BỘ CÔNG THƯƠNG

**TRƯỜNG ĐẠI HỌC KINH TẾ KỸ THUẬT CÔNG NGHIỆP**

**KHOA: CÔNG NGHỆ THÔNG TIN**

BÁO CÁO THỰC TẬP LẬP TRÌNH WEB

**Đề tài: Lập một trang Web bán nồi cơm điện**

**Giáo viên hướng dẫn: Trần Hồng Việt**

**Nhóm thực hiện :** NHÓM 5

**Lớp** **:** ĐH TIN 11A4 HN

**HÀ NỘI – 2020**

BỘ CÔNG THƯƠNG

**TRƯỜNG ĐẠI HỌC KINH TẾ KỸ THUẬT CÔNG NGHIỆP**

**KHOA: CÔNG NGHỆ THÔNG TIN**

BÁO CÁO THỰC TẬP LẬP TRÌNH WEB

**Đề tài: Lập một trang Web bán nồi cơm điện**

**Giáo viên hướng dẫn: Trần Hồng Việt**

**Sinh viên nhóm** **:** NHÓM 5

**Sinh viên :** Nguyễn Thị Thanh Thuỷ

**HÀ NỘI – 202**

**Mục lục**

[Link github: https://github.com/thanhthuy1999/doanweb 6](#_Toc57757910)

[Link hosting: http://thanhthuytin11a4.atwebpages.com/ 6](#_Toc57757911)

[CHƯƠNG 1. INTERNET, WORLD WIDE WEB VÀ HTML 6](#_Toc57757912)

[1.1. Khái niệm cơ bản về Internet 6](#_Toc57757913)

[1.1.1. Giới thiệu chung 6](#_Toc57757914)

[1.1.2. Phân loại 6](#_Toc57757915)

[1.2. World Wide Web 7](#_Toc57757916)

[1.2.1. Các khái niệm cơ bản về World Wide Web 7](#_Toc57757917)

[1.2.2. Cách tạo trang Web 7](#_Toc57757918)

[1.2.3. Trình duyệt Web (Web Client hay Web Browser) 7](#_Toc57757919)

[1.2.4. Webserver 8](#_Toc57757920)

[1.2.5. Phân loại Web 8](#_Toc57757921)

[1.2.5.1. Web tĩnh 8](#_Toc57757922)

[1.2.5.2. Website động 9](#_Toc57757923)

[1.3. HTML 9](#_Toc57757924)

[1.3.1. Cấu trúc chung của một trang HTML 9](#_Toc57757925)

[1.3.2. Các thẻ HTML cơ bản 10](#_Toc57757926)

[CHƯƠNG 2. TÌM HIỂU NGÔN NGỮ PHP 12](#_Toc57757927)

[2.1. Khái niệm PHP 12](#_Toc57757928)

[2.2. Hoạt động của PHP: 12](#_Toc57757929)

[2.3. Khai báo và sử dụng Session, Cookie 13](#_Toc57757930)

[CHƯƠNG 3. CƠ SỞ DỮ LIỆU MYSQL 17](#_Toc57757931)

[3.1. Giới thiệu cơ sở dữ liệu 17](#_Toc57757932)

[3.2. Mục đích sử dụng cơ sở dữ liệu 17](#_Toc57757933)

[3.3. Các kiểu dữ liệu trong cơ sở dữ liệu MySQL 18](#_Toc57757934)

[3.3.1. Loại dữ liệu numeric 18](#_Toc57757935)

[3.3.2. Loại dữ liệu kiểu Date and Time 19](#_Toc57757936)

[3.3.3. Loại dữ liệu String 20](#_Toc57757937)

[3.4. Các thao tác cập nhật dữ liệu 21](#_Toc57757938)

[3.5. Các hàm thông dụng trong MySQL 22](#_Toc57757939)

[3.5.1. Các hàm trong phát biểu GROUP BY 22](#_Toc57757940)

[3.5.2. Các hàm xử lí chuỗi: 22](#_Toc57757941)

[3.5.3. Các hàm xử lí về thời gian 22](#_Toc57757942)

[3.5.4. Các hàm về toán học 22](#_Toc57757943)

[CHƯƠNG 4 : THIẾT KẾ CƠ SỞ DỮ LIỆU 23](#_Toc57757944)

[4.1. Mô hình thực thể liên kết. 23](#_Toc57757945)

[4.2. Các bảng cơ sở dữ liệu. 24](#_Toc57757946)

[4.2.1. Nhà sản xuất 24](#_Toc57757947)

[4.2.2. Danh mục 24](#_Toc57757948)

[4.2.3. Sản phẩm 24](#_Toc57757949)

[4.2.4. Hóa đơn 25](#_Toc57757950)

[4.2.5. Chi tiết hóa đơn 26](#_Toc57757951)

[4.2.6. Người dùng 26](#_Toc57757952)

[CHƯƠNG 5 : THIẾT KẾ GIAO DIỆN 27](#_Toc57757953)

[5.1. Giao diện trang chủ. 27](#_Toc57757954)

[5.1.1. Giao diện trang chủ 27](#_Toc57757955)

[5.1.2. Giao diện tin tức 28](#_Toc57757956)

[5.1.3. Giao diện liên hệ 28](#_Toc57757957)

[5.1.4. Giao diện đăng nhập Admin 29](#_Toc57757958)

[5.2. Giao diện quản trị. 29](#_Toc57757959)

[5.2.1. Giao diện trang chủ quản trị 29](#_Toc57757960)

[5.2.2. Giao diện quản lý sản phẩm 30](#_Toc57757961)

[5.2.3. Giao diện quản lý hóa đơn. 30](#_Toc57757962)

[5.2.4. Giao diện quản lý người dùng. 31](#_Toc57757963)

[KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 32](#_Toc57757964)

[1. Những công việc làm được 32](#_Toc57757965)

[2. Hạn chế 32](#_Toc57757966)

[3. Kết luận 32](#_Toc57757967)

[PHỤ LỤC 33](#_Toc57757968)

[1. Code trang “Chi tiết sản phẩm”. 33](#_Toc57757969)

[2. Code trang “Hiển thị sản phẩm theo danh mục sản phẩm”. 34](#_Toc57757970)

[3. Code trang “Đăng ký người dùng”. 34](#_Toc57757971)

[4. Code trang “Đăng nhập”. 35](#_Toc57757972)

[5. Code trang “Đăng ký”. 35](#_Toc57757973)

Link github: <https://github.com/thanhthuy1999/doanweb>

Link hosting: <http://thanhthuytin11a4.atwebpages.com/>

CHƯƠNG 1. INTERNET, WORLD WIDE WEB VÀ HTML

1.1. Khái niệm cơ bản về Internet

1.1.1. Giới thiệu chung

Internet là mạng của các máy tính trên toàn cầu được thành lập từ những năm 80 bắt nguồn từ mạng APARRNET của bộ quốc phòng Mỹ.

Vào mạng Internet, bạn có thể tìm thấy nhiều tài liệu để nghiên cứu, học tập, trao đổi thư từ, đặt hàng, mua bán... Một trong những mục tiêu của Internet là chia sẻ thông tin giữa những người dùng trên nhiều lĩnh vực khác nhau.

Mỗi máy tính trên Internet được gọi là một host. Mỗi host có khả năng như nhau về truyền thông tới máy khác trên Internet. Một số host được nối tới mạng bằng đường nối Dial-up (quay số) tạm thời, số khác được nối bằng đường nối mạng thật sự 100% (như Ethernet, Tokenring...).

1.1.2. Phân loại

Các máy tính trên mạng được chia làm 2 nhóm Client và Server.

* Client: Máy khách trên mạng, chứa các chương trình Client
* Server: Máy phục vụ-Máy chủ. Chứa các chương trình Server, tài nguyên (tập tin, tài liệu... ) dùng chung cho nhiều máy khách. Server luôn ở trạng thái chờ yêu cầu và đáp ứng yêu cầu của Client.
* Internet Server: Là những server cung cấp các dịch vụ Internet như Web Server, Mail Server, FPT Server…

Các dịch vụ thường dùng trên Internet

* Dịch vụ World Wide Web (gọi tắt là Web)
* Dịch vụ Electronic Mail (viết tắt là Email)
* Dịch vụ FPT

Để truyền thông với những máy tính khác, mọi máy tính trên Internet phải hỗ trợ giao thức chung TCP/IP (Transmission Control Protocol/Internet Protocol- là giao thức điều khiển truyền dữ liệu và giao thức Internet), là một giao thức đồng bộ, cho truyền thông điệp từ nhiều nguồn và tới nhiều đích khác nhau. Ví dụ có thể vừ lấy thư từ trong hộp thư, đồng thời vừa truy cập trang Web. TCP đảm bảo tính an toàn dữ liệu, IP là giao thức chi phối phương thức vận chuyển dữ liệu trên Internet.

1.2. World Wide Web

1.2.1. Các khái niệm cơ bản về World Wide Web

World Wide Web (WWW) hay còn gọi là Web là một dịch vụ phổ biến nhất hiện nay trên Internet, 85% các giao dịch trên Internet ước lượng thuộc về WWW. Ngày nay số Website trên thế giới đã đạt tới con số khổng lồ. WWW cho phép bạn truy xuất thông tin văn bản, hình ảnh, âm thanh, video trên toàn thế giới. Thông qua Website, các quý công ty có thể giảm thiểu tối đa chi phí in ấn và phân phát tài liệu cho khách hàng ở nhiều nơi.

1.2.2. Cách tạo trang Web

Có nhiều cách để tạo trang Web:

* Có thể tạo trang Web trên bất kì chương trình xử lí văn bản nào.
* Tạo Web bằng cách viết mã nguồn bởi một trình soạn thảo văn bản như: Notepad, WordPad…là những chương trình soạn thảo văn bản có sẵn trong Window.
* Thiết kế bằng cách dùng Web Wizard và công cụ của Word 97, Word 2000.
* Thiết kế Web bằng các phần mềm chuyên nghiệp: FrontPage, Dreamweaver, Nescape Editor… Phần mềm chuyên nghiệp như DreamWeaver sẽ giúp bạn thiết kế trang Web dễ dàng hơn, nhanh chóng hơn, phần lớn mã lệnh HTML sẽ có sẵn trong phần Code cho bạn.

Để xây dựng một ứng dụng Web hoàn chỉnh và có tính thương mại, bạn cần kết hợp cả Client Script (kịch bản trình khách) và Server Script (kịch bản trên trình chủ) với một loại cơ sở dữ liệu nào đó, chẳng hạn như: MS Access, SQL Server, MySQL, Oracle,...

Khi bạn muốn triển khai ứng dụng Web trên mạng, ngoài các điều kiện về cấu hình phần cứng, bạn cần có trình chủ Web thường gọi là Web Server.

