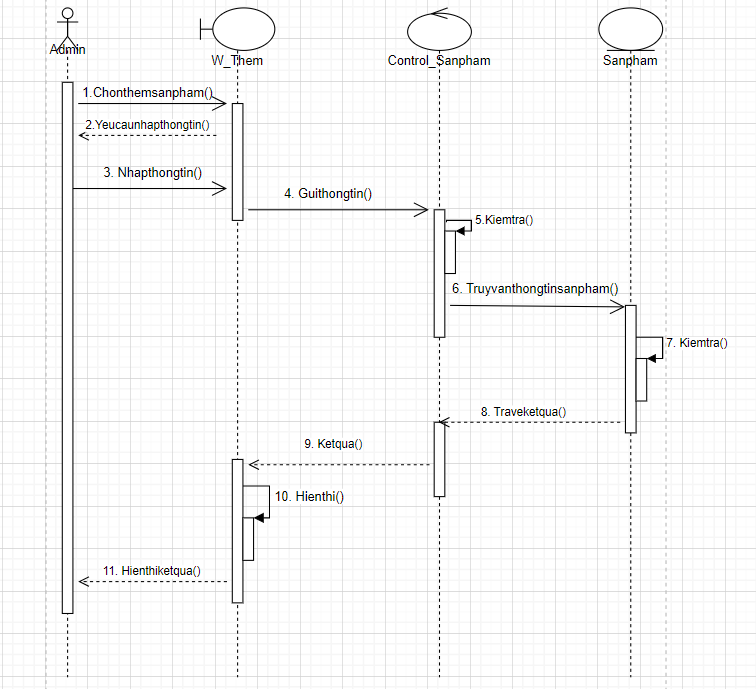
### *2.9.7.1. Tài liệu đặc tả chức năng thêm sản phẩm mới*

|  |  |
| --- | --- |
| **Tên chức năng** | Thêm sản phẩm |
| **Tác nhân** | Admin |
| **Mô tả** | Cho phép admin đăng sản phẩm mới |
| **Đầu vào** | Thông tin sản phẩm |
| **Đầu ra** | Sản phẩm mới đã được đăng và lưu vào cơ sở dữ liệu |
| **Điều kiện trước** | Đã đăng nhập vào hệ thống với tư cách quản trị viên |
| **Điều kiện sau** | Submit phần thêm sản phẩm |
| **Luồng sự kiện** | 1. Admin đăng nhập vào hệ thống 2. Chọn biểu tượng “Quản lý” trên thanh menu 3. Hệ thống chuyển qua giao diện chính của quản lý 4. Chọn chức năng “Thêm sản phẩm mới” 5. Nhập thông tin sản phẩm muốn thêm 6. Chọn “Thêm”. Thành công hệ thống quay về giao diện chính của quản lý 7. Kết thúc ca sử dụng |
| **Ngoại lệ** | 1. Admin không nhập thông tin vào ô textbox bắt buộc và ấn “Thêm” 2. Đăng nhập với tư cách không là admin |

### *2.9.7.2. Biểu đồ hoạt động cho chức năng thêm sản phẩm mới*



### *2.9.7.3. Biểu đồ trình tự cho chức năng thêm sản phẩm mới*

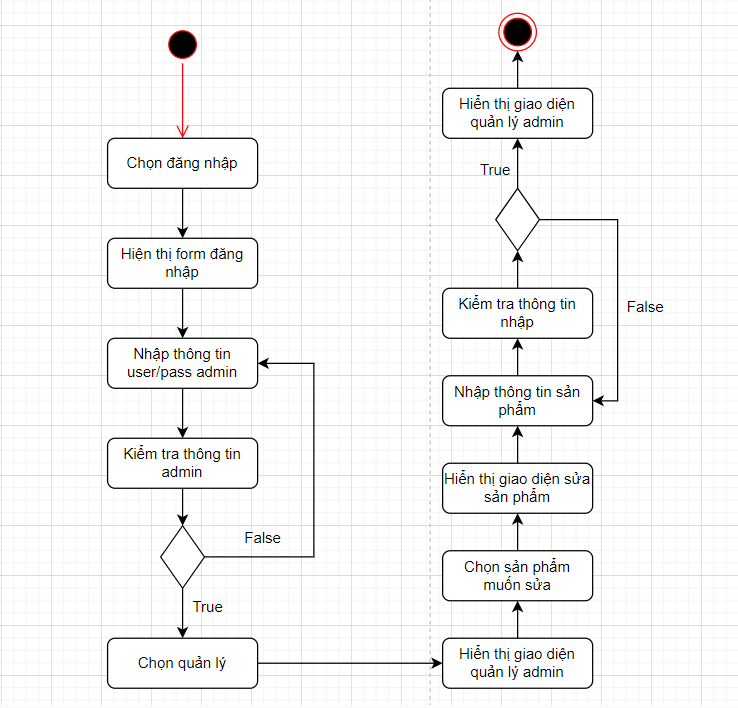


### *2.9.8. Chức năng sửa sản phẩm*

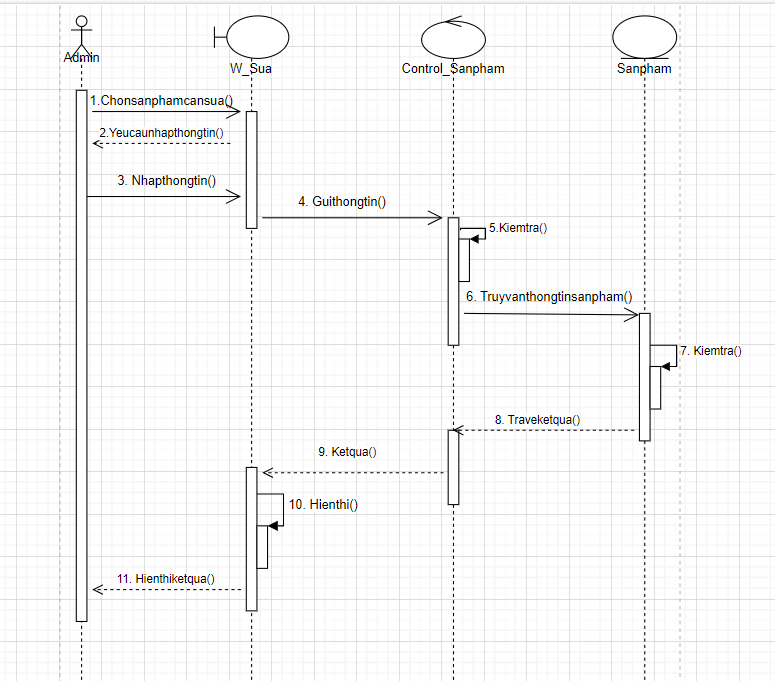
### *2.9.8.1. Tài liệu đặc tả chức năng sửa sản phẩm*

|  |  |
| --- | --- |
| **Tên chức năng** | Sửa sản phẩm |
| **Tác nhân** | Admin |
| **Mô tả** | Cho phép admin sửa sản phẩm đã đăng |
| **Đầu vào** | Thông tin sản phẩm |
| **Đầu ra** | Sản phẩm đã được sửa và cập nhật lại |
| **Điều kiện trước** | Đã đăng nhập hệ thống với tư cách admin |
| **Điều kiện sau** | Submit phần sửa sản phẩm |
| **Luồng sự kiện** | 1. Admin đăng nhập vào hệ thống 2. Chọn biểu tượng “Quản lý” trên thanh menu 3. Hệ thống chuyển qua giao diện chính của quản lý admin 4. Chọn sản phẩm muốn sửa. Chọn chức năng “Sửa” 5. Hệ thống sẽ hỏi “Bạn có chắc chắn muốn sửa không?” 6. Nhấn “Ok” để xác nhận 7. Nhập thông tin muốn sửa 8. Chọn “Sửa”. Thành công hệ thống quay về giao diện chính của quản lý admin 9. Kết thúc ca sử dụng |
| **Ngoại lệ** | 1. Admin không nhập thông tin vào ô textbox bắt buộc và ấn “Sửa” 2. Đăng nhập với tư cách không là admin |

### *2.9.8.2. Biểu đồ hoạt động cho chức năng sửa sản phẩm*



### *2.9.8.3. Biểu đồ trình tự cho chức năng sửa sản phẩm*

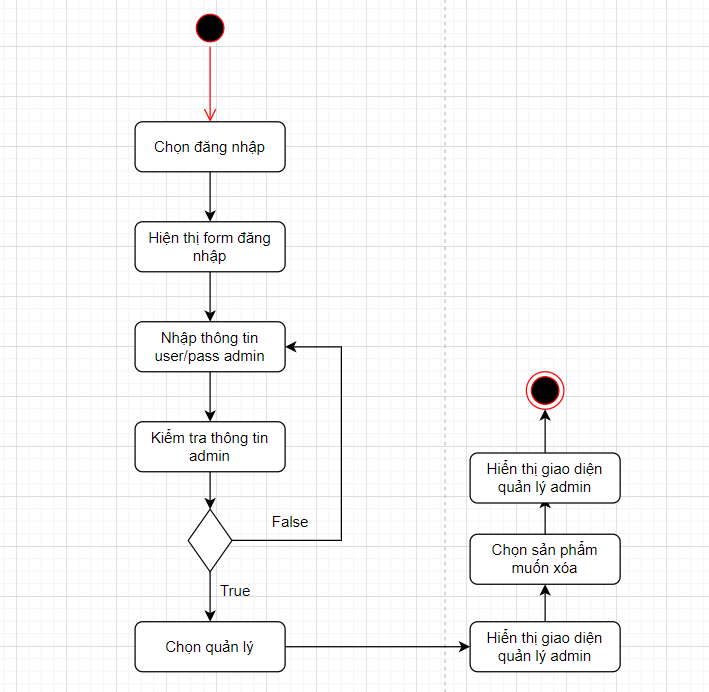
**

### *2.9.9. Chức năng xoá sản phẩm*

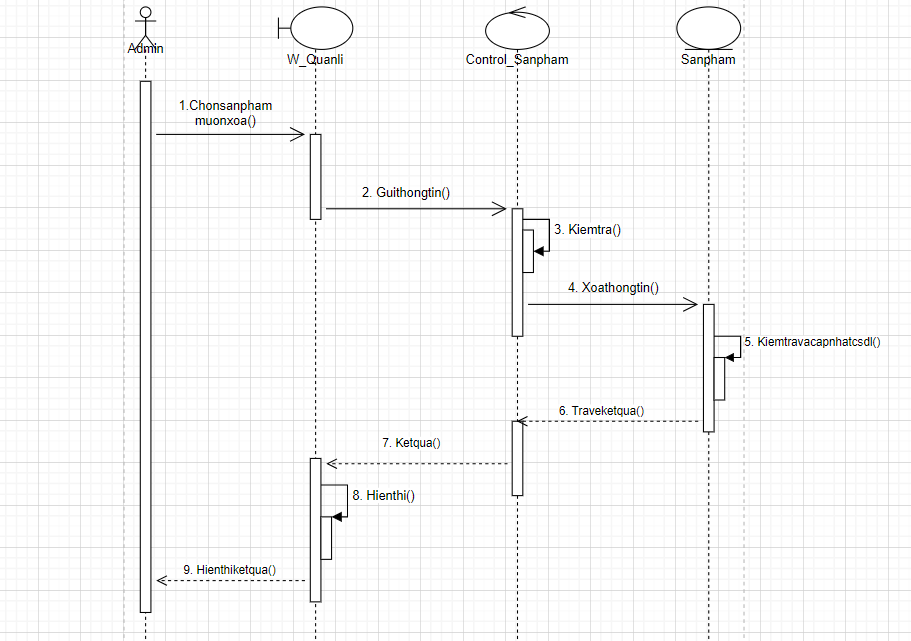
### *2.9.9.1. Tài liệu đặc tả chức năng xóa sản phẩm*

