|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | BỘ CÔNG THƯƠNG | KHOA CÔNG NGHỆ THÔNG TIN  TRƯỜNG ĐẠI HỌC KINH TẾ - KỸ THUẬT CÔNG NGHIỆP | |  |   BÁO CÁO TỔNG KẾT  ĐỀ TÀI THỰC TẬP THIẾT KẾ WEB  NĂM HỌC 2023  Tên đề tài:  ỨNG DỤNG NGÔN NGỮ LẬP TRÌNH PHP  XÂY DỰNG WEBSITE BÁN DỤNG CỤ NHÀ BẾP   |  |  |  | | --- | --- | --- | | Giảng viên hướng dẫn: | Điền Thị Hồng Hà |  | | Sinh viên thực hiện: | -Nguyễn Văn Đức  -Nguyễn Văn Hòa  -Bùi Minh Dương |  |   Lớp : DHTI14A12HN  HÀ NỘI 11/2023 |

# LỜI NÓI ĐẦU

Trong bối cảnh mạnh mẽ của công nghệ ngày nay, Internet đóng vai trò không thể phủ nhận trong mọi khía cạnh của cuộc sống và sự phát triển khoa học kỹ thuật. Internet không chỉ là một mạng máy tính toàn cầu, mà còn là nền tảng cho sự truyền đạt thông tin, giao tiếp và kinh doanh.

Trong thế giới hiện đại, nơi mà mọi người "người người làm Web, nhà nhà làm Web," việc sở hữu một trang web không chỉ là một xu hướng mà còn là một yếu tố quyết định quan trọng. Trang web không chỉ giúp chúng ta chia sẻ thông tin một cách dễ dàng mà còn là công cụ mạnh mẽ để quảng bá và phát triển kinh doanh.

Đồng hành cùng sự phát triển của website là sự xuất hiện của nhiều ngôn ngữ lập trình, trong đó PHP (Personal Home Page) là một trong những kịch bản phía máy chủ mạnh mẽ, cho phép chúng ta xây dựng và quản lý trang web kết hợp với cơ sở dữ liệu. PHP cùng với MySQL đã trở thành lựa chọn phổ biến, mang lại nhiều ưu điểm cho người sử dụng

Với lí do đó, được sự hướng dẫn và giúp đỡ của cô ***Điền Thị Hồng Hà***, em đã chọn đề tài “**Xây dựng Website bán hàng đồ dùng nhà bếp**” làm đề tài cho môn học **Thực hành web** của mình.

Trong quá trình thực hiện làm bài tập lớn này em đã nhận được sự giúp đỡ chỉ bảo tận tình của các thầy, cô giáo, Em xin chân thành cảm ơn cô ***Điền Thị Hồng Hà*** đã hướng dẫn trong quá trình làm bài tập lớn.

Tuy nhiên, do thời gian hạn hẹp, mặc dù đã nỗ lực hết sức mình nhưng chắc rằng bài làm khó tránh khỏi thiếu sót. Em rất mong nhận được sự thông cảm, những lời góp ý và chỉ bảo tận tình của Cô và các bạn.

# TÓM TẮT ĐỒ ÁN

Với mức độ phức tạp và quy mô ứng dụng, cộng thêm vấn đề thời gian cho nên đề tài “**Xây dựng Website bán hàng đồ dùng nhà bếp**” chỉ dừng ở mức tìm hiểu ngôn ngữ lập trình Web PHP & MYSQL, và áp dụng xây dựng ứng dụng thực nghiệm trang Web bán hàng trên mạng.

Website với mục đích cung cấp cho khách hàng các thông tin chính xác về các sản liên quan đến nhà bếp đang có trên thị trường của các hãng có uy tín và cách thức đặt mua hàng qua mạng. Các thông tin được cập nhật thường xuyên và nhanh chóng. Vì vậy, rút ngắn được khoảng cách giữa người mua và người bán, đưa thông tin về các sản phẩm mới nhanh chóng đến cho khách hàng.

Do còn một số hạn chế, nên website mới chỉ dừng lại ở chức năng thêm vào giỏ hàng. Trong tương lai, hệ thống sẽ phát triển theo hướng thương mại điện tử (với hình thức thanh toán trực tuyến qua thẻ tín dụng, tài khoản ngân hàng, thẻ mua hàng trực tuyến…).

**MỤC LỤC**

*Nội dung*

[LỜI NÓI ĐẦU i](#_Toc375574633)

[TÓM TẮT ĐỒ ÁN ii](#_Toc375574634)

[DANH SÁCH HÌNH VẼ vii](#_Toc375574635)

[DANH SÁCH CÁC BẢNG BIỂU viii](#_Toc375574636)

[1.1. Khái niệm cơ bản về Internet 1](#_Toc375574637)

[1.1.1. Giới thiệu chung 1](#_Toc375574638)

[1.1.2. Phân loại 1](#_Toc375574639)

[1.2. World Wide Web 2](#_Toc375574640)

[1.2.1. Các khái niệm cơ bản về World Wide Web 2](#_Toc375574641)

[1.2.2. Cách tạo trang Web 2](#_Toc375574642)

[1.2.3. Trình duyệt Web (Web Client hay Web Browser) 2](#_Toc375574643)

[1.2.4. Webserver 3](#_Toc375574644)

[1.2.5. Phân loại Web 3](#_Toc375574645)

[1.2.5.1. Web tĩnh 3](#_Toc375574646)

[1.2.5.2. Website động 3](#_Toc375574647)

[1.3. HTML 4](#_Toc375574648)

[1.3.1. Cấu trúc chung của một trang HTML 4](#_Toc375574649)

[1.3.2. Các thẻ HTML cơ bản 4](#_Toc375574650)

[CHƯƠNG 2. TÌM HIỂU NGÔN NGỮ PHP 6](#_Toc375574651)

[2.1. Khái niệm PHP 6](#_Toc375574652)

[2.2. Tại sao nên dùng PHP 6](#_Toc375574653)

[2.3. Hoạt động của PHP: 7](#_Toc375574654)

[2.4. Các loại thẻ PHP 8](#_Toc375574655)

[2.5. Các kiểu dữ liệu 8](#_Toc375574656)

[2.5.1. Số nguyên 8](#_Toc375574657)

[2.5.2. Số thực 8](#_Toc375574658)

[2.5.3. Xâu 9](#_Toc375574659)

[2.5.4. Mảng 9](#_Toc375574660)

[2.5.4.1. Mảng một chiều: 9](#_Toc375574661)

[2.5.4.2. Mảng 2 chiều 9](#_Toc375574662)

[2.6. Biến - giá trị 9](#_Toc375574663)

[2.6.1. Một số biến đã được khai báo sẵn 9](#_Toc375574664)

[2.6.2. Phạm vi giá trị 9](#_Toc375574665)

[2.6.3. Tên biến 10](#_Toc375574666)

[2.7. Các giá trị bên ngoài phạm vi PHP 10](#_Toc375574667)

[2.8. Hằng 11](#_Toc375574668)

[2.9. Biểu thức 11](#_Toc375574669)

[2.10. Các cấu trúc lệnh 11](#_Toc375574670)

[2.10.1. Các lệnh điều kiện và toán tử 11](#_Toc375574671)

[2.10.2. Phát biểu vòng lặp While 13](#_Toc375574672)

[2.10.3. Vòng lặp For: 13](#_Toc375574673)

[2.10.4. Vòng lặp do... while 14](#_Toc375574674)

[2.10.5. Vòng lặp while 14](#_Toc375574675)

[2.11. Hàm 14](#_Toc375574676)

[2.11.1. Tham trị 14](#_Toc375574677)

[2.11.2 Tham biến 14](#_Toc375574678)

[2.11.3 Tham số có giá trị mặc định 14](#_Toc375574679)

[2.11.4. Giá trị trả lại của hàm 15](#_Toc375574680)

[2.11.5 Hàm biến 15](#_Toc375574681)

[2.12. Các toán tử 15](#_Toc375574682)

[2.13. Lớp và đối tượng 16](#_Toc375574683)

[2.14. Tham chiếu 16](#_Toc375574684)

[2.15. Khai báo và sử dụng Session, Cookie 16](#_Toc375574685)

[2.16. MySQL và PHP 19](#_Toc375574686)

[CHƯƠNG 3. CƠ SỞ DỮ LIỆU MYSQL 22](#_Toc375574687)

[3.1. Giới thiệu cơ sở dữ liệu 22](#_Toc375574688)

[3.2. Mục đích sử dụng cơ sở dữ liệu 22](#_Toc375574689)

[3.3. Các kiểu dữ liệu trong cơ sở dữ liệu MySQL 23](#_Toc375574690)

[3.3.1. Loại dữ liệu numeric 23](#_Toc375574691)

[3.3.2. Loại dữ liệu kiểu Date and Time 23](#_Toc375574692)

[3.3.3. Loại dữ liệu String 24](#_Toc375574693)

[3.4. Các thao tác cập nhật dữ liệu 25](#_Toc375574694)

[3.5. Các hàm thông dụng trong MySQL 26](#_Toc375574695)

[3.5.1. Các hàm trong phát biểu GROUP BY 26](#_Toc375574696)

[3.5.2. Các hàm xử lí chuỗi: 26](#_Toc375574697)

[3.5.3. Các hàm xử lí về thời gian 27](#_Toc375574698)

[3.5.4. Các hàm về toán học 27](#_Toc375574699)

[CHƯƠNG 4. XÂY DỰNG ỨNG DỤNG THỰC TẾ 28](#_Toc375574700)

[4.1. Phân tích yêu cầu đề tài 28](#_Toc375574701)

[4.1.1 Tóm tắt hoạt động của hệ thống mà dự án sẽ được ứng dụng: 28](#_Toc375574702)

[4.1.2. Phạm vi dự án được ứng dụng 28](#_Toc375574703)

[4.1.3. Đối tượng sử dụng 28](#_Toc375574704)

[4.1.4. Mục đích của dự án 29](#_Toc375574705)

[4.2. Xác định yêu cầu của khách hàng 29](#_Toc375574706)

[4.2.1. Hệ thống hiện hành của cửa hàng 29](#_Toc375574707)

[4.2.2. Hệ thống đề nghị 29](#_Toc375574708)

[4.3. Yêu cầu giao diện của website 31](#_Toc375574709)

[4.3.1. Giao diện người dùng 31](#_Toc375574710)