1.2.3. Trình duyệt Web (Web Client hay Web Browser)

Trình duyệt Web là công cụ truy xuất dữ liệu trên mạng, là phần mềm giao diện trực tiếp với người sử dụng. Nhiệm vụ của Web Browser là nhận các yêu cầu của người dùng, gửi các yêu cầu đó qua mạng tới các Web Server và nhận các dữ liệu cần thiết từ Server để hiển thị lên màn hình. Để sử dụng dịch vụ WWW, Client cần có 1 chương trình duyệt Web, kết nối vào Internet thông qua một ISP. Các trình duyệt thông dụng hiện nay là: Microsoft Internet Explorer, Mozilla Firefox và Google Chrome… Trong đó Internet Explorer là một trình duyệt chuẩn cho phép trình bày nội dung do Web server cung cấp, cho phép đăng kí tới bất kì Website nào trên Internet, hỗ trợ trình bày trang Web.

1.2.4. Webserver

Webserver: là một máy tính được nối vào Internet và chạy các phần mềm được thiết kế. Webserver đóng vai trò một chương trình xử lí các nhiệm vụ xác định, như tìm trang thích hợp, xử lí tổ hợp dữ liệu, kiểm tra dữ liệu hợp lệ.... Webserver cũng là nơi lưu trữ cơ sở dữ liệu, là phần mềm đảm nhiệm vai trò Server cung cấp dịch vụ Web.

Webserver hỗ trợ các công nghệ khác nhau:

* IIS (Internet Information Service) : Hỗ trợ ASP, mở rộng hỗ trợ PHP
* Apache: Hỗ trợ PHP
* Tomcat: Hỗ trợ JSP (Java Servlet Page)

1.2.5. Phân loại Web

1.2.5.1. Web tĩnh

Tài liệu được phân phát rất đơn giản từ hệ thống file của Server

Định dạng các trang web tĩnh là các siêu liên kết, các trang định dạng Text, các hình ảnh đơn giản.

Ưu điểm: CSDL nhỏ nên việc phân phát dữ liệu có hiệu quả rõ ràng, Server có thể đáp ứng nhu cầu Client một cách nhanh chóng. Ta nên sử dụng Web tĩnh khi không thay đổi thông tin trên đó.

Nhược điểm: Không đáp ứng được yêu cầu phức tạp của người sử dụng, không linh hoat.

Hoạt động của trang Web tĩnh được thể hiện như sau:

Browser gửi yêu cầu

**B**rowser Server

Server gửi trả tài liệu

1.2.5.2. Website động

Về cơ bản nội dung của trang Web động như một trang Web tĩnh, ngoài ra nó còn có thể thao tác với CSDL để đáp ứng nhu cầu phức tập của một trang Web. Sau khi nhận được yêu cầu từ Web Client, chẳng hạn như một truy vấn từ một CSDL đặt trên Server, ứng dụng Internet Server sẽ truy vấn CSDL này, tạo một trang HTML chứa kết quả truy vấn rồi gửi trả cho người dùng.

Hoạt động của Web động:

Yêu cầu Kết nối

URL CGI CSDL

Form Dữ liệu trả về Dữ liệu trả về

1.3. HTML

Trang Web là sự kết hợp giữa văn bản và các thẻ HTML. HTML là chữ viết tắt của HyperText Markup Language được hội đồng World Wide Web Consortium (W3C) quy định. Một tập tin HTML chẳng qua là một tập tin bình thường, có đuôi .html hoặc .htm.

HTML giúp định dạng văn bản trong trang Web nhờ các thẻ. Hơn nữa, các thẻ html có thể liên kết từ hoặc một cụm từ với các tài liệu khác trên Internet. Đa số các thẻ HTML có dạng thẻ đóng mở. Thẻ đóng dùng chung từ lệnh giống như thẻ mở, nhưng thêm dấu xiên phải (/). Ngôn ngữ HTML qui định cú pháp không phân biệt chữ hoa chữ thường. Ví dụ, có thể khai báo **<html>** hoặc **<HTML>.**  Không có khoảng trắng trong định nghĩa thẻ.

1.3.1. Cấu trúc chung của một trang HTML

<html>

<head>

<title> Tiêu đề của trang Web </title>

</head>

<body>

<!-- Các thẻ Html và nội dung sẽ hiển thị -->

</body>

</html>

1.3.2. Các thẻ HTML cơ bản

**1. Thẻ <head>...</head>: T**ạo đầu mục trang

**2. Thẻ <title>...</title>:** Tạo tiêu đề trang trên thanh tiêu đề, đây là thẻ bắt buộc. Thẻ **title** cho phép bạn trình bày chuỗi trên thanh tựa đề của trang Web mỗi khi trang Web đó được duyệt trên trình duyệt Web.

**3. Thẻ <body>...</body>:** Tất cả các thông tin khai báo trong thẻ **<body>** đều có thể xuất hiện trên trang Web. Những thông tin này có thể nhìn thấy trên trang Web.

**4. Các thẻ định dạng khác. Thẻ <p>…</p>:**Tạo một đoạn mới. Thẻ **<font>... </font>:** Thay đổi phông chữ, kích cỡ và màu kí tự…

**5. Thẻ định dạng bảng <table>…</table>:** Đây là thẻ định dạng bảng trên trang Web. Sau khi khai báo thẻ này, bạn phải khai báo các thẻ hàng **<tr>** và thẻ cột **<td>** cùng với các thuộc tính của nó.

**6. Thẻ hình ảnh <img>:** Cho phép bạn chèn hình ảnh vào trang Web. Thẻ này thuộc loại thẻ không có thẻ đóng.

**7. Thẻ liên kết <a>... </a>:** Là loại thẻ dùng để liên kết giữa các trang Web hoặc liên kết đến địa chỉ Internet, Mail hay Intranet(URL) và địa chỉ trong tập tin trong mạng cục bộ (UNC).

**8. Các thẻ Input**: Thẻ Input cho phép người dùng nhập dữ liệu hay chỉ thị thực thi một hành động nào đó, thẻ Input bao gồm các loại thẻ như: text, password, submit, button, reset, checkbox, radio, image.

**9. Thẻ Textarea: < Textarea>.... < \Textarea>:** Thẻ Textarea cho phép người dùng nhập liệu với rất nhiều dòng. Với thẻ này bạn không thể giới hạn chiều dài lớn nhất trên trang Web.

**10. Thẻ Select**: Thẻ Select cho phép người dùng chọn phần tử trong tập phương thức đã được định nghĩa trước. Nếu thẻ Select cho phép người dùng chọn một phần tử trong danh sách phần tử thì thẻ Select sẽ giống như combobox. Nếu thẻ Select cho phép người dùng chọn nhiều phần tử cùng một lần trong danh sách phần tử, thẻ Select đó là dạng listbox.

**11. Thẻ Form:** Khi bạn muốn submit dữ liệu người dùng nhập từ trang Web phía Client lên phía Server, bạn có hai cách để làm điều nàu ứng với hai phương thức POST và GET trong thẻ form. Trong một trang Web có thể có nhiều thẻ Form khác nhau, nhưng các thẻ Form này không được lồng nhau, mỗi thẻ form sẽ được khai báo hành động (action) chỉ đến một trang khác.

# 

CHƯƠNG 2. TÌM HIỂU NGÔN NGỮ PHP

2.1. Khái niệm PHP

PHP là chữ viết tắt của “Personal Home Page” do Rasmus Lerdorf tạo ra năm 1994. Vì tính hữu dụng của nó và khả năng phát triển, PHP bắt đầu được sử dụng trong môi trường chuyên nghiệp và nó trở thành “PHP: Hypertext Preprocessor”

Thực chất PHP là ngôn ngữ kịch bản nhúng trong HTML, nói một cách đơn giản đó là một trang HTML có nhúng mã PHP, PHP có thể được đặt rải rác trong HTML.

PHP là một ngôn ngữ lập trình được kết nối chặt chẽ với máy chủ, là một công nghệ phía máy chủ (Server-Side) và không phụ thuộc vào môi trường (cross-platform). Đây là hai yếu tố rất quan trọng, thứ nhất khi nói công nghệ phía máy chủ tức là nói đến mọi thứ trong PHP đều xảy ra trên máy chủ, thứ hai, chính vì tính chất không phụ thuộc môi trường cho phép PHP chạy trên hầu hết trên các hệ điều hành như Windows, Unix và nhiều biến thể của nó... Đặc biệt các mã kịch bản PHP viết trên máy chủ này sẽ làm việc bình thường trên máy chủ khác mà không cần phải chỉnh sửa hoặc chỉnh sửa rất ít.

Khi một trang Web muốn được dùng ngôn ngữ PHP thì phải đáp ứng được tất cả các quá trình xử lý thông tin trong trang Web đó, sau đó đưa ra kết quả ngôn ngữ HTML.

Khác với ngôn ngữ lập trình, PHP được thiết kế để chỉ thực hiện điều gì đó sau khi một sự kiện xảy ra (ví dụ, khi người dùng gửi một biểu mẫu hoặc chuyển tới một URL).

2.2. Hoạt động của PHP:

Vì PHP là ngôn ngữ của máy chủ nên mã lệnh của PHP sẽ tập trung trên máy chủ để phục vụ các trang Web theo yêu cầu của người dùng thông qua trình duyệt.

Sơ đồ hoạt động:

Máy khách Yêu cầu URL Máy chủ

hàng HTML Web

HTML

PHP

Gọi mã kịch bản

Khi người dùng truy cập Website viết bằng PHP, máy chủ đọc mã lệnh PHP và xử lí chúng theo các hướng dẫn được mã hóa. Mã lệnh PHP yêu cầu máy chủ gửi một dữ liệu thích hợp (mã lệnh HTML) đến trình duyệt Web. Trình duyệt xem nó như là một trang HTML têu chuẩn. Như ta đã nói, PHP cũng chính là một trang HTML nhưng có nhúng mã PHP và có phần mở rộng là HTML. Phần mở của PHP được đặt trong thẻ mở <?php và thẻ đóng ?> .Khi trình duyệt truy cập vào một trang PHP, Server sẽ đọc nội dung file PHP lên và lọc ra các đoạn mã PHP và thực thi các đoạn mã đó, lấy kết quả nhận được của đoạn mã PHP thay thế vào chỗ ban đầu của chúng trong file PHP, cuối cùng Server trả về kết quả cuối cùng là một trang nội dung HTML về cho trình duyệt.

2.3. Khai báo và sử dụng Session, Cookie

Trang web HTML thông thường sẽ không thể truyền dữ liệu từ trang này sang trang khác. Nói 1 cách khác là: tất cả thông tin trở thành quên lãng khi một trang web mới được tải. Điều này gây khó khăn cho một số công việc như là mua hàng (shopping carts) những thứ cần thiết dữ liệu (sản phẩm người dùng đã chọn) ghi nhớ từ trang này sang trang khác.