|  |  |
| --- | --- |
| **Tên chức năng** | Xóa sản phẩm |
| **Tác nhân** | Admin |
| **Mô tả** | Cho phép người dùng xóa sản phẩm |
| **Đầu vào** | Thông tin sản phẩm cần xóa |
| **Đầu ra** | Sản phẩm đã được xóa trong cơ sở dữ liệu |
| **Điều kiện trước** | Đã đăng nhập hệ thống với tư cách admin |
| **Điều kiện sau** | Submit phần xóa |
| **Luồng sự kiện** | 1. Admin đăng nhập vào hệ thống 2. Chọn biểu tượng “Quản lý” trên thanh menu 3. Hệ thống chuyển qua giao diện chính của quản lý admin 4. Chọn sản phẩm muốn xóa. Chọn chức năng “Xóa” 5. Hệ thống sẽ hỏi “Bạn có chắc chắn muốn xóa không?” 6. Nhấn “Ok” để xác nhận 7. Thành công hệ thống quay về giao diện chính của quản lý admin 8. Kết thúc ca sử dụng |
| **Ngoại lệ** | 1. Đăng nhập với tư cách không phải là admin |

### *2.9.9.2. Biểu đồ hoạt động cho chức năng xóa sản phẩm*



### *2.9.9.3. Biểu đồ trình tự cho chức năng xóa sản phẩm*

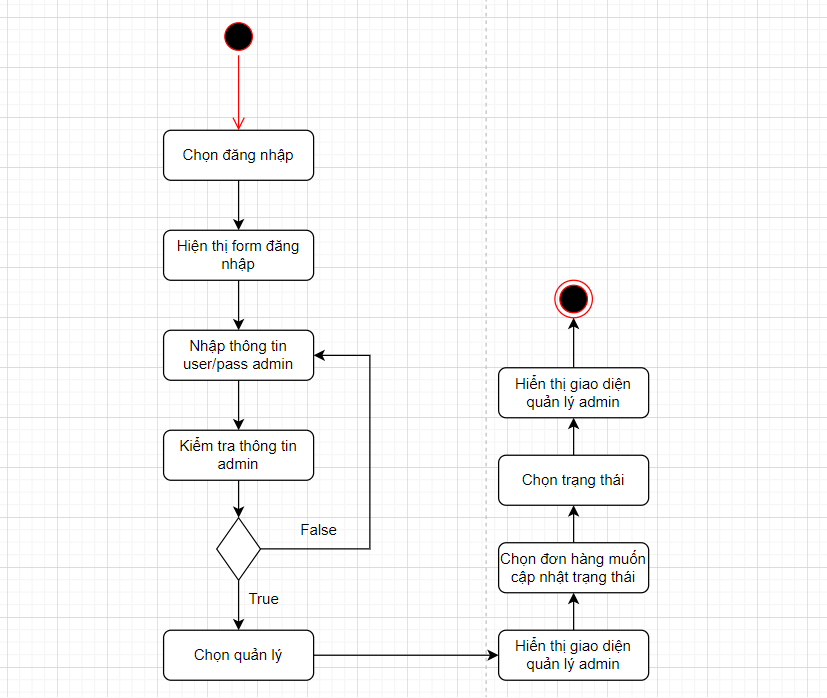


### *2.9.10. Chức năng quản lý đơn hàng*

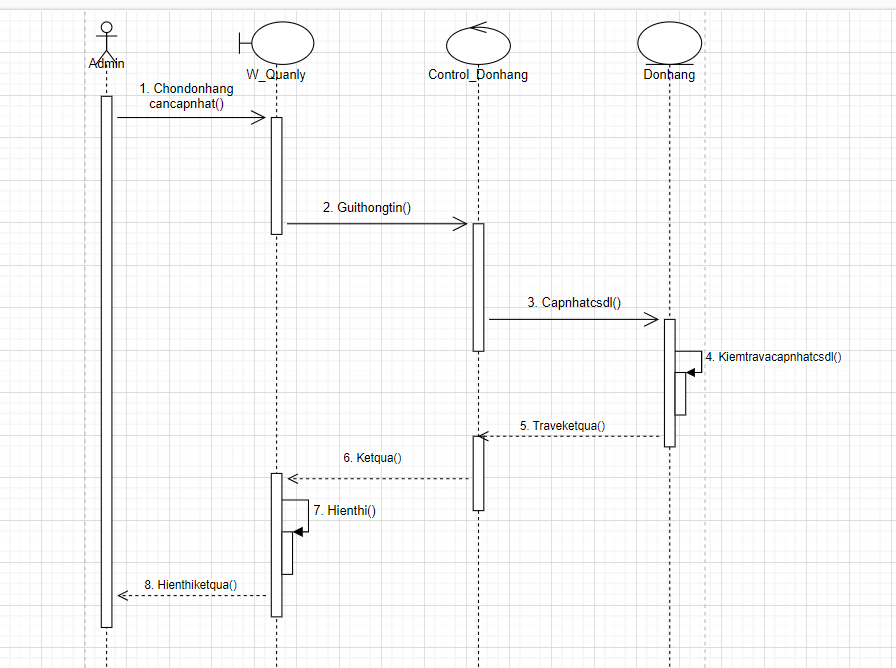
### *2.9.10.1. Tài liệu đặc tả chức năng quản lý đơn hàng*

|  |  |
| --- | --- |
| **Tên chức năng** | Quản lí đơn hàng |
| **Tác nhân** | Admin |
| **Mô tả** | Cho phép admin cập nhật trạng thái của đơn hàng |
| **Đầu vào** | Thông tin đơn hàng cần cập nhật trạng thái |
| **Đầu ra** | Đơn hàng đã được cập nhật trạng thái |
| **Điều kiện trước** | Đã đăng nhập hệ thống bằng admin |
| **Điều kiện sau** | Submit phần cập nhật đơn |
| **Luồng sự kiện** | 1. Admin đăng nhập vào hệ thống 2. Chọn biểu tượng “Quản lý” trên thanh menu 3. Hệ thống chuyển qua giao diện chính của quản lý admin 4. Chọn đơn hàng và trạng thái cần thay đổi 5. Chọn chức năng “Cập nhật” để thay đổi trạng thái đơn hàng 6. Thành công hệ thống quay về giao diện chính của quản lý admin 7. Kết thúc ca sử dụng |
| **Ngoại lệ** |  |

### *2.9.10.2. Biểu đồ hoạt động cho chức năng quản lý đơn hàng*



### *2.9.10.3. Biểu đồ trình tự cho chức năng quản lý đơn hàng*

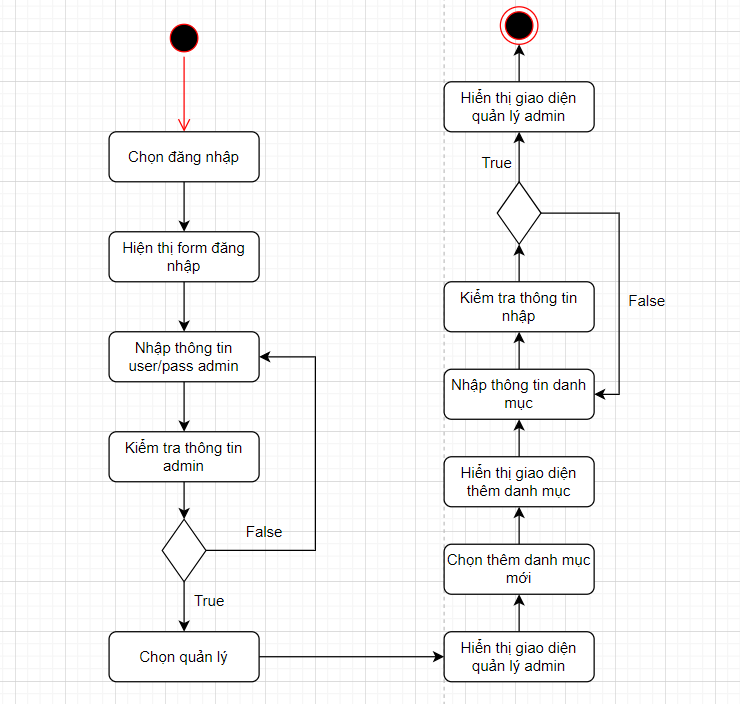
**

### *2.9.11. Chức năng thêm danh mục mới*

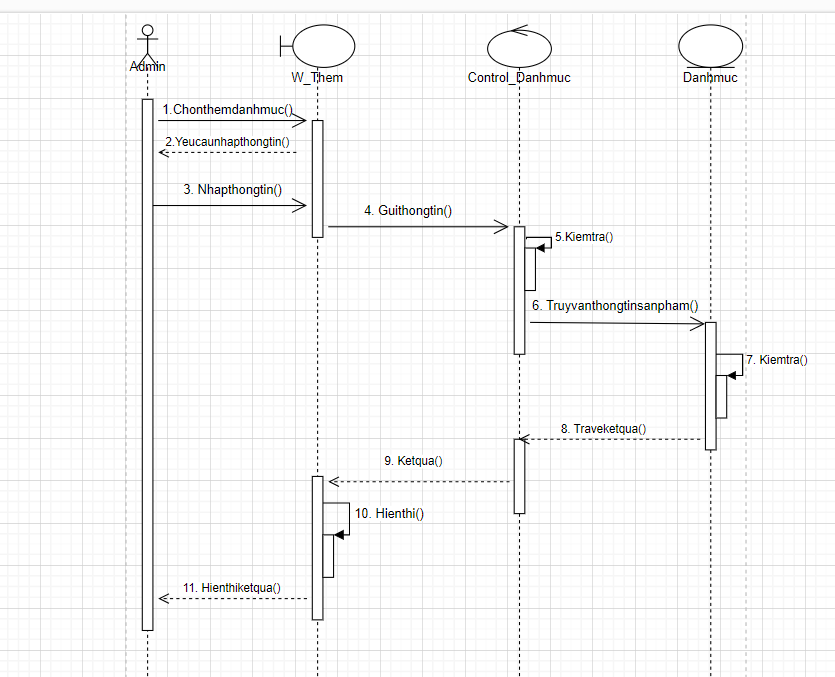
### *2.9.11.1. Tài liệu đặc tả chức năng thêm danh mục mới*

