[](https://www.studocu.com/vn?utm_campaign=shared-document&utm_source=studocu-document&utm_medium=social_sharing&utm_content=123doc-bao-cao-do-an-mon-lap-trinh-web-bang-ngon-ngu-php-sds)

# LỜI MỞ ĐẦU

Với sự phát triển nhảy vọt của công nghệ thông tin hiện nay, Internet ngày càng giữ vai trò quan trọng trong các lĩnh vực khoa học kĩ thuật và đời sống. Dĩ nhiên các bạn đã được nghe nói nhiều về Internet, nói một cách đơn giản, Internet là một tập hợp máy tính nối kết với nhau, là một mạng máy tính toàn cầu mà bất kì ai cũng có thể kết nối bằng máy PC của họ. Với mạng Internet, tin học thật sự tạo nên một cuộc cách mạng trao đổi thông tin trong mọi lĩnh vực văn hóa, xã hội, chính trị, kinh tế...

Trong thời đại ngày nay, thời đại mà “người người làm Web, nhà nhà làm Web” thì việc có một Website để quảng bá công ty hay một Website cá nhân không còn là điều gì xa xỉ nữa. Thông qua Website khách hàng có thể lựa chọn những sản phẩm mà mình cần một cách nhanh chóng và hiệu quả.

Với công nghệ Word Wide Web, hay còn gọi là Web sẽ giúp bạn đưa những thông tin mong muốn của mình lên mạng Internet cho mọi người cùng xem một cách dễ dàng với các công cụ và những ngôn ngữ lập trình khác nhau. Sự ra đời của các ngôn ngữ lập trình cho phép chúng ta xây dựng các trang Web đáp ứng được các yêu cầu của người sử dụng. PHP (Personal Home Page) là kịch bản trên phía trình chủ (Server Script) cho phép chúng ta xây dựng trang Web trên cơ sở dữ liệu. Với nhiều ưu điểm nổi bật mà PHP và MySQL được rất nhiều người sử dụng.

**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** **kinh** **doanh** **son** **môi** **trực** **tuyến**” 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 phẩm son môi 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 đặt hàng và hình thức thanh toán trực tiếp. 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

MỞ ĐẦU 1

1. Lý do chọn đề tài 1
2. [Mục đích nghiên cứu 1](#_TOC_250081)
3. [Đối tượng của đề tài 1](#_TOC_250080)
4. [Phạm vi nhiên cứu 1](#_TOC_250079)
5. [Phương pháp nghiên cứu 1](#_TOC_250078)
6. Ý nghĩa khoa học và thực tiễn của đề tài 1
7. Bố cục đề tài 2

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT 3](#_TOC_250077)

* 1. [Tìm hiểu ngôn ngữ PHP 3](#_TOC_250076)
     1. [Khái niệm PHP 3](#_TOC_250075)
     2. [Tại sao nên dùng PHP 3](#_TOC_250074)
     3. [Hoạt động của PHP 4](#_TOC_250073)
     4. [Các loại thẻ PHP 4](#_TOC_250072)
     5. [Các kiểu dữ liệu 4](#_TOC_250071)
        1. [Số nguyên 5](#_TOC_250070)
        2. [Số thực 5](#_TOC_250069)
        3. [Xâu 5](#_TOC_250068)
        4. [Mảng 5](#_TOC_250067)
     6. Biến - giá trị 5
        1. Một số biến đã được khai báo sẵn 5
        2. [Phạm vi giá trị 5](#_TOC_250066)
        3. [Tên biến 6](#_TOC_250065)
     7. Các giá trị bên ngoài phạm vi PHP 6
     8. [Hằng 6](#_TOC_250064)
     9. [Biểu thức 7](#_TOC_250063)
     10. [Các cấu trúc lệnh 7](#_TOC_250062)
         1. Các lệnh điều kiện và toán tử 7
         2. [Phát biểu vòng lặp While 8](#_TOC_250061)
         3. [Vòng lặp For 8](#_TOC_250060)
         4. [Vòng lặp do... while 8](#_TOC_250059)
         5. [Vòng lặp while 9](#_TOC_250058)
         6. [Exit 9](#_TOC_250057)
     11. [Hàm 9](#_TOC_250056)
     12. [Các toán tử 10](#_TOC_250055)
     13. [Lớp và đối tượng 10](#_TOC_250054)
     14. [Tham chiếu 10](#_TOC_250053)
     15. [Khai báo và sử dụng Session, Cookie 11](#_TOC_250052)
     16. [MySQL và PHP 13](#_TOC_250051)
  2. Tìm hiểu cơ sở dữ liệu My SQL 15
     1. Giới thiệu cơ sở dữ liệu 15
     2. Mục đích sử dụng cơ sở dữ liệu 15
     3. Các kiểu dữ liệu trong cơ sở dữ liệu MySQL 15
        1. [Loại dữ liệu numeric 15](#_TOC_250050)
        2. [Loại dữ liệu kiểu Date and Time 16](#_TOC_250049)
        3. [Loại dữ liệu String 17](#_TOC_250048)
     4. Các thao tác cập nhật dữ liệu 18
     5. [Các hàm thông dụng trong MySQL 18](#_TOC_250047)
        1. [Các hàm trong phát biểu GROUP BY 18](#_TOC_250046)
        2. [Các hàm xử lí chuỗi 18](#_TOC_250045)
        3. Các hàm xử lí về thời gian 19
        4. Các hàm về toán học 19

[CHƯƠNG 2. PHÂN TÍCH, THIẾT KẾ HỆ THỐNG 20](#_TOC_250044)

* 1. Phân tích yêu cầu đề tài 20
     1. Tóm tắt hoạt động của hệ thống mà dự án sẽ được ứng dụng: 20
     2. Phạm vi dự án được ứng dụng 20
  2. [Xác định yêu cầu của khách hàng 20](#_TOC_250043)
     1. [Hệ thống hiện hành của cửa hàng 20](#_TOC_250042)
     2. [Hệ thống đề nghị 21](#_TOC_250041)
  3. [Yêu cầu giao diện của website 22](#_TOC_250040)
     1. [Giao diện người dùng 22](#_TOC_250039)
     2. [Giao diện người quản trị 22](#_TOC_250038)
  4. [Phân tích các chức năng của hệ thống 22](#_TOC_250037)
     1. [Các chức năng của đối tượng Customer (Khách vãng lai) 23](#_TOC_250036)
        1. [Chức năng đăng ký thành viên 23](#_TOC_250035)
        2. [Chức năng tìm kiếm sản phẩm 24](#_TOC_250034)
        3. [Chức năng xem thông tin sản phẩm 24](#_TOC_250033)
     2. [Các chức năng của đối tượng Member (thành viên) 24](#_TOC_250032)
        1. [Chức năng đăng nhập 24](#_TOC_250031)
        2. [Chức năng đăng xuất 24](#_TOC_250030)
        3. [Chức năng đổi mật khẩu 24](#_TOC_250029)
        4. Chức năng xem hóa đơn đã được lập 25
     3. [Chức năng của Administrator 25](#_TOC_250028)
        1. [Các chức năng quản lý Member 25](#_TOC_250027)
        2. [Các chức năng quản lý Sản Phẩm 25](#_TOC_250026)
        3. [Các chức năng quản trị người dùng 25](#_TOC_250025)
  5. [Biểu đồ phân tích thiết kế hệ thống 26](#_TOC_250024)
     1. [Thuật toán 26](#_TOC_250023)
     2. [Đặc tả Use-case 38](#_TOC_250022)
  6. Cách tổ chức dữ liệu và thiết kế chương trình cho trang Web 43
     1. [Bảng giỏ hàng (giohang ) 44](#_TOC_250021)
     2. [Bảng liên hệ (lienhe) 44](#_TOC_250020)
     3. [Bảng sản phẩm (sanpham) 45](#_TOC_250017)
     4. [Bảng thành viên (thanhvien) 45](#_TOC_250016)
  7. Yêu cầu về bảo mật 45