* ***Khái niệm điều khiển Session***

PHP session giải quyết vấn đề này bằng cách cho phép lưu trữ dữ liệu của người dùng trên server để sử dụng về sau (như username, món hàng ...). Tuy nhiên những thông tin session này chỉ là tạm thời và thường bị xoá đi ngay khi người dùng rời khỏi trang web đã dùng session.

Chính vì tính tạm thời này mà nếu ứng dụng của bạn cần thiết lưu trữ dữ liệu 1 cách lâu dài, bạn hãy dùng các cách lưu trữ khác như là csdl Mysql.

Session hoạt động bằng cách tạo 1 chuỗi unique (UID) cho từng vistore và chứa thông tin dựa trên ID đó. Việc này sẽ giúp tránh tình trạng dữ liệu bị xung đột giữa các user

**Lưu ý**: Nếu bạn vẫn còn ít kinh nghiệm về việc dùng session trong ứng dụng thì không nên dùng nó trên các website đòi hỏi bảo mật cao vì rất dễ gây ra các lỗ hổng bảo mật nguy hiểm.

* ***Bắt đầu với PHP Session***

Việc đầu tiên trước khi bạn làm bất cứ việc gì với session là bạn phải chạy nó trước, và nó được đặt ngay trên đầu trong code của bạn, trước khi HTML được xuất ra.

Dưới đây là 1 ví dụ đơn giản về việc tạo session trong PHP

PHP Code:

<?php   
session\_start(); // start up your PHP session!    
?>

Đoạn mã ngắn trên sẽ đăng kí cho người dùng 1 session ở trên Server, cho phép bạn lưu thông tin của người dùng và đưa nó vào UID cho session của user đó.

* ***Lưu giá trị của session***

Khi bạn muốn lưu trữ 1 thông tin nào đó ở session, được dùng như 1 mảng kết hợp. đó là nơi bạn lưu và lấy dữ liệu ra. Sau đây là 1 ví dụ đơn giản cho việc đơn giản này.

PHP Code:

<?php   
session\_start();    
$\_SESSION['views'] = 1; // lưu views   
echo "Pageviews = ". $\_SESSION['views']; //lấy views và hiện thị   
?>

Output:  
Pageviews = 1

Trong ví dụ này, chúng ta đã học được cách lưu trữ của biến session thông qua mảng kết hợp $\_SESSION và cũng như cách lấy nó ra.

* ***Sử dụng SESSION***

Bây giờ chúng ta đã biết cách lưu và lấy dữ liệu từ biến $\_SESSION, mọi chuyện thật dễ dàng phải không, và tiếp theo ta sẽ tìm hiểu 1 vài hàm liên quan đến session.

Khi bạn tạo 1 biến và lưu nó vào 1 session, bạn có lẽ muốn dùng nó sau này, tuy nhiên, trước khi bạn dùng biến session đó, bạn nên kiểm tra nó đã được khởi tạo hay chưa.

Thao tác này được thực hiện thông qua hàm isset, isset là 1 hàm kiểm tra bất kì biến nào và nó đã được khởi tạo và gán giá trị hay chưa.

Qua nhưng ví dụ trên, chúng ta có thể tạo 1 bộ đếm đơn giản cho 1 trang bằng cách sử dùng isset để kiểm tra nó đã được tạo hay chưa và gán giá trị cho nó.

Ví dụ:

PHP Code:

<?php   
session\_start();     
if(isset($\_SESSION['views']))   
    $\_SESSION['views'] = $\_SESSION['views']+ 1;   
else   
    $\_SESSION['views'] = 1;   
  
echo "views = ". $\_SESSION['views'];    
?>

Trong lần chạy đầu tiền của trang câu lệnh If sẽ trả về false vì chưa có biến session [views] nào được tạo cả. Tuy nhiên, khi bạn refresh lại trang đó, thì câu lệnh if sẽ trả về giá trị true và biến đếm counter sẽ tăng lên 1 đơn vị, và sẽ tăng cho mỗi lần chạy của trang đó lên 1.

* ***Xóa và Hủy Session***

Mặt dù dữ liệu trong session chỉ mang tính chất tạm thời và nó không yêu cầu phải xóa sau khi sử dụng, nhưng có thể trong trường hợp nào đó bạn phải xóa dữ liệu của nó cho mục đích của bạn.

Hãy tưởng tượng bạn đang điều hành 1 doanh nghiệp online, và 1 thanh viên đang dùng website của bạn mua 1 món hàng. Thành viên đó đã hoàn tất việc mua hàng (phiên giao dịch) trên website, do đó , bạn phải xóa mọi thứ trong session sau khi việc này hoàn tất.

PHP Code:

<?php   
session\_start();     
if(isset($\_SESSION['cart']))   
    unset($\_SESSION['cart']);    
?>

Bạn cũng có thể hủy hoàn toàn các session bằng cách gọi hàng session\_destroy

PHP Code:

<?php   
session\_start();    
session\_destroy();   
?>

Destroy sẽ reset session của bạn, do đó không nên gọi hàm này trước khi bạn còn muốn thao tác lên dữ liệu chứa trong session đó.

* ***Khái niệm Cookie****:*

Cookie là một mẩu nhỏ thông tin được lưu trữ xuống từng máy tính truy cập đến Web Site của chúng ta. Có thể gán cookie trên máy tính của người dùng thông qua HTTP Header, bằng cách khai báo như sau:

Set – Cookie : Name = Value; [expires=Date;]

[path = Path;] [domain = Domain\_name;] [secure]

Khai báo trên tạo ra cookie với tên là Name, giá trị là Value, các tham số khác là tham số tuỳ chọn. Expires là thời gian giới hạn mà cookie này không thích hợp nữa. Nếu thời gian không cung cấp trong cookie, cookie này sẽ còn hiệu lực cho đến khi người dùng xoá tập tin Cookie.

* ***Gán Cookie từ PHP***

Có thể gán Cookie trong PHP bằng cách sử dụng hàm setcookie(), hàm setcookie() có cú pháp như sau:

Int setcookie (string name [,string value [, int expire[, string path [,string domain [ ,int secure ]]]]])

CHƯƠNG 3. CƠ SỞ DỮ LIỆU MYSQL

3.1. Giới thiệu cơ sở dữ liệu

MySQL là ứng dụng cơ sở dữ liệu mã nguồn mở phổ biến nhất hiện nay (theo www. mysql. com) và được sử dụng phối hợp với PHP. Trước khi làm việc với MySQL cần xác định các nhu cầu cho ứng dụng.

MySQL là cơ sở dữ có trình giao diện trên Windows hay Linux, cho phép người sử dụng có thể thao tác các hành động liên quan đến nó**.** Việc tìm hiểu từng công nghệ trước khi bắt tay vào việc viết mã kịch bản PHP, việc tích hợp hai công nghệ PHP và MySQL là một công việc cần thiết và rất quan trọng.

3.2. Mục đích sử dụng cơ sở dữ liệu

Mục đích sử dụng cơ sở dữ liệu bao gồm các chức năng như: lưu trữ (storage), truy cập (accessibility), tổ chức (organization) và xử lí (manipulation).

* Lưu trữ: Lưu trữ trên đĩa và có thể chuyển đổi dữ liệu từ cơ sở dữ liệu này sang cơ sở dữ liệu khác, nếu bạn sử dụng cho quy mô nhỏ, bạn có thể chọn cơ sở dữ liệu nhỏ như: Microsoft Exel, Microsoft Access, MySQL, Microsoft Visual FoxPro… Nếu ứng dụng có quy mô lớn, bạn có thể chọn cơ sở dữ liệu có quy mô lớn như: Oracle, SQL Server…
* Truy cập: Truy cập dữ liệu phụ thuộc vào mục đích và yêu cầu của người sử dụng, ở mức độ mang tính cục bộ, truy cập cơ sỏ dữ liệu ngay trong cơ sở dữ liệu với nhau, nhằm trao đổi hay xử lí dữ liệu ngay bên trong chính nó, nhưng do mục đích và yêu cầu người dùng vượt ra ngoài cơ sở dữ liệu, nên bạn cần có các phương thức truy cập dữ liệu giữa các cơ sở dử liệu với nhau như: Microsoft Access với SQL Server, hay SQL Server và cơ sở dữ liệu Oracle....
* Tổ chức: Tổ chức cơ sở dữ liệu phụ thuộc vào mô hình cơ sở dữ liệu, phân tích và thiết kế cơ sở dữ liệu tức là tổ chức cơ sở dữ liệu phụ thuộc vào đặc điểm riêng của từng ứng dụng. Tuy nhiên khi tổ chức cơ sở dữ liệu cần phải tuân theo một số tiêu chuẩn của hệ thống cơ sở dữ liệu nnhằm tăng tính tối ưu khi truy cập và xử lí.
* Xử lí: Tùy vào nhu cầu tính toán và truy vấn cơ sở dữ liệu với các mục đích khác nhau, cần phải sử dụng các phát biểu truy vấn cùng các phép toán, phát biểu của cơ sở dữ liệu để xuất ra kết quả như yêu cầu. Để thao tác hay xử lí dữ liệu bên trong chính cơ sở dữ liệu ta sử dụng các ngôn ngữ lập trình như: PHP, C++, Java, Visual Basic…

3.3. Các kiểu dữ liệu trong cơ sở dữ liệu MySQL

3.3.1. Loại dữ liệu numeric

|  |  |  |  |
| --- | --- | --- | --- |
| *Kiểu dữ liệu số nguyên* | | | |
| Loại | Range | Bytes | Diễn giải |
| Tinyint | -127->128 hay 0.. 255 | 1 | Số nguyên rất nhỏ |
| Smallint | -32768->32767 hay 0.. 65535 | 2 | Số nguyên nhỏ |
| Mediumint | -8388608->838860 hay 0.. 16777215 | 3 | Số nguyên vừa |
| Int | -231->231-1 hay 0.. 232-1 | 4 | Số nguyên |
| Bigint | -263->263-1 hay 0.. 264-1 | 8 | Số nguyên lớn |

*Kiểu dữ liệu số chấm động*

|  |  |  |  |
| --- | --- | --- | --- |
| Loại | Range | Bytes | Diễn giải |
| Float | phụ thuộc số thập phân | 4 | Số thập phân dạng Single hay Double |
| Float(M, D) | ±1. 175494351E-38  ±3. 40282346638 | 2 | Số thập phân dạng Single |
| Double(M, D) | ±1. 7976931348623157308  ±2. 2250738585072014E-308 | 8 | Số thập phân dạng Double |
| Float(M[, D]) |  |  | Số chấm động lưu dưới dạng char |

3.3.2. Loại dữ liệu kiểu Date and Time

Kiểu dữ liệu Date and Time cho phép bạn nhập dữ liệu dưới dạng chuỗi ngày tháng hay dạng số.

