

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG
-----o0o-----



ĐỒ ÁN TỐT NGHIỆP

NGÀNH CÔNG NGHỆ THÔNG TIN

HẢI PHÒNG 2011

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG
-----o0o-----

TÊN ĐỀ TÀI ĐỒ ÁN TỐT NGHIỆP

XÂY DỰNG WEBSITE CHO KHÁCH SẠN CAMELA

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành: Công nghệ Thông tin

HẢI PHÒNG - 2011

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG
-----o0o-----

TÊN ĐỀ TÀI ĐỒ ÁN TỐT NGHIỆP

XÂY DỰNG WEBSITE CHO KHÁCH SẠN CAMELA

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành: Công nghệ Thông tin

Sinh viên thực hiện:.....

Giáo viên hướng dẫn:.....

Mã số sinh viên:.....

HẢI PHÒNG - 2011

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập - Tự do - Hạnh phúc
-----oOo-----

NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP

Sinh viên: Bùi Thị Thu Hà.

Mã số: 110576.

Lớp: CT1102.

Ngành: Công nghệ Thông tin

Tên đề tài:

XÂY DỰNG WEBSITE CHO KHÁCH SẠN CAMELA

NHIỆM VỤ ĐỀ TÀI

1. Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp

a. Nội dung:

b. Các yêu cầu cần giải quyết

2. Các số liệu cần thiết để thiết kế, tính toán

3. Địa điểm thực tập

CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

Người hướng dẫn thứ nhất:

Họ và tên:.....

Học hàm, học vị:.....

Cơ quan công tác:.....

Nội dung hướng dẫn:

.....

.....

.....

Người hướng dẫn thứ hai:

Họ và tên:

Học hàm, học vị.....

Cơ quan công tác:

Nội dung hướng dẫn:

.....

.....

.....

Đề tài tốt nghiệp được giao ngày 12 tháng 04 năm 2010

Yêu cầu phải hoàn thành trước ngày 10 tháng 07 năm 2010

Đã nhận nhiệm vụ: Đ.T.T.N
Sinh viên

Đã nhận nhiệm vụ: Đ.T.T.N
Cán bộ hướng dẫn Đ.T.T.N

Hải Phòng, ngàytháng.....năm 2010

HIỆU TRƯỞNG

GS.TS.NGŨT Trần Hữu Nghị

PHẦN NHẬN XÉT TÓM TẮT CỦA CÁN BỘ HƯỚNG DẪN

1. Tinh thần thái độ của sinh viên trong quá trình làm đề tài tốt nghiệp:

.....

.....

.....

.....

.....

.....

.....

.....

2. Đánh giá chất lượng của đề tài tốt nghiệp (so với nội dung yêu cầu đã đề ra trong nhiệm vụ đề tài tốt nghiệp)

.....

.....

.....

.....

.....

.....

.....

.....

3. Cho điểm của cán bộ hướng dẫn:

(Điểm ghi bằng số và chữ)

.....

.....

Ngày.....tháng.....năm 2010

Cán bộ hướng dẫn chính

(Ký, ghi rõ họ tên)

**PHẦN NHẬN XÉT ĐÁNH GIÁ CỦA CÁN BỘ CHẤM PHẢN BIỆN ĐỀ TÀI
TỐT NGHIỆP**

1. Đánh giá chất lượng đề tài tốt nghiệp (về các mặt như cơ sở lý luận, thuyết minh chương trình, giá trị thực tế, ...)

2. Cho điểm của cán bộ phản biện

(Điểm ghi bằng số và chữ)

.....
.....

Ngày.....tháng.....năm 2010
Cán bộ chấm phản biện
(Ký, ghi rõ họ tên)

LỜI CẢM ƠN

Trước hết em xin gửi lời cảm ơn đến thầy Phùng Anh Tuấn, người đã hướng dẫn em rất nhiều trong suốt quá trình tìm hiểu nghiên cứu và hoàn thành đồ án này từ lý thuyết đến ứng dụng. Sự hướng dẫn của thầy đã giúp em có thêm những kiến thức về lập trình web về cách sử dụng mã nguồn mở. Qua những chỉ dẫn ân cần của thầy giúp em hiểu sâu hơn về những kiến thức đã được học. Đồng thời em cũng xin chân thành cảm ơn các thầy cô trong bộ môn cũng như các thầy cô trong trường đã trang bị cho em những kiến thức cơ bản cần thiết để em có thể hoàn thành tốt đề tài này. Em xin gửi lời cảm ơn đến các thành viên lớp CT1102, những người bạn đã luôn ở bên cạnh động viên, tạo điều kiện thuận lợi và cùng em tìm hiểu, hoàn thành tốt đồ án. Sau cùng, em xin gửi lời cảm ơn đến gia đình, bạn bè đã tạo mọi điều kiện để em xây dựng thành công đồ án này.

Hải Phòng ngày 10, tháng 6 năm 2010

Sinh viên thực hiện

LỜI NÓI ĐẦU

Cuộc cách mạng khoa học và công nghệ trên toàn thế giới thúc đẩy loài người bước sang một kỷ nguyên mới. Đó là kỷ nguyên của nền văn minh dựa trên cơ sở công nghiệp trí tuệ. Cuộc cách mạng khoa học kỹ thuật có thể được đánh dấu bằng sự ra đời và phát triển ồ ạt của máy tính cũng như sự đa dạng của công nghệ phần mềm, nên việc ứng dụng tin học vào đời sống xã hội nói chung và công tác quản lý nói riêng đang phát triển mạnh mẽ, nó góp một vai trò không nhỏ vào việc phát triển của công nghệ thông tin và ngày nay sự phát triển đó đang được ứng dụng nhiều và không thể thiếu được trong mọi ngành nghề văn phòng, quảng cáo, tài chính, công tác quản lý...

Sự bùng nổ Internet trên mọi lĩnh vực giúp mọi người có thể lấy các tin tức cần thiết, chia sẻ các thông tin trên mạng máy tính và liên lạc với người ở xa. Website là một trong những phương tiện hữu ích giúp người sử dụng làm những công việc trên, thông qua Web mọi người tìm được những gì mình cần rất nhanh chóng mà không thiếu thốn thời gian vì phát triển thương mại điện tử nên mọi cơ quan, văn phòng cũng như siêu thị đều có Web của riêng mình. Bạn nghĩ sao khi chỉ cần ngồi nhà mà có thể đặt hàng, đặt phòng cũng như lấy thông tin cần thiết. Có được những điều đó là nhờ sự kết hợp của Web và cơ sở dữ liệu nhằm đưa ra những cơ sở dữ liệu đơn lẻ độc lập trở nên hữu ích cho mọi người trên toàn thế giới, giúp mọi người có thể truy cập kho dữ liệu khổng lồ của nhân loại. Web trở thành một cuộc cách mạng vì nó làm cho Internet trở nên thân thiện với người dùng.

Xuất phát từ nhu cầu thực tiễn trên tôi đã lựa chọn đề tài tốt nghiệp "Xây dựng website cho khách sạn Camela" nhằm giúp cho mọi người có thể truy cập để biết các thông tin chi tiết về khách sạn để có sự lựa chọn phù hợp theo tiêu chuẩn mỗi người. Có thể truy cập tại nhà hoặc bất cứ nơi nào bạn muốn.

Chương 1: GIỚI THIỆU CÁC CÔNG CỤ HỖ TRỢ

1.1 INTERNET

1.1.1 Giới thiệu về internet

Internet là tài nguyên vô tận với con người trong mọi lĩnh vực. Internet là mạng máy tính khổng lồ được kết nối lại với nhau. Bất cứ vị trí , khoảng cách hoặc thời gian nào, một máy tính kết nối vào mạng Internet đều được coi là thành viên của mạng Internet.

1.1.2 Sự ra đời và phát triển của internet

Trong những thập niên 60 và 70, nhiều công nghệ mạng máy tính ra đời nhưng mỗi kiểu lại dựa trên những phần cứng riêng biệt. Một trong những kiểu này được gọi là mạng cục bộ (Local Area Networks - LAN), kết nối các máy tính với nhau trong phạm vi hẹp bằng dây dẫn và một thiết bị cài đặt trong mỗi máy. Các mạng lớn được gọi là mạng diện rộng (Wide Area Networks-WAN), nối nhiều máy tính với nhau trong phạm vi rộng thông qua một hệ thống dây truyền dẫn kiểu như trong các hệ thống điện thoại. Giao thức TCP/IP đảm bảo sự thông suốt trao đổi thông tin giữa các máy tính. Internet hiện nay đang liên kết hàng ngàn máy tính thuộc các công ty, cơ quan nhà nước, các trung tâm nghiên cứu khoa học, trường đại học, không phân biệt khoảng cách địa lý. Đó là ngân hàng dữ liệu khổng lồ của nhân loại.

1.1.3 Ứng dụng web

Khi nói đến Internet người ta thường nhắc đến bộ giao thức chuẩn TCP/IP và các dịch vụ điển hình nhất của nó là email, FPT (File Transfer Protocol) và WWW (World Wide Web). Tuy nhiên WWW chiếm vai trò quan trọng nhất vì nó quyết định mô hình của internet. Tìm hiểu về WWW ta có thể xác định phạm vi ứng dụng của internet trong thực tiễn, khoa học, công nghệ cũng như trong đời sống.

WWW rất dễ sử dụng và thú vị cho nên đã trở thành một dịch vụ quen thuộc không thể thiếu. Ngày nay, khi email và FPT đã được tích hợp vào hầu hết các trình duyệt thì WWW cũng đã trở thành một công cụ để khai thác các hoạt động tìm kiếm , giải trí... Với bản chất là một siêu tài liệu đa phương tiện., chứa đựng các thông tin về các dữ liệu multimedia, WWW đã làm cho internet trở nên thuận lợi, sinh động và hấp dẫn hơn nhiều.

1.2 JOOMLA

1.2.1 Giới thiệu về Joomla.

Joomla! là một hệ quản trị nội dung mã nguồn mở. Joomla! được viết bằng ngôn ngữ PHP và kết nối tới cơ sở dữ liệu MySQL, cho phép người sử dụng có thể dễ dàng xuất bản các nội dung của họ lên Internet hoặc Intranet.

Joomla! là một hệ quản trị nội dung - là phần mềm để tổ chức và tạo môi trường cộng tác thuận lợi nhằm mục đích xây dựng một hệ thống tài liệu và các loại nội dung khác một cách thống nhất (thường là quản lý nội dung trang web).

Các đặc điểm cơ bản của một hệ quản trị nội dung bao gồm:

- Phê chuẩn việc tạo hoặc thay đổi nội dung trực tuyến
- Chế độ Soạn thảo
- Quản lý người dùng
- Tìm kiếm và lập chỉ mục
- Lưu trữ
- Tùy biến giao diện
- Quản lý ảnh và các liên kết (URL)

Mã nguồn mở là mã nguồn được công bố và sử dụng một giấy phép nguồn mở. Giấy phép này cho phép bất cứ ai cũng có thể nghiên cứu, thay đổi và cải tiến phần mềm, và phân phối phần mềm ở dạng chưa thay đổi hoặc đã thay đổi.

Joomla! có các đặc tính cơ bản là: bộ đệm trang (page caching) để tăng tốc độ hiển thị, lập chỉ mục, đọc tin RSS (RSS feeds), trang dùng để in, bản tin nhanh, blog, diễn đàn, bình chọn, lịch biểu, tìm kiếm trong site và hỗ trợ đa ngôn ngữ.

Joomla! được sử dụng ở khắp mọi nơi trên thế giới, từ những website cá nhân cho tới những hệ thống website doanh nghiệp có tính phức tạp cao, cung cấp nhiều dịch vụ và ứng dụng. Joomla! có thể dễ dàng cài đặt, dễ dàng quản lý và có độ tin cậy cao.

1.2.2 Ứng dụng của joomla

- Các công thông tin điện tử hoặc các website doanh nghiệp
- Thương mại điện tử trực tuyến
- Báo điện tử, tạp chí điện tử
- Website của các doanh nghiệp vừa và nhỏ
- Website của các cơ quan, tổ chức phi chính phủ
- Website các trường học
- Website của gia đình hay cá nhân

1.2.3 Dòng phiên bản 1.5.x

Phiên bản Joomla! 1.5 là phiên bản cải tiến từ Joomla! 1.0.x được coi như Mambo 4.6. Joomla! 1.5 tiếp tục duy trì một giao diện người sử dụng đơn giản (nhìn và cảm nhận - look and feel).

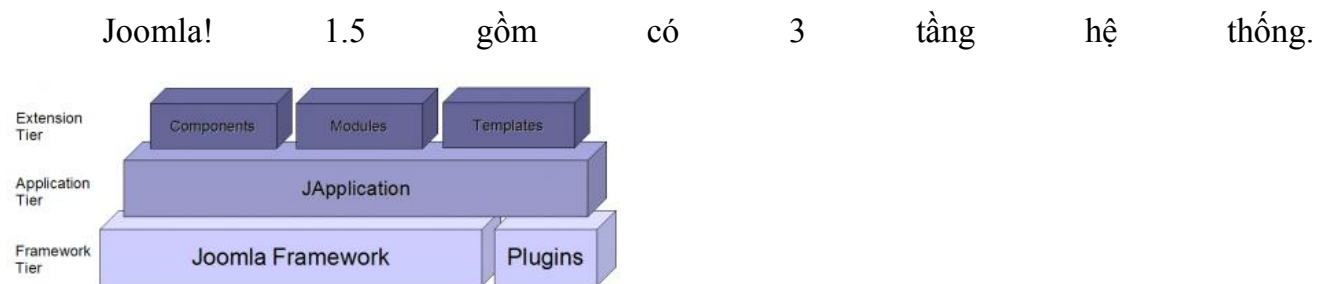
Cả Joomla! 1.5 và Mambo 4.6 đều hỗ trợ đa ngôn ngữ. Joomla! thì sử dụng file định dạng ".ini" để lưu các thông tin chuyển đổi ngôn ngữ, còn Mambo thì sử dụng file định dạng

".gettext". Joomla 1.5 hỗ trợ tất cả các ngôn ngữ có tập ký tự được biểu diễn bằng bảng mã UTF-8.

Joomla! 1.5 cũng bao gồm các đặc tính mới như các mô hình chứng thực (LDAP, Gmail...), hỗ trợ mô hình khách-chủ[7] xml-rpc. Nó cũng hỗ trợ các trình điều khiển cơ sở dữ liệu dành cho MySQL 4.1+ (trên nền PHP 5) và tăng cường hỗ trợ cho MySQL 5, cũng như hỗ trợ các loại cơ sở dữ liệu khác.

Điểm mạnh của Joomla! 1.5: Phần quản trị Website có sử dụng công nghệ Web 2.0, một số tính năng được cải tiến hơn so với Joomla 1.0.x.

1.2.4 Kiến trúc



Tầng dưới cùng là mức nền tảng, chứa các thư viện và các plugin (còn được biết với tên gọi mambot).

Tầng thứ hai là mức ứng dụng và chứa lớp JApplication. Hiện tại tầng này gồm 3 lớp con: JInstallation, JAdministrator và JSite.

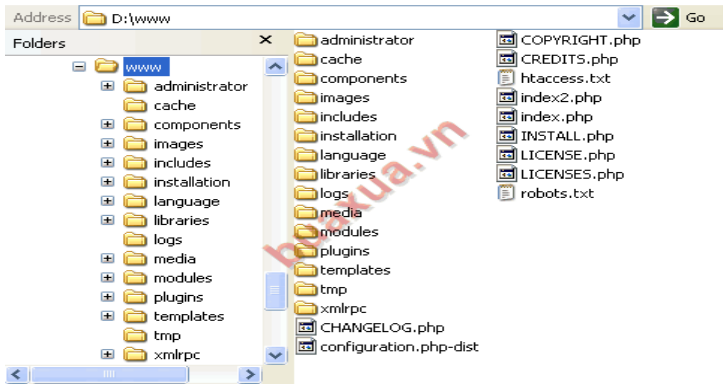
Tầng thứ ba là mức mở rộng. Tại tầng này có các thành phần (component), mô đun (module) và giao diện (template) được thực thi và thể hiện.