|  |  |
| --- | --- |
| **Tên chức năng** | Thêm danh mục mới |
| **Tác nhân** | Admin |
| **Mô tả** | Cho phép admin cập nhật thêm danh mục mới |
| **Đầu vào** | Thông tin danh mục mới |
| **Đầu ra** | Danh mục mới được lưu vào cơ sở dữ liệu |
| **Điều kiện trước** | Đăng nhập hệ thống với tư cách admin |
| **Điều kiện sau** | Submit phần thêm danh mục |
| **Luồng sự kiện** | 1. Admin đăng nhập vào hệ thống 2. Chọn biểu tượng “Quản lý” trên thanh menu 3. Hệ thống chuyển qua giao diện chính của quản lý 4. Chọn chức năng “Thêm danh mục mới” 5. Nhập thông tin danh mục muốn thêm 6. Chọn “Thêm”. Thành công hệ thống quay về giao diện chính của quản lý 7. Kết thúc ca sử dụng |
| **Ngoại lệ** | 1. Admin không nhập thông tin vào ô textbox bắt buộc và ấn “Thêm” 2. Đăng nhập với tư cách không là admin |

### *2.9.11.2. Biểu đồ hoạt động cho chức năng thêm danh mục mới*

****

### *2.9.11.3. Biểu đồ trình tự cho chức năng thêm danh mục mới*

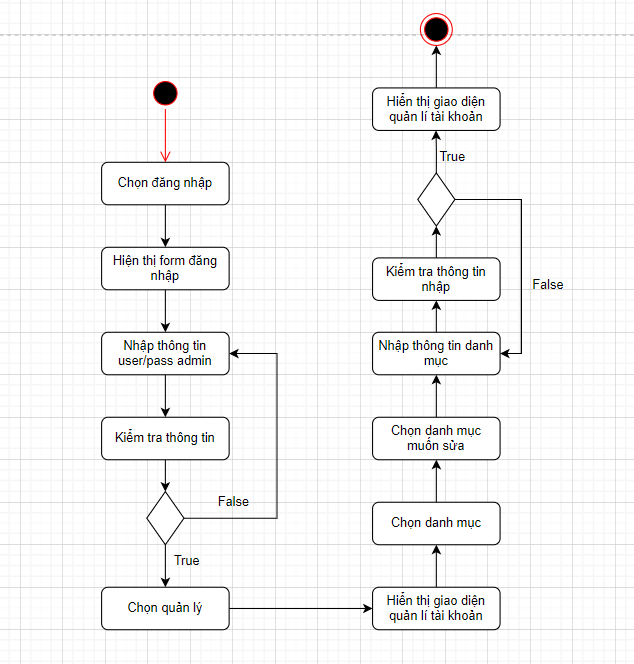


### *2.9.12. Chức năng sửa danh mục*

### *2.9.12.1. Tài liệu đặc tả chức năng sửa danh mục*

|  |  |
| --- | --- |
| **Tên chức năng** | Chỉnh sửa danh mục |
| **Tác nhân** | Admin |
| **Mô tả** | Cho phép admin chỉnh sửa lại danh mục |
| **Đầu vào** | Thông tin danh mục cần sửa |
| **Đầu ra** | Danh mục được cập nhật và lưu vào cơ sở dữ liệu |
| **Điều kiện trước** | Đăng nhập hệ thống bằng tài khoản admin |
| **Điều kiện sau** | Submit phần sửa danh mục |
| **Luồng sự kiện** | 1. Admin đăng nhập vào hệ thống 2. Chọn biểu tượng “Quản lý” trên thanh menu 3. Hệ thống chuyển qua giao diện chính của quản lý 4. Tại danh mục muốn chỉnh sửa, chọn “Sửa” 5. Hệ thống hiển thị thông báo “Bạn có chắc chắn muốn sửa?” 6. Nhấn “OK” để xác nhận 7. Hệ thống chuyển qua giao diện thông tin danh mục 8. Admin có thể sửa danh mục 9. Nhập thông tin muốn sửa 10. Chọn “Sửa” 11. Nếu thành công, hệ thống hiển thị giao diện quản lí 12. Kết thúc ca sử dụng |
| **Ngoại lệ** | 1. Admin không nhập thông tin vào ô textbox bắt buộc và ấn “Sửa” 2. Đăng nhập với tư cách không là admin |

### *2.9.12.2. Biểu đồ hoạt động cho chức năng sửa danh mục*



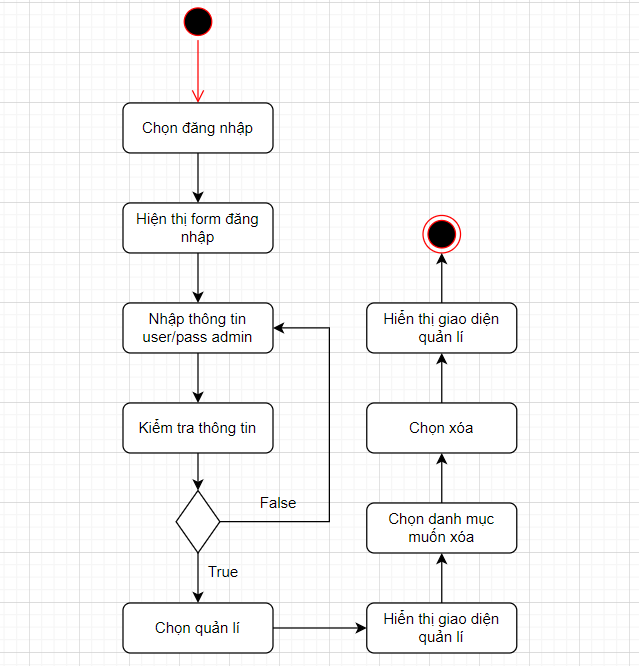
### *2.9.12.3. Biểu đồ trình tự cho chức năng sửa danh mục*

### *2.9.13. Chức năng xóa danh mục*

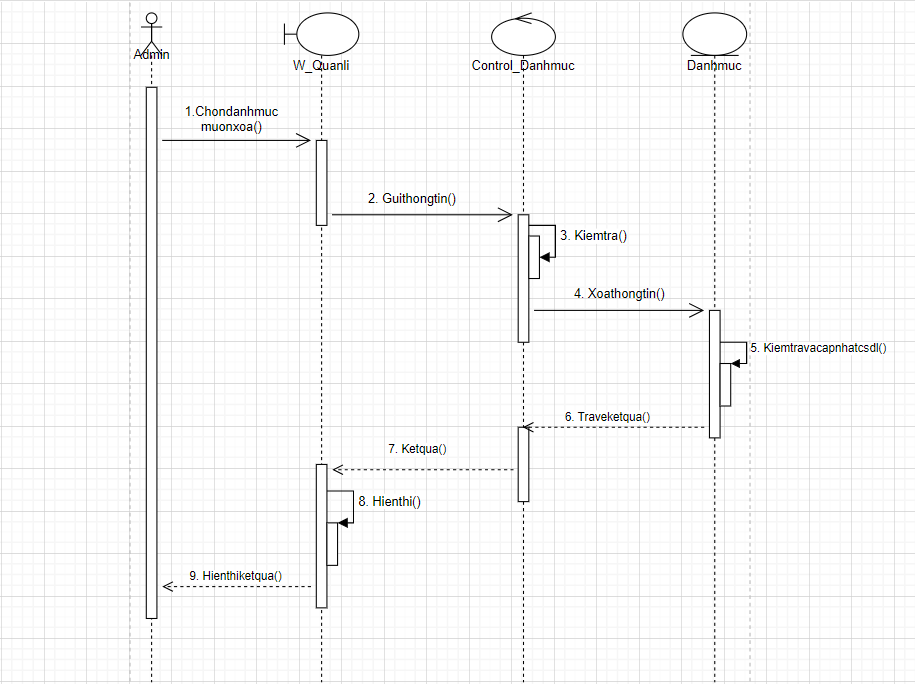
### *2.9.13.1. Tài liệu đặc tả chức năng xóa danh mục*

|  |  |
| --- | --- |
| **Tên chức năng** | Xóa danh mục |
| **Tác nhân** | Admin |
| **Mô tả** | Cho phép admin xóa danh mục |
| **Đầu vào** | Thông tin danh mục cần xóa |
| **Đầu ra** | Danh mục đã được xóa trong cơ sở dữ liệu |
| **Điều kiện trước** | Đăng nhập hệ thống bằng tài khoản admin |
| **Điều kiện sau** | Submit phần xóa danh mục |
| **Luồng sự kiện** | 1. Admin đăng nhập vào hệ thống 2. Chọn biểu tượng “Quản lý” trên thanh menu 3. Hệ thống chuyển qua giao diện chính của quản lý 4. Tại danh mục muốn xóa, chọn “Xóa” 5. Hệ thống hiển thị thông báo “Bạn có chắc chắn muốn xóa?” 6. Chọn “OK” để xác nhận xóa 7. Nếu xóa thành công, hệ thống cập nhật danh mục. Hiển thị giao diện quản lí 8. Kết thúc ca sử dụng |
| **Ngoại lệ** |  |