*Dữ liệu kiểu số nguyên*

|  |  |  |
| --- | --- | --- |
| Loại | Range | Diễn giải |
| Date | 1000-01-01 | Date trình bày dưới dạng yyyy-mm-dd. |
| Time | 00:00:00 23:59:59 | Time trình bày dưới dạng hh:mm:ss. |
| DateTime | 1000-01-01  00:00:00 9999-12-31 23:59:59 | Date và Time trình bày dưới dạng yyyy-mm-dd hh:mm:ss. |
| TimeStamp[(M)] | 1970-01-01  00:00:00 | TimeStamp trình bày dưới dạng yyyy-mm-dd hh:mm:ss. |
| Year[(2|4)] | 1970-2069  1901-2155 | Year trình bày dưới dạng 2 số hay 4 số |

Trình bày đại diện của TimeStamp

Loại hiển thị

---------------------------------------------------------------

TimeStamp YYYYMMDDHHMMSS

TimeStamp (14) YYYYMMDDHHMMSS

TimeStamp (12) YYMMDDHHMMSS

TimeStamp (10) YYMMDDHHMM

TimeStamp (8) YYYYMMDD

TimeStamp (6) YYMMDD

TimeStamp (4) YYMM

TimeStamp (2) YY

-----------------------------------------------------------------

(Y=năm, M=tháng, D=ngày)

3.3.3. Loại dữ liệu String

Kiểu dữ liệu String chia làm 3 loại: loại thứ nhất như char (chiều dài cố định) và varchar (chiều dài biến thiên); loại thứ hai là Text hay Blob, Text cho phép lưu chuỗi rất lớn, Blob cho phép lưu đối tượng nhị phân; loại thứ ba là Enum và Set.

*Kiểu dữ liệu String*

|  |  |  |
| --- | --- | --- |
| Loại | Range | Diễn giải |
| Char | 1-255 characters | Chiều dài của chuỗi lớn nhất 255 ký tự. |
| Varchar | 1-255 characters | Chiều dài của chuỗi lớn nhất 255 ký tự. |
| Tinyblob | 28-1 | Khai báo cho Field chứa kiểu đối tượng nhị phân cỡ 255 characters |
| Tinytext | 28-1 | Khai báo cho Field chứa kiểu chuỗi cỡ 255 characters. |
| Blob | 216-1 | Khai báo cho Field chứa kiểu blob cỡ 65, 535 characters.. |
| Text | 216-1 | Khai báo cho Field chứa kiểu chuỗi dạng văn bản cỡ 65, 535 characters. |
| Mediumblob | 224-1 | Khai báo cho Field chứa kiểu blob vừa khoảng 16, 777, 215 characters |
| Mediumtext | 224-1 | Khai báo cho Field chứa kiểu chuỗi dạng văn bản vừa khoảng 16, 777, 215 characters |
| Longblob | 232-1 | Khai báo cho Field chứa kiểu blob lớn khoảng 4, 294, 967, 295 characters. |
| Longtext | 232-1 | Khai báo cho Field chứa kiểu chuỗi dạng văn bản lớn khoảng 4, 294, 967, 295 characters. |

## 

3.4. Các thao tác cập nhật dữ liệu

* SELECT (Truy vấn mẫu tin):Select dùng để truy vấn từ một hay nhiều bảng khác nhau, kết quả trả về là một tập mẫu tin thỏa mãn các điều kiện cho trước nếu có, cú pháp của phát biểu SQL dạng SELECT như sau:

SELECT<danh sách các cột>

[FROM<danh sách bảng>]

[WHERE<các điều kiện ràng buộc>]

[GROUP BY<tên cột/biểu thức trong SELECT>]

[HAVING<điều kiện bắt buộc của GROUP BY>]

[ORDER BY<danh sách các cột>]

[LIMIT FromNumber |ToNumber]

* INSERT(Thêm mẫu tin):

Cú pháp: INSERT INTO Tên\_bảng VALUES(Bộ\_giá\_trị)

* UPDATE(Cập nhật dữ liệu):

Cú pháp: UPDATE TABLE Tên\_bảng

SET Tên\_cột=Biểu\_thức,...

[WHERE Điều\_kiện]

* DELETE(Xóa mẫu tin):

Cú pháp: DELETE FROM Tên\_bảng

[WHERE Điều\_kiện]

3.5. Các hàm thông dụng trong MySQL

3.5.1. Các hàm trong phát biểu GROUP BY

* Hàm AVG: Hàm trả về giá trị bình quân của cột hay trường trong câu truy vấn
* Hàm MIN: Hàm trả về giá trị nhỏ nhất của cột hay trường trong câu truy vấn
* Hàm MAX: Hàm trả về giá trị lớn nhất của cột hay trường trong câu truy vấn
* Hàm Count: Hàm trả về số lượng mẩu tin trong câu truy vấn
* Hàm Sum: Hàm trả về tổng các giá trị của trường, cột trong câu truy vấn.

3.5.2. Các hàm xử lí chuỗi:

* Hàm ASCII: Hàm trả về giá trị mã ASCII của kí tự bên trái của chuỗi.
* Hàm Char: Hàm này chuyển đổi kkiểu mã ASCII từ số nguyên sang dạng chuỗi.
* Hàm UPPER: Hàm này chuyển đổi chuỗi sang kiểu chữ hoa
* Hàm LOWER: Hàm này chuyển đổi chuỗi sang kiểu chữ thường.
* Hàm Len: Hàm này trả về chiều dài của chuỗi.
* Thủ tục LTRIM: Thủ tục loại bỏ khỏang trắng bên trái của chuỗi
* Thủ tục RTRIM: Thủ tục loại bỏ khỏang trắng bên phải của chuỗi
* Hàm Left(str, n): Hàm trả về chuỗi bên trái tính từ đầu cho đến vị trí n
* Hàm Right(str, n): Hàm trả về chuỗi bên phải tính từ đầu cho đến vị trí n
* Hàm Instrt: Hàm trả về chuỗi vị trí bắt đầu của chuỗi con trong chuỗi xét.

3.5.3. Các hàm xử lí về thời gian

* Hàm CurDate(): Hàm trả về ngày, tháng và năm hiện hành của hệ thống.
* Hàm CurTime(): Hàm trả về giờ, phút và giây hiện hành của hệ thống.
* Hàm Period\_Diff: Hàm trả về số ngày trong khoảng thời gian giữa 2 ngày.
* Hàm dayofmonth: Hàm trả về ngày thứ mấy trong tháng

3.5.4. Các hàm về toán học

* Hàm sqrt: Hàm trả về là căn bậc hai của một biểu thức.
* Hàm CurDate(): Hàm trả về ngày, tháng và năm hiện hành của hệ thống.

CHƯƠNG 4 : THIẾT KẾ CƠ SỞ DỮ LIỆU

4.1. Mô hình thực thể liên kết.

Danh mục

**MaDM**

TenDM

Dequi

Người dùng

**Idnd**

Tennd

Username

Password

Ngaysinh

Gioitinh

Email

Dienthoai

Diachi

Ngaydangky

Phanquyen

**Sản phẩm**

**IDSP**

MaDM

TenSP

Hinhanh

Mau

Chitiet

Soluong

Daban

Gia

Ngaycapnhat

Trạng thái

Chi tiết hóa đơn

**Mahd**

Idsp

Tensp

Soluong

Gia

phuongthucthanhtoan

Hóa đơn

**Mahd**

Idnd

Hoten

Diachi

Dienthoai

Email

Ngaydathang

Trangthai

Nhà sản xuất

**Idnsx**

Tennsx

Hinhanh

Diachi

Dienthoai

4.2. Các bảng cơ sở dữ liệu.

4.2.1. Nhà sản xuất

1. *Bảng 4.1. Bảng nhà sản xuất*

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** |
| Idnsx | int | Khóa chính |
| Tennsx | Varchar(50) |  |
| Hinhanh | Varchar(100) |  |
| Diachi | Varchar(50) |  |
| Dienthoai | Int |  |

4.2.2. Danh mục

1. *Bảng 4.2. Bảng danh mục*

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** |
| Madm | int | Khóa chính |
| Tendm | Varchar(50) |  |
| Dequi | Int |  |

4.2.3. Sản phẩm

1. *Bảng 4.3. Bảng sản phẩm*

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** |
| Idsp | int | Khóa chính |
| Madm | Varchar(10) |  |
| Idnsx | int |  |
| TenSP | Varchar(50) |  |
| Hinhanh | Varchar(100) |  |
| Mau | Varchar(20) |  |
| Chitiet | Text |  |
| soluong | Int |  |
| daban | int |  |
| gia | int |  |
| Ngaycapnhat | Date/Time |  |

4.2.4. Hóa đơn

1. *Bảng 4.5. Bảng hóa đơn*

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** |
| MaHD | int | Khóa chính |
| Idnd | int |  |
| Hoten | Varchar(50) |  |
| Diachi | Varchar(50) |  |
| Dienthoai | Int |  |
| Email | Varchar(50) |  |
| Ngaymuahang | Date/time |  |
| Trangthai | Int |  |

4.2.5. Chi tiết hóa đơn

1. *Bảng 4.6. Bảng chi tiết hóa đơn*

|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** |
| MaHD | int | Khóa chính |
| Idsp | int |  |
| Tensp | Varchar(50) |  |
| Soluong | Int |  |
| Gia | Int |  |
| Phuongthucthanhtoan | Int |  |