1.2.5 cách cài đặt joomla! Phiên bản 1.5.x trên localhost

Với sự ra đời phiên bản mới là 1.5.x hiện nay Joomla! có 2 dòng phiên bản đó là 1.0.x và 1.5.x. Phiên bản 1.5.x được lập trình mới hoàn toàn hứa hẹn sẽ đem lại sự lựa chọn và nhiều tính năng mới cho người sử dụng.

Chuẩn bị:

- Tải phiên bản Joomla! 1.5.x mới nhất
- Sau khi cài đặt XAMPP và download xong joomla các bạn có thể bắt đầu tiến hành cài đặt. Đầu tiên tạo thư mục với tên bất kì trong thư mục /htdocs của XAMPP và bạn giải nén joomla vào trong thư mục vừa tạo của XAMPP (VD:C:\xampp\htdocs\joomla).
- Sau khi giải nén, ví dụ thư mục giải nén của bạn là joomla thì bạn mở trình duyệt lên và gõ vào như sau :
<http://localhost/joomla/> và enter và bạn sẽ thấy trang cài đặt của Joomla 1.5.

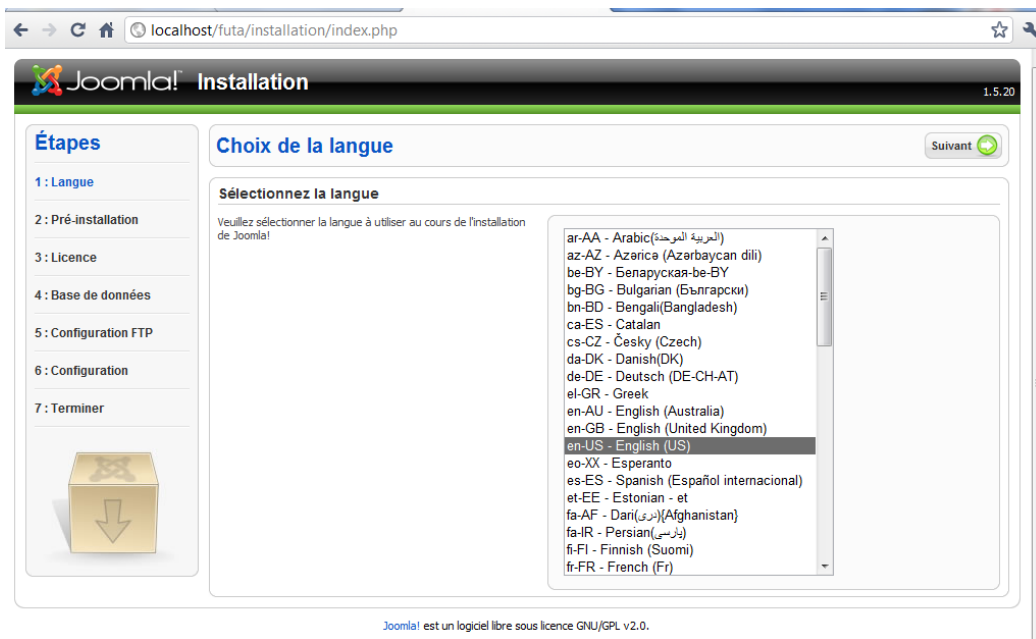


Cài đặt:

Mở trình duyệt Web (Internet Explorer, FireFox,...) nhập dòng địa chỉ `http://localhost/` và nhấn phím *Enter*.

Màn hình cài đặt Joomla! sẽ xuất hiện với các bước cài đặt sau:

Bước 1: Chọn ngôn ngữ



Bước này sẽ cho phép chọn ngôn ngữ hiển thị trong quá trình cài đặt, không phải ngôn ngữ của trang Web sau này. Nếu muốn trang Web sẽ tự động hiển thị ngôn ngữ Việt sau khi cài đặt thì giải nén gói ngôn ngữ Việt vào Joomla! trước khi thực hiện bước kế tiếp. Chọn **vn-VN** để hiển thị ngôn ngữ Việt và nhấn **Next**.

Bước 2: Kiểm tra trước khi cài đặt

Pre-installation Check

Pre-installation check for Joomla! 1.5.20 Stable [senu takaa] 18-July-2010 18:00 GMT:

If any of these items are not supported (marked as **No**), your system does not meet the minimum requirements for installation. Please take appropriate actions to correct the errors. Failure to do so could lead to your Joomla! installation not functioning properly.

PHP Version >= 4.3.10	Yes
- Zlib Compression Support	Yes
- XML Support	Yes
- MySQL Support	Yes
MB Language is Default	Yes
MB String Overload Off	Yes
configuration.php Writable	Yes

Recommended Settings:

These are the recommended settings for PHP in order to ensure full compatibility with Joomla!. Joomla! will still operate even if your settings do not match.

Directive	Recommended	Actual
Safe Mode:	Off	Off
Display Errors:	Off	On
File Uploads:	On	On
Magic Quotes Runtime:	Off	Off
Register Globals:	Off	Off
Output Buffering:	Off	Off
Session Auto Start:	Off	Off

Joomla! is free software released under the GNU/GPL v2.0 License.

- Bước này chương trình sẽ kiểm tra các thông số của máy chủ Web, nếu kết quả của các thông số có màu **xanh** là đáp ứng được hoạt động tối ưu của Joomla! 1.5.x. Nhấn **Tiếp** (Next) để tiếp tục.

Bước 3: Chứng nhận

License

GNU General Public License:

Table of Contents

- GNU GENERAL PUBLIC LICENSE
 - Preamble
 - TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION
 - How to Apply These Terms to Your New Programs

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place - Suite 330, Boston, MA 02111-1307, USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software—to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

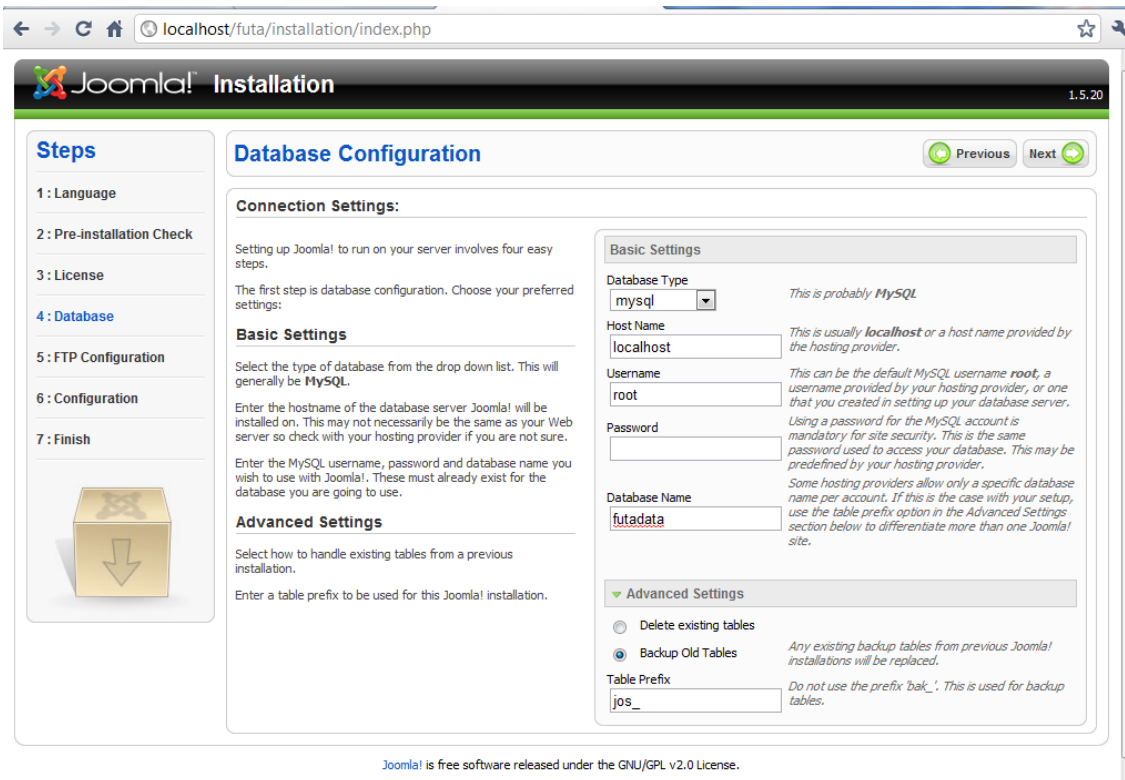
When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

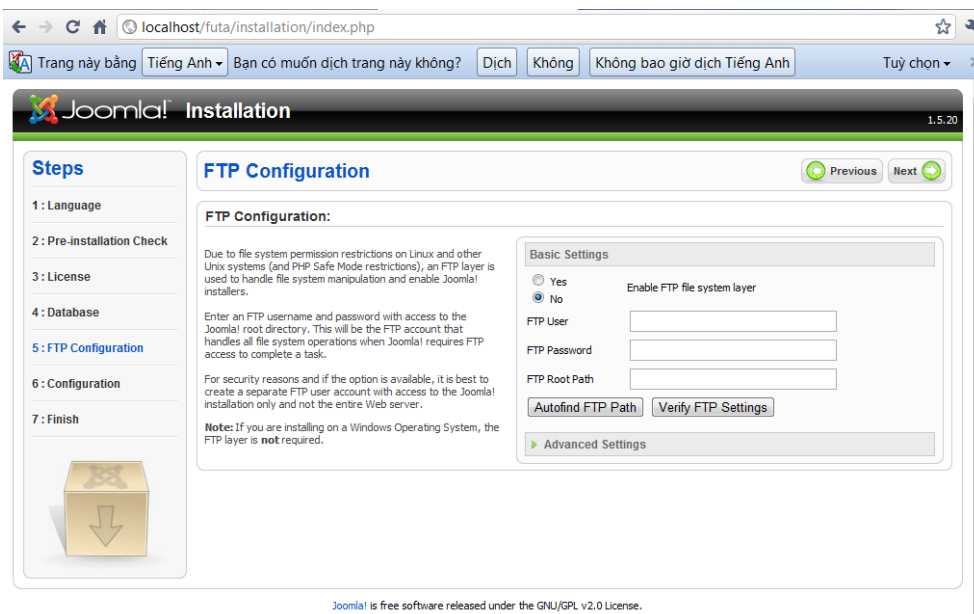
- Đây là các thông tin về bản quyền và điều kiện sử dụng của Joomla!, nhấn **Tiếp** (Next) để tiếp tục.

Bước 4: Cấu hình cơ sở dữ liệu



- Nhập các thông số của cơ sở dữ liệu đã được tạo trước ở phần chuẩn bị để chương trình tiến hành cài đặt các bảng dữ liệu mẫu cho Joomla! Nhấn **Tiếp** (Next) để tiếp tục, nếu các thông số được nhập đúng thì chương trình sẽ chuyển qua bước kế tiếp.

Bước 5: Cấu hình FTP



Bước này cho phép nhập các thông số để hỗ trợ việc truyền tải các tập tin (File) theo giao thức FTP. Thông số này không cần thiết đối với khi cài đặt Joomla! trên *Localhost*, sau này khi chuyển lên *Host* có thể thiết lập sau. Nhấn **Tiếp** (Next) để tiếp tục.

Bước 6: Cấu hình chính

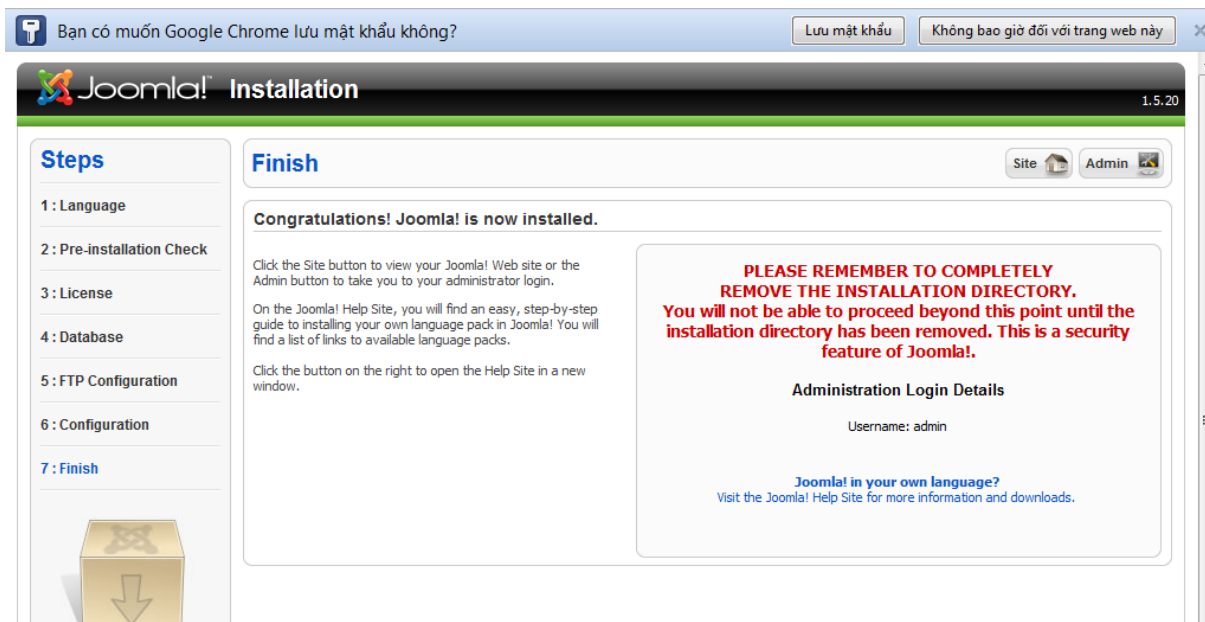
- Bước này sẽ cho phép thiết lập các thông số cho trang Web. Nhập tên cho trang Web, tên này sẽ được hiển thị ở thanh tiêu đề của trình duyệt, địa chỉ Email dùng để liên lạc giữa Ban quản trị (Admin) và các thành viên (User) của trang Web, chọn và *ghi nhớ mật khẩu quản trị* dùng truy cập vào phần Admin (Back-end) để quản lý trang Web.

Nhấn vào nút *Cài đặt dữ liệu mẫu* để chương trình cài đặt sẵn nội dung mẫu cho trang Web, việc này rất có ích đối với người mới dùng lần đầu, sau này chỉ cần sửa lại các dữ liệu mẫu này theo ý mình là được.