### *2.9.13.2. Biểu đồ hoạt động chức năng xóa danh mục*



### *2.9.13.3. Biểu đồ trình tự cho chức năng xóa danh mục*

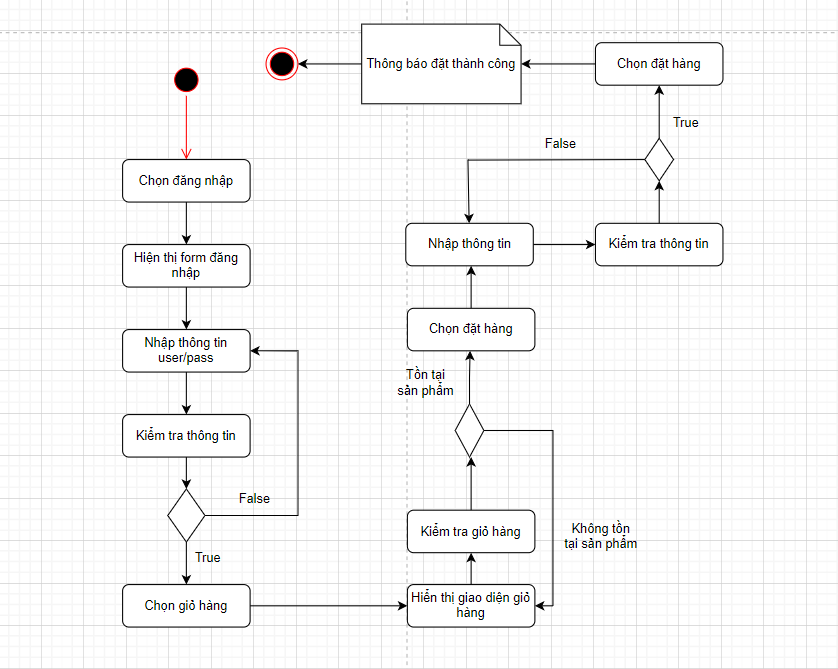


### *2.9.14. Chức năng đặt hàng*

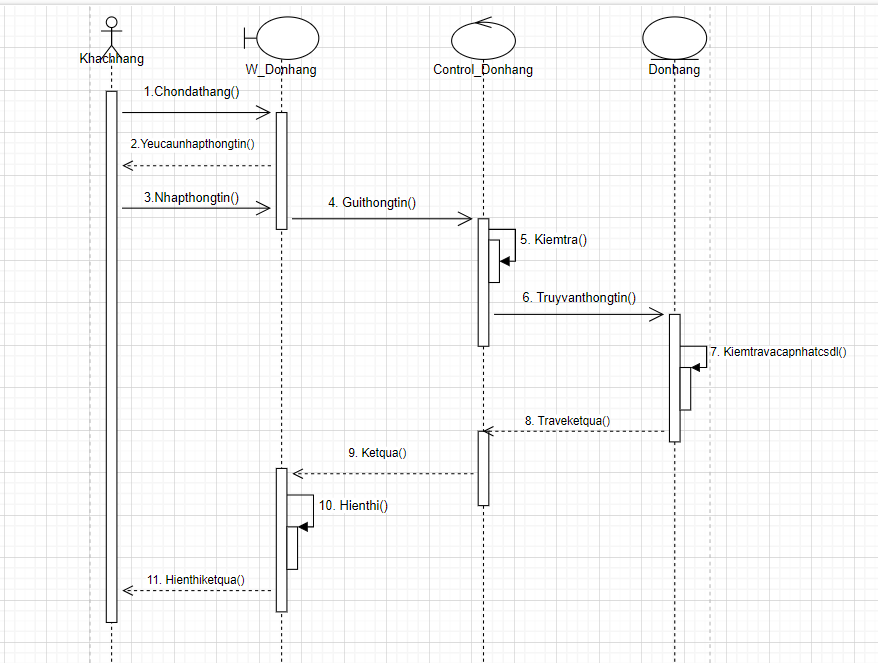
### *2.9.14.1. Tài liệu đặc tả chức năng đặt hàng*

|  |  |
| --- | --- |
| **Tên chức năng** | Đặt hàng |
| **Tác nhân** | Admin, khách hàng |
| **Mô tả** | Cho phép khách hàng đặt sản phẩm |
| **Đầu vào** | Thông tin sản phẩm và thông tin khách hàng |
| **Đầu ra** | Đơn hàng được đặt |
| **Điều kiện trước** | Đăng nhập hệ thống bằng tài khoản đã đăng ký |
| **Điều kiện sau** | Sản phẩm đã được đặt |
| **Luồng sự kiện** | 1. Khách hàng đăng nhập vào hệ thống 2. Chọn biểu tượng “Giỏ hàng” trên thanh menu 3. Hệ thống chuyển qua giao diện chính của giỏ hàng 4. Tại giỏ hàng, chọn “Đặt hàng” 5. Khách hàng điền thông tin cá nhân để nhận hàng 6. Chọn “Đặt hàng” để xác nhận đặt hàng 7. Nếu thành công, hệ thống thông báo cảm ơn 8. Kết thúc ca sử dụng |
| **Ngoại lệ** | 1. Khách hàng không đăng nhập vào hệ thống 2. Trong giỏ hàng không có sản phẩm |

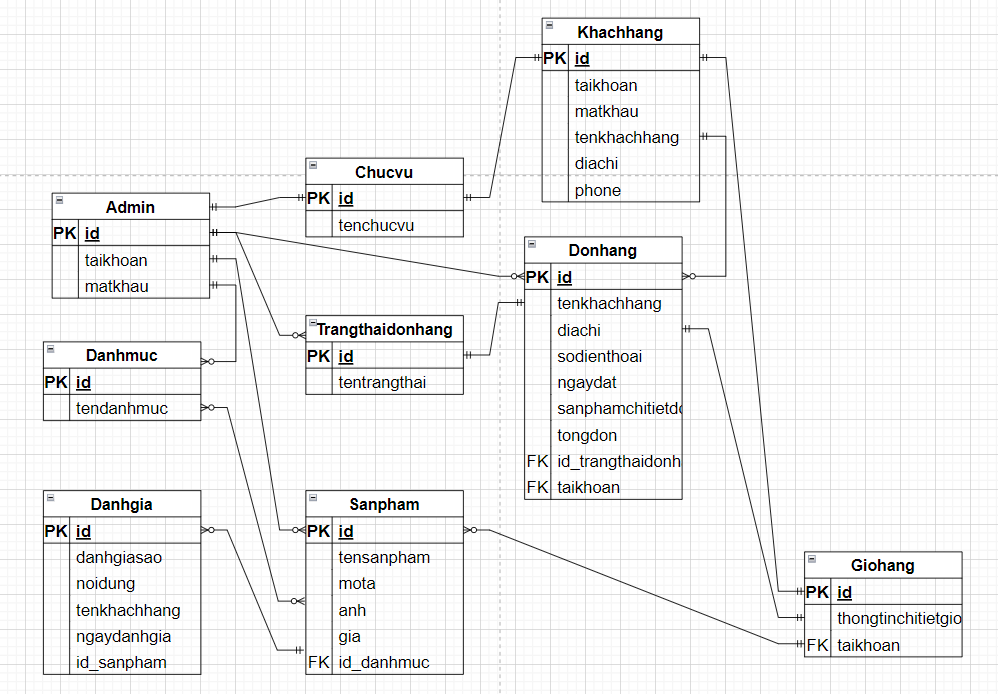
### *2.9.14.2. Biểu đồ hoạt động chức năng đặt hàng*



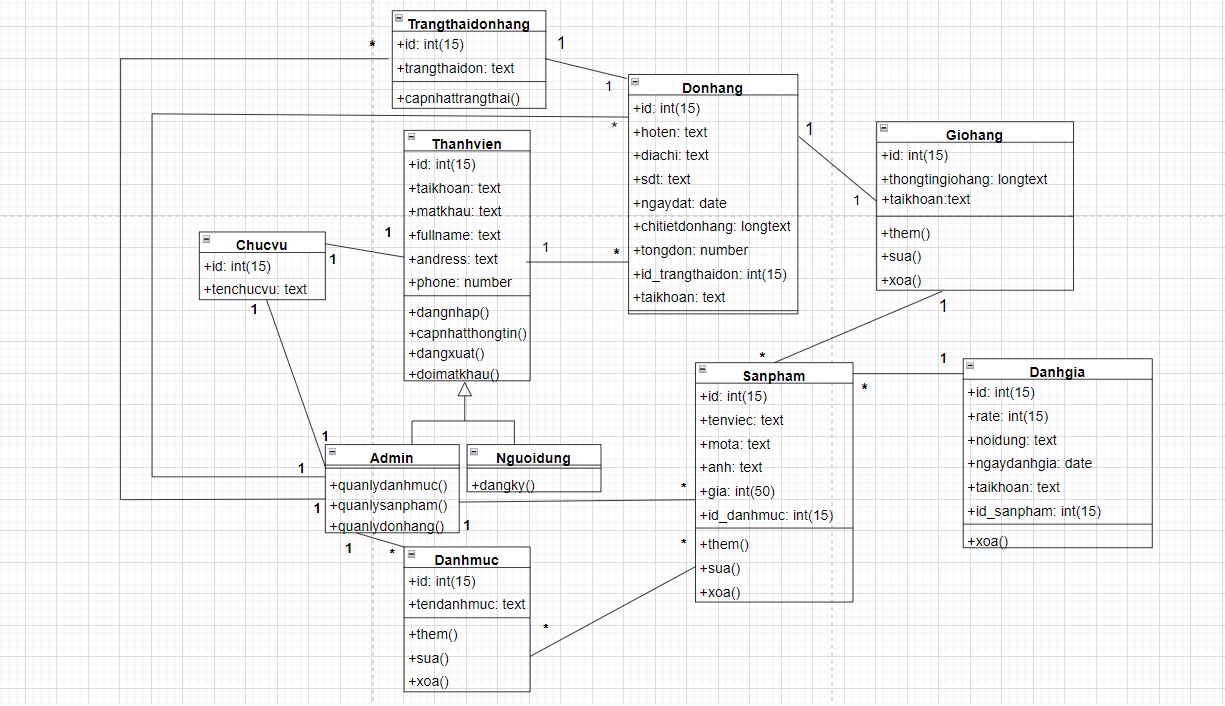
### *2.9.14.3. Biểu đồ trình tự chức năng đặt hàng*