[4.3.2. Giao diện người quản trị: 31](#_Toc375574711)

[4.4. Phân tích các chức năng của hệ thống 31](#_Toc375574712)

[4.4.1. Các chức năng của đối tượng Customer (Khách vãng lai) 33](#_Toc375574713)

[4.4.1.1. Chức năng đăng ký thành viên 33](#_Toc375574714)

[4.4.1.2. Chức năng tìm kiếm sản phẩm 33](#_Toc375574715)

[4.4.1.3. Chức năng xem thông tin sản phẩm 33](#_Toc375574716)

[4.4.1.4. Chức năng giỏ hàng 33](#_Toc375574717)

[4.4.1.5. Chức năng đặt hàng 34](#_Toc375574718)

[4.4.2. Các chức năng của đối tượng Member (thành viên) 34](#_Toc375574719)

[4.4.2.1. Chức năng đăng nhập 34](#_Toc375574720)

[4.4.2.2. Chức năng đăng xuất 34](#_Toc375574721)

[4.4.2.3. Chức năng đổi mật khẩu 34](#_Toc375574722)

[4.4.2.4. Chức năng xem hóa đơn đã được lập 34](#_Toc375574723)

[4.4.3 Chức năng của Administrator 35](#_Toc375574724)

[4.4.3.1. Các chức năng quản lý Member 35](#_Toc375574725)

[4.4.3.2. Các chức năng quản lý Sản Phẩm 35](#_Toc375574726)

[4.4.3.3. Các chức năng quản trị người dùng 35](#_Toc375574727)

[4.4.3.4. Các Chức Năng Đối Với Hóa Đơn Đặt Hàng 36](#_Toc375574728)

[4.5. Biểu đồ phân tích thiết kế hệ thống 36](#_Toc375574729)

[4.5.1. Biểu đồ hoạt động 36](#_Toc375574730)

[4.5.2. Đặc tả Use-case 48](#_Toc375574731)

[4.6. Cách tổ chức dữ liệu và thiết kế chương trình cho trang Web 53](#_Toc375574732)

[4.6.1. Bảng giohang 54](#_Toc375574733)

[4.6.2. Bảng hoadon 54](#_Toc375574735)

[4.6.3. Bảng lienhe 54](#_Toc375574737)

[4.6.4. Bảng loaisanpham 55](#_Toc375574739)

[4.6.5. Bảng nhomsanpham 55](#_Toc375574741)

[4.6.6. Bảng sanpham 55](#_Toc375574743)

[4.6.7. Bảng thanhvien 55](#_Toc375574745)

[4.6.11. Sơ đồ thực thể liên kết 56](#_Toc375574747)

[4.7. Yêu cầu về bảo mật 57](#_Toc375574748)

[4.8. Thiết kế giao diện 57](#_Toc375574749)

[4.8.2. Giao diện Form đăng nhập 58](#_Toc375574750)

[4.8.5. Giao diện Form đăng ký thành viên 59](#_Toc375574751)

[4.8.6. Giao diện trang chi tiết sản phẩm 60](#_Toc375574752)

[4.8.7. Giao diện trang quản trị Admin 60](#_Toc375574753)

[TÀI LIỆU THAM KHẢO 62](#_Toc375574754)

**CHƯƠNG 1****. INTERNET, WEBSITE VÀ HTML,JS**

## 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.

## 1.2. Website,Html và Js

### 1.2.1. Các khái niệm cơ bản về website

Website là một tập hợp các trang web liên kết với nhau và lưu trữ trên một máy chủ web. Đây là một không gian trực tuyến nơi mà người ta có thể truy cập để xem thông tin, tương tác với nội dung, và thậm chí làm các giao dịch trực tuyến

### 1.2.2. Khái niệm và thành phần tạo nên website

-Trang web (web page): Là một trang đơn của thông tin trên Internet. Nó có thể chứa văn bản, hình ảnh, video, và các yếu tố tương tác.

-Website (trang web): Là một bộ sưu tập các trang web liên kết với nhau thông qua các liên kết và menu. Mỗi trang web có thể có một chủ đề cụ thể hoặc phục vụ mục đích cụ thể.

-Máy chủ (server): Là một máy tính lưu trữ và cung cấp trang web cho người truy cập qua Internet. Máy chủ đáp ứng yêu cầu từ trình duyệt web và trả về thông tin tương ứng.

-Trình duyệt web (web browser): Là phần mềm trên máy tính hoặc thiết bị di động mà người dùng sử dụng để xem và tương tác với các trang web. Các trình duyệt phổ biến bao gồm Chrome, Firefox, Safari, và Edge.

-Tên miền (domain) : Là địa chỉ duy nhất của một trang web trên Internet. Ví dụ, "[www.example.com](http://www.example.com/)" là một tên miền.

-Hosting web: Là dịch vụ cung cấp không gian lưu trữ trên máy chủ để lưu trữ dữ liệu và thông tin của một trang web. Các công ty hosting web cung cấp các gói dịch vụ để đảm bảo trang web có thể truy cập mọi lúc.

-HTML (Hypertext Markup Language) : Là ngôn ngữ đánh dấu được sử dụng để tạo cấu trúc và hiển thị nội dung trên trang web.

-Css(Cascading Style Sheets) : Là ngôn ngữ quy định cách mà HTML được hiển thị trên trình duyệt. Nó điều chỉnh các phong cách, màu sắc, và bố cục của trang web.

-Javascript : Là một ngôn ngữ lập trình cho phía máy khách (client-side) giúp thêm tính tương tác động cho trang web.

### 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.

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.

## 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.

## 1.4. Javascript

JavaScript là ngôn ngữ lập trình được nhà phát triển sử dụng để tạo trang web tương tác. Từ làm mới bảng tin trên trang mạng xã hội đến hiển thị hình ảnh động và bản đồ tương tác, các chức năng của JavaScript có thể cải thiện trải nghiệm người dùng của trang web. Là ngôn ngữ kịch bản phía máy khách, JavaScript là một trong những công nghệ cốt lõi của World Wide Web. Ví dụ: khi duyệt internet, bất cứ khi nào bạn thấy quảng cáo quay vòng dạng hình ảnh, menu thả xuống nhấp để hiển thị hoặc màu sắc phần tử thay đổi động trên trang web cũng chính là lúc bạn thấy các hiệu ứng của JavaScript.

### 1.4.1. Các khái niệm cơ bản

**1.Biến (variables) :** Là các định danh được sử dụng để lưu trữ dữ liệu. Biến có thể được tạo bằng từ khóa **var**, **let**, hoặc **const**.

* Ví dụ:
  + var age = 25;
  + let name = “Nguyễn Văn Đức”;
  + const Pi = 3,14;

**2.Kiểu dữ liệu (data types) :** JavaScript có nhiều kiểu dữ liệu như số, chuỗi, boolean, mảng, đối tượng, v.v.

* Ví dụ:
  + var number = 32;
  + var text = “anhduc”;
  + var istrue = true;
  + var fruits = [“apple”, “banana”];
  + var person = {name: “anhduc”,age:20}

**3.Hàm(funtions):** Là một khối mã có thể được gọi để thực hiện một công việc cụ thể. Hàm có thể nhận tham số và trả về giá trị.

* Ví dụ:
  + Funtion(a,b){

Return a+b;

}

**4.Biểu thức điều kiện(Conditional Statements):** Được sử dụng để kiểm tra điều kiện và thực hiện các hành động khác nhau dựa trên kết quả kiểm tra.

* Ví dụ:
  + var age = 8;

if (age>18){

console.log(“hello”)

}else{

console.log(“hi”)

}

**5.Vòng lặp (Loops):** Cho phép thực hiện một khối mã nhiều lần.

* Ví dụ:
  + for (let i=0;i<5;i++){

console.log(i);

{

**6.Sự kiện (Events):** JavaScript có thể xử lý sự kiện từ người dùng, chẳng hạn như nhấn nút chuột, gõ phím, hoặc thay đổi kích thước cửa sổ.

* Ví dụ:
  + Document.queryselector(“button”).addEventListioner(“onclick

”,()=>{

Alert(“Bạn đã click vào button”);

})

**7.Promise và Asynchronus programming :** Cho phép xử lý các tác vụ không đồng bộ mà không làm chặn luồng chính.

* Ví dụ:
  + Function fetchData(){

Return new Promise(resolve,reject){

Resolve(“Data fetched successfully”);

}

}

Fetch().then(data){

Console.log(data);  
}

# CHƯƠNG 2. 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.

## 2.2. Tại sao nên dùng PHP

Để thiết kế Web động có rất nhiều ngôn ngữ lập trình khác nhau để lựa chọn, mặc dù cấu hình và tính năng khác nhau nhưng chúng vẵn đưa ra những kết quả giống nhau. Chúng ta có thể lựa chọn cho mình một ngôn ngữ: ASP, PHP, Java, Perl... và một số loại khác nữa. Vậy tại sao chúng ta lại nên chọn PHP. Rất đơn giản, có những lí do sau mà khi lập trình Web chúng ta không nên bỏ qua sự lựa chọn tuyệt vời này.

PHP được sử dụng làm Web động vì nó nhanh, dễ dàng, tốt hơn so với các giải pháp khác.

PHP có khả năng thực hiện và tích hợp chặt chẽ với hầu hết các cơ sở dữ liệu có sẵn, tính linh động, bền vững và khả năng phát triển không giới hạn.

Đặc biệt PHP là mã nguồn mở do đó tất cả các đặc tính trên đều miễn phí, và chính vì mã nguồn mở sẵn có nên cộng đồng các nhà phát triển Web luôn có ý thức cải tiến nó, nâng cao để khắc phục các lỗi trong các chương trình này

PHP vừa dễ với người mới sử dụng vừa có thể đáp ứng mọi yêu cầu của các lập trình viên chuyên nghiệp, mọi ý tuởng của các bạn PHP có thể đáp ứng một cách xuất sắc.

Cách đây không lâu ASP vốn được xem là ngôn ngữ kịch bản phổ biến nhất, vậy mà **b**ây giờ PHP đã bắt kịp ASP, bằng chứng là nó đã có mặt trên 12 triệu Website.

## 2.3. 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.

## 2.4. Các loại thẻ PHP

Có 4 loại thẻ khác nhau mà bạn có thể sử dụng khi thiết kế trang PHP:

* Kiểu **Short**: Thẻ mặc định mà các nhà lập trình PHP thường sử dụng.