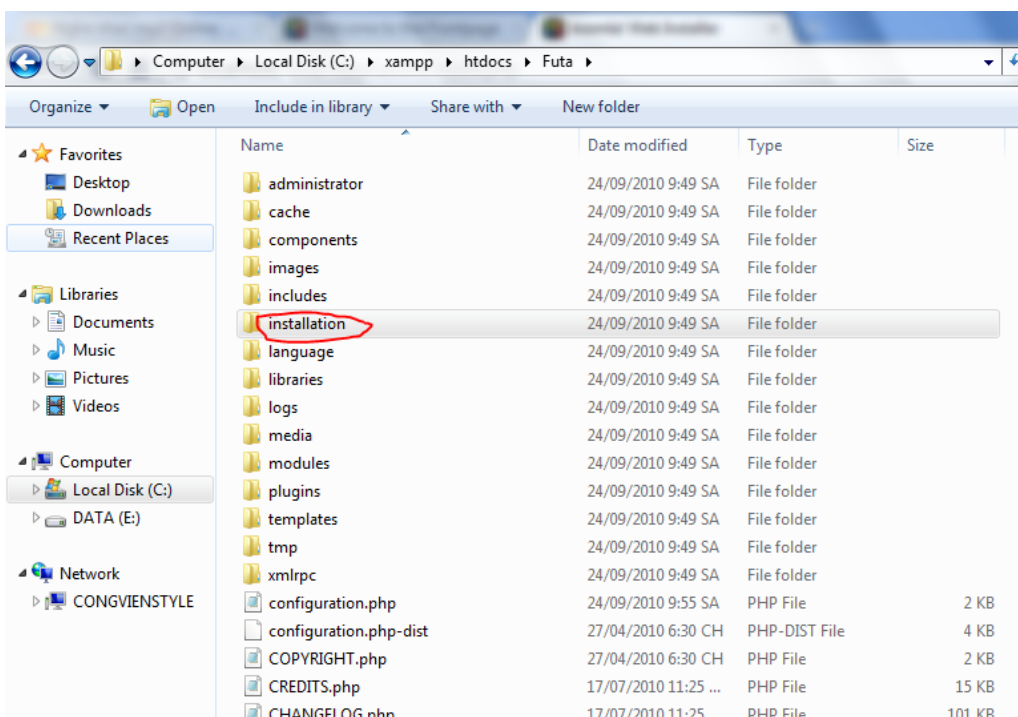
Sau khi cài đặt dữ liệu mẫu sẽ xuất hiện thông báo *Đã cài đặt thành công dữ liệu mẫu*. Lưu ý nếu không cài đặt được dữ liệu mẫu là do việc khai báo các thông số của Cơ sở dữ liệu ở *Bước 4* không đúng, cần phải nhấn nút Trước (Back) để quay lại khai báo cho đúng.

Sau khi thiết lập xong nhấn *Tiếp* (Next) để tiếp tục.

Bước 7: Kết thúc



Quá trình cài đặt đã hoàn tất, phải đổi tên hoặc xóa thư mục tên là *installation* ngay lập tức, thư mục này nằm trong nơi đã cài đặt Joomla!



Sau khi xóa xong quay trở lại trình duyệt, có 2 lựa chọn:

- *Quản trị*: Nhấn vào *Quản trị* để truy cập vào phần Quản lý trang Web (Admin) với tên truy cập (Username) là *admin* và mật khẩu (Password) được thiết lập ở *Bước 6*.

Trang Web: Nhấn vào *Trang Web* để truy cập vào phần nội dung của trang Web



Nếu thư mục *installation* chưa được đổi tên hay xóa thì khi nhấn *Quản trị* hoặc *Trang Web* sẽ xuất hiện thông báo yêu cầu đổi tên hay xóa thư mục này, sau khi thực hiện xong nhấn vào dòng chữ *Nhấn vào đây khi thư mục cài đặt (/installation) đã được xóa để tiếp tục.*



Nếu chọn cách *cài đặt ngôn ngữ Việt trước* thì mặc nhiên sau khi cài đặt trang Web sẽ tự động hiển thị tiếng Việt.



Đăng nhập

Dùng một tên đăng nhập và Mật mã hợp lý để đăng nhập vào khu vực quản trị.

[Trở về trang chủ](#)

Tên đăng nhập:

Mật mã:

Language:



1.2.6 Tổng quan trang quản trị trong Joomla

Joomla Administration – Quản trị Joomla: Là trung tâm điều khiển của website Joomla. Dùng giao diện tao nhã của Admin Section để xây dựng cấu trúc và biên tập, xuất bản nội dung.

Admin Section được chia thành bốn vùng : Menubar – thanh menu, Toolbar – thanh công cụ, Infobar – thanh thông tin, Workspace – vùng làm việc. Để truy cập vào vùng Administration, đăng nhập (log in) bằng tài khoản người dùng với đặc quyền quản trị (administrative).

Administrator Login – Đăng nhập Quản trị viên: Cài đặt mới Joomla sẽ có một người dùng được cài. Người dùng (user) này được biết như là Super Administrator (siêu quản trị viên). Super Administrator có tên truy cập (login name) là “admin”. Password cho Super Administrator được tạo trong quá trình cài đặt Joomla.

Mỗi người dùng trong Joomla được gán vào một nhóm, hiện tại có ba nhóm người dùng với đặc quyền truy cập tới Administrator Back-end : Super Administrator, Administrator, và Manager.

Chức năng của mỗi nhóm được giảng giải dưới đây :

Super Administrator – siêu quản trị : Super Administrator Group truy cập vào tất cả các chức năng Back-end (Administrator). Site phải có tối thiểu một người được bổ nhiệm là một Super Administrator để thực hiện Global Configuration (cấu hình tổng thể) của site. Những người dùng trong nhóm này không thể xóa và không thể chuyển đổi sang nhóm khác.

Administrator- Quản trị viên: Nhóm quản trị có một hạn chế nhỏ khi truy cập vào các chức năng Back-en (Administrator). Administrator không thể :

Thêm hay sửa một người dùng trong nhóm Super Administrator

Truy cập các cài đặt Global Configuration

Truy cập chức năng Mass Mail

Quản lý/cài đặt các Template

Quản lý/cài đặt các file language (ngôn ngữ)

Manager – Quản lý: Nhóm quản lý có tham số hầu hết, có ý nghĩa trong việc tạo nội dung. Một thành viên trong nhóm này, có thêm một số giới hạn so với một Administrator, họ không thể:

- Quản lý các User

- Cài đặt các Module

- Cài đặt các Component

- Truy cập một số Component (được xác định bởi Super Administrator)

Menubar – Thanh Menu: Thanh Menu được chia thành mười phần : Home (trang chủ), Site, Menu, Content (nội dung), Components, Modules, Mambots, Installers (cài đặt), Messages (thông điệp), System (hệ thống) và Help (trợ giúp). Mỗi phần có một menu đồ xuống phụ thuộc. Thanh menu là trung tâm điều hướng trong Admin Section.

Infobar – thanh thông tin: Infobar hiển thị những thông tin sau (theo thứ tự):

- Tên Web site
- Nơi hiện tại trong Admin Section
- Tên của người dùng đã đăng nhập (log in), như “Logged in as Admin”
- Số thông điệp chưa giải quyết mà có từ những Administrator khác, đang dùng chức năng thông điệp trong My Admin
- Số người dùng hiện tại đang online

Tên site về nơi chỉ ra trong Pathway (đường dẫn), một textbox (hộp nhập văn bản) nằm phía trên bên trái của trang Admin. Messages (những thông điệp), số người dùng đang online, và tên đăng nhập xuất phía trên bên phải.

Workspace – không gian làm việc: Workspace là một vùng nằm ngay dưới Menubar và Toolbar hiển thị những điều khiển cho mục menu được chọn hiện tại. workspace cập nhật khi lựa chọn một menu hay dùng một công cụ từ toolbar, cài đặt mặc định là Control Panel (bảng điều khiển)

Control Panel – bảng điều khiển: Bên trái của Control Panel hiện những biểu tượng cho những mục nội dung quản lý chính.

Bên phải của Control Panel hiện những thẻ tab truy cập những menu của component đã cài đặt, một danh sách các content item được ưa chuộng, thông tin về những item mới nhất thêm vào hệ thống (bao gồm tiêu đề, ngày, và tác giả), và menu statistics (thống kê menu). Tất cả những biểu tượng và dòng văn bản trên Control Panel đều có thể click và sẽ đưa các điều kiện của mục chọn vào Workspace.

Toolbar Icons – Các biểu tượng Toolbar:

Publish (xuất bản) : Đây là biểu tượng để xuất bản category hoặc những item. Những khác thăm site có thể những item này trên site.

Unpublish (thôi xuất bản): biểu tượng này sẽ thối xuất bản một category hoặc item, Việc không xuất bản sẽ xóa item khỏi trang xem công cộng, nhưng không xóa category hay item. Để xóa một .item, chọn nó và click biểu tượng “Delete” .

Archive: (kho lưu) : biểu tượng này chuyển một item vào Archive,

New: (tạo mới): biểu tượng này thêm một item, section, link, v.v.

Edit:(chỉnh sửa): biểu tượng này chỉnh sửa một item đã được tạo trước đó. Phải chắc đã lựa chọn một item trước khi click biểu tượng này.

Delete or Remove: (xóa): biểu tượng này xóa một category hay item trong site. Chú ý rằng tiến trình này không thể phục hồi được. Để xóa một item hay category tạm thời, click biểu tượng “Unpublish”.

Trash: (thùng rác): biểu tượng này xóa một item và đặt nó vào Trash Manager (quản lý rác). Ghi chú: có thể phục hồi những item đã đổ đi trở lại tình trạng trước đó. Đơn giản vào Trash Manager dưới menu Site.

Move: (di chuyển): biểu tượng này di chuyển một item đến một Section hay Category mới.

Apply: (áp dụng): biểu tượng này sẽ lưu nội dung mới hay những thay đổi nhưng trang vẫn mở để tiếp tục chỉnh sửa.

Save (lưu): biểu tượng này lưu nội dung mới hay những thay đổi cấu hình và quay trở lại trang trước.

Cancel (thôi) : biểu tượng này sẽ thôi tạo một trang hay thôi không lưu những thay đổi.

Chú ý: Nếu bạn muốn thôi tạo hay chỉnh sửa phải dùng nút này và không được dùng nút back trên trình duyệt vài như vậy sẽ khóa item và những content manager khác không thể chỉnh sửa trang được.

Preview: biểu tượng này dùng để xem, trong một cửa sổ pop-up, một cái nhìn trước của bất kỳ sự thay đổi nào. Nếu trình duyệt được cài khóa những của sổ pop-up hoặc đang dùng phần mềm khóa pop-up, cần đổi cài đặt hay vô hiệu phần mềm để xem trước.

Upload: biểu tượng này dùng để tải lên một file hay ảnh đến website .

User Groups và Access Control – những nhóm người dùng và điều khiển truy cập: Joomla có hai hệ thống phân cấp nhóm người dùng (user group) chính: một để truy cập Front-end (vì những user có thể log in và web site và xem những phần hay trang được chỉ định) và một cho truy cập Back-end Administrator .

Các nhóm được cung cấp mặc định là :

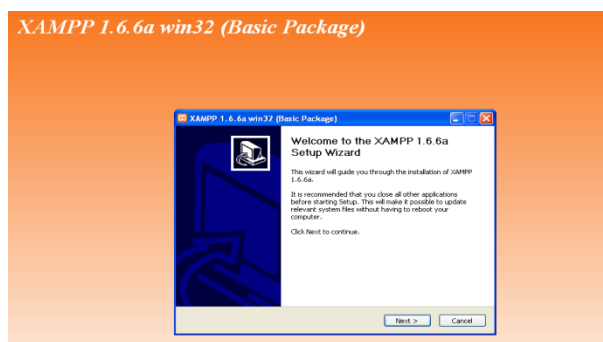
- Public Front-end (mặt trước dùng chung)
- Registered (đã đăng kí)
- Author (tác giả)
- Editor (người biên tập)
- Publisher (người xuất bản)
- Public Back-end (mặt sau dùng chung)
- Manager (người quản lý)
- Administrator (người quản trị)
- Super Administrator (siêu quản trị)

1.3 CÁCH CÀI ĐẶT CHƯƠNG TRÌNH XAMPP

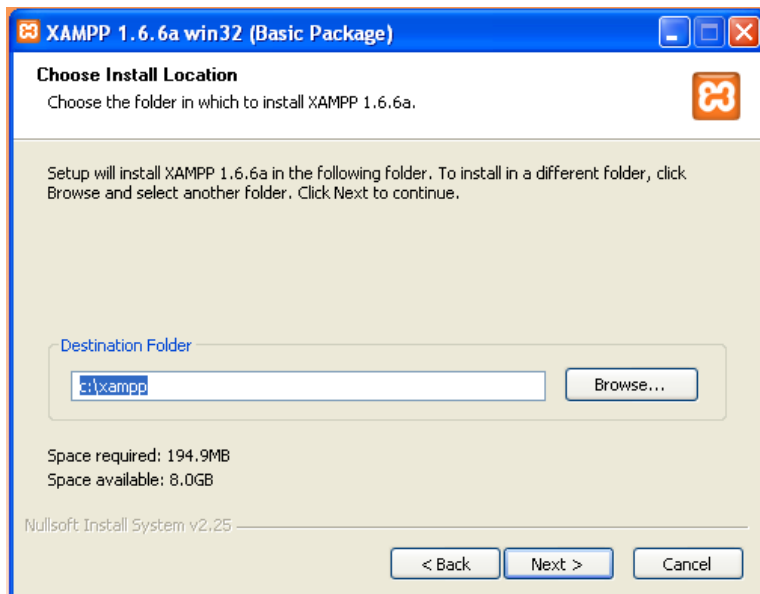
Xampp là chương trình tạo máy chủ Web (Web Server) trên máy tính cá nhân (Localhost) được tích hợp sẵn Apache, PHP, MySQL, FTP Server, Mail Server và các công cụ như PHPmyadmin.

Cách cài đặt như sau:

- Chạy chương trình cài đặt *XAMPP Installer*, khi xuất hiện bảng chọn ngôn ngữ, chọn *English* và nhấn *Ok*.
- Khi xuất hiện bảng *Welcome to the Xampp Setup Wizard*, nhấn *Next*.



- Trong *License Agreement* chọn *I accept the Agreement*, nhấn *Next*.
- Trong *Select Destination Location*, chọn nơi cài đặt *Xampp.C:\xampp* (cài trên thư mục *xampp* của ổ đĩa C), nhấn *Next*.



- Trong *XAMPP Options*, để nguyên mặc định của *Xampp* và nhấn *Install*.
- Chương trình sẽ tiến hành cài đặt và sau cùng nhấn *Finish* để kết thúc việc cài đặt, chương trình sẽ hỏi bạn có muốn mở bảng điều khiển của *Xampp* hay không, hãy chọn *No* vì chúng ta cần chỉnh thêm một vài thông số nữa.

Thư mục gốc (Web Root) để chứa các trang web mặc định của Xampp là *C:\xampp\htdocs*. Tất cả các Website hoặc thư mục chứa Website đều phải được đặt vào thư mục *Web Root* (*htdocs*) thì mới hoạt động được.

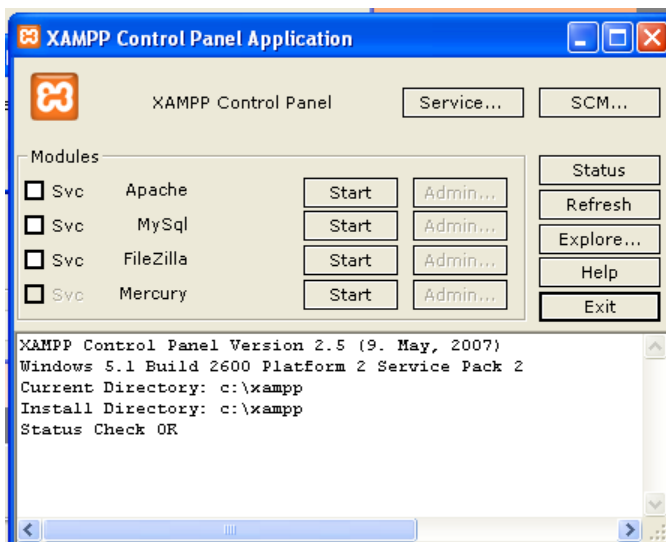
Nếu muốn bạn có thể thay đổi thư mục *Web Root*, bạn nên tạo một thư mục *Web Root* khác nằm ở bên ngoài thư mục *Xampp* để dễ sử dụng.