**2.10. Mô hình thực thể liên kết**



**2.11. Biểu đồ lớp**



**2.12. Thiết kế các bảng dữ liệu**

***2.12.1. Các bảng dữ liệu***

|  |  |  |
| --- | --- | --- |
| STT | Tên bảng | Diễn giải |
| 1 | products | Bảng sản phẩm |
| 2 | categories | Bảng danh mục |
| 3 | role\_user | Bảng chức vụ |
| 4 | status\_bill | Bảng trạng thái đơn hàng |
| 5 | user | Bảng thành viên |
| 6 | cart | Bảng giỏ hàng |
| 7 | bill | Bảng hóa đơn |
| 8 | reviews | Bảng đánh giá |

1. **Bảng products**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Diễn giải | Kiểu dữ liệu | Khóa |
| 1 | id | Id sản phẩm | int | Khóa chính |
| 2 | title | Tên sản phẩm | text |  |
| 3 | descrip | Mô tả sản phẩm | text |  |
| 4 | img | Ảnh sản phẩm | text |  |
| 5 | price | Giá sản phẩm | int |  |
| 6 | id\_category | Danh mục sản phẩm | int | Khóa ngoại |

1. **Bảng categories**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Diễn giải | Kiểu dữ liệu | Khóa |
| 1 | id | Id danh mục | int | Khóa chính |
| 2 | title | Tên danh mục | text |  |

1. **Bảng user**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Diễn giải | Kiểu dữ liệu | Khóa |
| 1 | id | Id user | int | Khóa chính |
| 2 | username | Tên tài khoản | text | Khóa chính |
| 3 | pass | Mật khẩu | text |  |
| 4 | fullname | Họ và tên | text |  |
| 4 | andress | Địa chỉ | text |  |
| 5 | id\_role | Chức vụ | int | Khóa ngoại |

1. **Bảng role\_user**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Diễn giải | Kiểu dữ liệu | Khóa |
| 1 | id | Id chức vụ | int | Khóa chính |
| 2 | role | Tên chức vụ | text |  |

1. **Bảng status\_bill**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Diễn giải | Kiểu dữ liệu | Khóa |
| 1 | id | Id trạng thái | int | Khóa chính |
| 2 | title | Tên trạng thái | text |  |

1. **Bảng cart**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Diễn giải | Kiểu dữ liệu | Khóa |
| 1 | id | Id giỏ hàng | int | Khóa chính |
| 2 | cart\_detail | Chi tiết giỏ hàng | longtext |  |
| 3 | id\_user | Tài khoản | int | Khóa ngoại |

1. **Bảng bill**

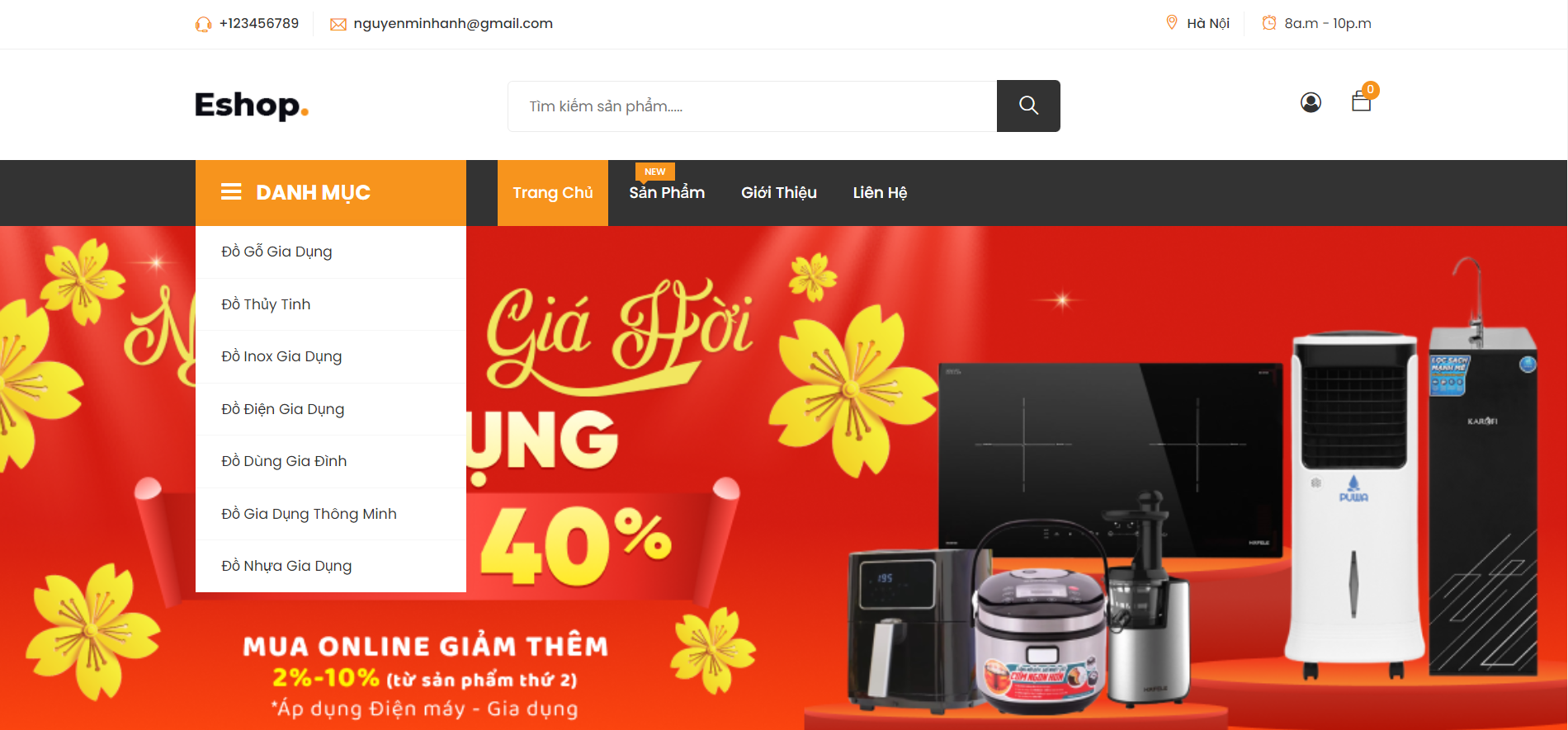
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Diễn giải | Kiểu dữ liệu | Khóa |
| 1 | id | Id đơn hàng | int | Khóa chính |
| 2 | fullname | Họ và tên | text |  |
| 3 | andress | Địa chỉ | text |  |
| 4 | phone | Số điện thoại | int |  |
| 5 | date\_order | Ngày đặt hàng | date |  |
| 6 | data\_order | Thông tin đơn đặt hàng | longtext |  |
| 7 | total\_bill | Tổng đơn hàng | int |  |
| 8 | id\_user | Tài khoản | int | Khóa ngoại |
| 9 | id\_status | Trạng thái đơn | int | Khóa ngoại |

1. **Bảng reviews**

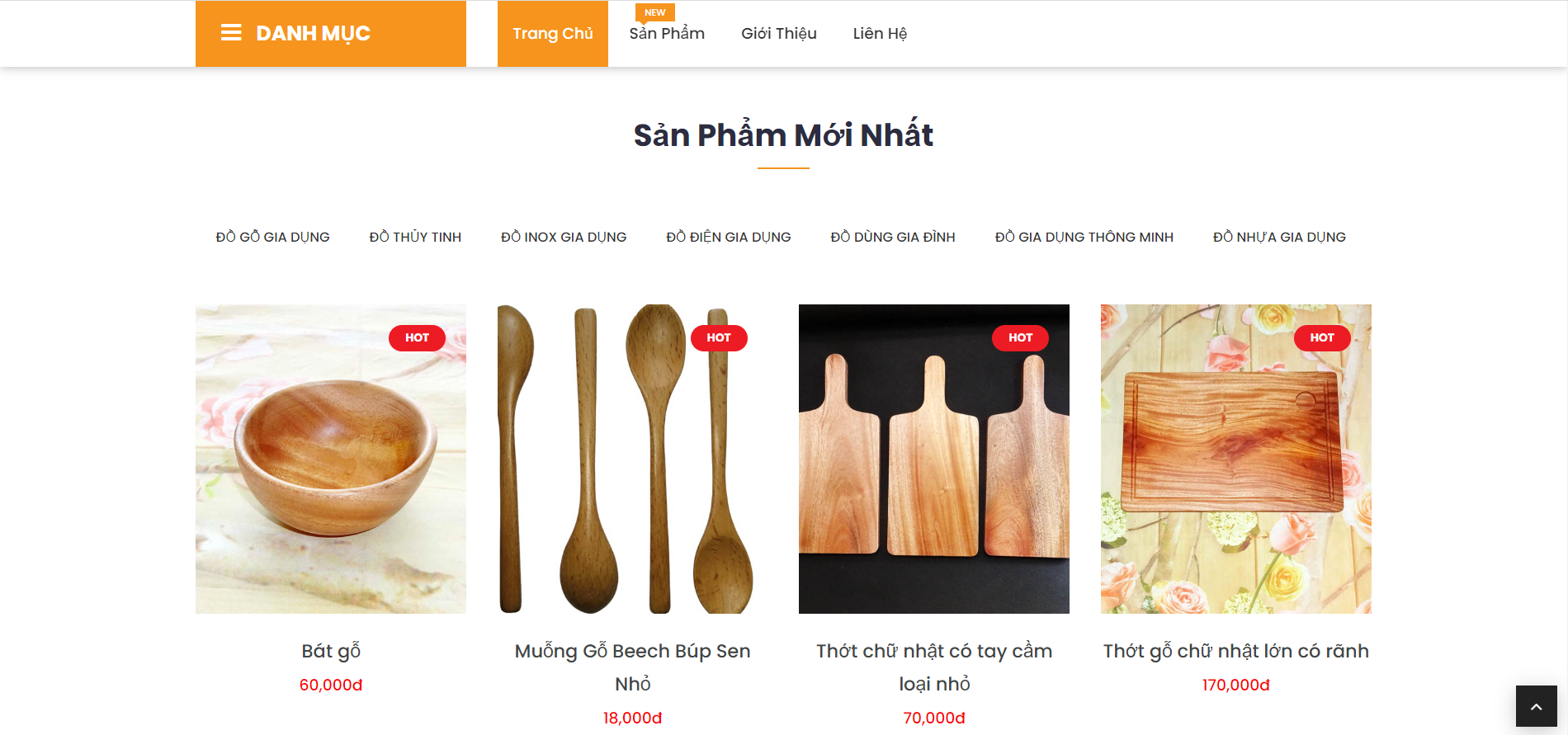
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên trường | Diễn giải | Kiểu dữ liệu | Khóa |
| 1 | id | Id đánh giá | int | Khóa chính |
| 2 | rate | Đánh giá sao | int |  |
| 3 | content\_reviews | Nội dung đánh giá | text |  |
| 4 | username | Tài khoản | text | Khóa ngoại |
| 5 | date\_review | Ngày đánh giá | date |  |
| 6 | id\_product | Sản phẩm đánh giá | int | Khóa ngoại |