[CHƯƠNG 3. XÂY DỰNG WEBSITE KINH DOANH SON MÔI TRỰC TUYẾN 46](#_TOC_250015)

* 1. [Thiết kế các bảng CSDL 46](#_TOC_250014)
     1. [Bảng giỏ hàng (giohang) 46](#_TOC_250013)
     2. [Bảng liên hệ (lienhe) 46](#_TOC_250012)
     3. [Bảng sản phẩm (sanpham) 47](#_TOC_250009)
  2. [Thiết kế giao diện 48](#_TOC_250006)
     1. [Giao diện trang chủ 48](#_TOC_250005)
     2. [Giao diện Form đăng nhập 49](#_TOC_250004)
     3. [Giao diện trang giỏ hàng 50](#_TOC_250003)
     4. [Giao diện Form đăng ký thành viên 51](#_TOC_250002)
     5. [Giao diện trang chi tiết sản phẩm 52](#_TOC_250001)
     6. [Giao diện trang quản trị Admin 53](#_TOC_250000)

KẾT LUẬN 54

TÀI LIỆU THAM KHẢO 55

**DANH** **SÁCH** **HÌNH** **VẼ**

Hình 2.1. Biểu đồ Use Case 23

Hình 2.2. Thuật toán đăng nhập vào hệ thống 26

Hình 2.3. Thuật toán thoát ra khỏi hệ thốn 27

Hình 2.5. Thuật toán xem thông tin về sản phẩm 29

Hình 2.6. Thuật toán thêm SP vào giỏ hàng 30

Hình 2.7. Thuật toán đặt hàng 31

Hình 2.8. Thuật toán đổi mật khẩu 32

Hình 2.9. Thuật toán xoá Member 33

Hình 2.10. Thuật toán thêm sản phẩm 34

Hình 2.11. Thuật toán sửa thông tin sản phẩm 35

Hình 2.12. Thuật toán xoá sản phẩm 36

Hình 2.13. Thuật toán xoá User 37

Hình 2.14. Use case tìm kiếm 38

Hình 2.15. Biểu đồ cộng tác chức năng tìm kiếm 38

Hình 2.16. Biểu đồ tuần tự chức năng tìm kiếm*Use-case* *quản* *lý* *sản* *phẩm* 38

Hình 2.17. Use-case quản lý sản phẩm 39

Hình 2.18. Biểu đồ cộng tác chức năng cập nhật 39

Hình 2.19. Biểu đồ tuần tự chức năng thêm sản phẩm mới 40

Hình 2.20. Biểu đồ tuần tự chức năng sửa thông tin sản phẩm 40

Hình 2.21. Biểu đồ tuần tự chức năng xóa sản phẩm 41

Hình 2.22. Biểu đồ tuần tự chức năng đăng nhập 42

Hình 2.23. Biểu đồ tuần tự chức năng đăng ký thành viên 43

Hình 3.1. Bảng giỏ hàng 46

Hình 3.2. Bảng liên hệ 46

Hình 3.3. Bảng loại sản phẩm 46

Hình 3.4. Bảng nhóm sản phẩm 46

Hình 3.5. Bảng sản phẩm 47

Hình 3.6. Bảng thành viên 47

Hình 3.7. Mối quan hệ giữa các bảng CSDL 48

Hình 3.8. Giao diện trang chủ 48

Hình 3.9. Giao diện form đăng nhập 49

Hình 3.10. Giao diện form liên hệ 50

Hình 3.11. Giao diện giỏ hàng 50

Hình3.12. Giao diện form đăng ký thành viên 51

Hình 3.13. Giao diện trang chi tiết sản phẩm 52

Hình 3.14. Giao diện trang quản trị Admin 53

[](https://www.studocu.com/vn?utm_campaign=shared-document&utm_source=studocu-document&utm_medium=social_sharing&utm_content=123doc-bao-cao-do-an-mon-lap-trinh-web-bang-ngon-ngu-php-sds)

### DANH SÁCH CÁC BẢNG BIỂU

Bảng 1. Bảng giohang 44

Bảng 3. Bảng lienhe 44

Bảng 4. Bảng loaisanpham 44

Bảng 5. Bảng nhomsanpham 44

Bảng 6. Bảng sanpham 45

Bảng 7. Bảng thanhvien 45

1. **Lý** **do** **chọn** **đề** **tài**

**MỞ** **ĐẦU**

Trong thời đại ngày nay, thời đại mà “người người làm Web, nhà nhà làm Web” thì việc có một Website để quảng bá công ty hay một Website cá nhân không còn là điều gì xa xỉ nữa. Thông qua Website khách hàng có thể lựa chọn những sản phẩm mà mình cần một cách nhanh chóng và hiệu quả.

Website với mục đích cung cấp cho khách hàng các thông tin chính xác về các sản phẩm son môi 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.

### Mục đích nghiên cứu

* + Vận dụng tổng hợp những kiến thức đã học để tiến hành xây dựng Xây dựng Website kinh doanh son môi trực tuyến.
  + Đáp ứng nhu cầu mua bán ngày càng phát triển của xã hội qua đó thúc đẩy phát triển buôn bán.

### Đối tượng của đề tài

Có 2 đối tượng sử dụng cơ bản là thành viên và nhà quản trị:

* *Thành* *viên:* Qua Website, khách hàng có thể xem thông tin, lựa chọn những sản phẩm ưa thích ở mọi nơi thậm chí ngay trong phòng làm việc của mình.
* *Nhà* *quản* *trị*: Nhà quản trị có toàn quyền sử dụng và cập nhật sản phẩm, hoá đơn, quản lý khách hàng, đảm bảo tính an toàn cho Website.

### Phạm vi nhiên cứu

* + 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

### Phương pháp nghiên cứu

* + Nghiên cứu lý thuyết.
  + Phân tích thiết kế hệ thống chương trình cho phù hợp.
  + Xây dựng chương trình theo những yêu cầu đã đặt ra.

1. **Ý** **nghĩa** **khoa** **học** **và** **thực** **tiễn** **của** **đề** **tài**
   * Giúp nhà quản trị dễ dàng hơn trong việc quản lý sản phẩm, khách hàng.
   * Nhà quản trị có thể cập nhật các sản phẩm một cách nhanh nhất giúp cho hoạt động của mình trở nên tốt hơn.

## 1

1. **Bố** **cục** **đề** **tài**

Nội dung đồ án gồm các thành phần sau:

MỞ ĐẦU

* + Chương 1. Cơ sở lý thuyết
  + Chương 2. Phân tích, thiết kế hệ thống
  + Chương 3. Xây dựng Website kinh doanh son môi trực tuyến KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

TÀI LIỆU THAM KHẢO

[](https://www.studocu.com/vn?utm_campaign=shared-document&utm_source=studocu-document&utm_medium=social_sharing&utm_content=123doc-bao-cao-do-an-mon-lap-trinh-web-bang-ngon-ngu-php-sds)2

# CHƯƠNG 1. CƠ SỞ LÝ THUYẾT

### Tìm hiểu ngôn ngữ PHP

### 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).

### 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.

## 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:

Yêu cầu URL Máy chủ

HTML Web

HTML

Gọi mã kịch bản

PHP

Máy khách

hàng

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 tiêu 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.

### 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).*

### 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.

## 4

### Số nguyên

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

$a=12345;

$a=-456;

### Số thực

Ví dụ: $a=2. 123;

$b=3. 1e3;

### Xâu

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

### Mảng

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

### 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..

### Mảng 2 chiều

Ví dụ: $a[1]=$f;

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

$a[“abc”][2]=$f;

...

* + 1. **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.

* + - 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?>...**

### 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;

## 5

$b=$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().

### 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”

* + 1. **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

### 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');

## 6

### Biểu thức

define('dbname', 'minhcanh); define('username', 'root');

define('pass', '');

?>

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.