Thí dụ tạo thư mục *Web Root* nằm ở ổ đĩa C: có tên là *WWW* bằng cách mở File *httpd.conf* nằm trong thư mục cài đặt Xampp *C:\xampp\apache\conf* (*C:\xampp\apache\conf\httpd.conf*) bằng chương trình *Notepad* hoặc bất cứ chương trình soạn thảo văn bản nào. Tìm đến dòng *DocumentRoot "C:/xampp/htdocs"* sửa lại thành *DocumentRoot "C:/www"* và dòng *<Directory "C:/xampp/htdocs">* sửa lại thành *<Directory "C:/www">*, *Save* và đóng File này lại. Lưu ý dấu số / và \ của đường dẫn phải chính xác.

Mở bảng điều khiển của chương trình Xampp bằng cách nhấn đúp chuột vào Icon *XAMPP Control Panel* nằm trên *Desktop*. Nếu xuất hiện bảng xuất hiện bảng *Windows Security Alert* thì chọn *Unblock*.

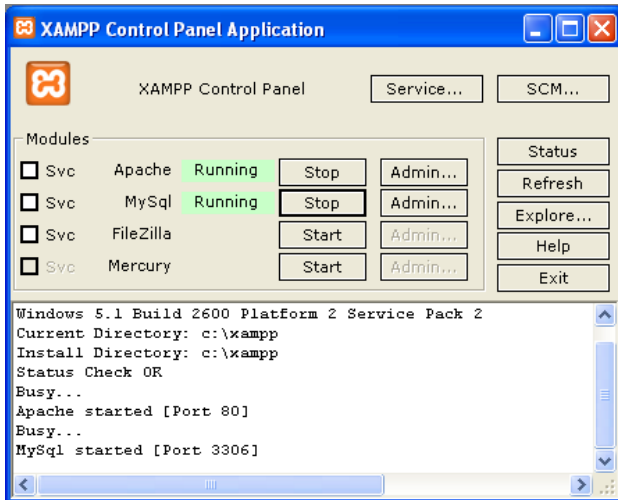


Trong bản điều khiển *Xampp*, chọn *Start* để khởi động *Apache* và *MySQL*. Nhấn vào nút **X** (góc trên bên phải) để tắt bảng điều khiển, bạn có thể mở bảng điều khiển này bất cứ lúc nào bằng cách nhấn chuột và biểu tượng của *Xampp* nằm ở khay hệ thống



Bây giờ thư mục gốc (Web Root) để chứa các trang web là *C:\www*. Bạn cũng có thể Copy các File trong thư mục *C:\xampp\htdocs* vào thư mục *C:\www*, sau đó mở chương trình duyệt web (Internet Explorer, Firefox...), trong ô địa chỉ bạn nhập vào: *http://localhost* nếu hiện lên bảng thông số của *Xampp* là việc cài đặt đã thành công.

Ta kích hoạt *apache* và *mysql* để chạy chương trình:



1.4 HTML

HTML là chữ viết tắt của Hyper Text Markup Language (Ngôn ngữ hiển thị siêu văn bản).

- Một file HTML là một file text bao gồm những tag nhỏ
- Những tag hiển thị nói cho trình duyệt biết nó phải hiển thị trang đó như thế nào
- Một file HTML phải có phần mở rộng là .htm hoặc .html
- Một file HTML có thể được tạo bởi một trình soạn thảo đơn giản.

1.4.1 Các dạng thẻ HTML

- Thẻ HTML dùng để viết lên những thành tố HTML
- Thẻ HTML được bao quanh bởi hai dấu lớn hơn < và > nhỏ hơn.
- Những thẻ HTML thường có một cặp giống như và
- Thẻ thứ nhất là thẻ mở đầu và thẻ thứ hai là thẻ kết thúc.
- Dòng chữ ở giữa hai thẻ bắt đầu và kết thúc là nội dung.
- Những thẻ HTML không phân biệt in hoa và viết thường, ví dụ dạng và đều như nhau,

1.4.2 Thành phần HTML

Xét ví dụ

```
<html>
```

```
<head>
```

```
<title>Viet Photoshop</title>
```

```
</head>
```

```
<body>
```

```
www.vietphotoshop.com and www.bantayden.com. <b>Photoshop and web design resources
```

```
</b>
```

```
</body>
```

```
</html>
```

*Photoshop and web design resources *

Thành phần của HTML bắt đầu với thẻ: ``

Nội dung của nó là: *Photoshop and web design resources*

Thành phần của HTML kết thúc với thẻ: ``

Mục đích của thẻ `` là để xác định một thành phần của HTML phải được thể hiện dưới dạng *in đậm*

Phần này bắt đầu bằng thẻ bắt đầu `<body>` và kết thúc bằng thẻ kết thúc `</body>`. Mục đích của thẻ `<body>` là xác định thành phần của HTML bao gồm nội dung của tài liệu.

1.4.3 Các thuộc tính của thẻ HTML

Những thẻ HTML đều có những thuộc tính riêng. Những thuộc tính này cung cấp thông tin về thành phần HTML của trang web. Tag này xác định thành phần thân của trang HTML: `<body>`. Với một thuộc tính thêm vào là `bgcolor`, bạn có thể báo cho trình duyệt biết rằng màu nền của trang này là màu đỏ, giống như sau: `<body bgcolor="red">` hoặc `<body bgcolor="#E6E6E6">` (`#E6E6E6` là giá trị hex của màu)

Thẻ này sẽ xác định dạng bảng HTML: `<table>` với một thuộc tính đường viền (`border`), bạn có thể báo cho trình duyệt biết rằng bảng sẽ không có đường viền: `<table border="0">`

Thuộc tính luôn luôn đi kèm một cặp như `name/value`: `name="value"` (tên="giá trị")
thuộc tính luôn luôn được thêm vào thẻ mở đầu của thành phần HTML.

1.4.4 Đoạn văn - paragraphs

Paragraphs được định dạng bởi thẻ `<p>`.

`<p>Đây là đoạn văn</p>`

`<p>Đây là một đoạn văn khác</p>`

HTML sẽ tự động thêm một dòng trắng trước và sau mỗi heading.

1.4.5 Những thẻ HTML cơ bản

Tag	Mô Tả
<code><html></code>	Xác định một văn bản dạng HTML
<code><body></code>	Xác định phần thân của tài liệu
<code><h1></code> to <code><h6></code>	Xác định header từ 1 đến 6
<code><p></code>	Xác định một đoạn văn
<code>
</code>	Chèn một dòng trắng
<code><hr></code>	Xác định một đường thẳng
<code><!--></code>	Xác định vùng chú thích

1.4.6 Những thẻ dùng để định dạng văn bản

Tag	Mô Tả
	Định dạng chữ đậm
<big>	Định dạng chữ lớn
	Định dạng kiểu chữ được nhấn mạnh
<i>	Chữ in <i>ngghiêng</i>
<small>	Chữ nhỏ
	Chữ đậm
<sub>	định dạng subscripted (chữ nhỏ)
<sup>	Định dạng superscripted (chữ lên cao)
<ins>	Dạng chữ mới chèn thêm
	Dạng chữ bị xóa
<s>	Hết hỗ trợ. Thay bằng
<strike>	Hết hỗ trợ. Thay bằng
<u>	Gạch dưới

1.4.7 Thẻ table

Tag	Mô Tả
<table>	Vẽ bảng
<th>	hàng đầu của bảng
<tr>	hàng trong bảng
<td>	ô trong hàng
<caption>	nhãn của bảng
<colgroup>	nhóm các cột
<col>	Định các thuộc tính của cột
<thead>	Hàng Đầu bảng
<tbody>	Thân của bảng
<tfoot>	Hàng cuối bảng

1.4.8 Thẻ của form

Tag	Mô Tả
<form>	Kiểu form để nhập thông tin
<input>	Một ô nhập liệu

<textarea>	Vùng nhập liệu có nhiều hàng
<label>	Nhãn
<fieldset>	Nhóm các vùng nhập với nhau
<legend>	nhãn của 1 fieldset
<select>	Danh sách chọn
<optgroup>	nhóm các phần tử trong danh sách chọn
<option>	1 phần tử trong danh sách chọn
<button>	Nút bấm
<isindex>	Hết hỗ trợ. Dùng <input> thay thế

1.4.9 Thẻ Image

Tag	Mô Tả
	Hình ảnh
<map>	Định nghĩa map (sơ đồ trên 1 hình)
<area>	Định Area, 1 vùng nhỏ tương ứng trên sơ đồ

1.5 PHP

1.5.1 Giới thiệu về ngôn ngữ php

PHP (viết tắt hội quy "PHP: Hypertext Preprocessor") là một ngôn ngữ lập trình kịch bản hay một loại mã lệnh chủ yếu được dùng để phát triển các ứng dụng viết cho máy chủ, mã nguồn mở, dùng cho mục đích tổng quát. Nó rất thích hợp với web và có thể dễ dàng nhúng vào trang HTML. Do được tối ưu hóa cho các ứng dụng web, tốc độ nhanh, nhỏ gọn, cú pháp giống C và Java, dễ học và thời gian xây dựng sản phẩm tương đối ngắn hơn so với các ngôn ngữ khác nên PHP đã nhanh chóng trở thành một ngôn ngữ lập trình web phổ biến nhất thế giới.

PHP là ngôn ngữ script trên server được thiết kế để dễ dàng xây dựng các trang Web động. Mã PHP có thể thực thi trên Webservice để tạo ra mã HTML và xuất ra trình duyệt web theo yêu cầu của người sử dụng. Ngôn ngữ PHP ra đời năm 1994 Rasmus Lerdorf sau đó được phát triển bởi nhiều người trải qua nhiều phiên bản. Phiên bản hiện tại là PHP 5 đã được công bố 7/2004. Có nhiều lý do khiến cho việc sử dụng ngôn ngữ này chiếm ưu thế xin nêu ra đây một số lý do cơ bản:

- Mã nguồn mở (open source code)
- Miễn phí, download dễ dàng từ Internet
- Ngôn ngữ rất dễ học, dễ viết.

-Mã nguồn không phải sửa lại nhiều khi viết chạy cho các hệ điều hành từ Windows, Linux, Unix.

- Rất đơn giản trong việc kết nối với nhiều nguồn DBMS, ví dụ như : MySQL, Microsoft SQL Server 2000, Oracle, PostgreSQL, Adabas, dBase, Empress, FilePro, Informix, InterBase, mSQL, Solid, Sybase, Velocis và nhiều hệ thống CSDL thuộc Hệ Điều Hành Unix (Unix dbm) cùng bất cứ DBMS nào có sự hỗ trợ cơ chế ODBC (Open Database Connectivity) ví dụ như DB2 của IBM.

1.5.2 Cấu trúc cơ bản

PHP cũng có thể bắt đầu và kết thúc giống với ngôn ngữ HTML. Chỉ khác, đối với PHP chúng ta có nhiều cách để thể hiện.

Cách 1 : Cú pháp chính: `<?php Mã lệnh PHP ?>`

Cách 2: Cú pháp ngắn gọn `<? Mã lệnh PHP ?>`

Cách 3: Cú pháp giống với ASP. `<% Mã lệnh PHP %>`

Cách 4: Cú pháp bắt đầu bằng script `<script language=php> </script>`

`<?`

```
Echo "PHP is simple"; //day la vi du ve code PHP
```

```
/*voi cu phap nay ta
```

```
co the chu thich mot cum ma lenh*/
```

```
?>
```

1.5.3 Xuất giá trị ra trình duyệt

Để xuất dữ liệu ra trình duyệt chúng ta có những dòng cú pháp sau :

```
+ Echo "Thông tin";
```

```
+ Printf "Thông tin";
```

Thông tin bao gồm : biến, chuỗi, hoặc lệnh HTML

```
<?php
```

```
Echo "hello word";
```

```
Printf"<br><font color=red>Who are you ?</font>";
```

```
?>
```

Nếu giữa hai chuỗi muốn liên kết với nhau ta sử dụng dấu "."

```
<?php
```

```
Echo "hello"."who are you ?"
```

```
?>
```

1.5.4 Khái niệm biến, hằng, chuỗi và các kiểu dữ liệu

a) Khai báo biến trong PHP

Biến được xem là vùng nhớ dữ liệu tạm thời. Và giá trị có thể thay đổi được. Biến được bắt đầu bằng ký hiệu "\$". Và theo sau chúng là một từ, một cụm từ nhưng phải viết liền hoặc có gạch dưới.

Một biến được xem là hợp lệ khi nó thỏa mãn các yếu tố :

+Tên của biến phải bắt đầu bằng dấu gạch dưới và theo sau là các ký tự, số hay dấu gạch dưới.

+Tên của biến không được phép trùng với các từ khóa của PHP. Trong PHP để sử dụng 1 biến chúng ta thường phải khai báo trước, tuy nhiên đối với các lập trình viên khi sử dụng họ thường xử lý cùng một lúc các công việc, nghĩa là vừa khai báo vừa gán dữ liệu cho biến. Bản thân biến cũng có thể gán cho các kiểu dữ liệu khác. Và tùy theo ý định của người lập trình mong muốn trên chúng.

ví dụ :

```
<?
```

```
$a= 100 //bien a co gia tri bang 100.
```

```
$a= "PHP is easy" //bien a co gia tri "PHP is easy".
```

```
Biena=123 //co loi vi bat dau 1 bien phai co dau "$"
```

```
$123a="PHP" // co loi vi phan ten bat dau dang so
```

```
?>
```

b) Khái niệm về hằng trong PHP

Nếu biến là cái có thể thay đổi được thì ngược lại hằng là cái chúng ta không thể thay đổi được. Hằng trong PHP được định nghĩa bởi hàm define theo cú pháp: define (string tên_hằng, giá_trị_hằng). Cũng giống với biến hằng được xem là hợp lệ thì chúng phải đáp ứng 1 số yếu tố :

+ Hằng không có dấu "\$" ở trước tên.

+ Hằng có thể truy cập bất cứ vị trí nào trong mã lệnh

+ Hằng chỉ được phép gán giá trị duy nhất 1 lần.

+ Hằng thường viết bằng chữ in để phân biệt với biến

Ví dụ :

```
<?
```

```
define ("C", "COMPANY");
```

```
define ("YELLOW", "ffff00");
```

```
echo "gia tri cua C la". C;
```

```
?>
```

c) Khái niệm về chuỗi.

Chuỗi là một nhóm các ký tự, số, khoảng trắng, dấu ngắt được đặt trong các dấu nháy. Ví dụ: 'Nam' "welcome to VietNam" Để tạo 1 biến chuỗi, chúng ta phải gán giá trị chuỗi cho 1 biến hợp lệ. Ví dụ: \$frist_name= "Nguyen"; \$last_name= 'Van A'; Để liên kết 1 chuỗi và 1 biến chúng ta thường sử dụng dấu "."

Ví dụ:

```
<?php
$test="QHOnline.Info";
echo "Welcome to".$test;
echo "<br><font color=red>welcome to".$test."</font><br>";
?>
```

d) Kiểu dữ liệu trong PHP

Các kiểu dữ liệu khác nhau chiếm các lượng bộ nhớ khác nhau và có thể được xử lý theo cách khác nhau khi chúng được theo tác trong 1 script. Trong PHP chúng ta có 6 kiểu dữ liệu chính như sau :

Kiểu dữ liệu	Ví dụ	Mô tả
Integer	10	Một số nguyên
Double	5.208	Kiểu số thực
String	"how are you ?"	Một tập hợp các ký tự
Boolean	True or false	Giá trị true hoặc false
Object	Hướng đối tượng trong php	
Array	Mảng trong php, chứa các phần tử	

1.5.5 Các toán tử

Toán tử kết hợp các giá trị hoặc biểu thức lại với nhau và tạo ra một giá trị mới. Ví dụ trong biểu thức 1+2 thì + là toán tử kết hợp hai giá trị 1 và 2 lại với nhau tạo ra giá trị mới là 3.