Ví dụ: <? **Echo** “*Well come to PHP*. ” ;?>

* Kiều đinh dạng **XML**: Thẻ này có thể sử dụng với văn bản đinh dạng XML

Ví dụ: <? **Php echo** “*Well come to PHP with XML*”;>?

* Kiểu **Script**: Trong trường hợp bạn sử dụng PHP như một script tương tự khai báo JavaScipt hay VBScript:

Ví dụ: **<script language= “php”>**

**echo** “*Php Script*”;

**</script>**

* Kiểu **ASP:** Trong trường hợp bạn khai báo thẻ PHP như một phần trong trang ASP.

Ví dụ: **<% echo “***PHP – ASP***”; %>**

**\****PHP và HTML là các ngôn ngữ không “nhạy cảm “với khoảng trắng, khoảng trắng có thể được đặt xung quanh để các mã lệnh để rõ ràng hơn. Chỉ có khoảng trắng đơn có ảnh hưởng đến sự thể hiện của trang Web (nhiều khoảng trắng liên tiếp sẽ chỉ thể hiện dưới dạng một khoảng trắng đơn).*

## 2.5. Các kiểu dữ liệu

Dữ liệu đến từ Script đều là biến PHP, bạn có thể nhận biết chúng bằng cách sử dụng dấu $ trước tên biến.

### 2.5.1. Số nguyên

Số nguyên được khai báo và sử dụng giá trị giống với C.

Ví dụ:

$a=12345;

$a=-456;

### 2.5.2. Số thực

Ví dụ:

$a=2. 123;

$b=3. 1e3;

### 2.5.3. Xâu

Xâu có hai cách để xác định 1 xâu: Đặt giữa 2 dấu ngoặc kép (“ ”) hoặc giữa 2 dấu ngoặc (‘ ’).

### 2.5.4. Mảng

Mảng thực chất gồm 2 bảng: Bảng chỉ số và bảng liên kết.

2.5.4.1. Mảng một chiều:

Có thể dùng hàm List() hoặc Array(). Có thể dùng các hàm aort(), ksort(), sort(), uaort(),... để sắp xếp mảng, tùy thuộc vào việc bạn định sắp xếp theo kiểu gì..

2.5.4.2. Mảng 2 chiều

Ví dụ:

$a[1]=$f;

$a[1][2]=$f;

$a[“abc”][2]=$f;

...

## 2.6. Biến - giá trị

PHP quy định một biến được biểu diễn bắt đầu bằng dấu $, sau đó là một chữ cái hoặc dấu gạch dưới.

### 2.6.1. Một số biến đã được khai báo sẵn

HTTP\_GET\_VARS: Mảng các giá trị nguyên truyền tới **script** thông qua phương thức HTTP GET. Chỉ có tác dụng nếu “***track\_vars***” .Trong cấu hình được đặt hoặc chỉ dẫn **<? Php\_track\_vars?>.**

HTTP\_POST\_VARS: Mảng các giá trị nguyên truyền tới script thông qua phương thức HTTP POST.

HTTP\_COOKIE\_VARS: Một mảng các giá trị được truyền tới script hiện tại bằng HTTP cookie. Chỉ có tác dụng nếu “***track\_vars***” trong cấu hình được đặt hoặc chỉ dẫn **<?php\_track\_vars?>...**

### 2.6.2. Phạm vi giá trị

PHP coi một biến có một giới hạn. Để xác định một biến toàn cục (global) có tác dụng trong một hàm ta cần khai báo lại. Nếu không có giá trị của biến sẽ được coi như là cục bộ trong hàm.

VD:

$a=1;

$b=2;

Function Sum(){

Global $a, $b;

$b=$a+$b;

}

Sum();

Echo $b;

Khi có khai báo global, $a và $b được biết đó là những biến toàn cục. Nếu không có khai báo global, $a và $b chỉ được coi là các biến bên trong hàm Sum().

### 2.6.3. Tên biến

Một biến có thể gắn với một cái tên

Ví dụ:

$a= “chao”;

$$a= “cacban”;

* $Chao= “cacban”

Và echo”$a${chao}”;

Kết quả sẽ là “chaocacban”

## 2.7. Các giá trị bên ngoài phạm vi PHP

HTML Form: Khi 1 giá trị gắn với 1 file php qua phương thức POST

Ví dụ:

<form action = “top. php” method= “post”>

Name: < input type = “text” name = “name” ><BR>

<input type = “Submit”>

</form>

PHP sẽ tạo 1 biến $ name bao gồm mọi giá trị trong trường Name của Form

PHP có thể hiểu được một mảng một chiều gồm các giá trị trong một Form

Vì vậy, bạn có thể nhóm những giá trị liên quan lại với nhau hoặc sử dụng đặc tính này để nhận các giá trị từ 1 đầu vào tuỳ chọn.

Khi tính chất track\_vars được đặt trong cấu hình hoặc có chỉ dẫn <?php\_track\_vars?>. Các giá trị được submit sẽ lấy ra qua phương thức GET và POST có thể lấy ra từ 2 mảng toàn cục $HTTP\_POST\_VARS và $HTTP\_GET\_VARS

## 2.8. Hằng

PHP định nghĩa sẳn các hằng số:

\_FILE\_: Tên của script file đang thực hiện

*\_*LINE\_: Số dòng của mã script đang được thực hiện trong script file hiện tại.

\_PHP\_VERSION\_: version của PHP đang chạy

TRUE

FALSE

E\_ERROR: Báo hiệu có lỗi

E\_PARSE: Báo lỗi sai khi biên dịch

E\_NOTICE: Một vài sự kiện có thể là lỗi hoặc không

Có thể định nghĩa một hằng số bằng hàm define()

VD:

<?php

define('sv', 'localhost');

define('dbname', 'minhcanh);

define('username', 'root');

define('pass', '');

?>

## 2.9. Biểu thức

Biểu thức là một thành phần quan trọng trong PHP. Một dạng cơ bản nhất của biểu thức bao gồm các biến và hằng số. PHP hỗ trợ 3 kiểu giá trị cơ bản nhất: Số nguyên, số thực và xâu. Ngoài ra còn có mảng và đối tượng. Mỗi kiểu giá trị này có thể gán cho các biến hay làm giá trị ra của các hàm.

## 2.10. Các cấu trúc lệnh

### 2.10.1. Các lệnh điều kiện và toán tử

Mỗi câu lệnh điều kiện bao gồm một mệnh đề **if:**

**If(điều kiện){**

**//thực hiện một điều gì đó**

**}**

Điều kiện này có thể được mở rộng thành:

**If(điều kiện){**

**//thực hiện một điều gì đó**

**} else {**

**//thực hiện một điều khác**

**}**

Và:

**If(điều kiện 1){**

**//thực hiện một điều gì đó**

**} elseif(điều kiện 2) {**

**//thực hiện một điều khác**

**}else {**

**//thực hiện một điều khác nữa**

**}**

*Các toán tử thường được sử dụng với câu lệnh điều kiện*

|  |  |  |  |
| --- | --- | --- | --- |
| Kí hiệu | Ý nghĩa | Dạng | Ví dụ |
| = | Được gán giá trị của | Gán | $n=1 |
| == | Bằng | So sánh | $x==$y |
| != | Không bằng | So sánh | $x!=$y |
| < | Nhỏ hơn | So sánh | $x<$y |
| > | Lớn hơn | So sánh | $x>$y |
| <= | Nhỏ hơn hoặc bằng | So sánh | $x<=$y |
| >= | Lớn hơn hoặc bằng | So sánh | $x>=$y |
| ! | Phủ định | Logic | !$x |
| && | AND | Logic | $x && $y |
| || | OR | Logic | $x || $y |

PHP còn có một dạng câu lệnh điều kiện nữa là **Swich**. Câu lệnh này rất phù hợp cho việc thay thế nếu câu lệnh **if-elseif-else** quá dài. Cú pháp của câu lệnh **switch** như sau:

**Switch($variable){**

**Case ‘giá trị 1’:**

**//thực hiện lệnh**

**Break;**

**Case ‘giá trị 2’:**

**//thực hiện lệnh**

**Break;**

**Default:**

**//thực hiện lệnh**

**}**

### 2.10.2. Phát biểu vòng lặp While

Phát biểu đơn giản nhất trong PHP là vòng lặp While, cho phép thực thi khối lệnh trong While cho đến điều kiện của While là True như cú pháp:

**While (condition)**

**{**

**Expression;**

**}**

Trong đó:

**Condition:** Biểu thức điều kiện, biến,...

**Expression:** Khối lệnh trong vòng lặp while

### 2.10.3. Vòng lặp For:

**For(expression1;condition;expression2**)

{

**expression3**;

}

Trong đó:

**Condition:** điều kiện giới hạn của vòng lặp for

**Expression1**: Giá trị khởi đầu của vòng lặp for

**Expression2**: Giá trị lặp của vòng lặp for

**Expression3**: Khối lệnh bên trong của vòng lặp for

### 2.10.4. Vòng lặp do... while

**Do**

**{ expression }**

**while(condition)**

Trong đó:

**Expression**: khối lệnh bên trong vòng lặp **do... while**

**Condition**: biểu thức điều kiện, biến...

### 2.10.5. Vòng lặp while

**While(condition)**

**{**

**expression;**

**}**

Trong đó: Biểu thức điều kiện, biến...

**Expression**: Khối lệnh bên trong vòng lặp while

**2.10. 6. Exit:**Thoát khỏi các vòng lặp hay phát biểu điều khiển nào đó.

## 2.11. Hàm

Dùng giống với C++. Ngoại trừ bạn không cần phải khai báo kiểu cho tham số của hàm:

### 2.11.1. Tham trị

Ví dụ: function takes\_array($input)

{

echo “$input[0] + $input[1] = “, $input[0]+$input[1];

}

### 2.11.2 Tham biến

Function add\_some\_extra (&$string)

{$string. =’and somthing extra’; }

### 2.11.3 Tham số có giá trị mặc định

Function makecoffee($type = “cappucino”)

{ Return “Making a cup of $type. \n”; }

Chú ý: Khi sử dụng hàm có đối số có giá trị mặc định, các biến này sẽ phải nằm về phía phải nhất trong danh sách đối số.