### Các cấu trúc lệnh

* + - 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* *kiệ*n

|  |  |  |  |
| --- | --- | --- | --- |
| 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 |

## 7

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**

**}**

### 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

### 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

### 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...

## 8

### Vòng lặp while

**While(condition)**

**{**

### expression;

**}**

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

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

### 6. Exit

Thoát khỏi các vòng lặp hay phát biểu điều khiển nào đó.

### Hàm

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

### \* Tham trị

Ví dụ: function takes\_array($input)

{

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

}

### \* Tham biến

Function add\_some\_extra (&$string)

{$string. =’and somthing extra’; }

**\*** **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”;

}

* **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.

### 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.

## 9

### 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”;

### 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

### 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;

## 10

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.

### 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

?>

## 11

[](https://www.studocu.com/vn?utm_campaign=shared-document&utm_source=studocu-document&utm_medium=social_sharing&utm_content=123doc-bao-cao-do-an-mon-lap-trinh-web-bang-ngon-ngu-php-sds)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']))

## 12

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 ]]]]])

### 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:

lieu)".

<?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

mysql\_error());

?>

## 13

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.

## 14

* 1. **Tìm** **hiểu** **cơ** **sở** **dữ** **liệu** **My** **SQL**
     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.](http://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.

* + 1. **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…

* + 1. **Các** **kiểu** **dữ** **liệu** **trong** **cơ** **sở** **dữ** **liệu** **MySQL**

### Loại dữ liệu numeric

## 15

*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* *độ*ng

|  |  |  |  |
| --- | --- | --- | --- |
| 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 |

### 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* *nguyê*n

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

## 16

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)

### 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* *liệ*u 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. |

## 17

* + 1. **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]

### Các hàm thông dụng trong MySQL

### 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.

### 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

## 18

* 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.
  + - 1. **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
  + - 1. **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.

19

# CHƯƠNG 2. PHÂN TÍCH, THIẾT KẾ HỆ THỐNG

* 1. **Phân** **tích** **yêu** **cầu** **đề** **tài**
     1. **Tóm** **tắt** **hoạt** **động** **của** **hệ** **thống** **mà** **dự** **án** **sẽ** **được** **ứng** **dụng:**
* Là một Website chuyên bán các loại sản phẩm về son môi
* 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
* Các sản phẩm được sắp xếp hợp lý. Vì vậy người dùng sẽ có cái nhìn tổng quan về tất cả các sản phẩm hiện có.
* 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.
* Người dùng sẽ chọn các hình thức vận chuyển, thanh toán hàng do hệ thống đã định.
  + 1. **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.

### Xác định yêu cầu của khách hàng

### 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…

## 20

### 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*).
* Giao diện người quản trị.
  + Đơn giản, dễ quản lý và không thể thiếu các mục như:
    - Quản lý sản phẩm.
    - Quản lý hoá đơn.
    - Quản lý khách hàng.
    - Quản lý đăng nhập.
    - Quản lý mọi thay đổi của website.

*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 loại Son: Có thể đưa ra đựơc một cái nhìn tổng quát, sâu rộng về sản phẩm.

*Về* *bảo* *mật:*

* Người Admin có toàn quyền giữ bảo mật cho website bằng mật khẩu riêng.
* Quản lý User & Password của khách hàng an toàn: Thông tin của khách hàng được bảo mật

*Về* *hệ* *thống*

* Phần cứng: PC bộ vi xử lý pentium III, ram 128 trở lên, ổ cứng 10G trở lên.
* Hệ điều hành: Windown XP trở lên
* Phần mềm hỗ trợ: XAMPP, mySQL.

*Lựa* *chọn* *giải* *pháp*

* Chương trình sử dụng ngôn ngữ PHP và cơ sở dữ liệu MySQL.

## 21

* 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.
  + Dreamweaver: Tạo form, thiết kế giao diện.

### Yêu cầu giao diện của website

### 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.
* Sản phẩm của cửa hàng: sản phẩm đang có, sản phẩm bán chạy, sản phẩm mới nhất, sản phẩm sắp ra mắt…

### Giao diện người quản trị:

* Giao diện đơn giản, dễ quản lý dữ liệu.
* Phải được bảo vệ bằng User & Password riêng của Admin.

### Phân tích các chức năng của hệ thống

Các tác nhân của hệ thống gồm có:

* Admin:là thành viên quản trị của hệ thống, có các quyền và chức năng như: tạo các tài khoản, quản lý sản phẩm, quản trị người dùng, quản lý hoá đơn…
* Member (thành viên) có chức năng: Đăng kí, đăng nhập, tìm kiếm, xem, sửa thông tin cá nhân, xem giỏ hàng, đặt hàng, xem thông tin về các hóa đơn đã lập.
* Customer (khách hàng) có chức năng: Đăng kí, tìm kiếm, xem thông tin sản phẩm.

## 22

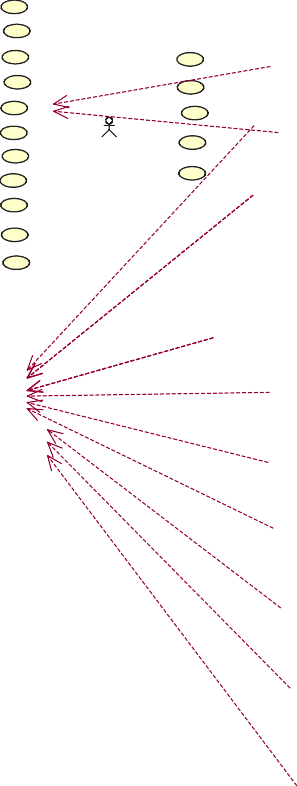




Customer

TKTT san pham

Xem TT san pham



Chuyen HD dat hang -> Da Mua

Liet ke hoa don

Xem gio hang

Xoa Thanh Vien

Them san pham vào gio hang

Xoa Sp

Member

Dang ki thanh vien

Sua TT Sp

Dat hang

Admin

Them Hang

Dang nhap

Them SP

Dang xuat

Doi mat khau

Sua TT thanh vien

LS mua hang

Hình 2.1. Biểu đồ Use Case

### Các chức năng của đối tượng Customer (Khách vãng lai)

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.

### 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 vãng lai 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.

## 23

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

### 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.

### 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.

### Các chức năng của đối tượng Member (thành viên)

Thành viên có tất cả các chức năng giống như khách vãng lai và còn có thêm một số chức năng khác như:

### Chức năng đăng nhập

+ 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.

### Chức năng đăng xuất

+ Description: Cho phép người dùng đăng xuất khỏi hệ thống khi không còn nhu cầu sử dụng hệ thống.

+ Input: Người dùng click vào nút thoát trên hệ thống.

+ Process: Tiến hành xóa session lưu thông tin đăng nhập để dừng phiên làm việc của tài khoản Member trong hệ thống.

+ Output: Quay trở lại trang hiện hành. Ẩn hết các chức năng của Member.

### Chức năng đổi mật khẩu

+ Description: Cho phép thay đổi mật khẩu.

+ Input: Người dùng điền thông tin mật khẩu cũ và mới.

+ Process: Kiểm tra thông tin người dùng nhập vào. Nếu đúng thì cập nhật các thông tin mới, ngược lại thì không.

+ Output: Hiển thị thông báo thành công nếu thông tin nhập vào chính xác hoặc hiển thị thông báo yêu cầu nhập lại nếu thông tin không chính xác

## 24

* + - 1. **Chức** **năng** **xem** **hóa** **đơn** **đã** **được** **lập**

+ Description: Cho phép Member xem các hóa đơn mua hàng của mình.

+ Input: Click chọn lịch sử giao dịch trên menu của Member.

+ Process: Gọi trang hiển thị thông tin hóa đơn đã được lập của Member đó

+ Output: Hiển thị thông tin chi tiết các hóa đơn.