Các toán tử trong PHP được chia thành 3 nhóm:

Các toán tử áp dụng trên 1 giá trị, ví dụ như toán tử ++ hoặc --

Các toán tử kết hợp 2 hoặc nhiều giá trị, ví dụ như toán tử +, -, *, /

Toán tử ?: dùng để chọn 1 trong 2 giá trị tùy thuộc vào 1 điều kiện cho trước

Thứ tự ưu tiên của toán tử

Quy tắc liên kết	Toán tử	Ghi chú
	new	Tạo một đối tượng từ 1 class, toán tử này chỉ áp

		dụng trên 1 toán hạng nên không có quy tắc liên kết
Bên phải trước	[Toán tử truy cập một phần tử trong mảng
	++, --	Tăng/giảm một đơn vị toán tử này chỉ áp dụng trên 1 toán hạng nên không có quy tắc liên kết
	!, ~, -, (int),(float),(string),(array),(object), @	Các toán tử này chỉ áp dụng trên 1 toán hạng nên không có quy tắc liên kết
Bên trái trước	*, /, %	
Bên trái trước	+, -, .	
Bên trái trước	<<, >>	
	==, !=, ===, !==	Toán tử so sánh chỉ áp dụng trên 2 toán hạng nên không có quy tắc liên kết
Bên trái trước	&	
Bên trái trước	^	
Bên trái trước		
Bên trái trước	&&	
Bên trái trước		
Bên trái trước	?:	
Bên trái trước	+=, -=, *=, /=, .+=, %=, &=, =, ^=, <<=, >>=	
Bên trái trước	and	
Bên trái trước	xor	
Bên trái trước	or	
Bên trái trước	,	

1.5.6 Các câu lệnh điều khiển

a) Câu lệnh if

Cú pháp đơn giản nhất của câu lệnh if có dạng như sau:

if (biểu thức)

câu lệnh;

Câu lệnh if trên được diễn giải như sau: nếu biểu thức trả về giá trị TRUE (hoặc tương đương với TRUE sau khi chuyển đổi) thì câu lệnh sẽ được thực thi; ngược lại (khi biểu thức trả về giá trị FALSE) thì bỏ qua không thực thi câu lệnh nữa.

Cú pháp nâng cao của câu lệnh if có dạng như sau:

if (biểu thức)

câu lệnh 1;

else

câu lệnh 2;

Câu lệnh if trên được diễn giải như sau: nếu biểu thức trả về giá trị TRUE thì câu lệnh 1 sẽ được thi hành, ngược lại thì câu lệnh 2 sẽ được thi hành.

Ngoài ra PHP còn cung cấp từ khoá elseif, chính là ghép giữa từ khoá else và if.

b) Câu lệnh while

Câu lệnh while dùng để tạo 1 vòng lặp, cú pháp của câu lệnh này như sau:

while (biểu thức)

câu lệnh;

Được diễn giải như sau: trong khi biểu thức còn trả về giá trị TRUE thì tiếp tục thực hiện câu lệnh, sau khi thực hiện câu lệnh thì kiểm tra lại biểu thức, nếu vẫn còn trả về giá trị TRUE thì lại tiếp tục thực hiện câu lệnh... cứ tiếp tục như vậy cho tới khi nào biểu thức trả về giá trị FALSE thì ngưng.

Một ví dụ in ra các số từ 1 tới 10 với câu lệnh while:

```
<?php
```

```
 $\$i = 1;$ 
```

```
while (  $\$i \leq 10$  ) {
```

```
echo  $\$i$ , "\n";
```

```
 $\$i++;$ 
```

```
} //end while
```

```
?>
```

Ghi chú: Câu lệnh $\$i++;$ tương đương với $\$i = \$i+1;$, câu lệnh này sẽ tăng giá trị của $\$i$ lên 1 qua mỗi lần lặp.

c) Câu lệnh do-while

Câu lệnh do-while cũng tương tự như câu lệnh white, chỉ khác một điểm là câu lệnh được thực hiện trước rồi biểu thức mới được kiểm tra sau, nếu biểu thức còn trả về giá trị TRUE thì tiếp tục thực hiện câu lệnh. Cú pháp của câu lệnh do-while như sau:

```
do {
```

câu lệnh;

```
} while ( biểu thức );
```

Một ví dụ in ra các số từ 1 tới 10 với câu lệnh do-while:

```
<?php  
$i = 1;  
do {  
echo $i, "\n";  
$i++;  
} while ( $i < 10 );  
?>
```

d) Câu lệnh for

Câu lệnh for cũng dùng để tạo vòng lặp. Đây là một trong những câu lệnh phức tạp nhất của PHP, cú pháp của nó như sau:

```
for ( biểu thức 1; biểu thức 2; biểu thức 3 )
```

câu lệnh;

Được diễn giải như sau:

Đầu tiên biểu thức 1 được thực hiện,

Tiếp theo biểu thức 2 được kiểm tra

Nếu trả về TRUE thì câu lệnh được thực hiện và sau đó thực hiện biểu thức 3.

Nếu trả về FALSE thì kết thúc câu lệnh for.

Kiểm tra lại biểu thức 2 và lặp lại quá trình như trên.

Một ví dụ in ra các số từ 1 tới 10 với câu lệnh for:

```
<?php  
for ( $i = 0; $i < 10; $i++ ) {  
echo $i, "\n";  
} //end for  
?>
```

e) Câu lệnh foreach

Câu lệnh foreach chỉ làm việc với array. Câu lệnh foreach có 2 dạng cú pháp như sau:

```
foreach ( $array as $value )
```

câu lệnh;

```
foreach ( $array as $key => $value )
```

câu lệnh;

Ví dụ :

```

<?php
$a = array('a' => 1, 'b' => '2', 'c' => '3');
foreach ( $a as $key => $value ) {
echo $key, "=", $value, "\n";
} //end foreach
?>

```

Chương trình trên sẽ in ra 3 chuỗi a=1, b=2 và c=3.

f) Câu lệnh switch

Câu lệnh switch hoạt động như là 1 loạt câu lệnh if ghép lại với nhau.

```

switch ( $a ) {
case "abc";
echo "Giá trị của a là abc";
break;
case "def";
echo "Giá trị của a là def";
break;
case "123";
echo "Giá trị của a là 123";
break;
default:
echo "Giá trị khác";
} //end switch

```

g) Câu lệnh break

Câu lệnh break sẽ dừng việc thực thi của các vòng lặp for, foreach, while, do-while và switch.

Ở phần trước ta đã thấy câu lệnh break được sử dụng trong câu lệnh switch. Nếu không có break, câu lệnh switch ở phần trước sẽ thành:

```

switch ( $a ) {
case "abc";
echo "Giá trị của a là abc";
case "def";
echo "Giá trị của a là def";
case "123";
echo "Giá trị của a là 123";
default:

```

```
echo "Giá trị khác";
```

```
} //end switch
```

Nếu giá trị của \$a là "abc" thì cả 4 chuỗi "Giá trị của a là abc", "Giá trị của a là def", "Giá trị của a là 123" và "Giá trị khác" sẽ được in ra.; nếu \$a mang giá trị "def" thì 3 chuỗi "Giá trị của a là def", "Giá trị của a là 123" và "Giá trị khác" sẽ được in ra.

Ở đây ta muốn chỉ có 1 dòng duy nhất in ra tương ứng với giá trị của biến \$a, nên ta thêm các câu lệnh break vào các phân case, để khi in ra chuỗi tương ứng với giá trị \$a thì ta thoát ra khỏi câu lệnh switch.

Một ví dụ sử dụng câu lệnh break trong vòng lặp for:

```
for ( $i=1; $i<=10; $i++ ) {  
echo $i;  
if ( $i == 5 ) break;  
}
```

Vòng lặp for ở trên thay vì in ra 10 số từ 1 đến 10, vòng lặp chỉ in ra 5 số từ 1 đến 5 mà thôi vì khi \$i đạt giá trị 5, vòng lặp sẽ kết thúc do câu lệnh break.

Cách dùng câu lệnh break trong các vòng lặp foreach, while và do-while cũng tương tự.

h) Câu lệnh continue

Câu lệnh continue áp dụng lên các vòng lặp, lệnh continue sẽ bỏ qua lần lặp hiện thời và tiếp tục thực hiện các lần lặp tiếp theo. Để hiểu rõ hơn ta hãy xem ví dụ sau:

```
for ( $i=1; $i<=5; $i++ ) {  
if ( $i == 2 ) continue;  
echo $i;  
}
```

Khi \$i đạt giá trị 2, câu lệnh echo \$i; sẽ được bỏ qua không thì hành nữa do câu lệnh continue. Và như vậy, đoạn lệnh trên khi chạy sẽ in ra các giá trị 1,3,4,5 (không có giá trị 2).

Cách dùng của câu lệnh continue trong các vòng lặp foreach, while, do-while cũng tương tự.

1.5.7 Hàm

Trong lập trình, có một số đoạn mã được dùng nhiều lần ở nhiều nơi khác nhau trong chương trình. Sẽ rất phiền và khó sửa lỗi nếu như ta phải viết lặp đi lặp lại 1 đoạn mã đó ở nhiều nơi. PHP cung cấp một giải pháp đó là hàm do người dùng định nghĩa. Ta có thể đưa đoạn mã đó vào trong 1 hàm, và ở chỗ nào cần dùng đoạn mã đó ta chỉ cần gọi hàm, khi cần sửa đổi, ta chỉ cần sửa đổi 1 chỗ duy nhất là nội dung của hàm chứ không cần phải sửa ở nhiều nơi trong chương trình.

a) Cú pháp để tạo 1 hàm do người dùng định nghĩa như sau:

```
function tênHàm($tham_số1, $tham_số2, ..., $tham_sốn) {
//thân hàm
echo "Testing";
return $kết_quả_trả_về;
} //end
```

Khi cần sử dụng hàm ở chỗ nào, ta chỉ cần dùng cú pháp *tênHàm(các tham số cần thiết)*;

Lưu ý: Tên hàm cũng như tên biến chỉ bao gồm các ký tự chữ cái (a..z, A..Z), chữ số (0..9) và ký tự gạch dưới (_), ngoài ra tên hàm không được bắt đầu bằng chữ số, nhưng được phép bắt đầu bằng ký tự gạch dưới (tên hàm khác với tên biến chỗ này). Tên hàm trong PHP phân biệt chữ hoa và chữ thường, tức là testing và Testing là 2 tên hàm khác nhau.

b) Tham số của hàm

Hàm có thể nhận vào các tham số, ví dụ:

```
<?php
function testing($a) {
echo "Tham số là $a";
} //end testing
testing(123);
testing("abc");
?>
```

Ta có thể gán giá trị mặc định cho tham số của hàm:

```
<?php
function testing($a="mặc định") {
echo "Tham số là $a";
} //end testing
testing();
?>
```

Khi tham số tương ứng của hàm không được truyền, tham số đó sẽ nhận giá trị mặc định. Đoạn chương trình ví dụ ở trên khi chạy sẽ in ra dòng Tham số là mặc định.

1.6 CSS

1.6.1 CSS là gì?

Còn CSS (Cascading Style Sheets mà Pearl tạm dịch là tờ mẫu theo Style Sheets) là một ngôn ngữ quy định cách trình bày cho các tài liệu viết bằng HTML, XHTML, XML, SVG, hay UML,...

1.6.2 Ưu điểm khi sử dụng CSS

Sử dụng các mã định dạng trực tiếp trong HTML tốn hao nhiều thời gian thiết kế cũng như dung lượng lưu trữ trên đĩa cứng. Trong khi đó CSS đưa ra phương thức “tờ mẫu ngoài” giúp áp dụng một khuôn mẫu chuẩn từ một file CSS ở ngoài. Nó thật sự có hiệu quả đồng bộ khi bạn tạo một website có hàng trăm trang hay cả khi bạn muốn thay đổi một thuộc tính trình bày nào đó.

CSS còn cho phép bạn áp đặt những kiểu trình bày thích hợp hơn cho các phương tiện khác nhau như màn hình máy tính, máy in, điện thoại,...

Cú pháp CSS cơ bản: Selector { property:value; } Trong đó:

+ *Selector*: Các đối tượng mà chúng ta sẽ áp dụng các thuộc tính trình bày. Nó là các tag HTML, class hay id (chúng ta sẽ học về 2 thành phần này ở bài học sau). Ví dụ: body, h2, p, img, #title, #content, .username,... Trong CSS ngoài viết tên selector theo tên tag, class, id. Chúng ta còn có thể viết tên selector theo phân cấp như để chỉ các ảnh ở trong #entry, chúng ta viết selector là #entry img, như vậy thì các thuộc tính chỉ định sẽ chỉ áp dụng riêng cho các ảnh nằm trong #entry.

Khi viết tên cho class, đôi khi sẽ có nhiều thành phần có cùng class đó, ví dụ như thẻ img và thẻ a cùng có class tên vistors nhưng đây lại là hai đối tượng khác nhau, 1 cái là ảnh của người thăm, 1 cái là liên kết tới trang người thăm.

+ *Property*: Chính là các thuộc tính quy định cách trình bày. Ví dụ: background-color, font-family, color, padding, margin,...

+ *Value*: Giá trị của thuộc tính. Ví dụ: như ví dụ trên value chính là #FFF dùng để định màu trắng cho nền trang.

Chú thích trong CSS: Cũng như nhiều ngôn ngữ web khác. Trong CSS, chúng ta cũng có thể viết chú thích cho các đoạn code để dễ dàng tìm, sửa chữa trong những lần cập nhật sau. Chú thích trong CSS được viết như sau /* Nội dung chú thích */ Ví dụ: /* Màu chữ cho trang web */ body { color:red }

1.6.3 Đơn vị CSS

Trong CSS2 hỗ trợ các loại đơn vị là đơn vị đo chiều dài và đơn vị đo góc, thời gian, cường độ âm thanh và màu sắc.

Đơn vị chiều dài

Đơn vị	Mô tả	Đơn vị	Mô tả
%	Phần trăm	ex	1 ex bằng chiều cao của chữ x in thường của font hiện hành. Do đó, đơn vị này không những phụ thuộc trên kích cỡ

			font chữ mà còn phụ thuộc loại font chữ vì cùng 1 cỡ 14px nhưng chiều cao chữ x của font Times và font Tohama là khác nhau.
In	Inch (1 inch =2.54 cm)	pt	Point (1 pt = 1/72 inch)
cm	Centimeter	Pc	Pica (1 pc = 12 pt)
mm	milimeter	Px	Pixels (điểm ảnh trên màn hình máy tính)
em	1 em tương đương kích thước font hiện hành, nếu font hiện hành có kích cỡ 14px thì 1 em = 14 px. Đây là một đơn vị rất hữu ích trong việc hiển thị trang web.		

Đơn vị màu sắc

Đơn vị	Mô tả
Color-name	Tên màu tiếng Anh. Ví dụ: black, white, red, green, blue, cyan, magenta,...
RGB (r,g,b)	Màu RGB với 3 giá trị R, G, B có trị từ 0 – 255 kết hợp với nhau tạo ra vô số màu.
RGB (%r,%g,%b)	Màu RGB với 3 giá trị R, G, B có trị từ 0 – 100% kết hợp.
Hexadecimal RGB	Mã màu RGB dạng hệ thập lục. Ví dụ: #FFFFFF: trắng, #000000: đen, #FF00FF: đỏ tươi.