VD: Sai

Function makeyogurt($type=”acidophilus”, $flavour)

{

Return “Making a bowl of $type $flavour. \n”;

}

Đúng:

Function makeyogurt($flavour, $type=”acidophilus”)

{

Return “Making a bowl of $type $flavour. \n”;

}

### 2.11.4. Giá trị trả lại của hàm

Có thể là bất kỳ giá trị nào. Tuy vậy, không thể trả lại nhiều giá trị riêng lẻ nhưng có thể trả lại một mảng các giá trị.

### 2.11.5 Hàm biến

PHP cho phép sử dụng hàm giá trị nghĩa là khi một biến được gọi có kèm theo dấu ngoặc đơn, PHP sẽ tìm hàm có cùng tên với giá trị biến đó thực hiện.

## 2.12. Các toán tử

**Các phép số học:** +, -, \*, /%

**Các toán tử logic:** And, or, xor: &&, ||, !

**Toán tử thao tác với bít:** &, |, ^, ~, <<, >>

**Toán tử so sánh:** ==, !=, <, >, <=, >=, ===

**Toán tử điều khiển lỗi:** @

Khi đứng trước 1 biểu thức thì các lỗi của biểu thức sẽ bị bỏ qua và lưu trong $PHP\_errormsg

<?php

$link=mysql\_connect(sv, username, pass);

if(!$link)

die("khong ket noi duoc mysql");

?>

Toán tử thực thi:’ ‘PHP sẽ thực hiện nội dung nằm giữa 2 dấu ‘ như 1 lệnh shell. Trả ra giá trị là kết quả thực hiện lệnh.

VD:

$output=’Is-al’; //liệt kê các file bằng lệnh Linux

Echo”$output”;

## 2.13. Lớp và đối tượng

**Class**: là tập hợp các biến và hàm làm việc với các biến này. Một lớp có định dạng như sau:

<php

Class Cart{

Var $items;

Function add\_item($a)

Lớp Cart ở đây là một kiểu dữ liệu

## 2.14. Tham chiếu

Tham chiếu trong PHP có nghĩa là lấy cùng giá trị bằng nhiều tên biến khác nhau. Khác với con trỏ C, tham chiếu là một bảng các bí danh. Chú ý: Trong PHP, tên biến và nội dung của các biến khác nhau. Vì vậy, cùng một nội dung có thể có nhiều tên khác nhau.

Tham chiếu PHP cho phép bạn tạo 2 biến có cùng nội dung.

Ví dụ:

$a=&$b; ==> $a, $b trỏ tới cùng một giá trị

Tham chiếu truyền giá trị bằng tham chiếu. Thực hiện việc này bằng cách tạo một hàm cục bộ và truyền giá trị được tham chiếu.

Ví dụ:

function f(&$var)

{$var++; }

$a=5;

f($a);

--> Kết quả: $a=6;

Giá trị trả lại của một hàm bằng tham chiếu rất tiện lợi khi bạn muốn sử dụng hàm để tìm 1 giá trị trong một phạm vi nào đó.

## 2.15. 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 ]]]]])

## 2.16. MySQL và PHP

Để connect tới 1 CSDL trên MySQL Server rất đơn giản. Chỉ cần sử dụng hàm:mysql\_connect (host, user, password) để mở 1 kết nối tới MSQL Server với kết quả là giá trị trả về của hàm (Giả sử là biến $dbname). Sau đó, dùng hàm mysql\_select\_db(dbname, $link)

VD:

<?php

define('sv', 'localhost');

define('dbname', 'minhcanh');

define('username', 'root');

define('pass', '');

$link=mysql\_connect(sv, username, pass);

mysql\_select\_db("minhcanh", $link)or die("khong ket noi duoc co so du lieu)".

mysql\_error());

?>

Để thực hiện một câu lệnh truy vấn, dùng hàm mysql\_query(query, link). Giá trị trả lại của hàm là kết quả của câu truy vấn. Nếu bỏ qua link thì kết nối cuối cùng tới MySQL server sẽ được thực hiện.

VD:

<?php

mysql\_query("insert into hang value('$mahang', '$tenhang')", $link)or die("khong chen duoc co so du lieu". mysql\_error());

?>

Để lấy kết quả thực hiện, ta có thể dùng hàm mysql\_result(), mysql\_fetch\_row(), mysql\_fetch\_array(), mysql\_fetch\_object().

\* **mysql\_fetch\_row():**Trả về một mảng các dữ liệu lấy từ 1 dòng trong CSDL. Nếu đã ở cuối CSDL, giá trị trả về là false. Bạn phải dùng chỉ số của các trường trong CSDL nếu muốn lấy dữ liệu.

VD:

<?php

mysql\_select\_db(dbname, $link)or die("Loi chon co so du lieu ket noi(bangdl\_hang. php):". mysql\_error());

$result=mysql\_query("SELECT \* FROM hang")or die("Loi chon co so du lieu(bangdl. php):". mysql\_error());

$sohang=mysql\_num\_rows($result);

echo"số hãng nhập vào hiện nay là:$sohang. <p>";

echo"<table width=\"100%\" boder=1>";

echo "<tr>";

echo "<td width=\"50%\">Mã hãng</td><td width=\"50%\">Tên hãng</td>";

echo "</tr>";

while($thongtinvao=mysql\_fetch\_row($result))

{echo "<tr>";

echo "<td width=\"50%\"><font face=arial size=1/> $thongtinvao[0] </font></td>";

echo "<td width=\"50%\"><font face=arial size=1/><a href =\"nhaphang. php\"> $thongtinvao[1] </a></font></td>";

echo "</tr>";

}

echo "</table>";

mysql\_close($link);

?>

**\* mysql\_fetch\_array(int result[, int result\_type]):** Trả về một mảng chứa dữ liệu lấy từ một dòng trong CSDL. Tương tự như mysql\_fetch\_row() nhưng bạn có thể lấy thêm trường để lấy dữ liệu.

**\* mysql\_fetch\_object:**Chuyển kết quả thành một đối tượng với các trường là các trường trong cơ sở dữ liệu.

# 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. XÂY DỰNG ỨNG DỤNG THỰC TẾ

## 4.1. Phân tích yêu cầu đề tài

### 4.1.1 Tóm tắt hoạt động của hệ thống mà dự án sẽ được ứng dụng:

* Là một Website chuyên bán các sản phẩm về dụng cụ nhà bếp và phòng ăn
* Là một Website động, các thông tin được cập nhật theo định kỳ.
* Người dùng truy cập vào Website có thể xem, tìm kiếm, mua các sản phẩm,hoặc đặt hàng sản phẩm
* Người dùng có thể xem chi tiết từng sản (*có hình ảnh minh hoạ sản phẩm*).
* Khi đã chọn được món hàng vừa ý thì người dùng click vào nút cho vào giỏ hàng hoặc chọn biểu tượng giỏ hàng để sản phẩm được cập nhật trong giỏ hàng.
* Người dùng vẫn có thể quay trở lại trang sản phẩm để xem và chọn tiếp, các sản phẩm đã chọn sẽ được lưu vào trong giỏ hàng.
* Khách hàng cũng có thể bỏ đi những sản phẩm không vừa ý đã có trong giỏ hàng.
* Nếu đã quyết định mua các sản phẩm trong giỏ hàng thì người dùng click vào mục thanh toán để hoàn tất việc mua hàng, hoặc cũng có thể huỷ mua hàng.
* Đơn giá của các món hàng sẽ có trong giỏ hàng.

### **4.1.2**. Phạm vi dự án được ứng dụng

* Dành cho mọi đối tượng có nhu cầu mua sắm trên mạng.
* Do nghiệp vụ của cửa hàng kết hợp với công nghệ mới và được xử lý trên hệ thống máy tính nên công việc liên lạc nơi khách hàng cũng như việc xử lý hoá đơn thực hiện một cách nhanh chóng và chính xác. Rút ngắn được thời gian làm việc, cũng như đưa thông tin về các sản phẩm mới nhanh chóng đến cho khách hàng.

### 4.1.3. Đối tượng sử dụng

\* *Người dùng:* Qua Website, khách hàng có thể xem thông tin, lựa chọn những sản phẩm ưa thích ở mọi.

### 4.1.4. Mục đích của dự án

* Đáp ứng nhu cầu mua bán ngày càng phát triển của xã hội, xây dựng nên một môi trường làm việc hiệu quả.
* Thúc đẩy phát triển buôn bán trực tuyến.
* Rút ngắn khoảng cách giữa người mua và người bán, tạo ra một Website trực tuyến có thể đưa nhanh thông tin cũng như việc trao đổi mua bán các loại điện thoại qua mạng.
* Việc quản lý hàng trở nên dễ dàng.
* Sản phẩm được sắp xếp có hệ thống nên người dùng dễ tìm kiếm sản phẩm

## 4.2. Xác định yêu cầu của khách hàng

### 4.2.1. Hệ thống hiện hành của cửa hàng

Hiện tại cửa hàng đang sử dụng cơ chế quảng cáo, hợp đồng trưng bày, đặt hàng trực tiếp tại trung tâm do các nhân viên bán hàng trực tiếp đảm nhận (*hoặc có thể gọi điện đến nhà phân phối để đặt hàng và cung cấp địa chỉ để nhân viên giao hàng tận nơi cho khách hàng*). Trong cả hai trường hợp trên khách hàng đều nhận hàng và thanh toán trực tiếp bằng tiền mặt tại nơi giao hàng.Và trong tờ hoá đơn khách hàng phải điền đầy đủ thông tin cá nhân và số lượng mặt hàng cần mua. Kế toán phải chuẩn bị hoá đơn thanh toán, nhập một số dữ liệu liên quan như ngày, giờ, tên khách hàng, mã số mặt hàng, số lượng mua, các hợp đồng trưng bày sản phẩm, tổng số các hoá đơn vào trong sổ kinh doanh.

Dựa vào những ràng buộc cụ thể của hệ thống hiện hành chúng ta có thể đề nghị một hệ thống khác tiên tiến hơn, tiết kiệm được thời gian…

### 4.2.2. Hệ thống đề nghị

Để có thể vừa quảng cáo, bán hàng và giới thiệu sản phẩm của cửa hàng trên mạng thì website cần có các phần như:

*Về giao diện:*

* Giao diện người dùng
  + Phải có một giao diện thân thiện dễ sử dụng.
  + Phải nêu bật được thế mạnh của website, cũng như tạo được niềm tin cho khách hàng ngay từ lần viếng thăm đầu tiên.
  + Giới thiệu được sản phẩm của cửa hàng đang có.
  + Có biểu mẫu đăng ký để trở thành thành viên.
  + Có danh mục sản phẩm.
  + Chức năng đặt hàng, mua hàng.
  + Có biểu mẫu liên hệ.
  + Chức năng tìm kiếm sản phẩm.
  + Mỗi loại sản phẩm cần phải có trang xem chi tiết sản phẩm.
  + Mục login của khách hàng khi đã trở thành thành viên gồm có: Tên đăng nhập *(Username*), mật khẩu đăng nhập (*password*).