4.2.6. Người dùng

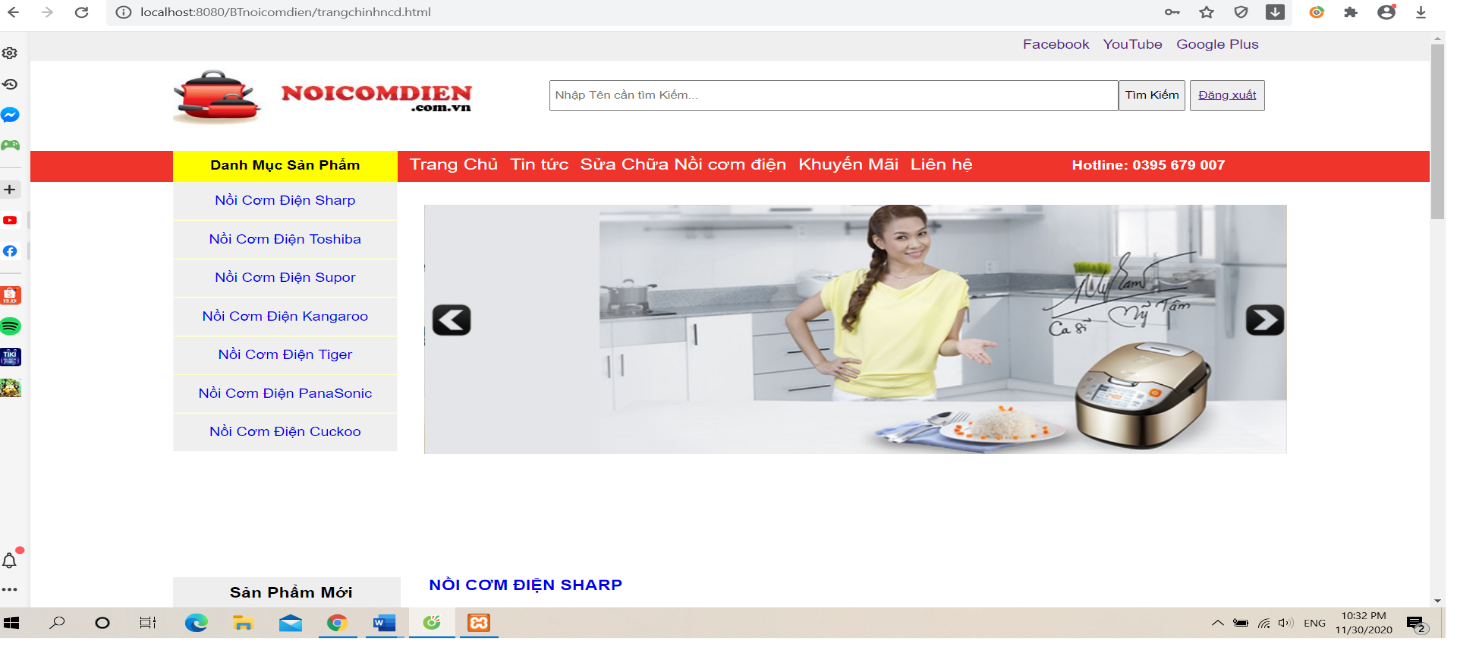
1. *Bảng 4.7. Bảng người dùng*

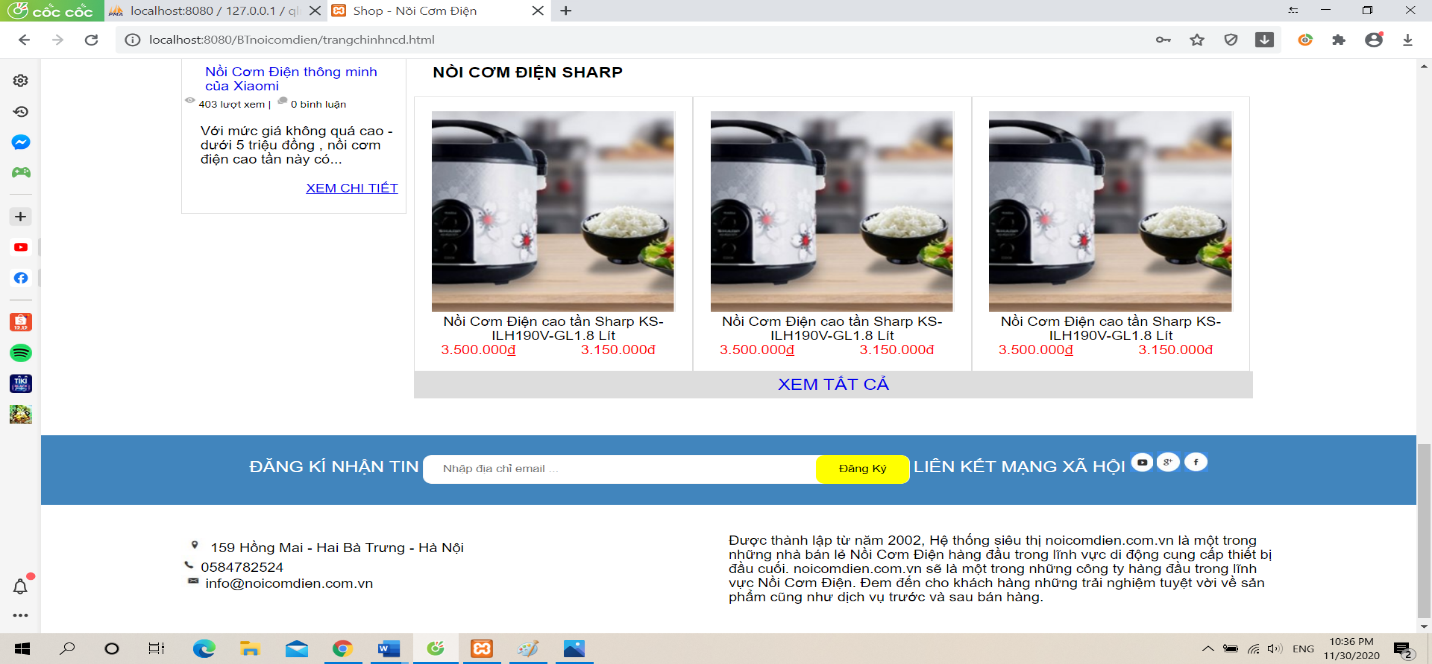
|  |  |  |
| --- | --- | --- |
| **Thuộc tính** | **Kiểu dữ liệu** | **Ràng buộc** |
| Idnd | Varchar(10) | Khóa chính |
| Tennd | Varchar(50) |  |
| Username | Varchar(30) |  |
| Password | Varchar(30) |  |
| Ngaysinh | Date/time |  |
| Gioitinh | Varchar(10) |  |
| Email | Varchar(30) |  |
| Dienthoai | Int |  |
| Diachi | Varchar(255) |  |
| Ngaydangky | Date/Time |  |
| Phanquyen | int |  |