1.6.4 Vị trí đặt CSS

+ *Cách 1*: Nội tuyến (kiểu thuộc tính)

Ở ví dụ sau chúng ta sẽ tiến hành định nền màu trắng cho trang và màu chữ xanh lá cho đoạn văn bản như sau:

```
<html>
```

```
<head>
```

```
<title>Ví dụ</title>
```

```
</head>
```

```
<body style="background-color=#FFF;">
<p style="color:green">^_^ Welcome To WallPearl's Blog ^_^</p>
</body>
</html>
```

+ *Cách 2*: Bên trong (thẻ style)

Cũng ví dụ làm trang web có màu nền trắng, đoạn văn bản chữ xanh lá, chúng ta sẽ thể hiện như sau:

```
<html>
<head>
<title>Ví dụ</title>
<style type="text/css">
body { background-color:#FFF }
p { color:#00FF00 } </style>
</head>
<body> <p>^_^ Welcome To WallPearl's Blog ^_^</p> </body> </html>
```

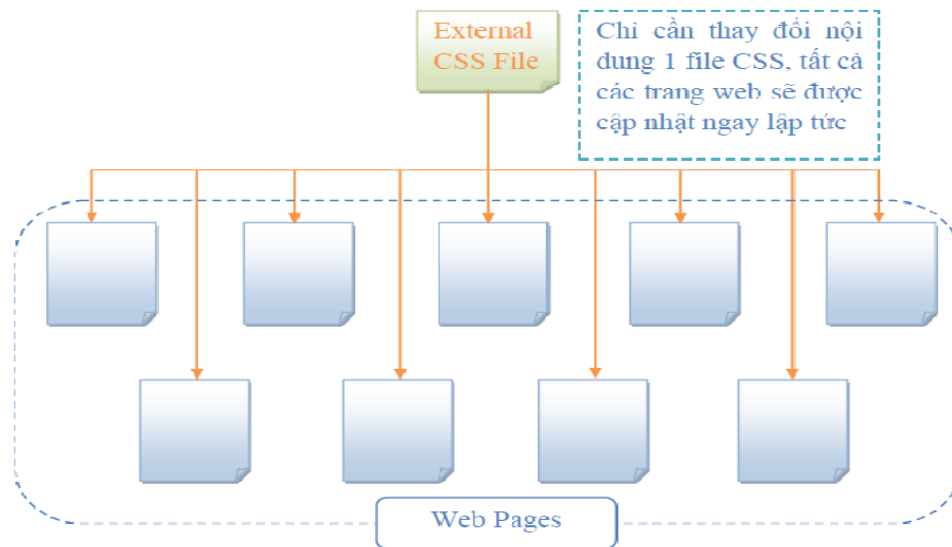
Lưu ý: Thẻ style nên đặt trong thẻ head.

Như ở ví dụ trên, nếu trình duyệt không hỗ trợ thẻ style thì 2 dòng CSS: body { background-color:#FFF } p { color:#00FF00 } sẽ hiện ra trên trình duyệt. Để tránh tình trạng này, bạn nên đưa vào thêm dấu <!-- ở trước và --> ở sau khối code CSS. Như ví dụ trên sẽ viết lại là:

```
<style type="text/css">
<!-- body { background-color:#FFF }
      p { color:#00FF00 } -->
</style>
```

+ *Cách 3*: Bên ngoài (liên kết với một file CSS bên ngoài)

Tương tự như cách 2 nhưng thay vì đặt tất cả các mã CSS trong thẻ style chúng ta sẽ đưa chúng vào trong một file CSS (có phần mở rộng .css) bên ngoài và liên kết nó vào trang web bằng thuộc tính href trong thẻ link.



1.6.5 Màu nền (thuộc tính *background-color*)

Thuộc tính `background-color` giúp định màu nền cho một thành phần trên trang web. Các giá trị mã màu của `background-color` cũng giống như `color` nhưng có thêm giá trị `transparent` để tạo nền trong suốt. Ví dụ sau đây sẽ chỉ cho chúng ta biết cách sử dụng thuộc tính `background-color` để định màu nền cho cả trang web, các thành phần `h1`, `h2` lần lượt là xanh lơ, đỏ và cam.

```
body {
    background-color:cyan
}
h1 {
    background-color:red
}
h2 {
    background-color:orange
}
```

1.6.6 Ảnh nền (thuộc tính *background-image*)

Ví dụ sẽ viết CSS để đặt một ảnh có tên `logo.png` làm ảnh nền trang web như sau:

```
body {
    background-image:url(logo.png)
}
h1 {
    background-color:red
```

```

    }
h2 {
    background-color:orange
}
p {
    background-color: FDC689
}

```

Như các bạn đã thấy chúng ta sẽ phải chỉ định đường dẫn của ảnh trong cặp ngoặc đơn sau url. Do ảnh đặt trong cùng thư mục với file style3.css nên chúng ta chỉ cần ghi abc.jpg. Nhưng nếu chúng ta tạo thêm một thư mục img trong thư mục thì chúng ta sẽ phải ghi là background-image:url(img/abc.jpg).

1.6.7 Lặp lại ảnh nền (thuộc tính background-repeat)

- + repeat-x: Chỉ lặp lại ảnh theo phương ngang.
- + repeat-y: Chỉ lặp lại ảnh theo phương dọc.
- + repeat: Lặp lại ảnh theo cả 2 phương, đây là giá trị mặc định.
- + no-repeat: Không lặp lại ảnh

1.6.8 Khóa ảnh nền (thuộc tính background-attachment)

Background-attachment là một thuộc tính cho phép bạn xác định tính cố định của ảnh nền so với nội dung trang web. Thuộc tính này có 2 giá trị:

- + scroll: Ảnh nền sẽ cuộn cùng nội dung trang web, đây là giá trị mặc định.
- + fixed: Cố định ảnh nền so với nội dung trang web. Khi áp dụng giá trị này, ảnh nền sẽ đứng yên khi bạn đang cuộn trang web.

1.6.9 Định vị ảnh nền (thuộc tính background-position)

Theo mặc định ảnh nền khi được chèn sẽ nằm ở góc trên, bên trái màn hình. Tuy nhiên với thuộc tính background-position bạn sẽ có thể đặt ảnh nền ở bất cứ vị trí nào (trong không gian của thành phần mà nó làm nền).

Background-position sẽ dùng một cặp 2 giá trị để biểu diễn tọa độ đặt ảnh nền. Có khá nhiều kiểu giá trị cho thuộc tính position. Như đơn vị chính xác như centimeters, pixels, inches,... hay các đơn vị qui đổi như %, hoặc các vị trí đặt biệt như top, bottom, left, right.

Giá trị	Ý nghĩa
Background-position:5cm 2cm	Ảnh được định vị 5cm từ trái qua và 2cm từ trên xuống.
Background-position:20% 30%	Ảnh được định vị 20% từ trái qua và 30% từ trên xuống.

1.6.10 Font Chữ

a) Thuộc tính font-family

Có công dụng định nghĩa một danh sách ưu tiên các font sẽ được dùng để hiển thị một thành phần trang web. Theo đó, thì font đầu tiên được liệt kê trong danh sách sẽ được dùng để hiển thị trang web.

Có hai loại tên font được dùng để chỉ định trong font-family: family-names và generic families.

+ Family-names: Tên cụ thể của một font. Ví dụ: Arial, Verdana, Tohama,...

+ Generic families: Tên của một họ gồm nhiều font. Ví dụ: sans-serif, serif,...

Ví dụ sau chúng ta sẽ viết CSS để quy định font chữ dùng cho cả trang web là Times New Roman, Tohama, sans-serif, và font chữ dùng để hiển thị các tiêu đề h1, h2, h3 sẽ là Arial, Verdana và các font họ serif.

```
body { font-family:"Times New Roman",Tohama,sans-serif }
```

```
h1, h2, h3 { font-family:arial,verdana,serif }
```

Mở trang web trong trình duyệt và kiểm tra kết quả. Chúng ta thấy phần tiêu đề sẽ được ưu tiên hiển thị bằng font Arial, nếu trên máy không có font này thì font Verdana sẽ được ưu tiên và kế đó sẽ là các font thuộc họ serif.

Chú ý: Đối với các font có khoảng trắng trong tên như Times New Roman cần được đặt trong dấu ngoặc kép.

b) Thuộc tính font-style:

Thuộc tính font-style định nghĩa việc áp dụng các kiểu in thường (normal), in nghiêng (italic) hay xiên (oblique) lên các thành phần trang web.

Trong ví dụ bên dưới chúng ta sẽ thử thực hiện áp dụng kiểu in nghiêng cho thành phần h1 và kiểu xiên cho h2.

```
h1
```

```
{
  font-style:italic;
}
```

```
h2
```

```
{
  font-style:oblique;
}
```

c) Thuộc tính font-variant:

Thuộc tính font-variant được dùng để chọn giữa chế độ bình thường và small-caps của một font chữ. Một font small-caps là một font sử dụng chữ in hoa có kích cỡ nhỏ hơn in hoa chuẩn để thay thế những chữ in thường.

Trong ví dụ sau chúng ta sẽ sử dụng kiểu small-caps cho phần h1

```
h1
{
  font-variant:small-caps
}
```

d) Thuộc tính font-weight:

Thuộc tính font-weight mô tả cách thức thể hiện của font chữ là ở dạng bình thường (normal) hay in đậm (bold). Ngoài ra, một số trình duyệt cũng hỗ trợ mô tả độ in đậm bằng các con số từ 100 – 900.

Thử in đậm phần p:

```
p
{
  font-weight:bold
}
```

e) Thuộc tính font-size:

Kích thước của một font được định bởi thuộc tính font-size. Thuộc tính này nhận các giá trị đơn vị đo hỗ trợ bởi CSS bên cạnh các giá trị xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger. Tùy theo mục đích sử dụng của website bạn có thể lựa chọn những đơn vị phù hợp

Ở ví dụ sau trang web sẽ có kích cỡ font là 20px, h1 là 3em = 3 x 20 = 60px, h2 là 2em = 40px.

```
body
{
  font-size:20px
}
```

```
h1 {
  font-size:3em
}
```

```
h2 {
```

```
font-size:2em
}
```

1.6.11 Text

a) Màu chữ (thuộc tính color):

Để định màu chữ cho một thành phần nào đó trên trang web chúng ta sử dụng thuộc tính color. Giá trị của thuộc tính này là các giá trị màu CSS hỗ trợ.

Ví dụ sau chúng ta sẽ viết CSS để định màu chữ chung cho một trang web là đen, cho tiêu đề h1 màu xanh da trời, cho tiêu đề h2 màu xanh lá chúng ta sẽ làm như

```
body {
    color:#000
}

h1 {
    color:#0000FF
}

h2 {
    color:#00FF00
}
```

b) Thuộc tính text-align

Thuộc tính text-align giúp bạn thêm các canh chỉnh văn bản cho các thành phần trong trang web. Thuộc tính này có tất cả 4 giá trị : left (canh trái – mặc định), right (canh phải), center (canh giữa) và justify (canh đều).

Trong ví dụ sau chúng ta sẽ thực hiện canh phải các thành phần h1, h2 và canh đều đối với thành phần <p> :

```
h1, h2 {
    text-align:right
}

p {
    text-align:justify
}
```

c) Thuộc tính letter-spacing:

Thuộc tính letter-spacing được dùng để định khoảng cách giữa các ký tự trong một đoạn văn bản. Muốn định khoảng cách giữa các ký tự trong thành phần h1, h2 là 7px và thành phần <p> là 5px chúng ta sẽ viết CSS sau:

```
h1, h2 {  
    letter-spacing:7px  
}  
p {  
    letter-spacing:5px  
}
```

d) Thuộc tính text-decoration:

Thuộc tính text-decoration giúp bạn thêm các hiệu ứng gạch chân (underline), gạch xiên (line-through), gạch đầu (overline), và một hiệu ứng đặc biệt là văn bản nhấp nháy (blink).

Ví dụ sau chúng ta sẽ định dạng gạch chân cho thành phần h1, gạch đầu thành phần h2

```
h1 {  
    text-decoration:underline  
}  
  
h2 {  
    text-decoration:overline  
}
```

e) Thuộc tính text-transform:

Text-transform là thuộc tính qui định chế độ in hoa hay in thường của văn bản mà không phụ thuộc vào văn bản gốc trên HTML. Thuộc tính này có tất cả 4 giá trị: uppercase (in hoa), lowercase (in thường), capitalize (in hoa ở ký tự đầu tiên trong mỗi từ) và none (không áp dụng hiệu ứng – mặc định).

Trong ví dụ dưới đây chúng ta sẽ định dạng cho thành phần h1 là in hoa, h2 là in hoa đầu mỗi ký tự.

```
h1 {  
    text-transform:uppercase  
}  
h2 {  
    text-transform:capitalize  
}
```

1.6.12 Pseudo-classes For Links

Một thành phần rất quan trọng trong mọi website chính là liên kết. Cũng như một đối tượng văn bản thông thường, chúng ta hoàn toàn có thể áp dụng các thuộc tính định dạng đã học ở 2 bài trước như định font chữ, gạch chân, màu chữ, ... cho một liên kết.

Pseudo-classes cho phép bạn xác định các hiệu ứng định dạng cho một đối tượng liên kết ở một trạng thái xác định như khi liên kết chưa được thăm (a:link), khi rê chuột lên liên kết (a:hover), khi liên kết được thăm (a:visited) hay khi liên kết đang được kích hoạt – đang giữ nhấn chuột (a:active).

Ví dụ 2: Ví dụ này tạo cho liên kết hiệu ứng màu các hiệu ứng tương ứng với trình trạng liên kết: các liên chưa thăm có màu xanh lá, kích cỡ font 14px; liên kết mouse over có màu đỏ tươi, kích cỡ font 1.2em, hiệu ứng nhấp nháy; liên kết đã thăm sẽ có màu xanh da trời, không có đường gạch chân; các liên kết đang kích hoạt có màu tím và font dạng small-caps, các liên kết sẽ có khung viền màu đen, kích cỡ font 14px; liên kết mouse over có nền light cyan; các liên kết đã thăm có nền light yellow.

```
a      {
        border:1px solid #000; font-size:14px
      }
a:link {
        color:#00FF00;
      }
a:hover {
        background-color:#00BFF3;
        color:#FF00FF;
        font-size:1.2em;
        text-decoration:blink
      }
a:visited {
        background-color:#FFF568;
        color:#FF0000;
        text-decoration:none
      }
a:active {
        color:#662D91;
        font-variant:small-caps
```

}

1.6.13 Class & ID

a) Nhóm các phần tử với class

Ví dụ chúng ta có một đoạn mã HTML sau đây :

```
<p>Danh Sách Các Tỉnh, Thành Phố Của Việt Nam</p>
<ul>
<li>Hà Nội</li>
<li>TP. Hồ Chí Minh</li>
<li>Đà Nẵng</li>
<li>Thừa Thiên Huế</li>
</ul>
```

Yêu cầu đặt ra là làm thế nào để tên các thành phố là màu đỏ và tên các tỉnh là màu xanh da trời. Để giải quyết vấn đề này chúng ta sẽ dùng một thuộc tính là class để tạo thành 2 nhóm là thành phố và tỉnh. Ta sẽ viết lại đoạn HTML sau thành như thế này:

```
<p>Danh Sách Các Tỉnh, Thành Phố Của Việt Nam</p>
<ul>
<li class="tp">Hà Nội</li>
<li class="tp">TP. Hồ Chí Minh</li>
<li class="tp">Đà Nẵng</li>
<li class="tinh">Thừa Thiên Huế</li>
</ul>
```

Với việc dùng class để nhóm các đối tượng như trên thì công việc của chúng ta sẽ trở nên đơn giản hơn nhiều:

```
li .tp {
    color:FF0000
}
li .tinh {
    color:0000FF
}
```

Lưu ý: Không nên đặt tên class với ký tự đầu là chữ số, nó sẽ không làm việc cho Firefox.

b) Nhận dạng phần tử với id:

Ví dụ: Cũng với đoạn HTML như ví dụ về class. Nhưng yêu cầu đặt ra là Hà Nội sẽ có màu đỏ sậm, TP. Hồ Chí Minh màu đỏ, Đà Nẵng màu đỏ tươi còn các tỉnh màu xanh da trời.

Để giải quyết vấn đề này chúng ta sẽ sử dụng thuộc tính HTML là id để nhận dạng mỗi thành phần và dùng class để nhóm các tính.