.

*Về nghiệp vụ*

* Người xây dựng nên hệ thống phải am hiểu về thương mại điện tử, hiểu rõ cách thức mua bán hàng qua mạng.
* Người xây dựng nên hệ thống đặc biệt phải hiểu rõ các thông số kĩ thuật về máy tính: Có thể đưa ra đựơc một cái nhìn tổng quát, sâu rộng về sản phẩm.

*Về bảo mật:*

* Quản lý User & Password của khách hàng an toàn: Thông tin của khách hàng được bảo mật

*Lựa chọn giải pháp*

* Chương trình sử dụng ngôn ngữ PHP ở bên phía server ,Javascript bên phía client và cơ sở dữ liệu MySQL.
* Các công cụ mà hệ thống sử dụng:
  + Gói XAMPP: Cài đặt các thành phần Apache, PHP, MySQL.
    - Apache 2 - Server
    - PHP 5.5.1 - Ngôn ngữ lập trình
    - MySQL - Cơ sở dữ liệu
    - Và các tính năng chuyên sâu khác...
  + mySQL: Dùng để lưu cơ sở dữ liệu.
  + Javascript: xử lí sự kiện lấy dữ liệu từ bên phía server về
  + Photoshop, flash: Thiết kế giao diện.

## 4.3. Yêu cầu giao diện của website

### 4.3.1. Giao diện người dùng

Phải có một giao diện thân thiện dễ sử dụng:

* Màu sắc hài hoà làm nổi bật hình ảnh của sản phẩm, font chữ thống nhất, tiện lợi khi sử dụng.
* Giới thiệu những thông tin về cửa hàng: Địa chỉ liên hệ của cửa hàng hoặc các đại lý trực thuộc, các số điện thoại liên hệ, email liên hệ.
* Sản phẩm của cửa hàng: sản phẩm đang có, sản phẩm bán chạy, sản phẩm mới nhất, sản phẩm sắp ra mắt…

## 4.4. Phân tích các chức năng của hệ thống

* Customer: Là khách hàng có chức năng: Đăng kí, đăng nhập,tìm kiếm, xem thông tin sản phẩm, xem giỏ hàng, đặt hàng.

### 4.4.1. Các chức năng của đối tượng Customer

Khi tham gia vào hệ thống thì họ có thể xem thông tin, tìm kiếm sản phẩm, đăng ký là thành viên của hệ thống, đăng nhập, chọn sản phẩm vào giỏ hàng, xem giỏ hàng, đặt hàng.

4.4.1.1. Chức năng đăng ký thành viên

Description: Dành cho khách vãng lai đăng ký làm thành viên để có thêm nhiều chức năng cho việc mua bán sản phẩm, cũng như tạo mối quan hệ lâu dài với công ty. Qua đó, thành viên sẽ được hưởng chế độ khuyến mại đặc biệt.

Input: Khách hàng phải nhập đầy đủ các thông tin (\*: là thông tin bắt buộc)

Tên đăng nhập: (\*)

Mật khẩu: (\*)

Nhập lại mật khẩu (\*)

Họ và tên: Nhập họ và tên. (\*)

Email: nhập email. (\*)

Địa chỉ: Nhập địa chỉ của Member. (\*)

Điện thoại: Nhập điện thoại. (\*)

Process: Kiểm tra thông tin nhập. Nếu thông tin chính xác sẽ lưu thông tin vào CSDL và thêm thông tin của thành viên đó vào CSDL.

Output: Đưa ra thông báo đăng ký thành công hoặc yêu cầu nhập lại nếu thông tin không hợp lệ

4.4.1.2. Chức năng đăng nhập thành viên

+ Description: Cho Member login vào hệ thống.

+ Input: Người dùng nhập vào các thông tin về username, password để login.

+ Process: Kiểm tra username và password của người dùng nhập vào và so sánh với username và password trong CSDL.

+ Output: nếu đúng cho đăng nhập và hiển thị các chức năng của Member, ngược lại hiển thị thông báo yêu cầu nhập lại nếu thông tin không chính xác.

4.4.1.3. Chức năng tìm kiếm sản phẩm

+ Input: Nhập vào hoặc lựa chọn thông tin tìm kiếm theo Hãng, theo Giá, theo tên sản phẩm, theo chủng loại…

+ Process: Lấy thông tin sản phẩm từ CSDL theo các trường tìm kiếm.

+ Output: Hiện chi tiết thông tin sản phẩm thỏa mãn yêu cầu tìm kiếm.

4.4.1.4. Chức năng xem thông tin sản phẩm

+ Description: Cho phép xem chi tiết thông tin của sản phẩm.

+ Input: Chọn sản phẩm cần xem.

+ Process: Lấy thông tin sản phẩm từ CSDL thông qua ID.

+ Output: Hiện chi tiết thông tin sản phẩm.

4.4.1.5. Chức năng giỏ hàng

+ Description: Cho phép xem chi tiết giỏ hàng.

+ Input: Click chọn vào giỏ hàng.

+ Process: Lấy thông tin sản phẩm từ CSDL thông qua ID. ID này được lấy từ biến session lưu các thông tin về mã sản phẩm và số lượng của từng sản phẩm mà khách hàng chọn vào giỏ hàng.

+ Output: Hiển thị thông tin về tên sản phẩm, ảnh, đơn giá, số lượng, thành tiền, tổng tiền của sản phẩm.

4.4.1.6. Chức năng đặt hàng

+ Description: Cho phép tất cả các khách hàng có thể đặt hàng mà không cần phải là thành viên của hệ thống.

+ Input: Sau khi xem giỏ hàng, khách hàng có nhu cầu đặt hàng qua mạng thì có thể click vào nút đặt hàng và điền một số thông tin cần thiết để người quản trị có thể xác nhận thông tin và giao hàng trực tiếp đến đúng địa chỉ một cách nhanh nhất có thể.

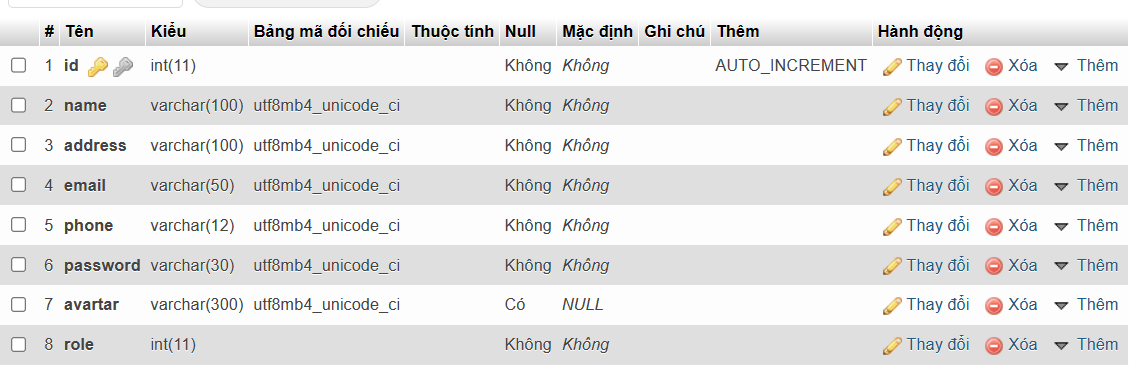
+ Process: Lưu thông tin về khách hàng và thông tin hóa đơn đặt hàng vào các bảng trong cơ sở dữ liệu.

+ Output: Đưa ra thông báo đơn đặt hàng đã được lập thành công hoặc không thành công.

## 4.5. Cơ sở dữ liệu

Việc tổ chức dữ liệu phải giải quyết được các yêu cầu đã phân tích để lúc hiển thị lên trang Web có giao diện dễ nhìn, dễ dàng, hấp dẫn khách hàng... Một phương pháp sử dụng khá rộn rãi từ trước đến nay là quản lí theo từng nhóm sản phẩm. Mỗi nhóm sẽ có một mã nhóm và tên nhóm để nhận biết. Đối với Website này, với các sản phẩm, các sản phẩm sẽ tương ứng với mã hãng và mã loại sản phẩm của nó và mã đó sẽ làm khóa chính trong bảng đó. Sau đây là một số cơ sở dữ liệu chính trong toàn bộ cơ sở dữ liệu:

4.5.1.Bảng khách hàng:lưu thông tin khách hàng



Bảng 1. Bảng khách hàng

4.5.2.Bảng sản phẩm:lưu thông tin sản phẩm



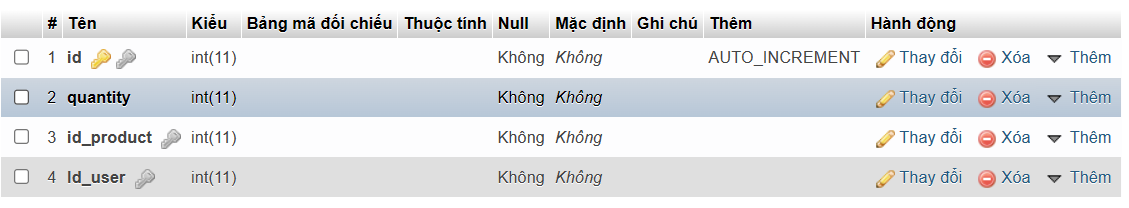
Bảng 2. Bảng sản phẩm

4.5.3.Bảng danh mục sản phẩm:lưu thông tin danh mục sản phẩm



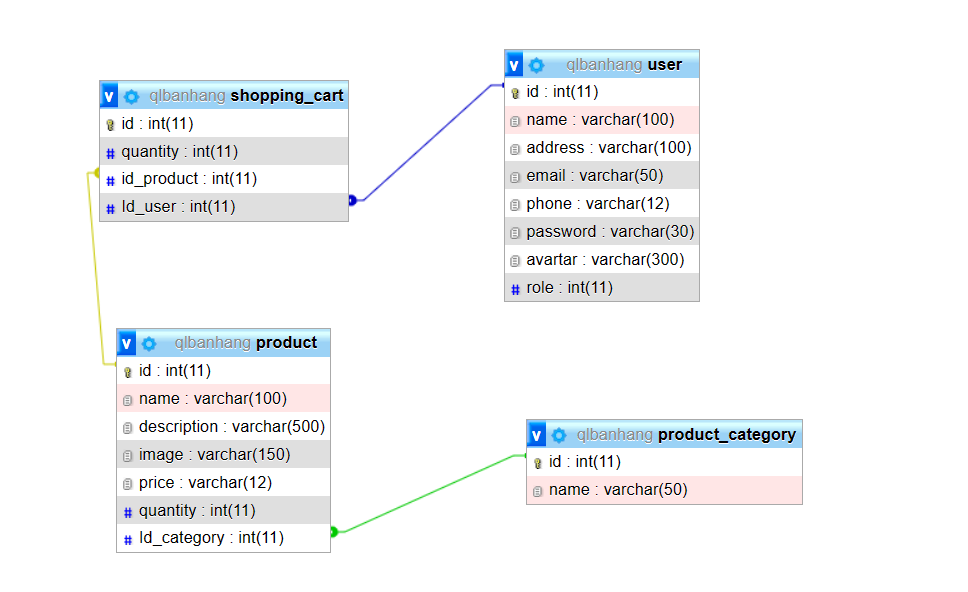
Bảng 3. Bảng danh mục sản phẩm

4.5.4.Bảng giỏ hàng:lưu thông tin sản phẩm khách hàng đã thêm vào giỏ hàng



Bảng 4. Bảng giỏ hàng

4.5.4.Mô hình ERD (Entity-Relationship Diagram)



## 4.6. Yêu cầu về bảo mật

Có tính bảo mật cao, đảm bảo nếu không có được username, password và được cho phép hoạt động của Admin thì không có một User nào có thể thay đổi thông tin về sản phẩm cũng như toàn bộ Website ngoài việc thay đổi thông tin cá nhân của User vì vậy việc quản lý dữ liệu bằng mật khẩu đăng nhập:

+ Tên đăng nhập

+ Mật khẩu đăng nhập

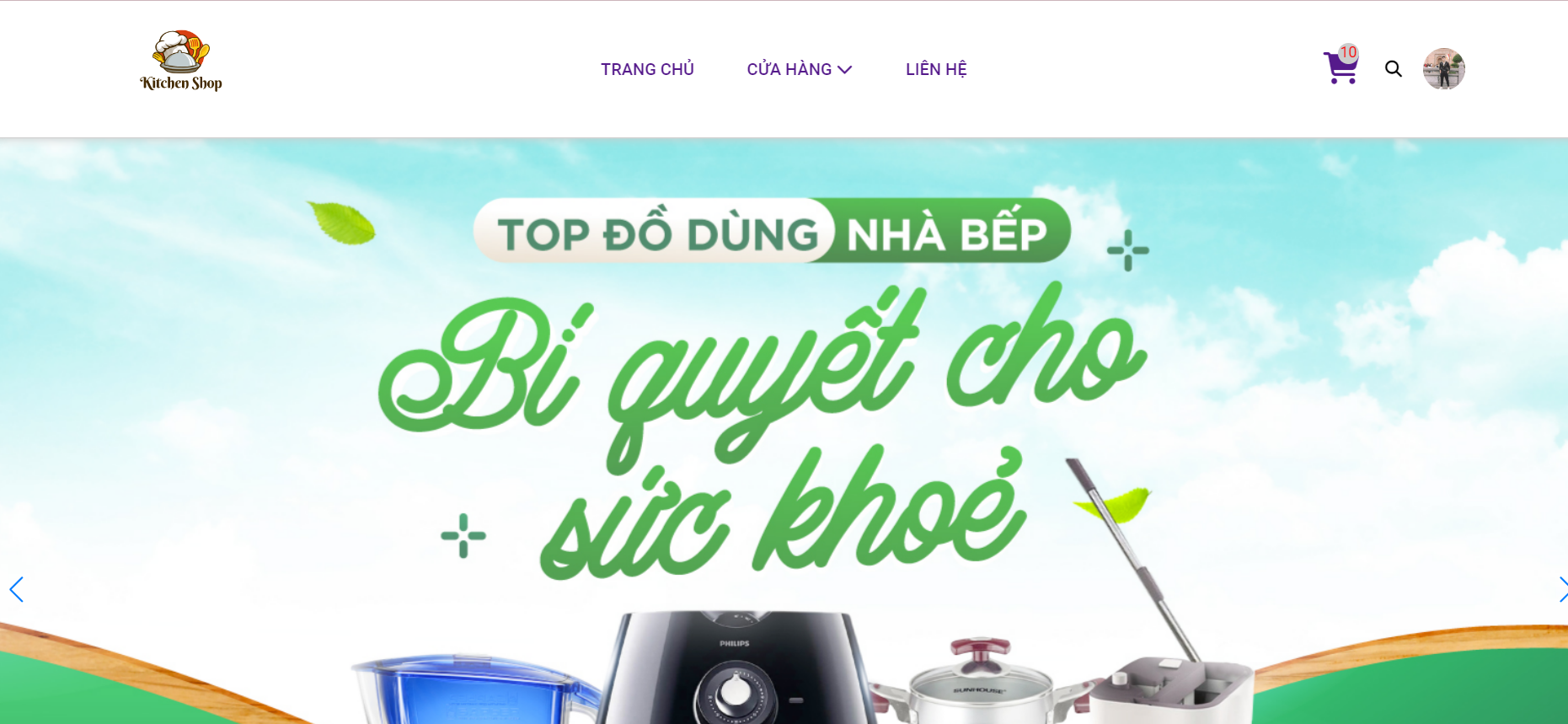
Các thông tin của khách hàng được bảo mật.

Hệ thống quản trị tuyệt đối an toàn, không thể bị truy cập do lỗi của hệ thống.

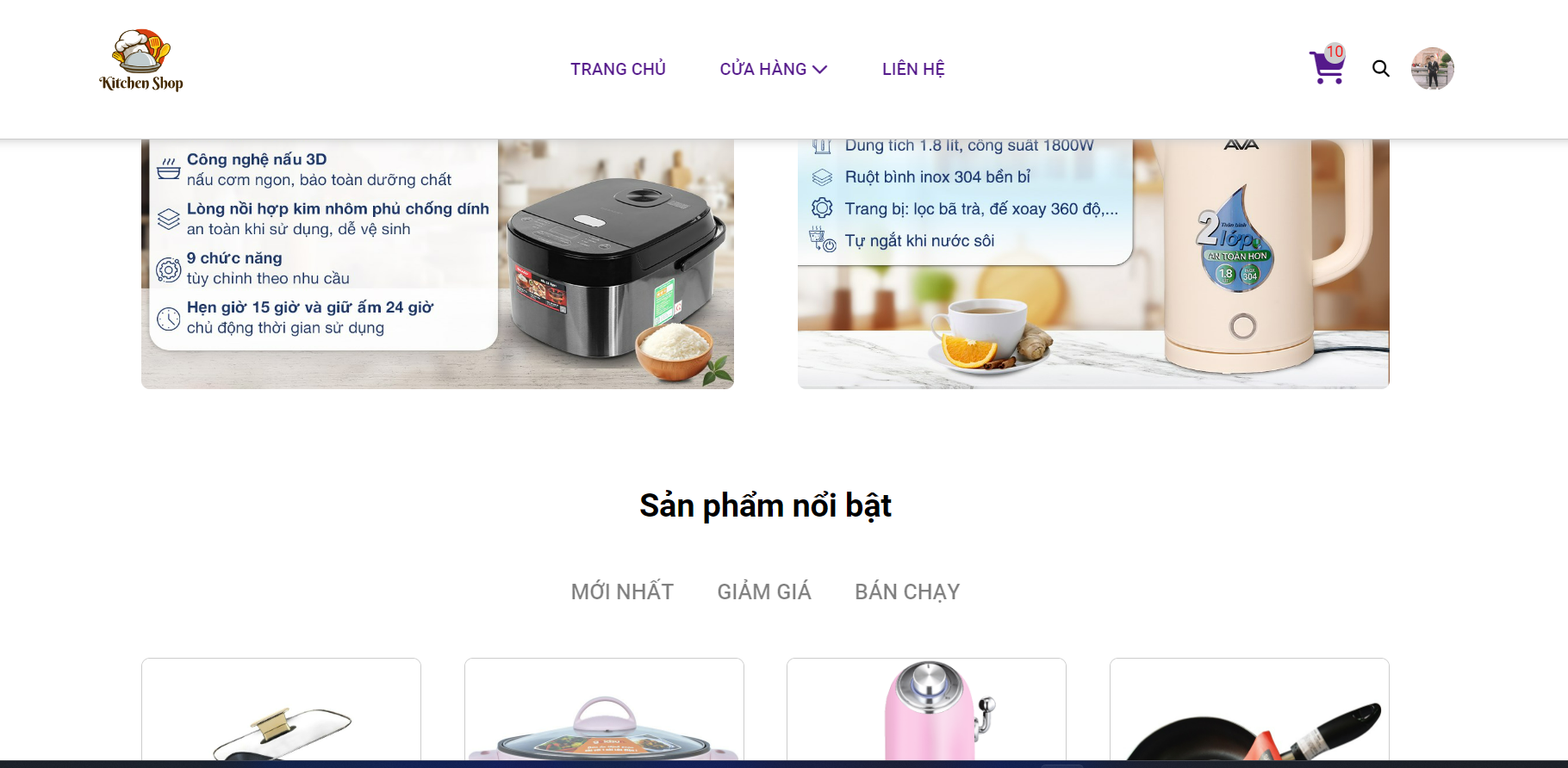
Như vậy, từ các biểu đồ Use Case và biểu đồ hoạt động của hệ thống đã thể hiện được các chức năng của các tác nhân và hệ thống làm việc như thế nào, xuất phát từ hệ thống hiện tại và các yêu cầu về hệ thống, yêu cầu khách hàng, chúng ta nên xây dựng cho trang Website có đầy đủ chức năng của người quản trị và người dùng và phải đảm bảo có được tính bảo mật cao.