### Chức năng của Administrator

### Các chức năng quản lý Member

* + - * + *Chức* *năng* *xoá* *Member:*

+ Description: Giúp Admin có thể xóa Member ra khỏi CSDL.

+ Input: Chọn Member cần xóa.

+ Process: Lấy các thông tin của Member và hiển thị ra màn hình để chắc chắn rằng Admin xóa đúng Member cần thiết.

+ Output: Load lại danh sách Member để biết được đã xoá thành công Member ra khỏi CSDL

### Các chức năng quản lý Sản Phẩm

* + - * + *Chức* *năng* *chỉnh* *sửa* *thông* *tin* *sản* *phẩm:*

+ Description: Giúp Admin thay đổi thông tin của sản phẩm đã có trong CSDL.

+ Input: Admin nhập thông tin mới của sản phẩm.

+ Process: Cập nhật thông tin mới cho sản phẩm.

+ Output: Hiển thị thông báo đã cập nhật sản phẩm.

* + - * + *Chức* *năng* *xoá* *sản* *phẩm:*

+ Description: Giúp Admin có thể xoá sản phẩm

+ Input: Chọn sản phẩm cần xoá

+ Process: Xoá trong CSDL

+ Output: Load lại danh sách sản phẩm

* + - * + *Chức* *năng* *thêm* *sản* *phẩm:*

+ Description: Giúp Admin có thể thêm sản phẩm mới.

+ Input: Admin nhập vào những thông tin cần thiết của sản phẩm mới.

+ Process: Kiểm tra xem những trường nào không được để trống. Nếu tất cả đều phù hợp thì thêm vào database. Ngược lại thì không thêm vào database.

+ Output: Load lại danh sách sản phẩm để xem sản phẩm mới đã được thêm vào CSDL

### Các chức năng quản trị người dùng

* + - * + *Chức* *năng* *xoá* *với* *người* *dùng:*

+ Description: Giúp Admin có thể xoá người dùng.

+ Input: Tìm User cần xoá và chọn biểu tượng xoá .

+ Process: Xoá User ra khỏi CSDL

+ Output: Load lại danh sách người dùng để xem thông tin vừa thay đổi.

## Các Chức Năng Đối Với Hóa Đơn Đặt Hàng

25

* + - * + *Chức* *năng* *xem* *thông* *tin* *chi* *tiết* *các* *đơn* *đặt* *hàng* (đang chờ được xử lý, đang xử lý, đã hoàn thành, hoặc hoá đơn bị huỷ bỏ):

+ Description: Xem chi tiết đơn đặt hàng có trong CSDL

+ Input: Chọn tên hoá đơn hoặc tên khách hàng của hoá đơn đó.

+ Process: Lấy toàn bộ thông tin chi tiết của đơn đặt hàng có trong CSDL.

+ Output: Hiển thị chi tiết thông tin trong đơn đặt hàng.

* + - * + *Chức* *năng* *chuyển* *các* *đơn* *đặt* *hàng* *chưa* *thanh* *toán* *thành* *hóa* *đơn* *đã* *thanh* *toán*:

+ Description: Sau khi khách hàng xác nhận đặt hàng, hoá đơn sẽ được lưu vào trong CSDL. Admin gọi điện cho khách hàng để xác nhận thông tin. Nếu đúng hoá đơn sẽ được chuyển sang bộ phận giao hàng, sau khi thực hiện giao dịch thành công hoá đơn được hoàn tất, trong quá trình xử lý hoá đơn khách hàng có thể huỷ bỏ hoá đơn đặt hàng đó.

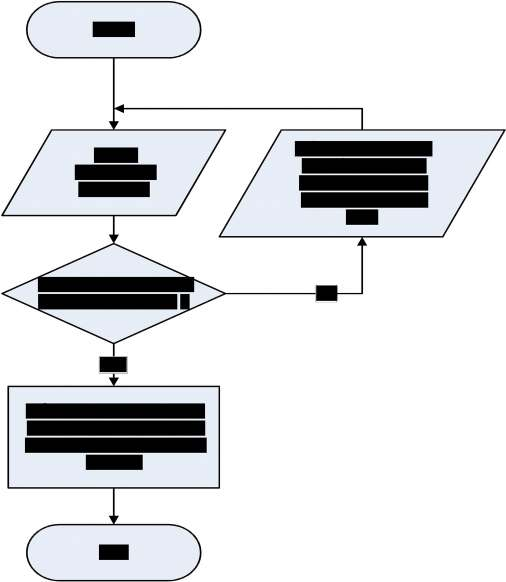
+ Input: Admin chọn những hoá đơn cần xử lý

+ Process: Xứ lý thay đổi của hoá đơn.

+ Output: Load lại danh sách hoá đơn để xem lại sự thay đổi của hoá đơn.

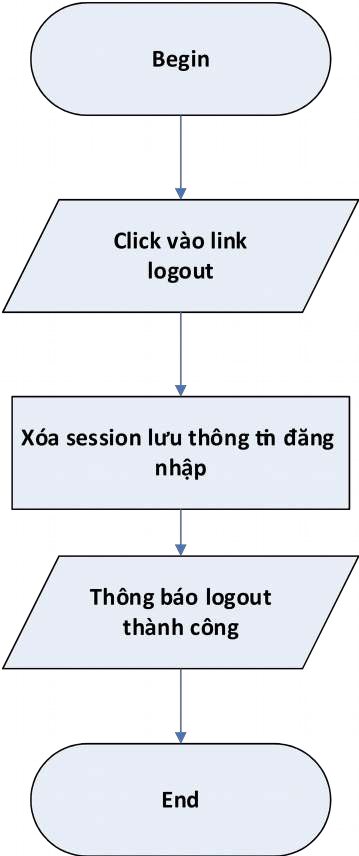
### Biểu đồ phân tích thiết kế hệ thống

### Thuật toán

* Đăng nhập

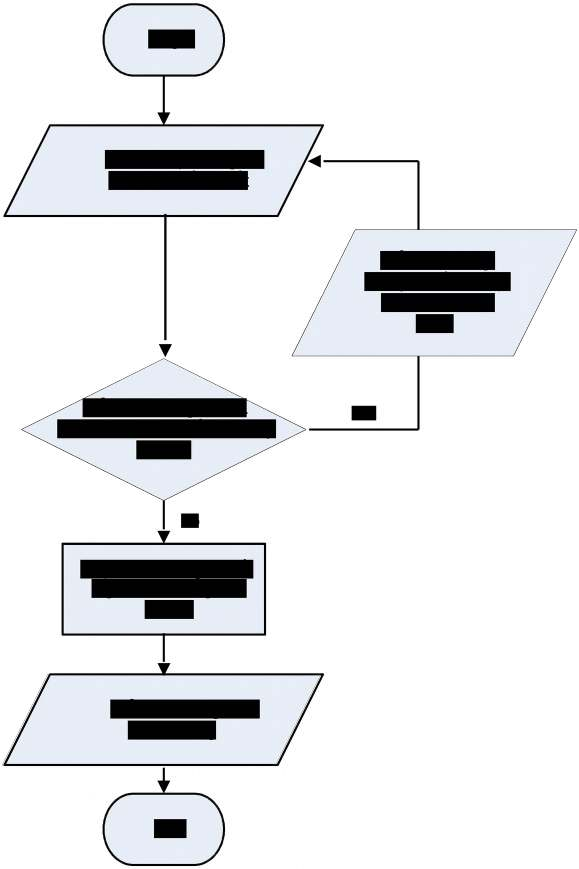
Hình 2.2. Thuật toán đăng nhập vào hệ thống