Đoạn HTML của chúng ta bây giờ sẽ là :

```
<p>Danh Sách Các Tỉnh, Thành Phố Của Việt Nam</p>
<ul>
<li id="hanoi">Hà Nội</li>
<li id="hcmc">TP. Hồ Chí Minh</li>
<li id="danang">Đà Nẵng</li>
<li class="tinh">Thừa Thiên Huế</li>
</ul>
```

Và đoạn CSS cần dùng sẽ là :

```
#hanoi { color:# 790000 }
#hcmc { color:#FF0000 }
#danang { color:#FF00FF }
.tinh { color:#0000FF }
```

- Class dùng để nhóm các đối tượng có cùng thuộc tính, do tính chất đó nó có thể được sử dụng nhiều lần.

- Id dùng để nhận dạng một đối tượng đặc trưng, id có tính duy nhất.

1.6.14 Span & Div

a) Nhóm phân tử với thẻ :

Thẻ trong HTML thật ra là một thẻ trung hòa, nó không thêm hay bớt bất cứ một thứ gì vào một tài liệu HTML cả. Nhưng chính nhờ tính chất trung hòa này mà nó lại là một công cụ đánh dấu tuyệt vời để qua đó chúng ta có thể viết CSS định dạng cho các phân tử mong muốn.

Ví dụ: Chúng ta có đoạn HTML sau trích dẫn câu nói của chủ tịch Hồ Chí Minh <p>Không có gì quý hơn độc lập, tự do.</p>

Yêu cầu ở đây là chúng ta hãy dùng CSS tô đậm 2 từ độc lập, tự do.

Để giải quyết vấn đề này, chúng ta sẽ thêm thẻ vào đoạn HTML như sau:

```
<p>Không có gì quý hơn <span class="nhanmanh">độc lập</span>, <span class="nhanmanh">tự do</span>.</p>
```

Và bây giờ chúng ta có thể viết CSS cho yêu cầu trên:

```
.nhanmanh {
    font-weight:bold
}
```

b) Nhóm khối phần tử với thẻ <div>:

Cũng như , <div> cũng là một thẻ trung hòa và được thêm vào tài liệu HTML với mục đích nhóm các phần tử lại cho mục đích định dạng bằng CSS. Tuy nhiên, điểm khác biệt là dùng để nhóm một khối phần tử trong khi đó <div> có thể nhóm một hoặc nhiều khối phần tử.

Trở lại ví dụ về danh sách tỉnh, thành trong phần class bài trước chúng ta sẽ giải quyết vấn đề bằng cách nhóm các phần tử với <div> như sau:

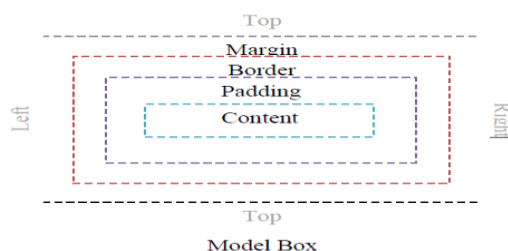
```
<p>Danh Sách Các Tỉnh, Thành Phố Việt Nam:</p>
<ul>
<div id="tp">
<li>Hà Nội</li>
<li>TP. Hồ Chí Minh</li>
<li>Đà Nẵng</li> </div> <div id="tinh">
<li>Thừa Thiên Huế</li>
</div>
</ul>
```

Và đoạn CSS cho mục đích này sẽ là:

```
#tp { color:#FF0000 }
#tinh { color:0000FF }
```

1.6.15 Box Model

Trong CSS, box model (mô hình hộp) mô tả cách mà CSS định dạng khối không gian bao quanh một thành phần. Nó bao gồm padding (vùng đệm), border (viền) và margin (canh lề) và các tùy chọn. Hình bên dưới mô tả cấu trúc minh họa mô hình hộp cho một thành phần web.



Mô hình hộp trên chỉ là một mô hình lý thuyết lý tưởng. Bên dưới đây chúng ta sẽ xét mô hình hộp của một đối tượng web cụ thể:

Ví dụ: Chúng ta có một đoạn HTML sau

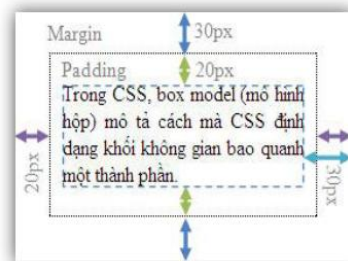
```
<p>
```

Trong CSS, mô tả cách mà CSS định dạng khối không gian bao quanh một thành phần.

</p>

Phần CSS cho đoạn HTML trên:

```
p {  
    width:200px;  
    margin:30px 20px;  
    padding:20px 10px;  
    border:1px solid #000;  
    text-align:justify  
}
```



Kết quả là:

1.6.16 Margin & Padding

a) Thuộc tính margin:

Như tất cả những ai đã học qua MS Word đều biết là trong phần thiết lập Page Setup của Word cũng có một thiết lập margin để định lề cho trang in. Tương tự, thuộc tính margin trong CSS cũng được dùng để canh lề cho cả trang web hay một thành phần web này với các thành phần web khác hay với viền trang.

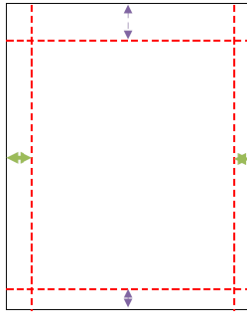
Ví dụ sau sẽ chỉ cho chúng ta biết cách canh lề cho một trang web.

```
body {  
    margin-top:80px;  
    margin-bottom:40px;  
    margin-left:50px;  
    margin-right:30px;  
    border:1px dotted #FF0000  
}
```

Hoặc gọn hơn chúng ta sẽ viết như sau:

```
body {  
    margin:80px 30px 40px 50px;  
    border:1px dotted #FF0000  
}
```

Kết quả của ví dụ trên sẽ được mô hình hóa như sau:



Cú pháp như sau: `margin:<margin-top> | <margin-right> | <margin-bottom> | <margin-left>`
 Hoặc: `margin:<value1>|< value2>` – với value 1 là giá trị `margin-top` và `margin-bottom` và value2 là giá trị `margin-left` và `margin-right`.

b) Thuộc tính padding:

Padding có thể hiểu như là một thuộc tính đệm. Padding không ảnh hưởng tới khoảng cách giữa các các đối tượng như margin mà nó chỉ quy định khoảng cách giữa phần nội dung và viền của một đối tượng

Cú pháp: Tương tự margin.

`Padding:<padding-top> | <padding-right> | <padding-bottom> | <padding-left>`

c) Border

Border là một thành phần quan trọng trong một trang web. Nó thường được dùng trong trang trí, đóng khung cho một đối tượng cần nhấn mạnh, phân cách các đối tượng giúp trang web trông dễ nhìn hơn,...

d) Thuộc tính border-width:

Border-width là một thuộc tính CSS quy định độ rộng cho viền của một đối tượng web. Thuộc tính này có các giá trị: thin (mảnh), medium (vừa), thick (dày), hay là một giá trị đo cụ thể như pixels. Xem hình minh họa bên dưới.

d) Thuộc tính border-color:

Border-color là thuộc tính CSS quy định màu viền cho một đối tượng web. Thuộc tính này nhận tất cả đơn vị màu CSS hỗ trợ.

e) Thuộc tính border-style:

Border-style là thuộc tính CSS quy định kiểu viền thể hiện của một đối tượng web. CSS cung cấp tất cả 8 kiểu viền tương ứng với 8 giá trị: dotted, dashed, solid, double, groove, ridge, inset và outset. Ngoài ra, hai giá trị none hay hidden dùng để ẩn đường viền.

1.6.17 Height & Width

a) Thuộc tính width

Width là một thuộc tính CSS dùng để quy định chiều rộng cho một thành phần web. Ví dụ sau chúng ta sẽ định chiều rộng cho thành phần p của một trang web. `p { width:700px; }`

b) Thuộc tính max-width

Max-width là thuộc tính CSS dùng để quy định chiều rộng tối đa cho một thành phần web.

c) Thuộc tính min-width

Min-width là thuộc tính CSS dùng để quy định chiều rộng tối thiểu cho một thành phần web.

d) Thuộc tính height

Height là một thuộc tính CSS dùng để quy định chiều cao cho một thành phần web. Ví dụ sau chúng ta sẽ định chiều cao cho thành phần p của một trang web.

```
p { height:300px }
```

e) Thuộc tính max-height:

Max-height là thuộc tính CSS dùng để quy định chiều cao tối đa cho một thành phần web.

f) Thuộc tính min-height

Min-height là thuộc tính CSS dùng để quy định chiều cao tối thiểu cho một thành phần web.

1.6.18 Position

a) Absolute position

Định vị tuyệt đối là sự định vị mà trong đó các thành phần được định vị không để lại bất cứ một khoảng trống nào trong tài liệu. Một thành phần được định vị tuyệt đối sẽ nhận giá trị position là absolute. Các đối tượng đã định vị tuyệt đối sẽ dùng kết hợp với các thuộc tính top, left, right, bottom để xác định tọa độ.

Ví dụ sau sẽ chỉ cho chúng ta cách đặt bốn ảnh ở bốn góc tài liệu bằng định vị tuyệt đối.

```
#logo1 {  
    position:absolute;  
    top:50px; left:70px  
}
```

```
#logo2 {  
    position:absolute;  
    top:0; right:0  
}
```

```
#logo3 {  
    position:absolute;  
    bottom:0; left:0
```

```
    }  
#logo4 {  
    position:absolute;  
    bottom:70px; right:50px  
}
```

b) Layers

CSS hoạt động trên cả 3 chiều: cao, rộng, sâu. Hai chiều đầu tiên, chúng ta đã được nhìn thấy trong các bài học trước. Trong bài học này, chúng ta sẽ được học về cách đặt các thành phần web ở các lớp khác nhau với thuộc tính z-index. Nói đơn giản hơn thì đó là cách bạn đặt một thành phần này lên trên một thành phần khác. Với mục đích này, bạn sẽ gán cho mỗi phần tử một con số. Theo đó, phần tử có số cao hơn sẽ nằm trên, phần tử có số thấp hơn sẽ nằm dưới.

1.7 KẾT HỢP PHP VÀ MYSQL TRONG ỨNG DỤNG WEBSITE

Để làm việc với mysql và PHP chúng ta cần nắm 6 hàm cơ bản:

a) Kết nối cơ sở dữ liệu:

Cú pháp:

```
mysql_connect("hostname","user","pass")
```

b) Lựa chọn cơ sở dữ liệu:

Cú pháp:

```
mysql_select_db("tên_CSDL")
```

Ví dụ:

```
$conn=mysql_connect("localhost","root","root") or die(" không the ket noi");
```

```
mysql_select_db("demo");
```

c) Thực thi câu lệnh truy vấn:

Cú pháp:

```
mysql_query("Câu truy vấn ở đây");
```

d) Đếm số dòng dữ liệu trong bảng:

Cú pháp:

```
mysql_num_rows();
```

e) Lấy dữ liệu từ bảng đưa vào mảng:

Cú pháp:

```
mysql_fetch_array();
```

f) Đóng kết nối cơ sở dữ liệu:

Cú pháp:

```
mysql_close);
```

Kiểm tra xem trong bảng dữ liệu đã tồn tại user nào chưa ?. Nếu chưa thì xuất ra thông báo lỗi, ngược lại thì đưa chúng vào mảng và lặp cho đến hết bảng dữ liệu.

```
<?
```

```
if(mysql_num_rows($query) == 0)
```

```
{
```

```
echo "Chua co du lieu";
```

```
}
```

```
else
```

```
{
```

```
while($row=mysql_fetch_array($query))
```

```
{
```

```
echo $row[username] ." - ".$row[password]."<br />";
```

```
}
```

```
}
```

```
?>
```

Và cuối cùng chúng ta đóng kết nối và kết thúc thao tác xử lý.

```
<?
```

```
mysql_close($conn);
```

```
?>
```

Và cuối cùng là file hoàn chỉnh của ứng dụng trên:

```
<?
```

```
$conn=mysql_connect("localhost", "root", "root") or die("can't connect database");
```

```
mysql_select_db("demo_mysql",$conn);
```

```
$sql="select * from user";
```

```
$query=mysql_query($sql);
```

```
if(mysql_num_rows($query) == 0)
```

```
{
```

```
echo "Chua co du lieu";
```

```
}
```

```
else
```

```
{
```

```
while($row=mysql_fetch_array($query))
```

```
{
```

```

echo $row[username] ." - ".$row[password]."<br />";
}
}
mysql_close($conn);
?>

```

1.8 CƠ BẢN VỀ SQL VÀ MYSQL

Cơ sở dữ liệu (viết tắt CSDL; tiếng Anh là database) được hiểu theo cách định nghĩa kiểu kỹ thuật thì nó là một tập hợp thông tin có cấu trúc. Tuy nhiên, thuật ngữ này thường dùng trong công nghệ thông tin và nó thường được hiểu rõ hơn dưới dạng một tập hợp liên kết các dữ liệu, thường đủ lớn để lưu trên một thiết bị lưu trữ như đĩa hay băng. Dữ liệu này được duy trì dưới dạng một tập hợp các tập tin trong hệ điều hành hay được lưu trữ trong các hệ quản trị cơ sở dữ liệu.

Mysql là hệ quản trị dữ liệu miễn phí, được tích hợp sử dụng chung với apache, PHP. Chính yếu tố phát triển trong cộng đồng mã nguồn mở nên mysql đã qua rất nhiều sự hỗ trợ của những lập trình viên yêu thích mã nguồn mở. Mysql cũng có cùng một cách truy xuất và mã lệnh tương tự với ngôn ngữ SQL.

Nhưng Mysql không bao quát toàn bộ những câu truy vấn cao cấp như SQL. Về bản chất Mysql chỉ đáp ứng việc truy xuất đơn giản trong quá trình vận hành của Website nhưng hầu hết có thể giải quyết các bài toán trong PHP.

1.8.1 Những định nghĩa cơ bản

a) Định nghĩa cơ sở dữ liệu, bảng, cột

Cơ sở dữ liệu: là tên của cơ sở dữ liệu chúng ta muốn sử dụng

Bảng: Là 1 bảng giá trị nằm trong cơ sở dữ liệu.

Cột là 1 giá trị nằm trong bảng. Dùng để lưu trữ các trường dữ liệu. Thuộc tính

Ví dụ:

Bảng user	
User_id	1234
First_name	Bui
Last_name	Thu ha
Username	Ha thu
Password	123456
Email	Cucthuytinh14209@gmail.com
Date	2011-9-2 16:25:30

Như vậy ta có thể hiểu như sau: 1 cơ sở dữ liệu có thể bao gồm nhiều bảng. 1 bảng có thể bao gồm nhiều cột 1 cột có thể có hoặc không có những thuộc tính.

b) Định nghĩa 1 số thuật ngữ

NULL : Giá trị cho phép rỗng.

AUTO_INCREMENT : Cho phép giá trị tăng dần (tự động).

UNSIGNED : Phải là số nguyên dương

PRIMARY KEY : Cho phép nó là khóa chính trong bảng.

c)Loại dữ liệu trong Mysql

Ở đây chúng ta chỉ giới thiệu 1 số loại thông dụng: 1 số dữ liệu khác có thể tham khảo trên trang chủ của mysql.