CHƯƠNG 5 : THIẾT KẾ GIAO DIỆN

5.1. Giao diện trang chủ.

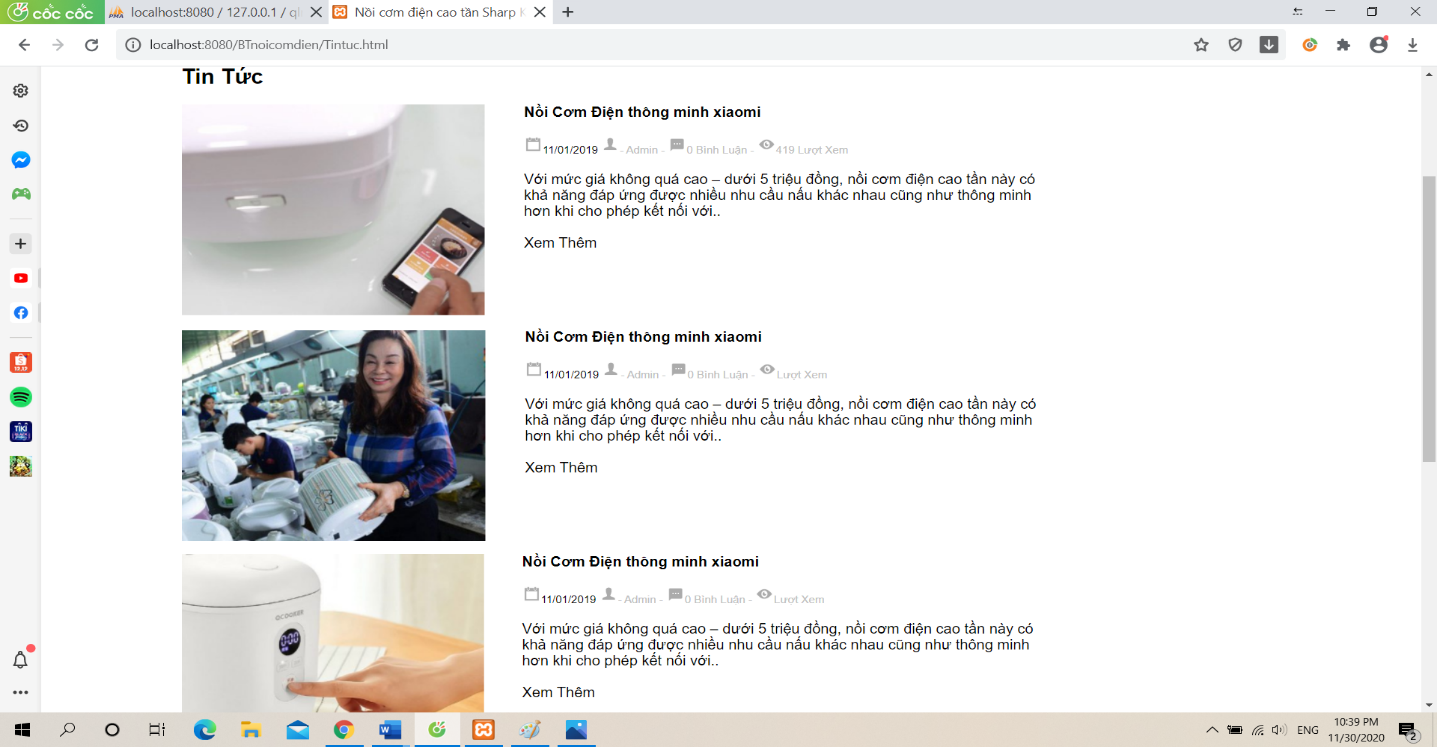
5.1.1. Giao diện trang chủ



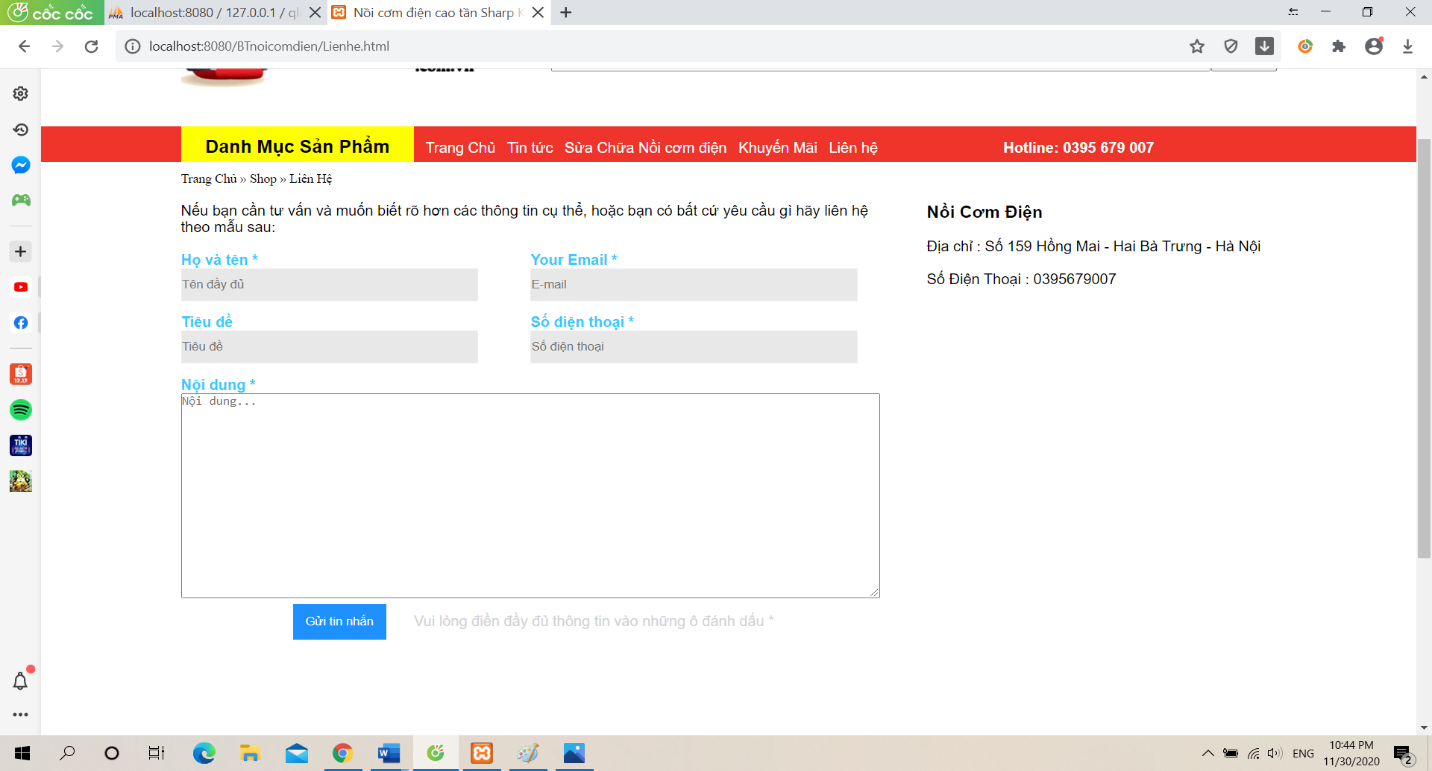


* Trang chủ là nơi tương tác giữa người dùng với website.
* Chứa các chức năng: đăng nhập, mua hàng, góp ý,…

5.1.2. Giao diện tin tức

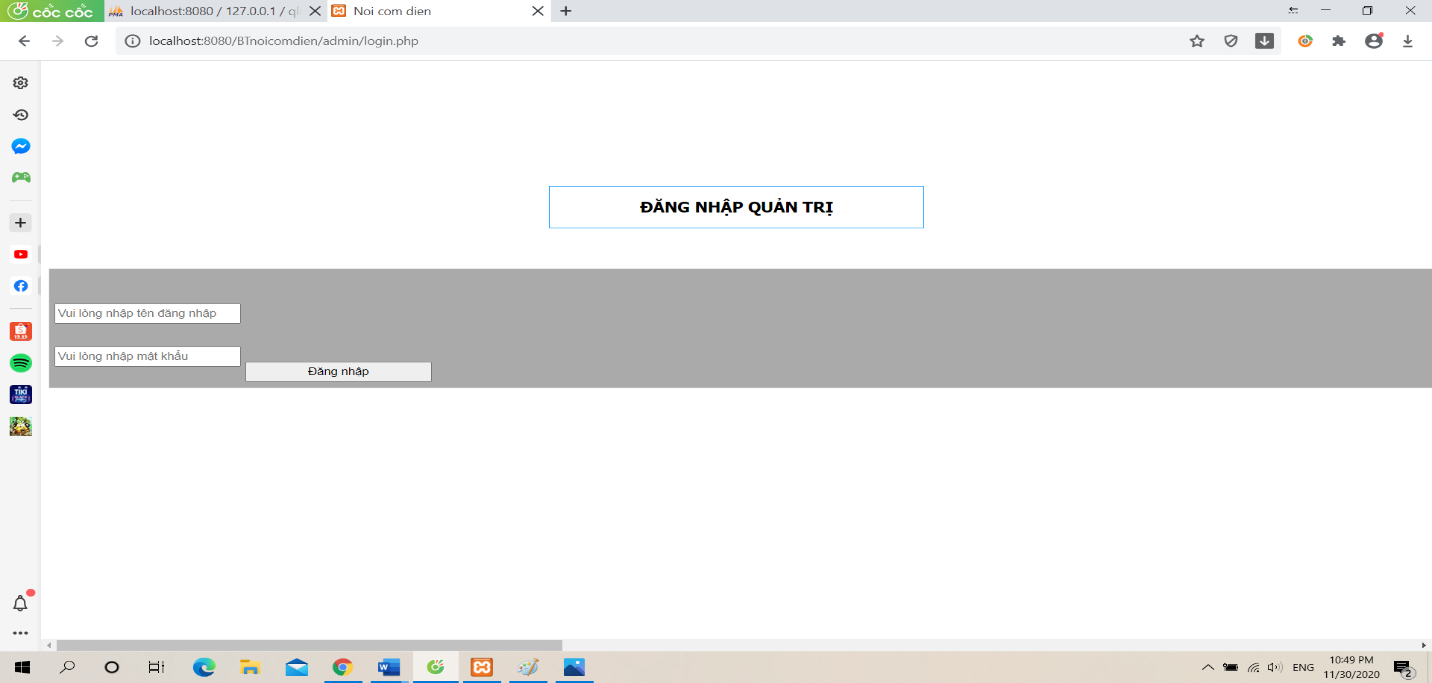


5.1.3. Giao diện liên hệ



* Người dùng muốn góp ý hoặc muốn liên hệ với quản trị viên sẽ vào giao diện góp ý để gửi.

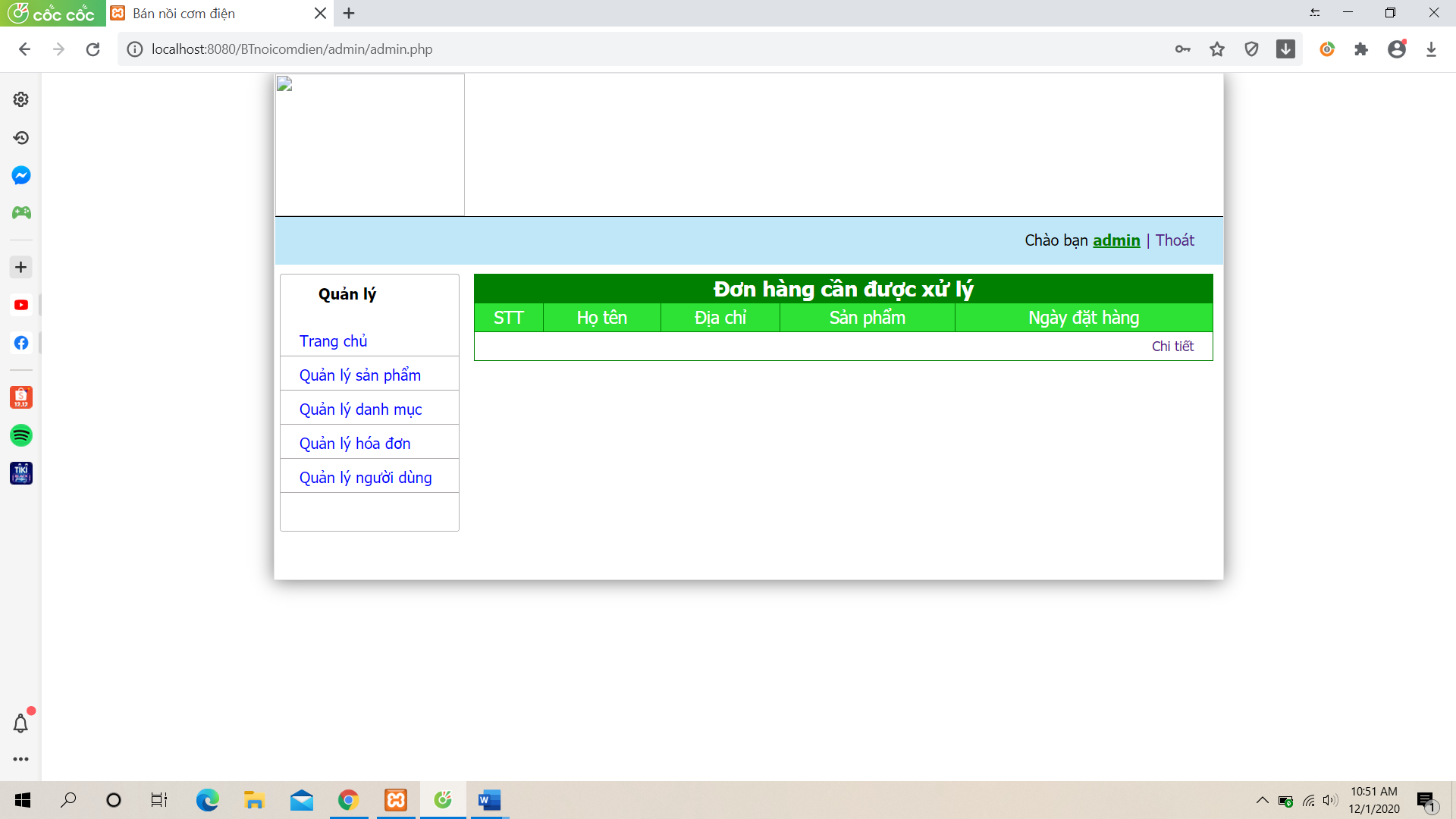
5.1.4. Giao diện đăng nhập Admin



## 

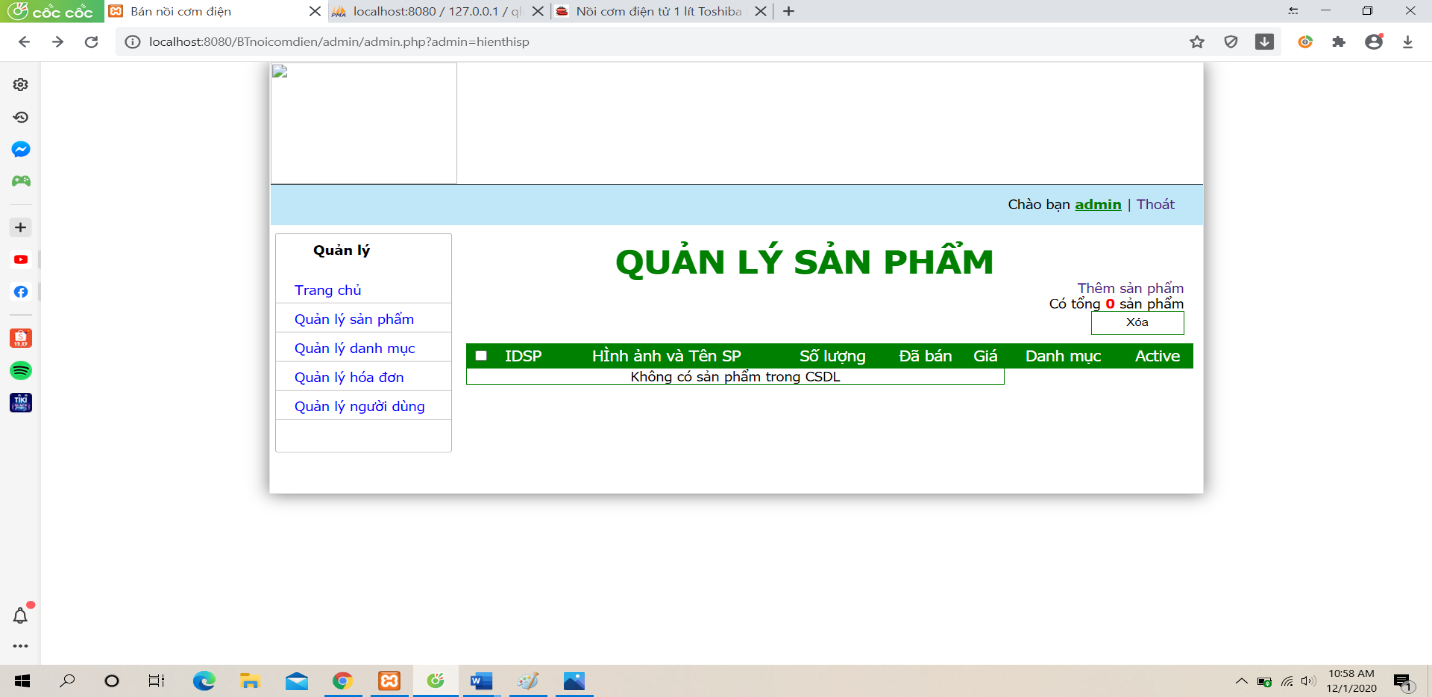
5.2. Giao diện quản trị.