**Chương III. Giao diện demo chương trình**

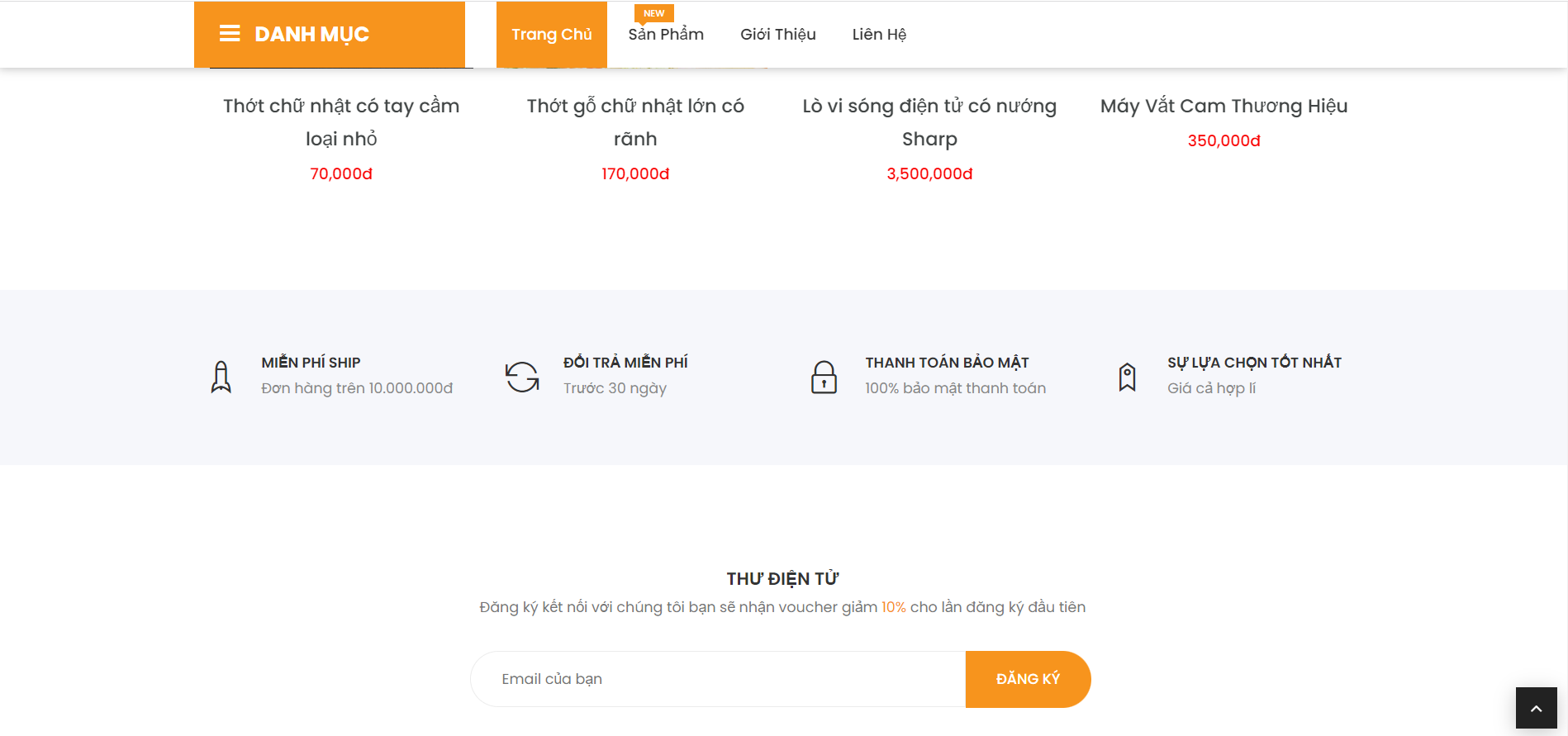
**3.1. Một số giao diện trang chủ**

****

Hình 3. Giao diện trang chủ

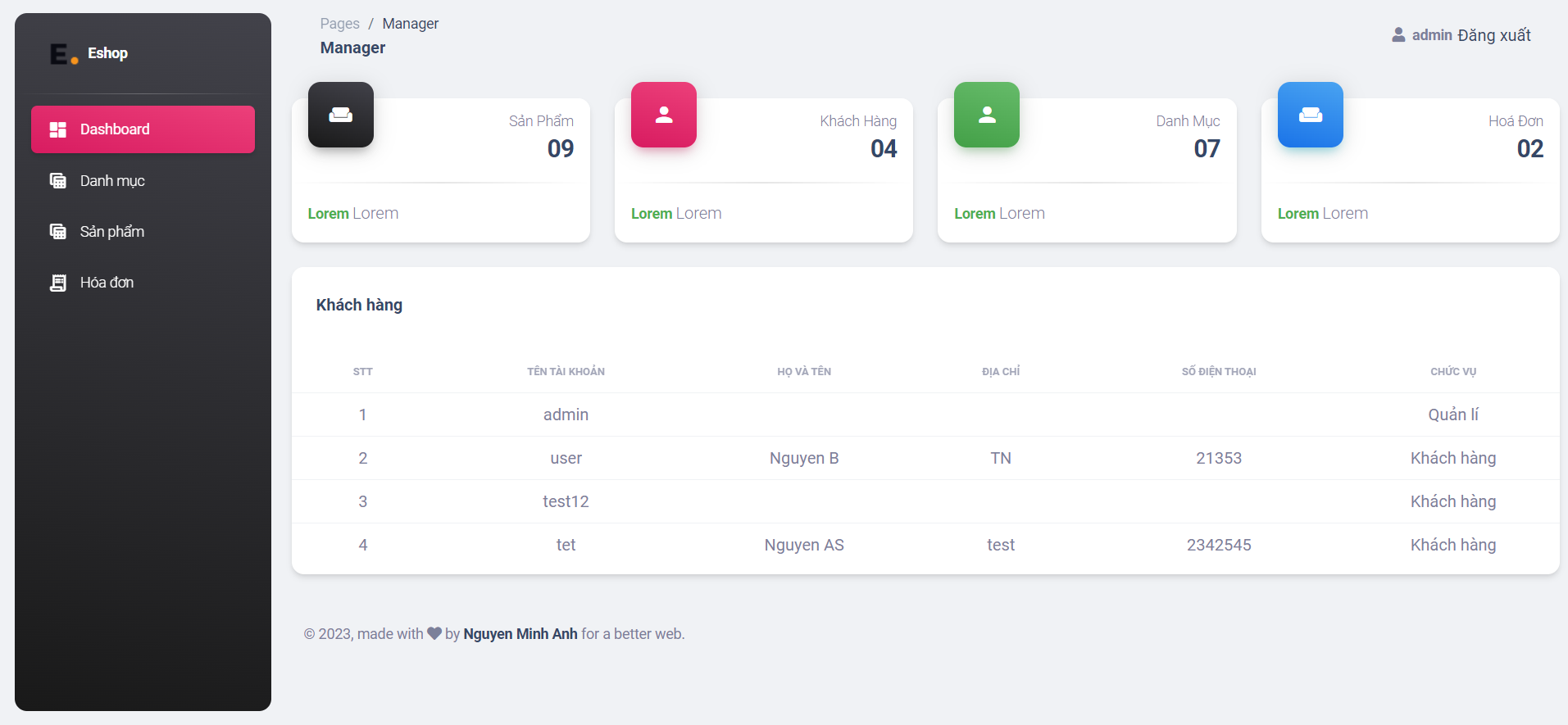


Hình 3. Giao diện trang chủ

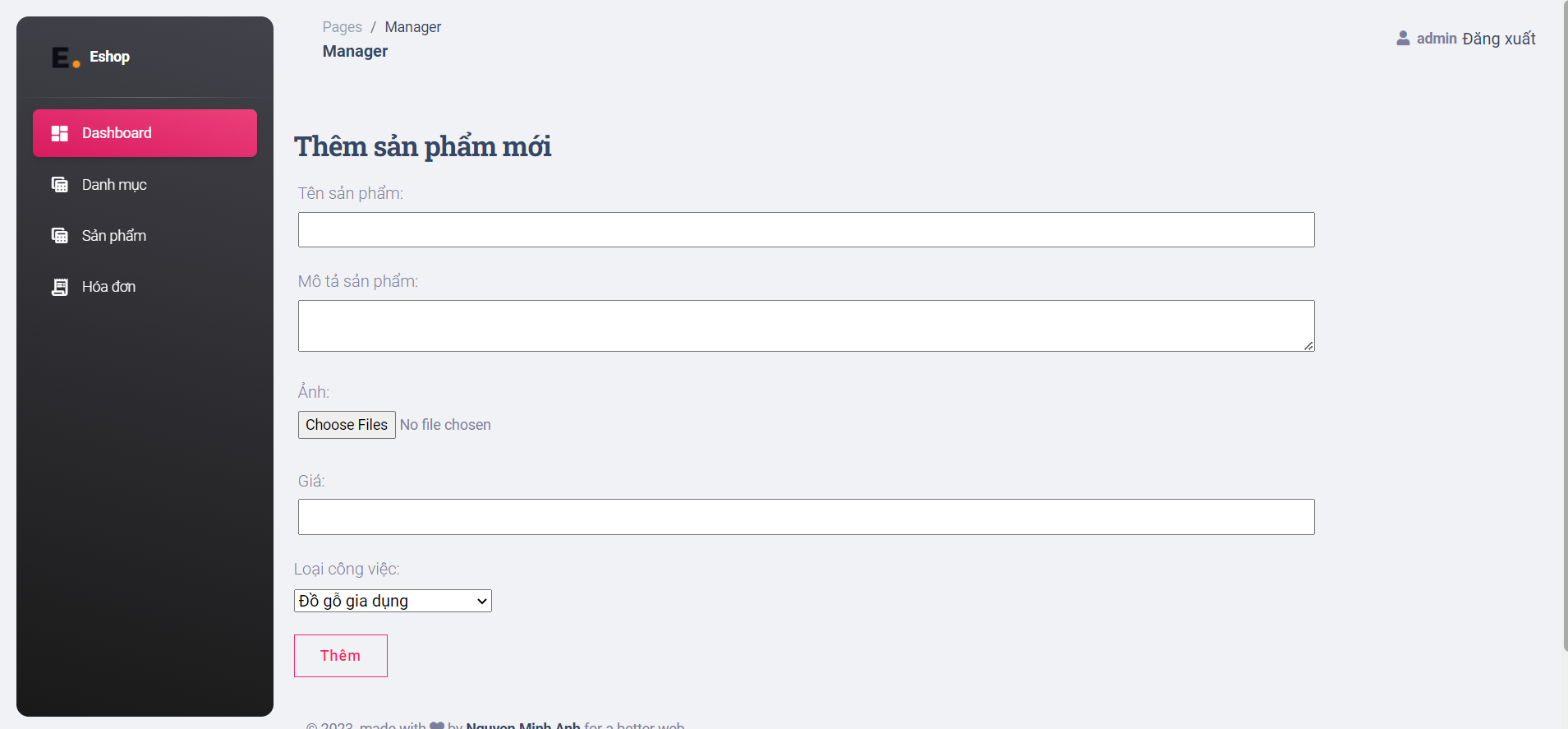


Hình 3. Giao diện trang chủ

**3.2. Chức năng thêm sản phẩm mới**

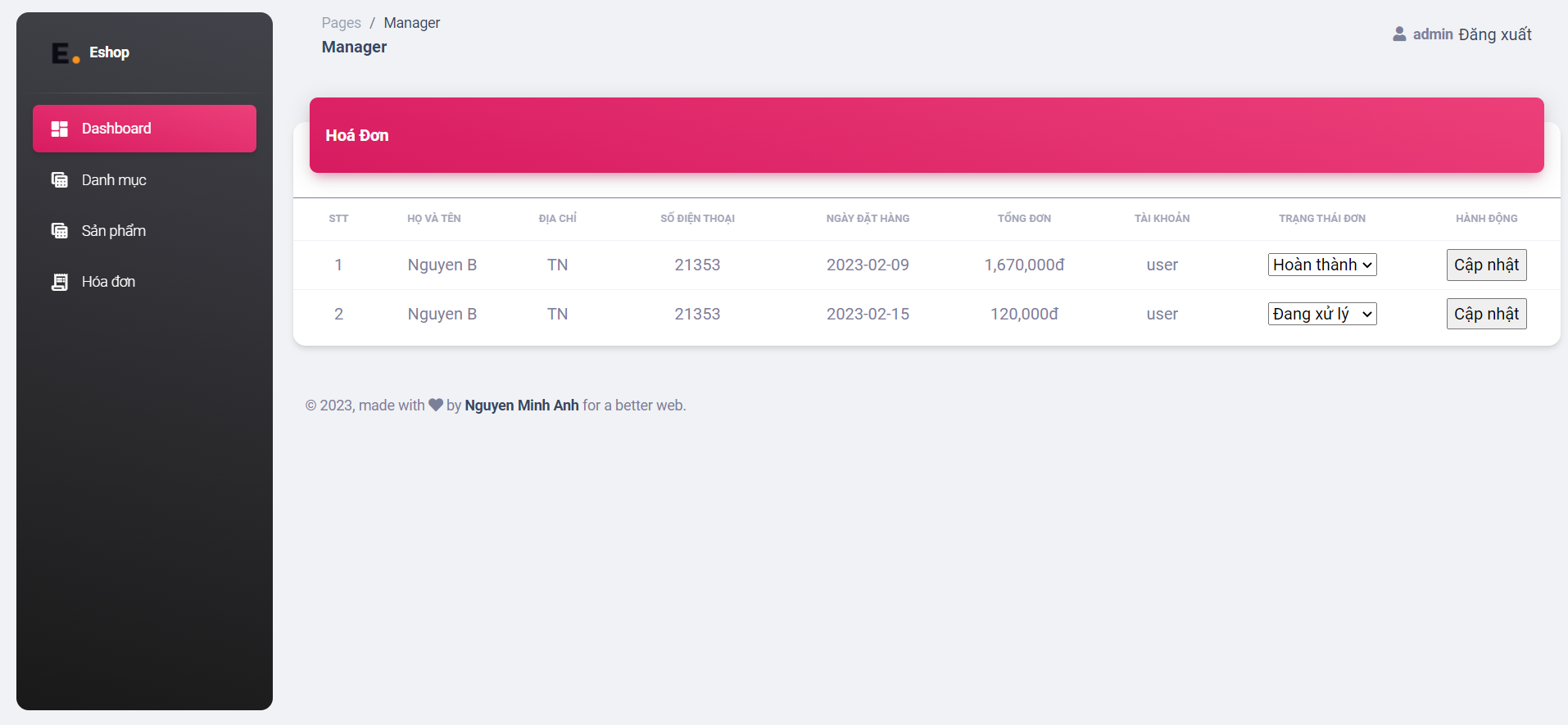
****

Hình 3. Giao diện quản lý của quản trị viên



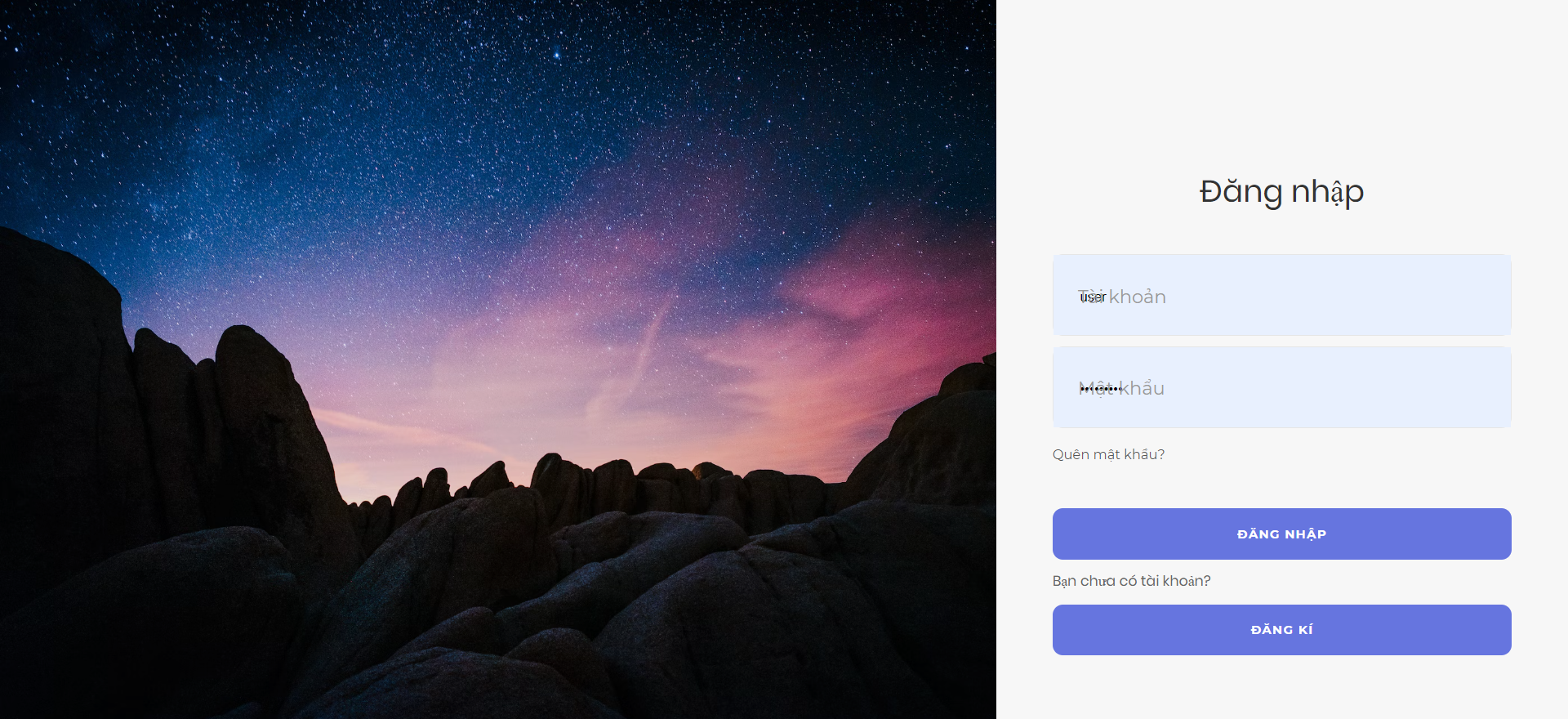
Hình 3. Giao diện thêm sản phẩm

**3.3. Chức năng quản lý hóa đơn**

****

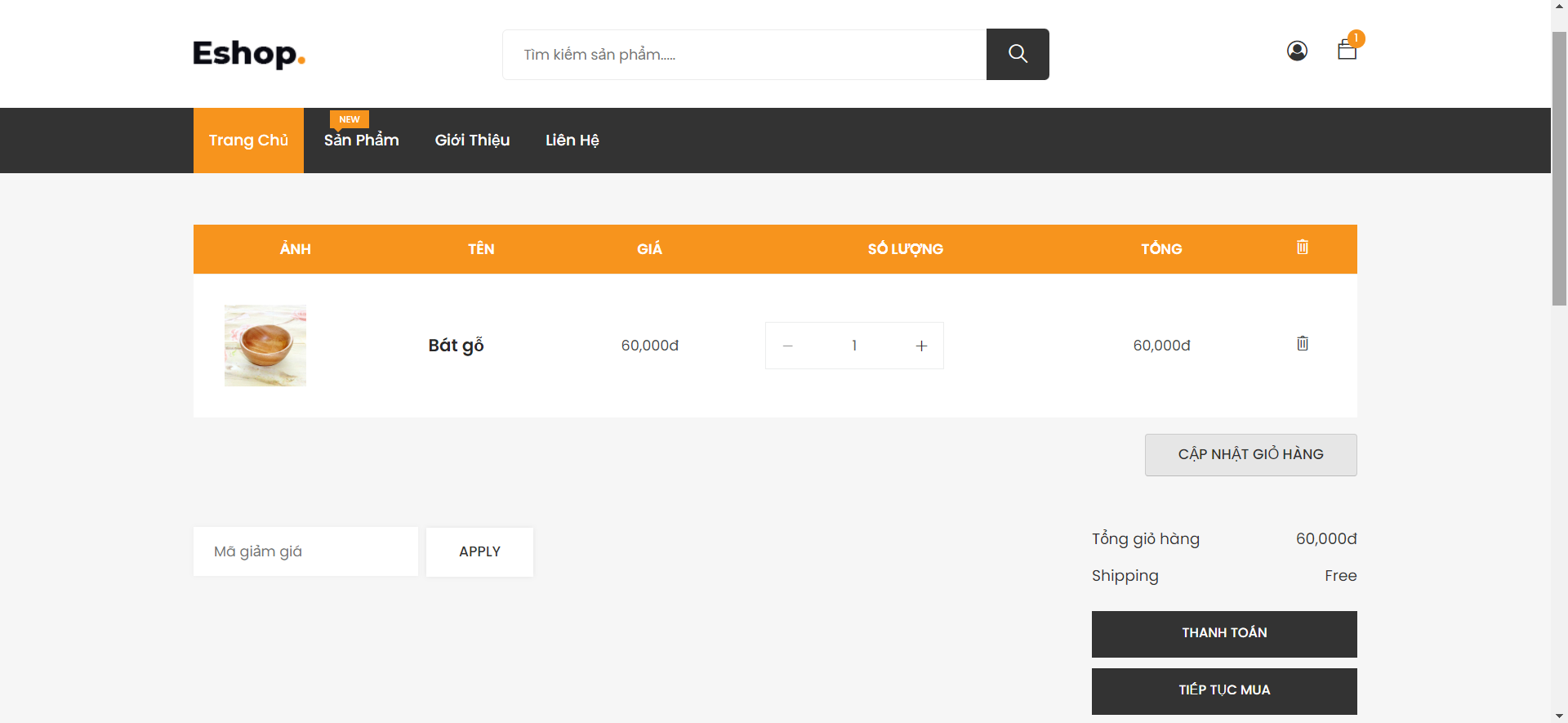
Hình 3. Giao diện quản lý hóa đơn

**3.4. Chức năng đăng nhập**



Hình 3. Giao diện chức năng đăng nhập

**3.5. Giao diện giỏ hàng**



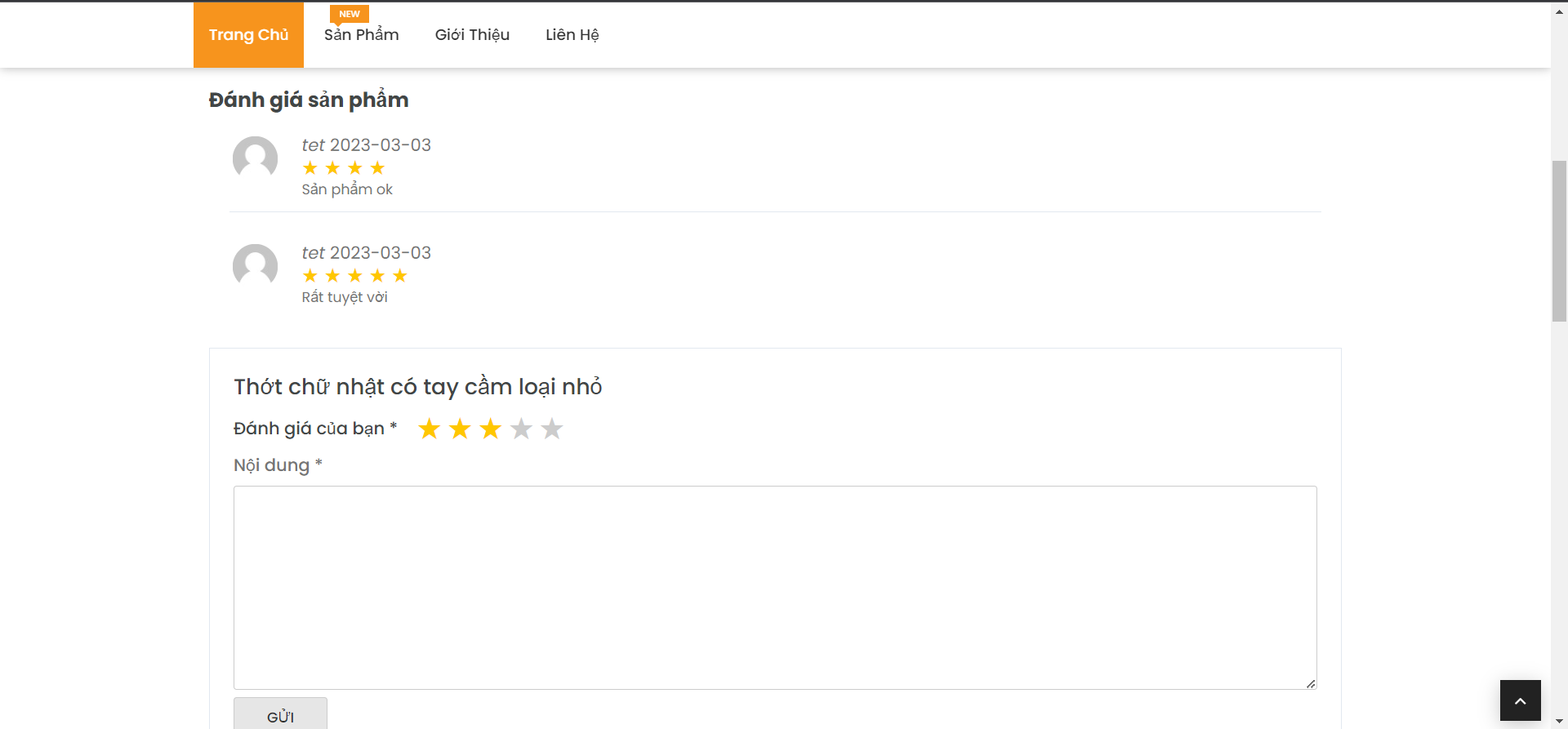
Hình 3. Giao diện giỏ hàng

**3.6. Giao diện thanh toán**

****

Hình 3. Giao diện thanh toán

**3.7. Giao diện đánh giá sản phẩm**

****

Hình 3. Giao diện đánh giá sản phẩm

Kết luận

**Kết quả đạt được**

Sau khi tự mình phân tích và thiết kế một website, em có những thu hoạch sau:

* Tự cải thiện năng lực sử dụng ngôn ngữ PHP và mô hình MVC để tạo ra website.
* Nắm bắt sơ bộ về cách thức phân tích và thiết kế website cũng như phát triển một website.
* Tăng cường khả năng làm việc độc lập do đề tài thực tập không thuộc hoạt động nhóm.
* Xây dựng và phát triển được website bán đồ gia dụng có đầy đủ các chức năng cơ bản đối với khách hàng