## 26

* + Đăng xuất

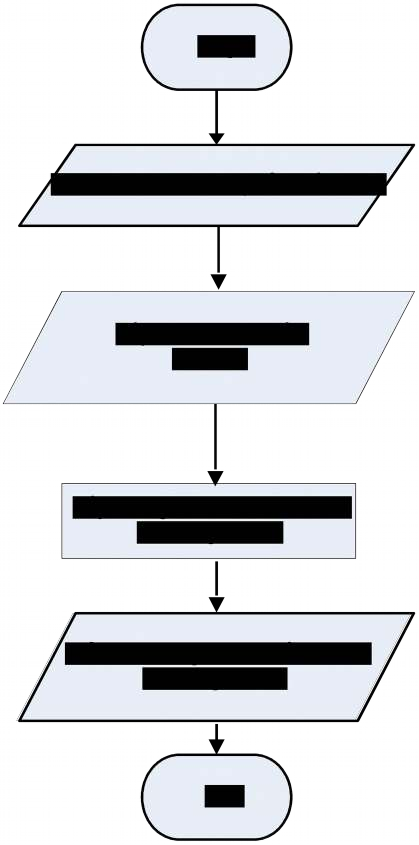
Hình 2.3. Thuật toán thoát ra khỏi hệ thốn