5.2.1. Giao diện trang chủ quản trị



* Giao diện trang quản lý chứa các chức năng quản trị như: Quản lý sản phẩm, Quản lý danh mục, Quản lý hóa đơn,…..
* Hiển thị những đơn hàng chưa được xử lý.

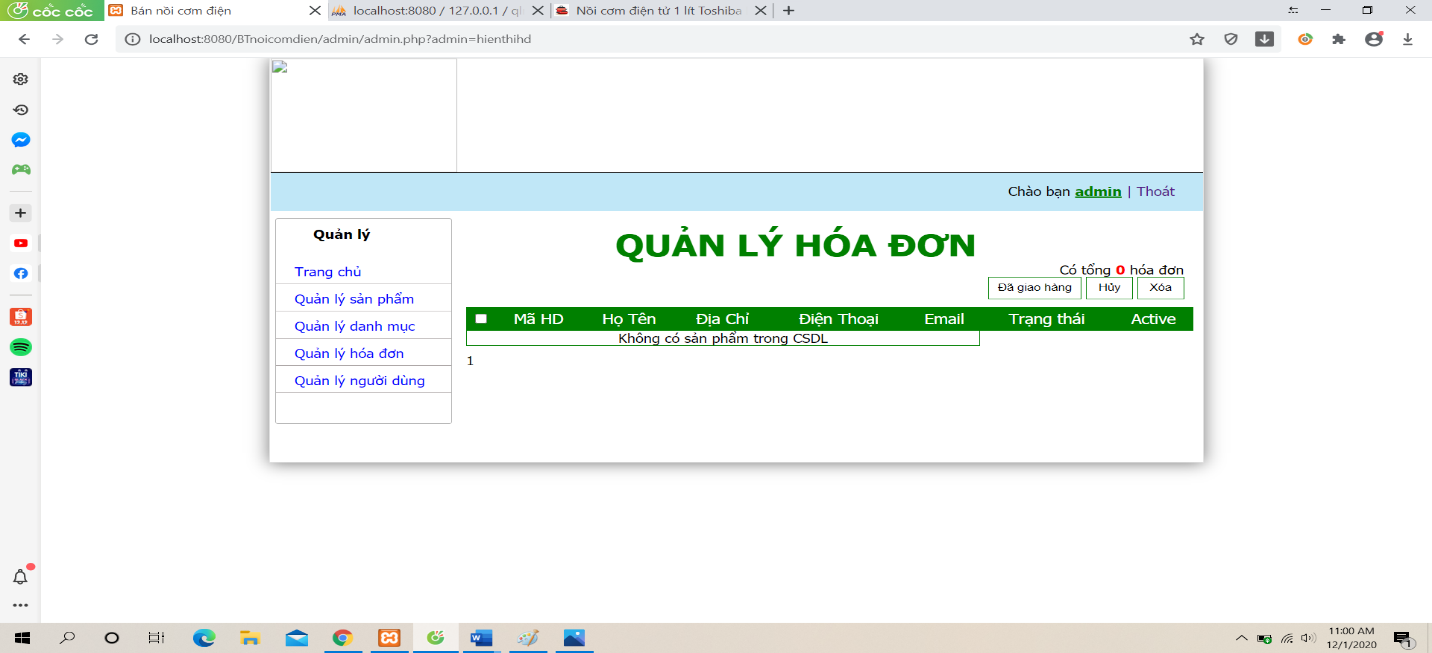
5.2.2. Giao diện quản lý sản phẩm



*Hình 4.8. Giao diện quản lý sản phẩm*

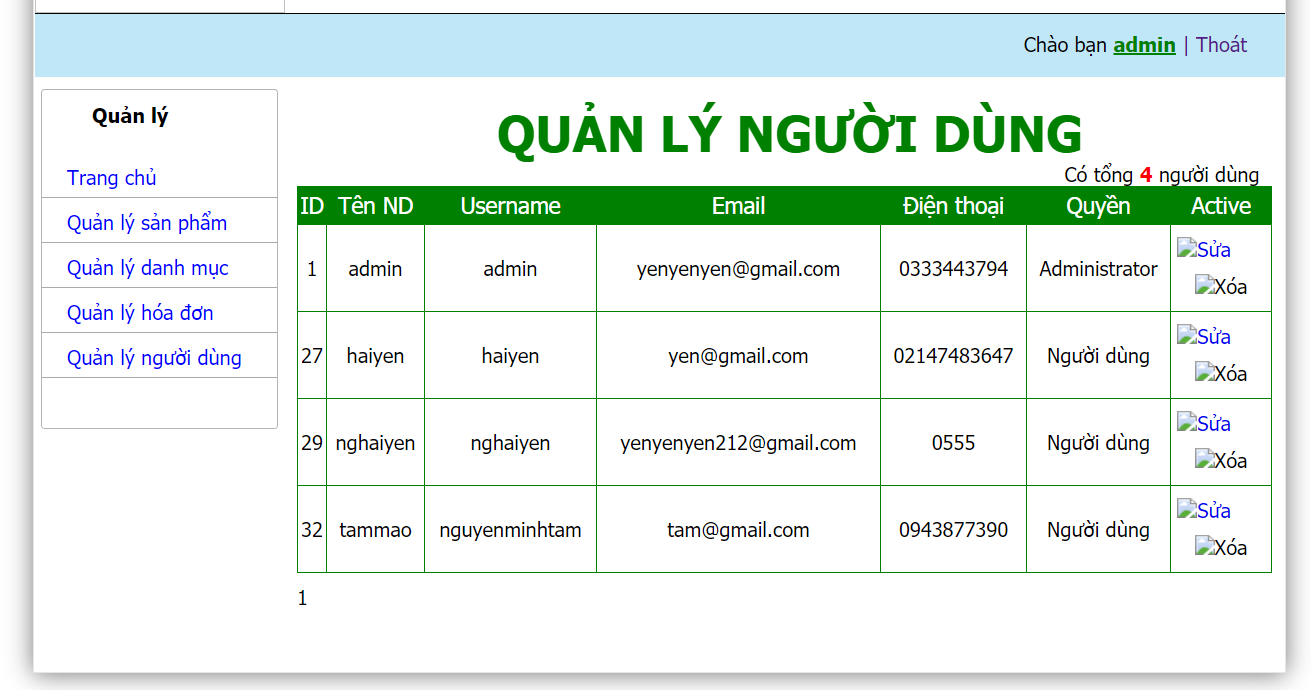
* Các thông tin sản phẩm sẽ được Admin quản lý và có thể thêm, sửa, xóa sản phẩm.

5.2.3. Giao diện quản lý hóa đơn.



Khi khách hàng gửi đơn hàng thì nó sẽ được hiển thị và Admin sẽ xử lý đơn hàng này

5.2.4. Giao diện quản lý người dùng.



* Trang quản lý người dùng sẽ cho người quản lý sửa, xóa và phân quyền cho người dùng.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

1. Những công việc làm được

* Đã phân tích được hiện trạng hệ thống để đưa ra được sơ đồ phân cấp các chức năng của hệ thống mới, đưa ra được các biểu đồ mức ngữ cảnh, mức đỉnh và mức dưới đỉnh để phục vụ cho quá trình xây dựng hệ thống mới của nhà hàng.
* Thiết kế được mô hình cơ sở dữ liệu cho website.
* Thiết kế giao diện website thân thiện, dễ sử dụng cho người dùng.
* Thiết kế được giao diện giỏ hàng.

2. Hạn chế

* Do kinh nghiệm thực tế chưa có nhiều do vậy quá trình phân tích hệ thống cho website của cửa hàng còn nhiều chỗ chưa đúng với thực tế hay chưa đảm bảo tính đúng đắn. Vấn đề này em xin phép hoàn thiện thêm trong quá trình phát triển hệ thống sau đó.
* Giao diện trang web còn chưa được đẹp mắt.
* Ngôn ngữ và phần mềm soạn thảo là còn mới mẻ nên còn nhiều chức năng, chưa vận dụng và kiểm soát được.