**Ưu và nhược điểm của website**

* Ưu điểm:
* Khách hàng có thể thay đổi mật khẩu và mật khẩu thanh toán trong thông tin cá nhân. Khách hàng có thể xem thông tin đơn hàng của mình.
* Cho phép khách hàng đánh giá sản phẩm.
* Xây dựng trên mô hình MVC dễ dàng quản lý việc thay đổi cấu trúc sau này.
* Hoàn thành hầu hết các chức năng cơ bản của một trang website bán hàng.
* Hiệu quả chi phí: Vì đang trong quá trình phát triển và thử nghiệm nên một website vẫn chạy trên localhost rẻ hơn nhiều so với lưu trữ nó trên một máy chủ website.
* Website chạy nhanh, dễ dàng upload, test và debug.
* Cho phép admin và người dùng dễ dàng quản lý thông tin, đối tượng.
* Nhược điểm:
* Dữ liệu chưa được đầy đủ.
* Giao diện khá cứng, không thể thay đổi template cho giao diện.
* Vì đang trong quá trình phát triển và thử nghiệm nên một website vẫn chạy trên localhost nên:
* Trang web không thể truy cập công khai trừ khi có địa chỉ IP tĩnh, điều này gây khó khăn cho việc chia sẻ với những người dùng khác.
* Vấn đề bảo mật.
* Bị hạn chế về tài nguyên, hiệu suất

**Hướng phát triển của đề tài**

Với những thiếu sót như phân tích trên, để website ngày càng hoàn thiện, phục vụ tốt cho nhu cầu của người dùng thì nên có những cập nhật sau đây:

* Thiết kế thêm vị trí để cập nhật giá cả thị trường.
* Nâng cao khả năng bảo mật cho chương trình.
* Ngăn ngừa tối đa các trường hợp đột nhập dữ liệu hệ thống.