## 27

* + Đăng kí thành viên

Hình 2.4. Thuật toán đăng kí thành viên

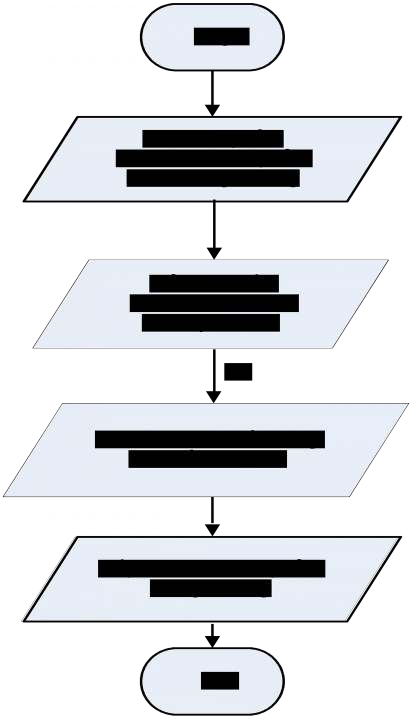
## 28

Xem thông tin sản phẩm

Hình 2.5. Thuật toán xem thông tin về sản phẩm

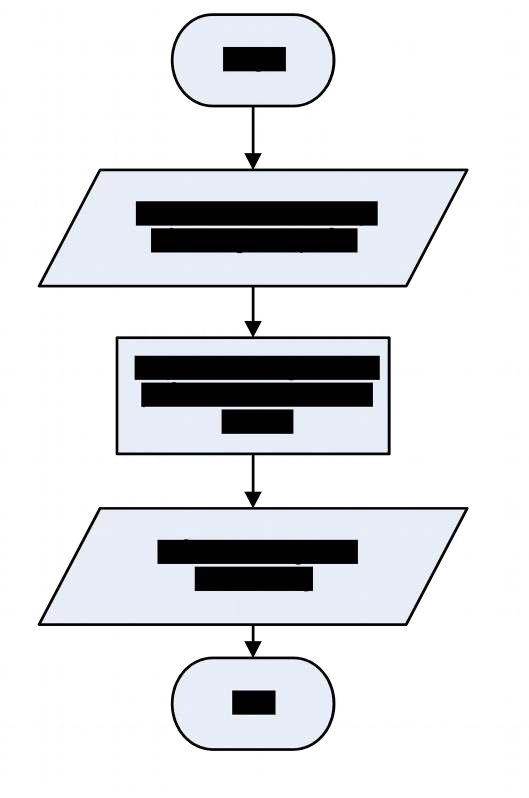
## 29

* + Thêm sản phẩm vào giỏ hàng



Hình 2.6. Thuật toán thêm SP vào giỏ hàng

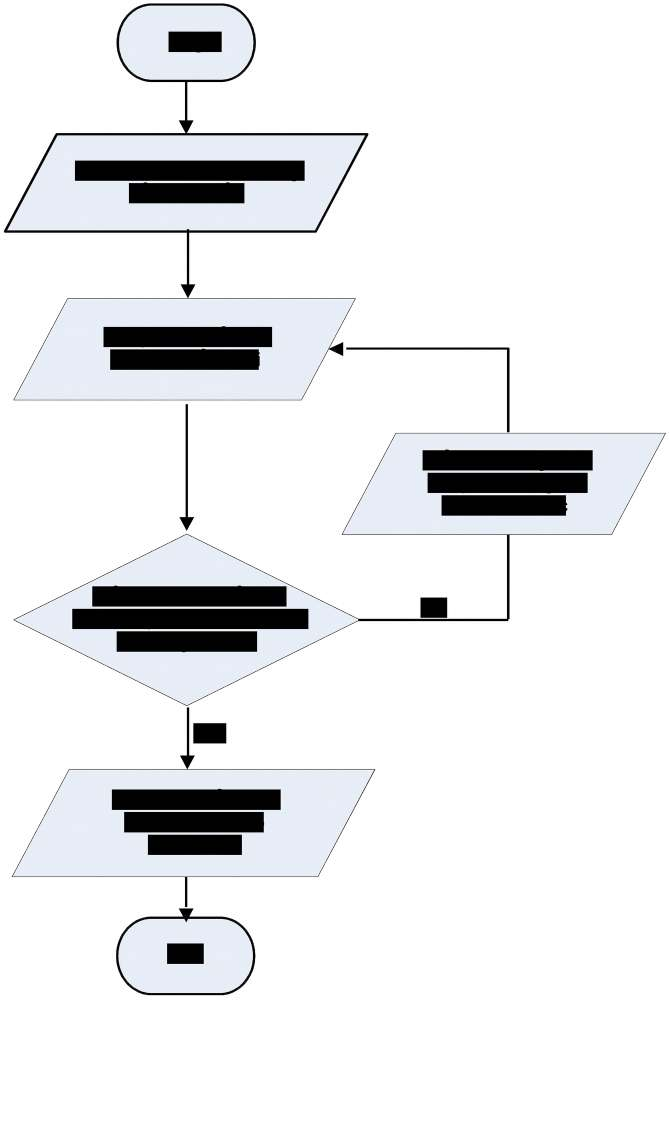
## 30

* + - Đặt hàng

Hình 2.7. Thuật toán đặt hàng

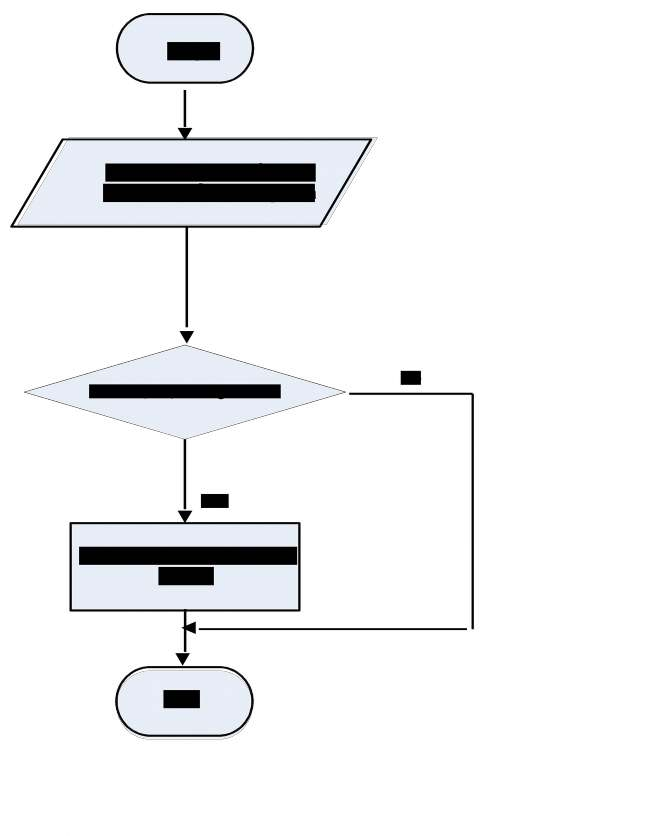
## 31

* + Đổi mật khẩu



Hình 2.8. Thuật toán đổi mật khẩu

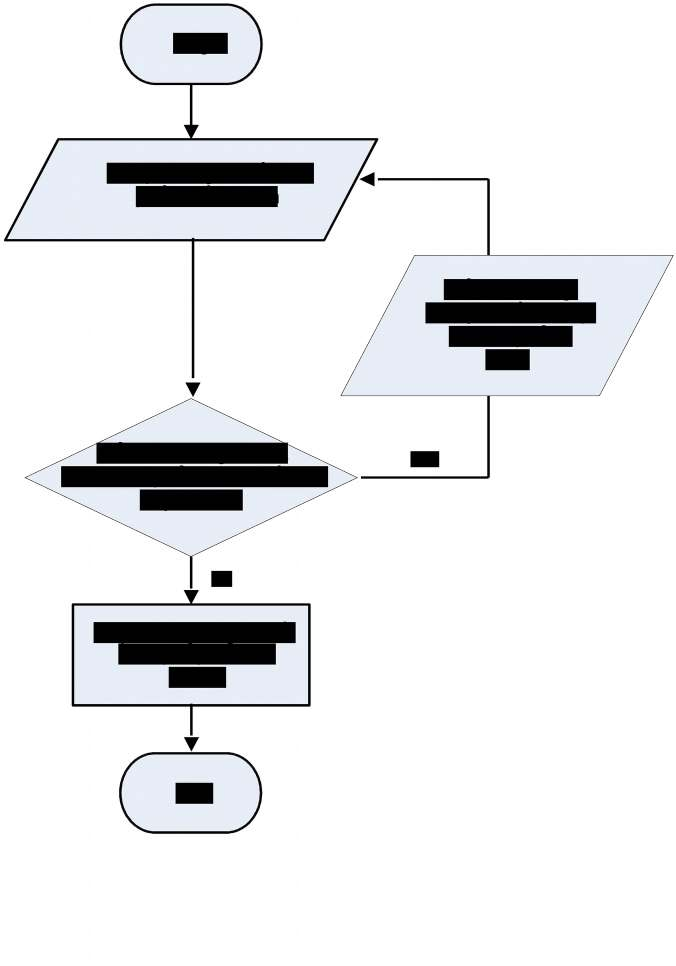
## 32

* + Xoá Member

Hình 2.9. Thuật toán xoá Member

## 33

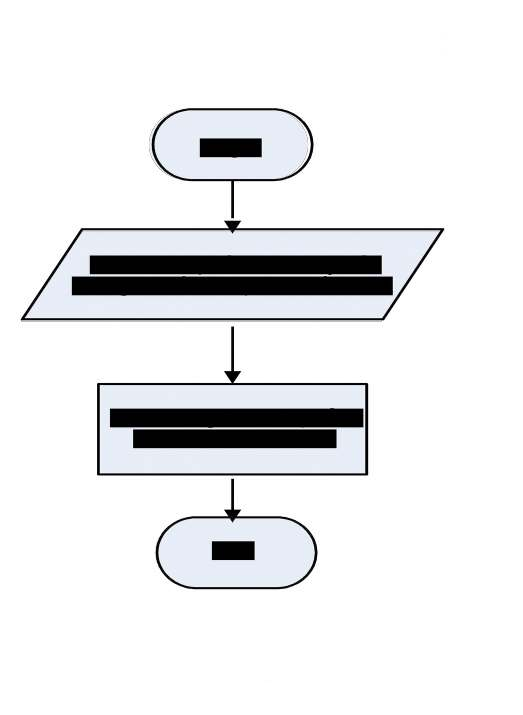
* + Thêm sản phẩm



Hình 2.10. Thuật toán thêm sản phẩm

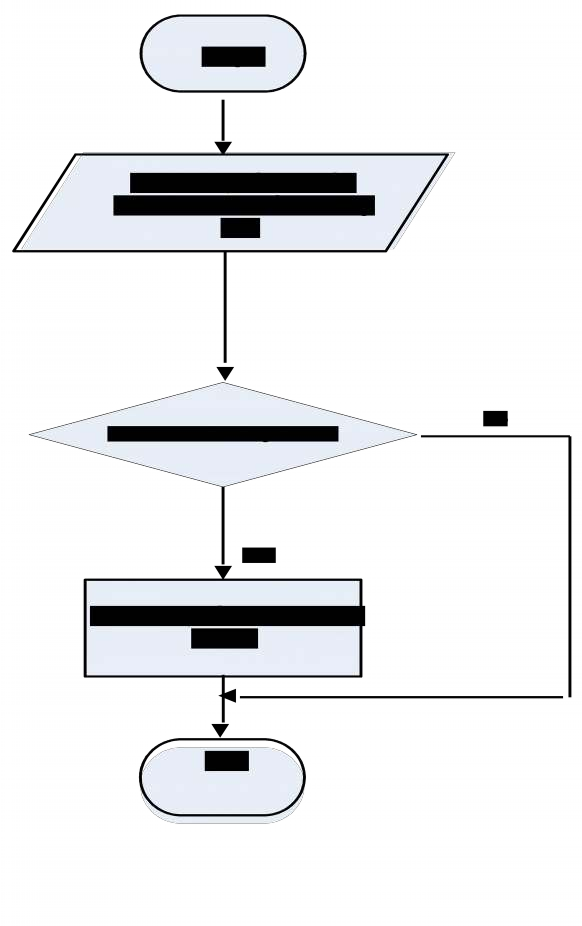
## 34

* + Sửa thông tin sản phẩm



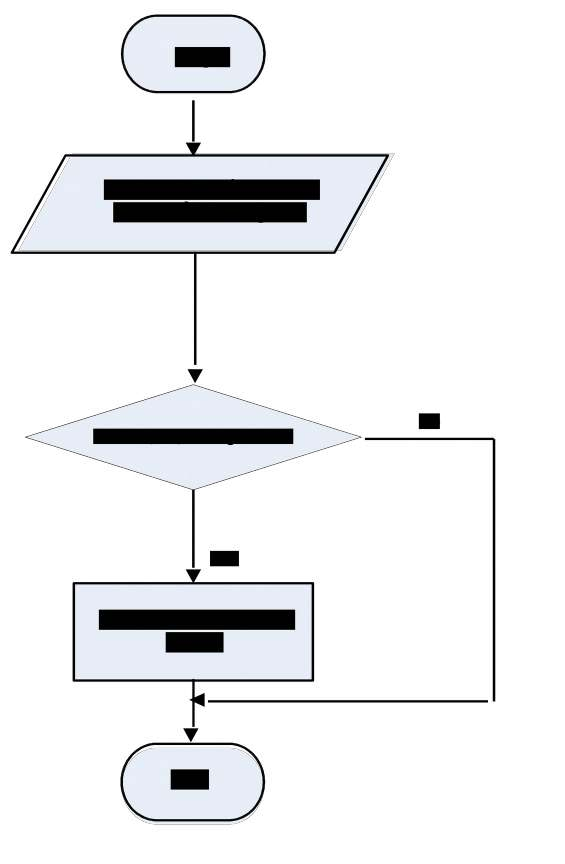
Hình 2.11. Thuật toán sửa thông tin sản phẩm