## 4.7. Thiết kế giao diện

**4.7.1. Giao diện trang chủ**

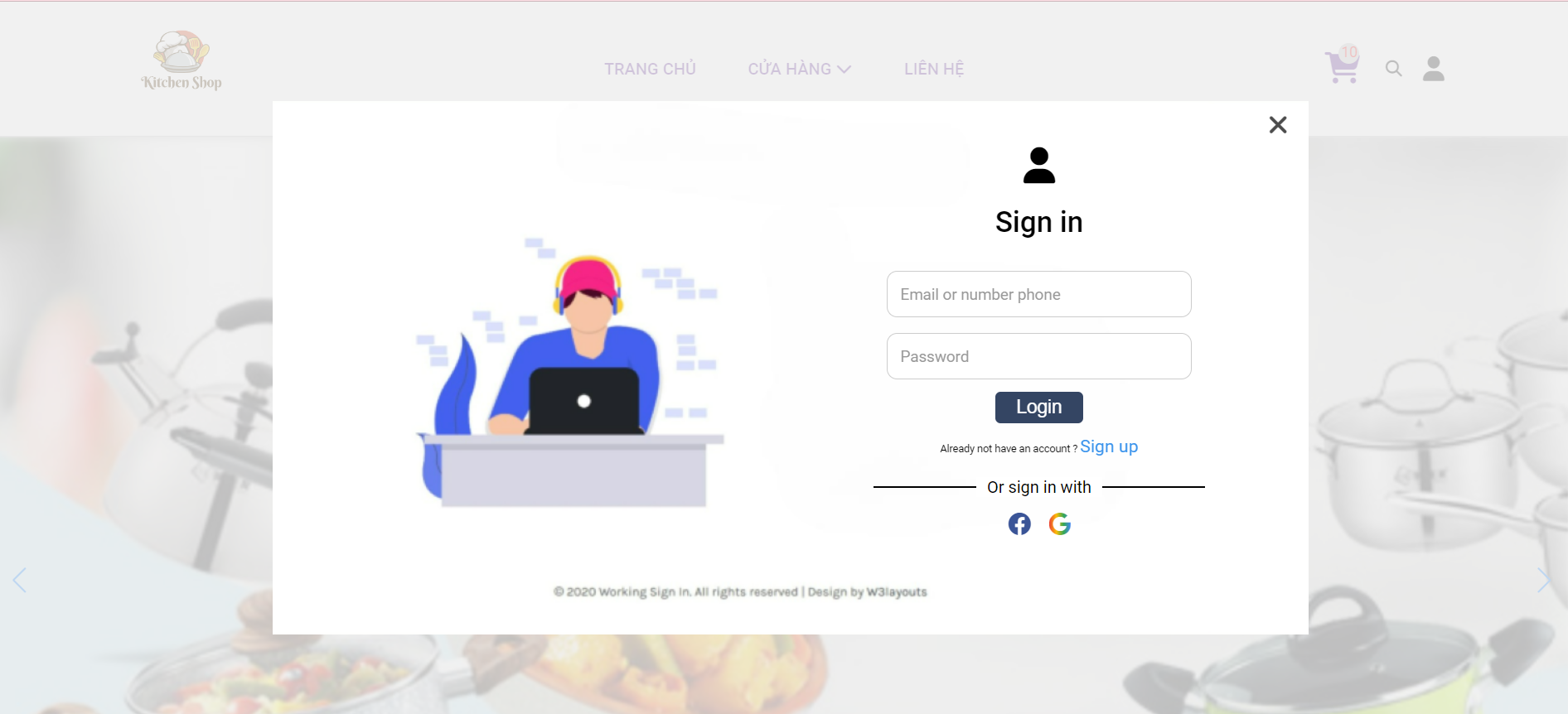


Hình 1. Giao diện trang chủ



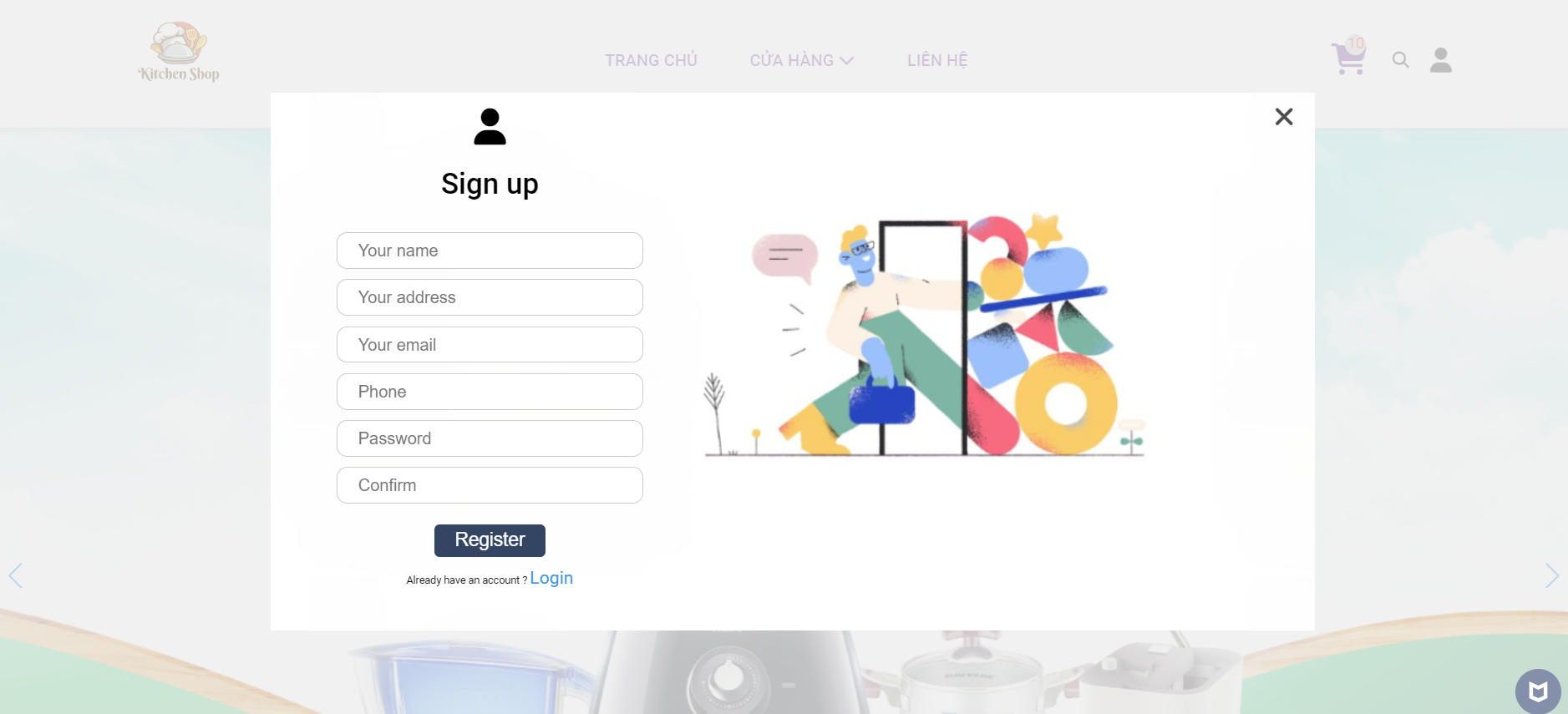
Hình 2. Giao diện trang chủ

### 4.8.2. Giao diện Form đăng nhập



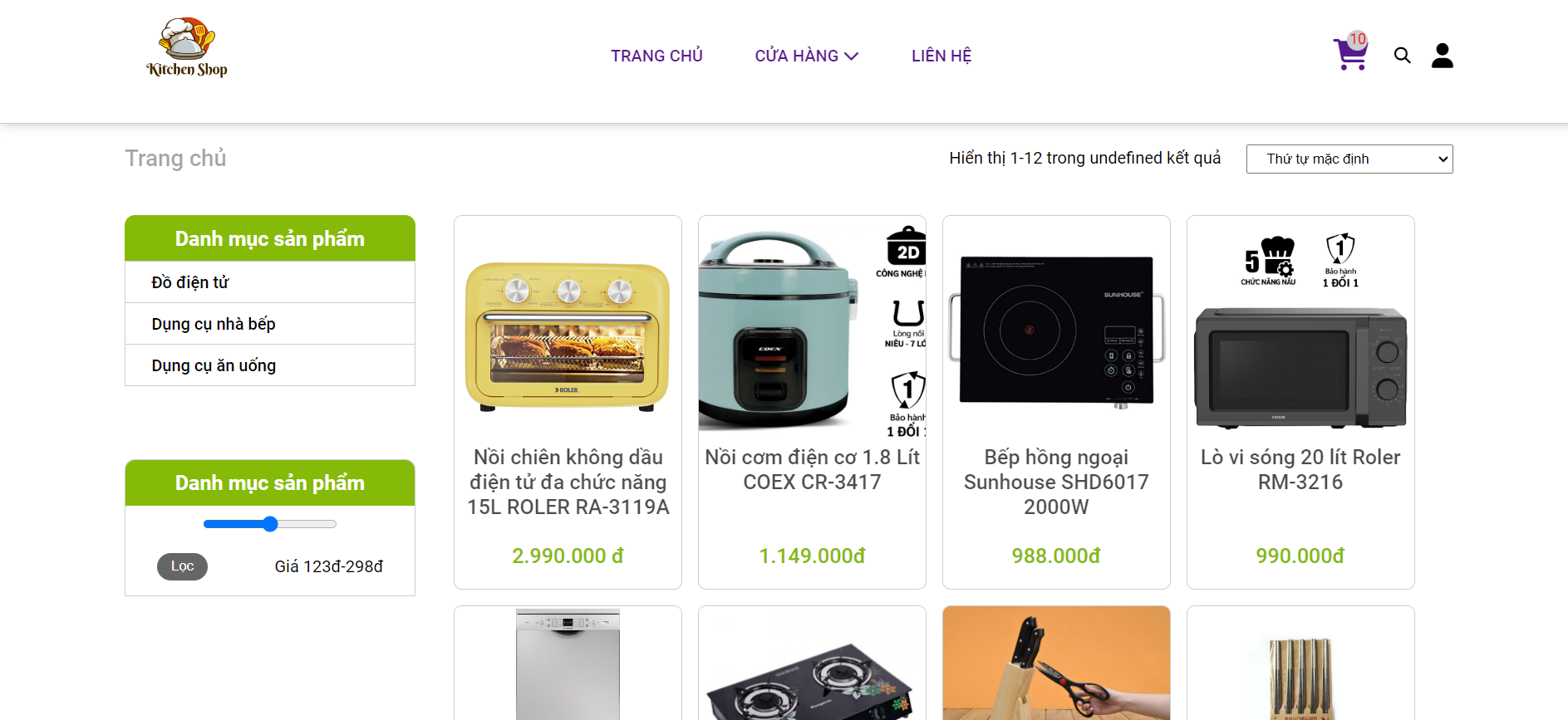
Hình 3. Giao diện đăng nhập

### 4.8.3. Giao diện Form đăng ký



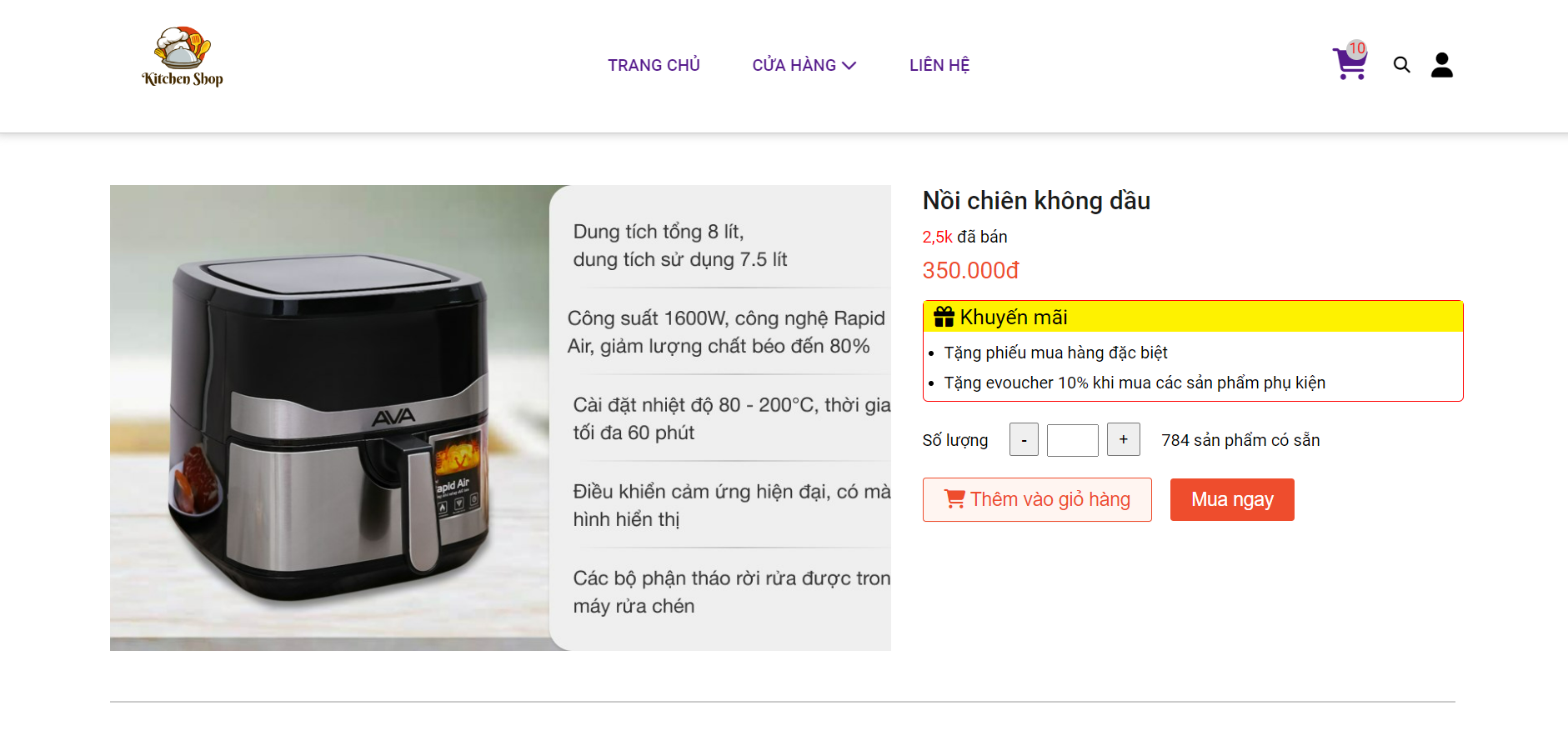
Hình 4. Giao diện đăng ký

**4.8.4. Giao diện cửa hàng**



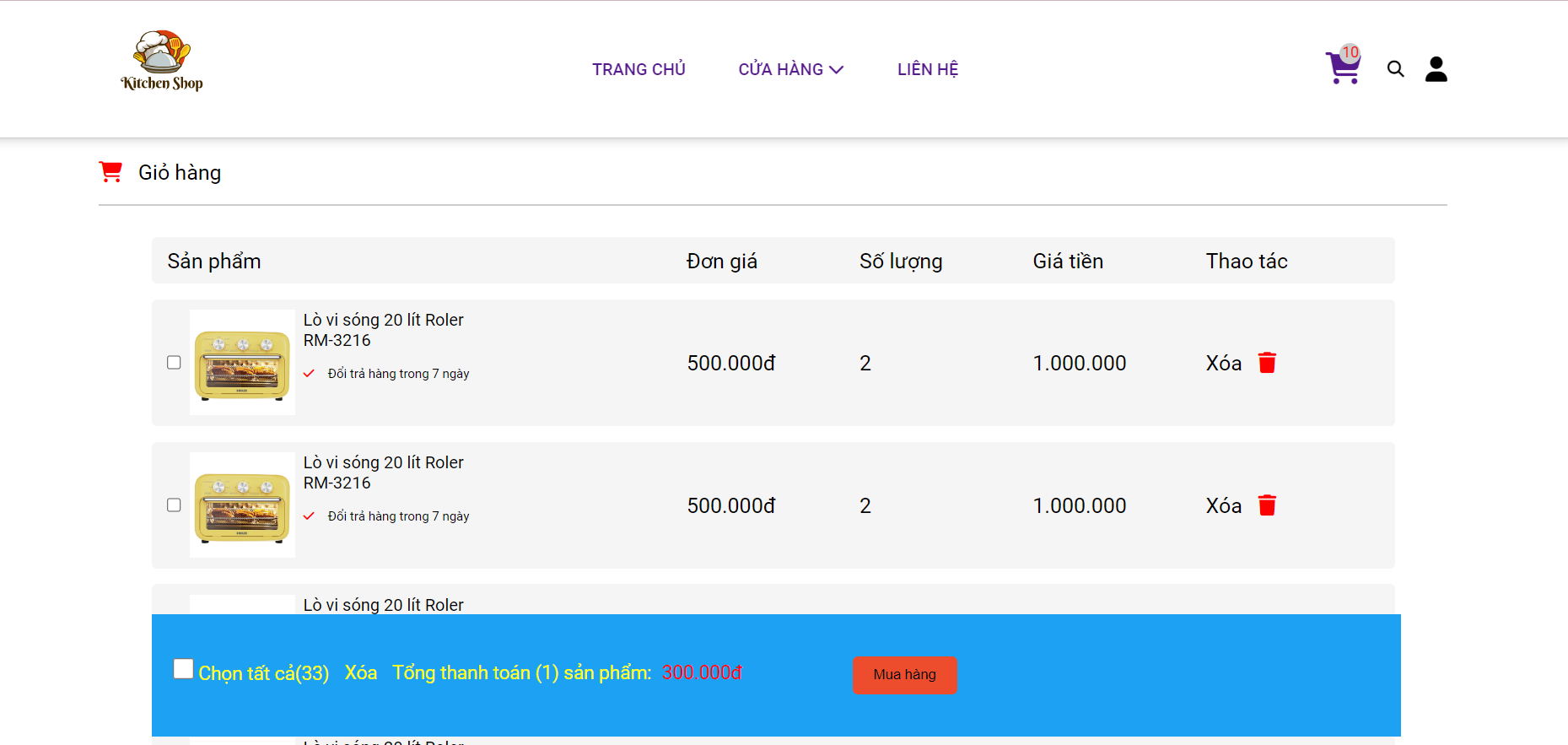
Hình 5. Giao diện cửa hàng

**4.8.5. Giao diện chi tiết sản phẩm**

****

Hình 6. Giao diện chi tiết sản phẩm

**4.8.6. Giao diện giỏ hàng**

****

Hình 7. Giao diện giỏ hàng

**KẾT LUẬN**

Đề tài “**Xây dựng Website bán hàng dụng cụ nhà bếp**” cũng xuất phát từ thực tế ngày nay nhằm tạo một nền tảng cơ sở ban đầu để có thể hỗ trợ thêm cho các bạn muốn thiết kế một Website cho công ty hay một Website cá nhân.

Mặc dù đã có nhiều cố gắng, tìm hiểu các kiến thức đã học, kết hợp tra cứu các tài liệu chuyên nghành nhưng do còn hạn chế về thời gian, khả năng và kinh nghiệm nên không tránh khỏi những thiếu sót nhất định nên đề tài đã hoàn thành ở mức độ sau:

- Tìm hiểu ngôn ngữ lập trình Web PHP & MySQL

- Áp dụng xây dựng ứng dụng thực nghiệm trang **Website kinh doanh trực tuyến**

**Hướng nghiên cứu phát triển:**

* Tìm hiểu sâu hơn về ngôn ngữ PHP & MySQL để có thể đáp ứng nhiều hơn nữa nhu cầu của người sử dụng, phát triển và tối ưu hóa hệ thống,
* Kết hợp ngôn ngữ PHP với các hệ quản trị cơ sở dữ liệu lớn hơn: SQL Server, Oracle…
* Tìm hiểu thêm một số ngôn ngữ, các phần mềm ứng dụng để nâng cao giao diện đồ họa đẹp mắt, thân thiện hơn…
* Xây dựng trang Web quy mô lớn hơn với nhiều ứng dụng, chức năng...

Để hoàn thành đề tài này, một lần nữa em xin chân thành cảm ơn cô ***Điền Thị Hồng Hà*** là người đã quan tâm, giúp đỡ em trong suốt thời gian qua. Em xin chân thành cảm ơn !

# TÀI LIỆU THAM KHẢO

[1]. http://www.php.net, truy cập cuối cùng ngày 10/06/2016

[2]. http://www. mysql.com, truy cập lần cuối ngày 10/06/2016

[3]. http://www.w3school.com, truy cập lần cuối ngày 10/06/2016