* Chỉnh sữa dữ liệu cho chính xác với thực tế và sẽ nâng cấp dần website để đáp ứng được những nhu cầu của khách hàng trong thời đại mới.
* Thiết kế thanh toán online cho khách hàng có thể thanh toán bằng nhiều cách khác nhau
* Thiết kế trang web đẹp hơn, tiện dụng hơn, có nhiều chức năng và sinh động hơn. Xây dựng thêm một diễn đàn trên trang web để khách hàng có thể trao đổi bình luận về sản phẩm.
* Một trong những hướng đi gần nhất và khả quan nhất là phát triển website bán đồ gia dụng trở thành website với việc cải tiến cơ sở dữ liệu cũng như phát triển thêm các tính năng cho người dùng và admin quản lý sử dụng.

**Tổng kết**

Một lần nữa, em xin chân thành cảm ơn Th.s Trần Quang Quý, trong thời gian qua đã giúp đỡ, chỉ bảo để em có thể hoàn thành để tài một cách tốt nhất. Tuy nhiên, do kiến thức còn hạn chế, tầm hiểu biết hạn hẹp nên bài làm còn nhiều thiếu xót, em mong thầy cô cảm thông ạ!

*Em xin chân thành cảm ơn!*

**Tài liệu tham khảo**

1. <https://codeignter.com>/

1. <https://www.w3school.com>/

1. <https://topdev.vn/>

1. <https://business-letters.com/>
2. <https://freetuts.net/hoc-html-css>
3. <https://monamedia.co/>
4. <https://hocphp.net/>
5. <https://freetuts.net/hoc-php>
6. <https://hocwebchuan.com/tutorial/php/>
7. <https://tuhocict.com/huong-dan-tu-hoc-lap-trinh-php/>
8. <https://vietnix.vn/>
9. https://topdev.vn/blog/

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

*Thái Nguyên, ngày …. tháng …. năm …….*

**GIÁO VIÊN HƯỚNG DẪN**