## 35

* + Xoá sản phẩm

Hình 2.12. Thuật toán xoá sản phẩm

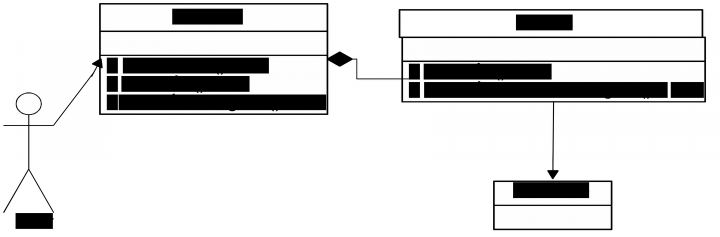
## 36

* + Xoá User

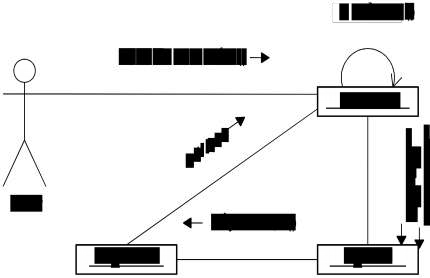
Hình 2.13. Thuật toán xoá User

## 37

### Đặc tả Use-case

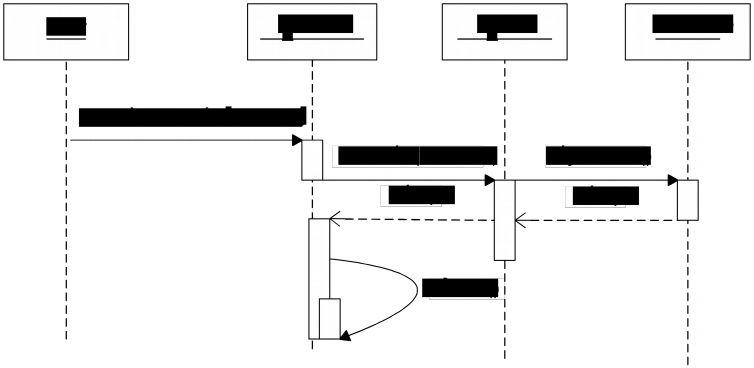
* *Use-case* *tìm* *kiếm*

Hình 2.14. Use case tìm kiếm



Hình 2.15. Biểu đồ cộng tác chức năng tìm kiếm

* Lớp biên: Interface
* Lớp điều khiển: Control
* Lớp thực thể: Database

Dưới đây là biểu đồ tuần tự.

Hình 2.16. Biểu đồ tuần tự chức năng tìm kiếm

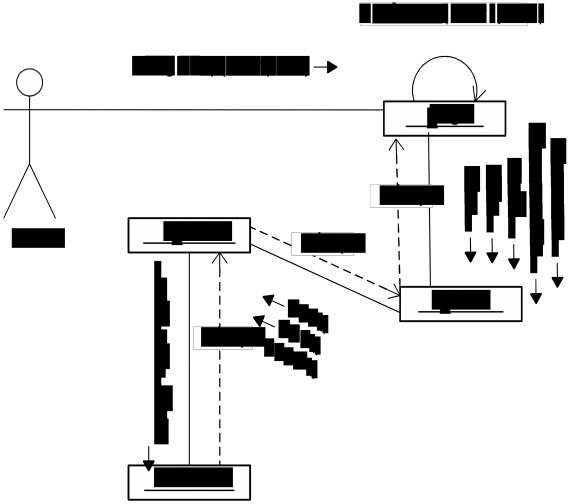
## 38

*Use-case* *quản* *lý* *sản* *phẩm*



Hình 2.17. Use-case quản lý sản phẩm

* Lớp biên: Login, Interface
* Lớp điều khiển: Control
* Lớp thực thể: Database



Hình 2.18. Biểu đồ cộng tác chức năng cập nhật

## 39

### Chức năng thêm mới sản phẩm

Hình 2.19. Biểu đồ tuần tự chức năng thêm sản phẩm mới

### Chức năng sửa thông tin sản phẩm

Hình 2.20. Biểu đồ tuần tự chức năng sửa thông tin sản phẩm

## 40

### Chức năng xóa sản phẩm:

Hình 2.21. Biểu đồ tuần tự chức năng xóa sản phẩm

Tương tự đối với Use-case quản lý người dùng hệ thống và quản lý khách hàng (với các chức năng thêm, sửa, xóa).

## 41

### Chức năng đăng nhập:

Hình 2.22. Biểu đồ tuần tự chức năng đăng nhập

## 42

### Chức năng đăng ký thành viên:

Hình 2.23. Biểu đồ tuần tự chức năng đăng ký thành viên

* 1. **Cách** **tổ** **chức** **dữ** **liệu** **và** **thiết** **kế** **chương** **trình** **cho** **trang** **Web**

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 bảng cơ sở dữ liệu chính trong toàn bộ cơ sở dữ liệu:

## 43

### Bảng giỏ hàng (giohang )

*Dùng* *để* *lưu* *thông* *tin* *về* *giỏ* *hàng* *của* *khách* *hàng*

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Null | Chú thích |
| **Id\_giohang** | Int(10) | No | Mã giỏ hàng |
| id | Int(10) | No | Mã sản phẩm |
| User | Varchar(255) | No | Tên user |
| soluong | Int(10) | No | Số lượng |
| tinhtrang | Text | No | Tình trạng |
| ngaydat | datetime | No | Ngày đặt hàng |

### Bảng 1. Bảng giohang

### Bảng liên hệ (lienhe)

*Dùng* *để* *quản* *lí* *các* *thắc* *mắc* *liên* *hệ* *của* *khách* *hàng*

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Null | Chú thích |
| **Id-lienhe** | Int(10) | No | Mã liên hệ |
| Hoten | Text | No | Họ tên |
| Cty | Text | No | Công ty |
| Email | Text | No | Email |
| dienthoai | Int(20) | No | Điện thoại |
| Fax | Int(20) | No | Fax |
| diachi | Text | No | Địa chỉ |
| noidung | Text | No | Nội dung |
| ngaylienhe | Datetime | No | Ngày liên hệ |

### Bảng 3. Bảng lienhe

### Bảng loại sản phẩm (loaisanpham)

*Dùng* *để* *quản* *lí* *danh* *mục* *loại* *sản* *phẩm*

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Null | Chú thích |
| **Id-loai** | Int(10) | No | Mã loại sp |
| Id\_nhom | Int(11) | No | Mã nhóm sản phẩm |
| tenloaisp | Text | No | Tên loại sản phẩm |
| ghichu | Varchar(255) | No | Ghi chú |

### Bảng 4. Bảng loaisanpham

### Bảng nhóm sản phẩm (nhomsanpham)

*Dùng* *để* *quản* *lý* *nhóm* *sản* *phẩm*

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Null | Chú thích |
| **Id\_nhom** | Int(10) | No | Mã nhóm sản phẩm |
| tennhom | Varchar(255) | No | Tên nhóm sản phẩm |
| Chuthich | Text | No | Chú thích |

**Bảng** **5.** **Bảng** **nhomsanpham**

## 44

### Bảng sản phẩm (sanpham)

*Dùng* *để* *quản* *lý* *sản* *phẩm*

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Null | Chú thích |
| **Id** | Int(20) | No | Mã sản phẩm |
| Id\_loai | Int(10) | No | Mã loại sản phẩm |
| Tensp | Text | No | Tên sản phẩm |
| Mota | Text | No | Mô tả |
| Hinh | Text | No | Hình ảnh |
| Gia | Int(15) | No | Đơn giá |
| ghichu | Text | No | Ghi chú |
| soluongban | Int(10) | No | Số lượng |
| Id\_menu | Int(10) | No | Mã menu |

### Bảng 6. Bảng sanpham

### Bảng thành viên (thanhvien)

*Dùng* *để* *quản* *lý* *thành* *viên*

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Null | Chú thích |
| Hoten | Varchar(255) | No | Họ tên |
| diachi | Varchar(255) | No | Địa chỉ |
| Email | Varchar(255) | No | Email |
| dienthoai | Int(20) | No | Điện thoại |
| User | Varchar(255) | No | Tên đăng nhập |
| Pass | Varchar(32) | No | password |
| hieuluc | Int(10) | No | Hiệu lực |
| capquyen | Int(10) | No | Quyền |

### Bảng 7. Bảng thanhvien

* 1. **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.