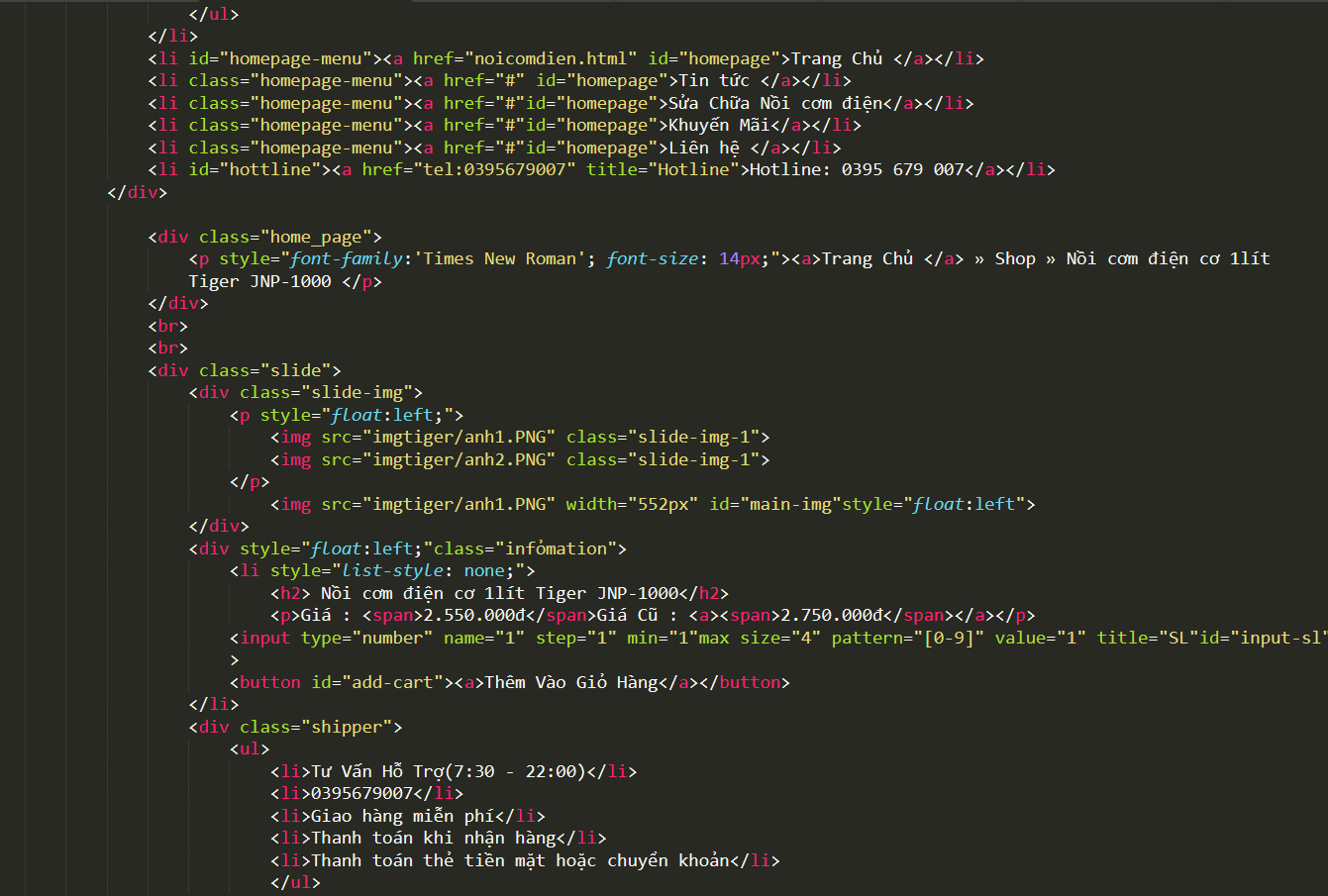
3. Kết luận

* Với sự nỗ lực của bản thân và sự giúp đỡ, chỉ bảo tận tình của **thầy Lê Chí Luận** hướng dẫn cuối cùng em cũng hoàn thành xong đề tài. Tuy vậy,Với những thuận lợi và khó khăn trong quá trình làm việc, bài làm về cơ bản đã hoàn thành nhưng không thể tránh khỏi sai sót.

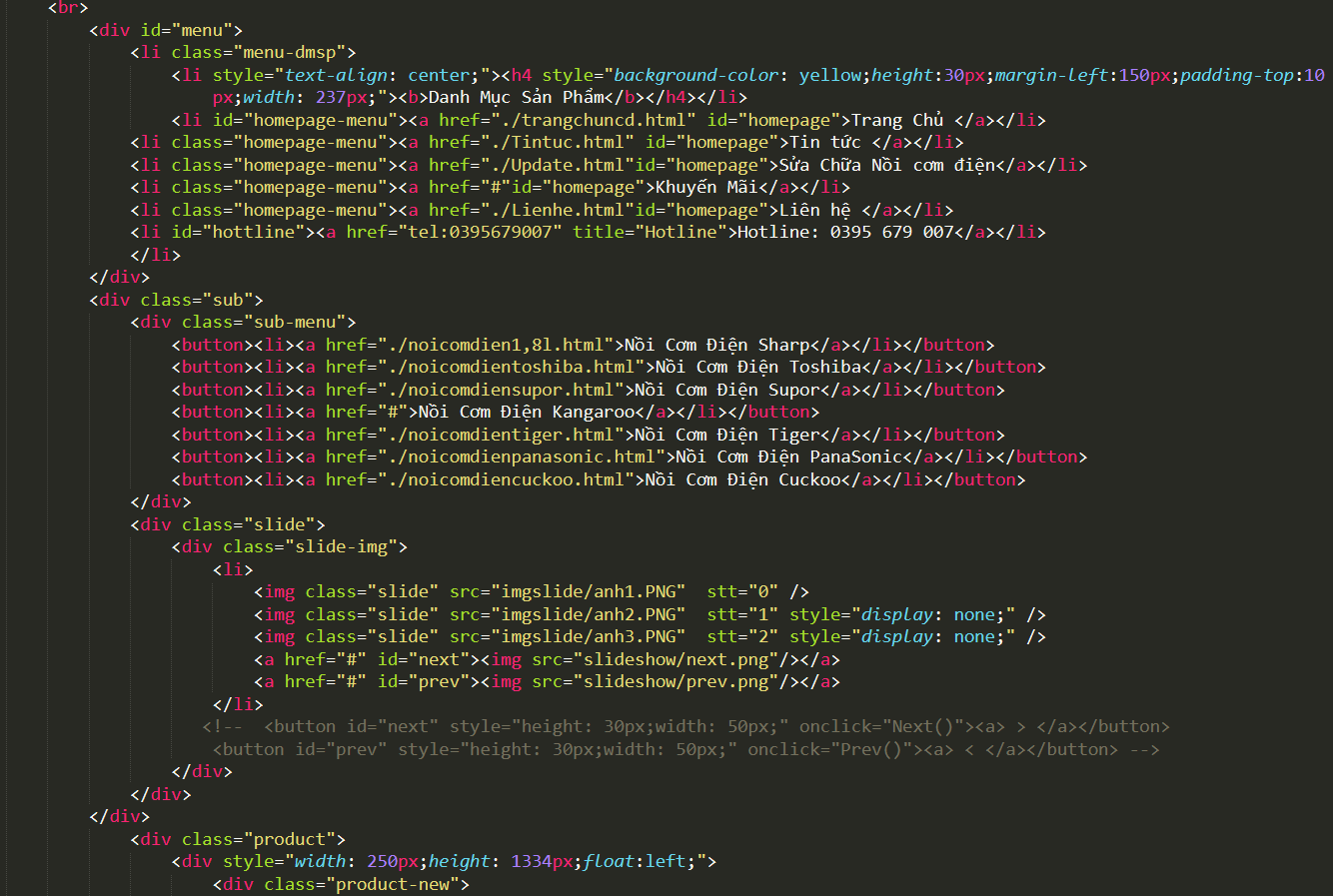
# 

PHỤ LỤC

1. Code trang “Chi tiết sản phẩm”.



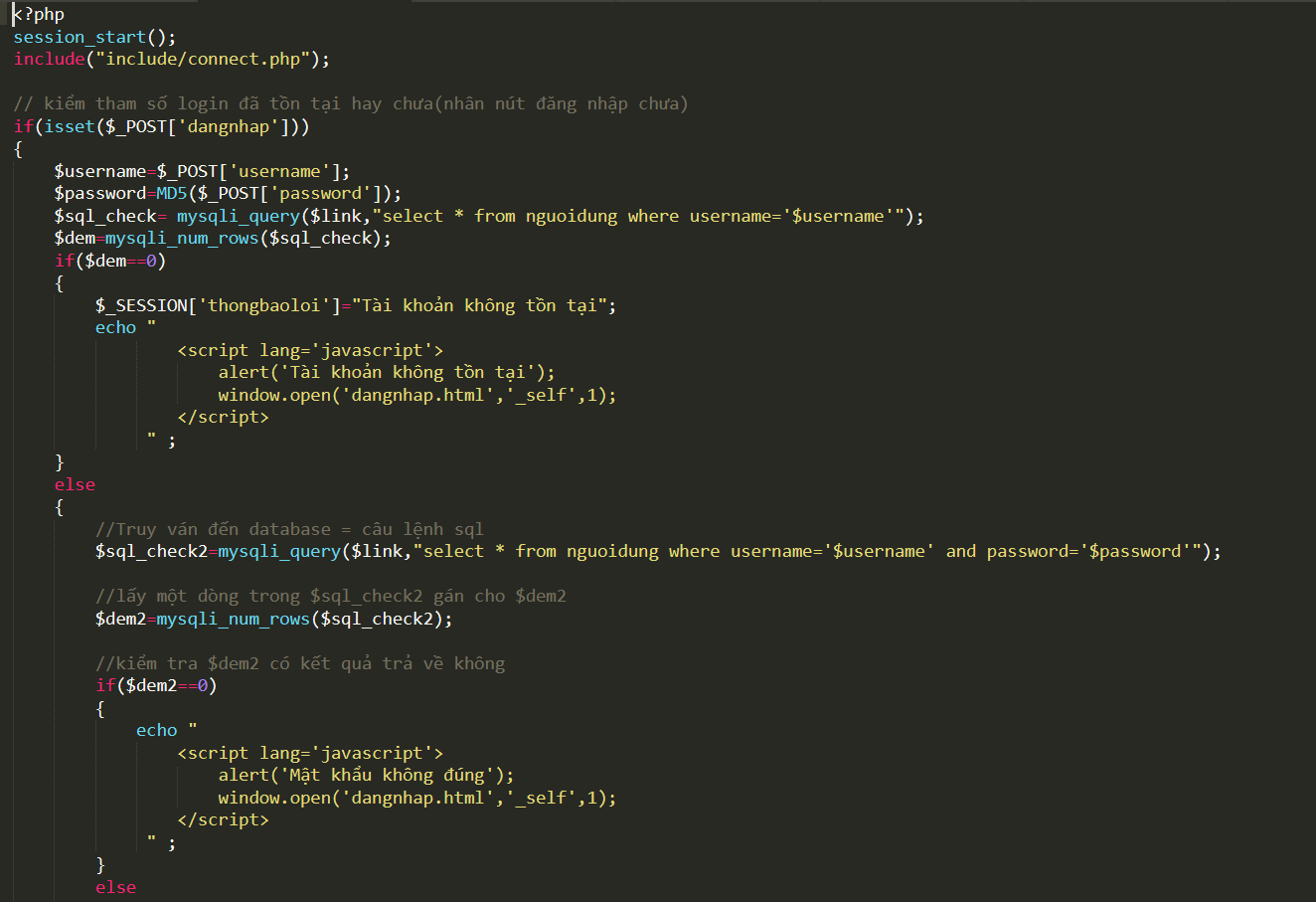
2. Code trang “Hiển thị sản phẩm theo danh mục sản phẩm”.



3. Code trang “Đăng ký người dùng”.



4. Code trang “Đăng nhập”.



5. Code trang “Đăng ký”.