45

# CHƯƠNG 3. XÂY DỰNG WEBSITE KINH DOANH SON MÔI TRỰC TUYẾN

### Thiết kế các bảng CSDL

### Bảng giỏ hàng (giohang)

Hình 3.1. Bảng giỏ hàng

### Bảng liên hệ (lienhe)

Hình 3.2. Bảng liên hệ

### Bảng loại sản phẩm (loaisanpham)

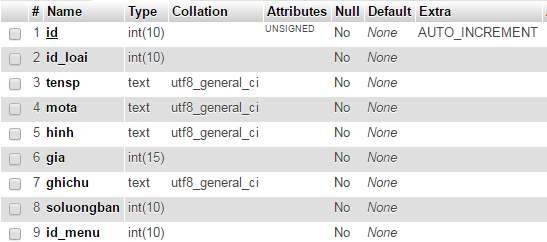
Hình 3.3. Bảng loại sản phẩm

### Bảng nhóm sản phẩm (nhomsanpham)

Hình 3.4. Bảng nhóm sản phẩm

## 46

### Bảng sản phẩm (sanpham)



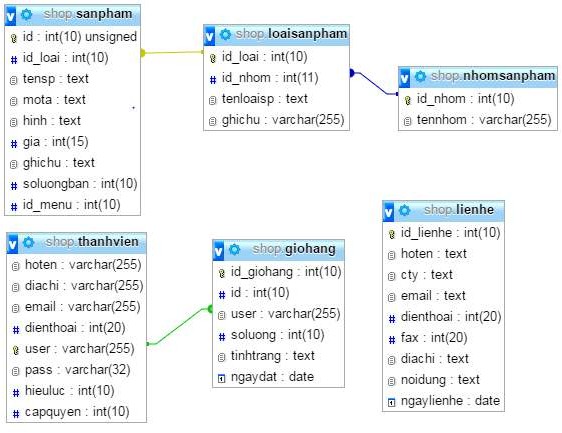
Hình 3.5. Bảng sản phẩm

### Bảng thành viên (thanhvien)

Hình 3.6. Bảng thành viên

## 47

### Mối quan hệ giữa các bảng



Hình 3.7. Mối quan hệ giữa các bảng CSDL

### Thiết kế giao diện

### Giao diện trang chủ

Hình 3.8 Giao diện trang chủ

## 48

Trang chủ *“Index.php”*: Là trang chính, chứa tất cả các mục để lựa chọn. Từ trang chủ khách hàng có thể tìm thấy mọi thông tin cần thiết:

* Thông tin về các loại sản phẩm với giao diện gần gũi, thân thiện và dễ sử dụng... từ đó khách hàng có thể đặt mua cho mình những sản phẩm yêu thích, phù hợp với túi tiền...
* Các bài viết hay về thể giới máy tính được quản trị hệ thống chọn lọc...
* Bạn có thắc mắc, góp ý với công ty…

### Giao diện Form đăng nhập

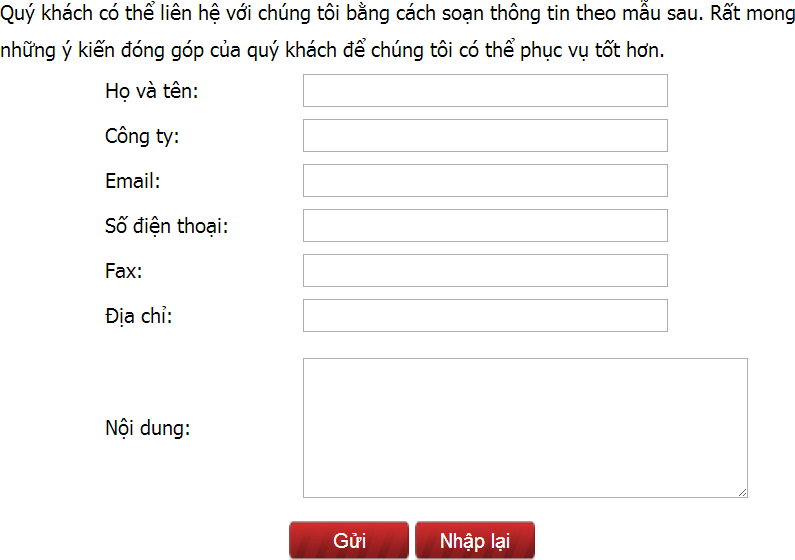


Hình 3.9 Giao diện form đăng nhập

Trang “login.php” là trang mà khách hàng sẽ nhập User và Pass của mình để đăng nhập vào website, nếu chưa có tài khoản khách hàng nhấn vào nút Đăng ký để tạo cho mình một tài khoản mới.

## Giao diện Form liên hệ

49



Hình 3.10 Giao diện form liên hệ

Trang “*lienhe.php”* là cầu nối giữa khách hàng và cửa hàng, tại đây khách hàng có thể gửi các thắc mắc cũng như ý kiến của mình tới cửa hàng. khách hàng phải nhập đầy đủ các thông tin cá nhân trong form trên.

### Giao diện trang giỏ hàng

Hình 3.11 Giao diện giỏ hàng

## 50

Trang *“giohang.php”* là trang mà thành viên có thể xem được những sản phẩm của mình đã chọn. Tại đây thành viên có thể biết được số lượng mặt hàng mình đã chọn cũng như tổng số tiền phải thanh toán. Thành viên có thể thêm, cập nhập cũng như xóa sản phẩm ngay tại giỏ hàng. Tại đây cũng có chức năng lưu lịch sử mua hàng của thành viên.

### Giao diện Form đăng ký thành viên



Hình3.12 Giao diện form đăng ký thành viên

Trang “*register.php*” Tại đây khách hàng sẽ điền đầy đủ thông tin cá nhân và đăng ký cho mình một tài khoản, để dùng cho việc mua hàng tại website.

## 51

### Giao diện trang chi tiết sản phẩm



Hình 3.13. Giao diện trang chi tiết sản phẩm

Trang “*chitiet.php*” là trang hiển thị thông tin chi tiết về sản phẩm: giá, bảo hành, thông số kỹ thuật…giúp cho khách hàng có cái nhìn tổng quan về sản phẩm. Là cơ sở để khách hàng có quyết định mua sản phẩm hay không.

## 52

### Giao diện trang quản trị Admin



Hình 3.14. Giao diện trang quản trị Admin

Đây là trang quản trị dành cho người Admin của cửa hàng, cung cấp đầy đủ tính năng cho người quản trị như: quản lý sản phẩm, quản lý thành viên, quản lý các đơn đặt hàng…

## 53

### Kết quả đạt được

**KẾT** **LUẬN**

* + Tìm hiểu ngôn ngữ lập trình Web PHP & MySQL
  + Áp dụng xây dựng ứng dụng thực nghiệm trang **Website** **kinh** **doanh** **son** **trực** **tuyến**
  + Quản lý chính xác các, nhanh chóng và kịp thời các hoạt động
  + Trang web dễ sử dụng
  + Tạo sự chuyên nghiệp trong môi trường làm việc.

### Hạn chế

* + Các chức năng tìm kiếm thống kê chưa được sinh động, còn gò bó theo 1 hình dạng khuôn mẫu.
  + Còn hạn chế về việc thanh toán tiền cho khách hàng.

### 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...

## 54

### Tiếng Việt

**TÀI** **LIỆU** **THAM** **KHẢO**

[1]. Nguyễn Trường Sinh, *Sử* *dụng* *PHP* *và* *MySQL* *thiết* *kế* *web* *động*, NXB Thống kê, 2008.

[2]. Giáo trình: *Phát* *tiển* *phẩn* *mềm* *mã* *nguồn* *mở*, Trường đại học Sao Đỏ

### Trang web

[1]. [http://www.php.net](http://www.php.net/)

[2]. [http://www. mysql.com](http://www.mysql.com/)

55