Kiểu dữ liệu	Mô tả
Char	Định dạng text có chiều dài từ 0->255
Varchar	Định dạng text có chiều dài từ 0->255
Text	Định dạng text có chiều dài từ 0->65535
Longtext	Định dạng text có chiều dài từ 0->4294967215
Int	Định dạng số có chiều dài từ 0->4294967215
Float	Định dạng số thập phân có chiều dài nhỏ
Double	Định dạng số thập phân có chiều dài lớn
Date	Định dạng thời gian theo định dạng:YYYY-MM-DD
Datetime	Định dạng thời gian theo định dạng:YYYY-MM-DD HH:MM:SS

1.8.2 Những cú pháp cơ bản

Cú pháp tạo 1 cơ sở dữ liệu:

```
CREATE DATABASE tên_cơ_sở_dữ_liệu;
```

Cú pháp sử dụng cơ sở dữ liệu: Use tên_database;

Cú pháp thoát khỏi cơ sở dữ liệu: Exit

Cú pháp tạo 1 bảng trong cơ sở dữ liệu: CREATE TABLE user (<tên_cột>

<mô_tả>,...,<tên_cột_n>.....<mô_tả_n>)

Ví dụ:

```
mysql> create table user(user_id INT(15) UNSIGNED NOT NULL AUTO_INCREMENT,  
username VARCHAR(255) NOT NULL, password CHAR(50) NOT NULL, email  
VARCHAR(200) NOT NULL, PRIMARY KEY (user_id));
```

Hiển thị có bao nhiêu bảng: show tables;

Hiển thị có bao nhiêu cột trong bảng: show columns from table;

Thêm 1 cột vào bảng : ALTER TABLE tên_bảng ADD <tên_cột> <thuộc_tính> AFTER <tên_cột>

1.8.3 Thêm giá trị vào bảng

Cú pháp: INSERT INTO Tên_bảng(tên_cột) VALUES(Giá_trị_tương_ứng);

Ví dụ:

```
mysql>insertintouser(username,password,email,sex,home)
values("Lanna","12345","lanna@yahoo.com","F","www.abc.com");
```

1.8.4 Truy xuất dữ liệu

Cú pháp: SELECT tên_cột FROM Tên_bảng;

Ví dụ: mysql> select user_id,username from user;

1.8.5 Truy xuất dữ liệu với điều kiện

Cú pháp:

SELECT tên_cột FROM Tên_bảng WHERE điều kiện;

Ví dụ:

```
mysql> select user_id,username from user where user_id=2;
```

1.8.6 Truy cập dữ liệu và sắp xếp theo trình tự

Cú pháp:

SELECT tên_cột FROM Tên_bảng WHERE điều kiện (có thể có where hoặc không)
ORDER BY Theo quy ước sắp xếp.

Trong đó quy ước sắp xếp bao gồm hai thông số là ASC (từ trên xuống dưới), DESC (từ dưới lên trên).

```
mysql> select user_id,username from user order by username ASC ;
```

1.8.7 Truy cập dữ liệu có giới hạn

Cú pháp:

SELECT tên_cột FROM Tên_bảng WHERE điều kiện (có thể có where hoặc không)
LIMIT vị trí bắt đầu, số record muốn lấy ra

Ví dụ:

```
mysql> select user_id,username from user order by username ASC limit 0,10 ;
```

1.8.8 Cập nhật dữ liệu trong bảng

Cú pháp:

Update tên_bảng set tên_cột=Giá trị mới WHERE (điều kiện).

Nếu không có ràng buộc điều kiện, chúng sẽ cập nhật toàn bộ giá trị mới của các record trong bảng.

Ví dụ:

```
mysql> update user set email="admin@qhonline.info" where user_id=1 ;
```

1.8.9 Xóa dữ liệu trong bảng

Cú pháp:

DELETE FROM tên_bảng WHERE (điều kiện). Nếu không có ràng buộc điều kiện, chúng sẽ xóa toàn bộ giá trị của các record trong bảng.

Ví dụ

```
mysql>delete from user where user_id=1 ;
```

1.9 VIỆC BẢO TRÌ TRANG WEB

- Joomla rất thân thiện cho người sử dụng cũng như người quản trị. Tất cả mọi người có kiến thức cơ bản về website đều có thể dễ dàng học cách quản trị. Joomla! không đòi hỏi người quản trị phải có kỹ thuật cao để thêm hay chỉnh sửa nội dung, hình ảnh, để tổ chức sắp xếp các dữ liệu quan trọng của công ty bạn hiệu quả.

- Thông qua giao diện đơn giản và thân thiện qua trình duyệt web, bạn có thể dễ dàng cập nhật thông cáo báo chí mới, danh mục tin tức, quản lý sản phẩm, sắp xếp lịch làm việc...

1.10 ADOBE DREAMWEAVERCS4

Nội dung trang web trong một website động bao gồm những phần cố định và những phần có thể thay đổi theo yêu cầu của người xem hay theo yêu cầu cập nhật. Vì vậy, hầu hết các website cần có sự cập nhật thường xuyên đều ở dạng động để tiết kiệm công sức thiết kế và điều hành. Đối với website tĩnh, phần mềm thiết kế phổ biến là FrontPage, nhưng để thiết kế website động, người ta thường dùng Dreamweaver của hãng Adobe.

Adobe Dreamweaver CS4 là một chương trình hỗ trợ thiết kế website được sử dụng rất phổ biến vì với những công cụ mạnh mẽ, được bố trí hợp lý trong một giao diện thân thiện, nó rất thích hợp cho mọi đối tượng, từ những nhà thiết kế website chuyên nghiệp cho đến những ai mới vào nghề. Với Adobe Dreamweaver CS3, bạn sẽ dễ dàng nhận biết được rằng mình nên bắt đầu thiết kế một website ra sao bằng cách sử dụng các template sẵn có của nó hoặc tải về từ Internet rồi từ đó hiệu chỉnh lại cho phù hợp với nhu cầu. Bên cạnh đó, với khả năng can thiệp trực tiếp vào database (cơ sở dữ liệu), Adobe Dreamweaver CS3 sẽ là một trợ thủ đắc lực cho bạn trong việc thiết kế các website động với các ngôn ngữ lập trình mạnh mẽ và phổ biến như: PHP, ASP/ASP.NET, ColdFusion v.v... mà không cần biết nhiều về lập trình web.

1.11 ADOBE PHOTOSHOP CS4

Là phần mềm số 1 trong việc biên tập và xử lý hình ảnh chuyên nghiệp, bạn sẽ có thể làm được hơn nhiều điều với Photoshop.

Những công cụ sáng tạo của nó giúp bạn làm được những kết quả thật phi thường. Những khả năng tuyệt vời chưa hề có sẽ giúp bạn dành trọn thời gian với Photoshop cho mọi

công việc của bạn. Đồng thời với nhiều hiệu ứng biên tập, xử lý và biến đổi hình ảnh công việc của bạn sẽ được giải quyết một cách thật sự nhanh chóng.

Xin điểm qua một vài tính năng mới mà bạn có thể tìm thấy trong CS4.

Khung Adjustment nay được thay thế bằng một bảng khung bao gồm những tác vụ chính giúp việc sử dụng công cụ thiết lập được nhanh chóng.

Tính năng Auto-blending được cải thiện trong việc thiết lập trộn nhiều layer cho hình ảnh được đẹp với việc loại bỏ các điểm vỡ ảnh.

Khung Mask được thêm mới lần này giúp người sử dụng dễ dàng tạo được những hình ảnh theo yêu cầu thông qua cân bằng màu sắc, độ dither...

Diễn hình nhất trong phiên bản CS4 là tính năng Adobe Photoshop Lightroom. Tính năng này cho bạn ghép nhiều ảnh từ những hình ảnh nhỏ thông qua thiết lập tự động hình ảnh theo một chế độ riêng cân bằng trên cả từng layer hoặc trên từng hình ảnh (panorama).

Ngoài ra, tính năng Content-Aware (*ảnh bên*) mới trong CS4 lần này sẽ giúp bạn co giãn hình ảnh nhưng không làm mất tính logic của hình ảnh lẫn về nội dung và màu sắc.

1.12 Tổng quan ngôn ngữ JavaScript

a) Javascript là một ngôn ngữ thông dịch (interpreter), chương trình nguồn của nó được nhúng (embedded) hoặc tích hợp (integrated) vào tập tin HTML chuẩn. Khi file được load trong Browser (có support cho JavaScript), Browser sẽ thông dịch các Script và thực hiện các công việc xác định. Chương trình nguồn JavaScript được thông dịch trong trang HTML sau khi toàn bộ trang được load nhưng trước khi trang được hiển thị. Javascript là một ngôn ngữ có đặc tính:

Đơn giản.

Động (Dynamic).

Hướng đối tượng (Object Oriented).

b) Nguyên tắc trong JavaScript:

Mỗi câu lệnh trong JavaScript đều kết thúc bằng dấu ";" . Các điều kiện đều được khai báo trong dấu bao đơn ().

Mỗi hàm hay phương thức do người dùng định nghĩa được bắt đầu với từ khóa Function, tham số truyền vào phương thức không cần khai báo kiểu dữ liệu. Khi khai báo biến trong JavaScript, chúng ta cần phải sử dụng từ khóa var trước tên biến. Biến có thể khởi tạo giá trị hay không đều được chấp nhận.

Trong phát biểu có điều kiện, nếu có hơn một phát biểu, phải sử dụng cặp dấu ngoặc nhọn { }.

c) Tham chiếu đến thẻ HTML

Để tham chiếu đến thẻ HTML bằng JavaScript, ta cần phải xác định loại thẻ và thuộc tính của thẻ, để thẻ được tham chiếu, chúng ta phải có tên (Name) hay ID nhận dạng.

Đối với một số thẻ không có thuộc tính Value, khi tham chiếu đến giá trị của chúng, chúng ta phải tham chiếu đến thuộc tính value như một phần giá trị của chúng.

Thẻ	Ví dụ
Text	Document.form.txtUser.value
Hidden	Document.form.txtForm.value
Password	Document.form.txtPwd.value
Textarea	Document.form.txtDesc.value
Select	Document.form.cbCity.value
Select	Document.form.cbIndu[i].value
Checkbox	Document.form.chkid.value
Checkbox	Document.form.chkid[i].value
Radio	Document.form.rdGender.value
Radio	Document.form.rdGender.value

d) Một số biến cố thường dùng trong JavaScript

Khi người dùng thực hiện một hành động nào đó trên một số thẻ HTML, chúng ta có thể sử dụng một số phương thức của JavaScript hay phương thức do người dùng định nghĩa thông qua biến cố phát sinh.

Sự kiện	Diễn giải
OnClick	Khi người dùng nhấn chuột trên thẻ
OnSubmit	Khi người dùng submit một form
OnReset	Khi người dùng reset một form
OnActive	Khi form hoạt động
OnLoad	Khi form nạp lên trình duyệt
Onunload	Khi form đóng lại
Onmousemove	Khi mouse di chuyển vào thẻ
Onmouseout	Khi mouse di chuyển ra khỏi thẻ
Onmouseover	Khi mouse di chuyển trên thẻ

Onblur	Khi con trở di chuyển ra khỏi thẻ
Onkeypress	Khi phím trên thẻ

f) Một số phương thức thường dùng trong JavaScript

Ngoài một số phương thức *JavaScript* do người dùng định nghĩa, bản thân *JavaScript* cung cấp một số phương thức, cho phép người dùng thực hiện một số hành động thay vì sử dụng thông qua biến cố.

Phương thức	Diễn giải
Alert(str)	Xuất hiện hộp thoại với nội dung là chuỗi Str
Confirm(str)	Xuất hiện hộp thoại với nội dung cần được xác nhận với nút Ok và Canel
Prompt(str,default)	Xuất hiện hộp thoại với nội dung là str ,giá trị mệnh đề default cho phép người dùng nhập giá trị
f.submit()	Submit form có tên là f
f.reset()	Reset form có tên là f
t.focus()	Trở con nháy vào thẻ có tên là t
s.toString()	Chuyển giá trị của đối tượng sang dạng chuỗi
s.indexOf(i)	Trả về vị trí đầu tiên của chuỗi i tìm thấy trong chuỗi s
s.lastIndexOf(i)	Trả về vị trí cuối cùng của chuỗi i tìm thấy trong chuỗi s
s.substr(i,j)	Trả về chuỗi con trong chuỗi s từ vị trí thứ i dài j ký tự
s.charAt(i)	Trả về ký tự tại vị trí thứ i trong chuỗi s

g) Một số phương thức điều khiển khác

Phương thức	Diễn giải
Window.close()	Đóng cửa sổ hiện tại
Window.open(URL)	Mở một cửa sổ trình duyệt với địa chỉ URL
Window.opener(URL)	Mở một cửa sổ trình duyệt với địa chỉ URL trên cửa sổ popup
Window.history.back(i)	Trở về cửa sổ trang web trình duyệt trước đó I lần, chẳng hạn muốn về trang trước đó window..history.back(i)
Window.history.go(-i)	Trở về cửa sổ trang web trình duyệt trước đó i lần,

	chẳng hạn muốn về trang trước đó window.history.go(i)
Window.print()	Mở hộp thoại print để in trang web
Document.write(str)	Cho phép in chuỗi str ra vị trí hiện tại trên trang web
Window.external.AddFavorite (url)	Thêm địa chỉ URL vào Fevorite của trình duyệt web

Chương 2: KHẢO SÁT VÀ PHÂN TÍCH BÀI TOÁN XÂY DỰNG WEBSITE CHO KHÁCH SẠN CAMELA

2.1 KHÁI QUÁT ĐỀ TÀI

Website giới thiệu và đặt phòng trực tuyến qua mạng của khách sạn Camela được xây dựng bằng ngôn ngữ lập trình PHP, CSS và hệ quản trị cơ sở dữ liệu MySQL. Webstie được xây dựng nhằm ứng dụng công nghệ thông tin vào đời sống. Website là một hệ thống hoàn chỉnh trong việc giới thiệu và đặt phòng qua mạng của khách sạn. Qua đó giúp khách hàng có thể chọn các loại hình dịch vụ, và chọn Camela là nơi nghỉ dưỡng của mình. Có thể chọn cho mình một căn phòng ưng ý mà không phải tới tận nơi để xem và đặt phòng.

Chức năng chính của website:

- + Giới thiệu về khách sạn
- + Xây dựng hệ thống đặt phòng qua mạng
- + Giải đáp thắc mắc của khách hàng qua phần liên hệ
- + Gửi danh sách khách đặt phòng cho bộ phận kinh doanh

2.2 KHẢO SÁT HOẠT ĐỘNG CỦA KHÁCH SẠN

Website luôn luôn cập nhật tin tức nên ban giám đốc của khách sạn quyết định giao việc quản trị web cho những bộ phận khác nhau như sau:

-Bộ phận kinh doanh:chịu trách nhiệm quản lý đơn đặt phòng khách sạn khi khách đặt phòng trực tuyến.Người quản trị có nhiệm vụ kiểm tra và xác nhận thông tin khách hàng và đơn đặt phòng.

-Bộ phận văn phòng: Đảm nhiệm việc thêm mới,cập nhật thông tin các dịch vụ,quảng bá hình ảnh của khách sạn.Xóa bỏ những thông tin cũ,thông tin không còn giá trị.Người quản trị cập nhật lại dữ liệu.

Quy trình nghiệp vụ:

Công ty chỉ đạo những thông tin hình ảnh sẽ đưa lên web.Bộ phận văn phòng soạn thảo thông tin,hình ảnh về các loại phòng,căn hộ,dịch vụ,nhà hàng &bar,khu sinh thái,mà khách sạn sẽ cung cấp cho khách,sẽ luôn cập nhật thông tin của khách sạn lên website.

Ngoài ra bộ phận này còn đưa những tin tức sự kiện đã và đang xảy ra ở khách sạn,những ưu đãi,khuyến mại nếu có.

Khách vào trang web sẽ được xem thông tin từ tổng quan đến chi tiết của khách sạn.Về các loại phòng nghỉ,các loại căn hộ về giá thành,diện tích,tiện nghi,tiện ích đi kèm.sau đó khách có thể đặt phòng trực tuyến trên website.

Các đơn đặt phòng sẽ được cập nhật vào database và hiển thị cho người quản trị được xem,xóa hoặc xác nhận,kiểm tra tính chính xác của thông tin đặt phòng.

Khách cũng thông qua trang web xem các dịch vụ gồm các phòng hội thảo về số chỗ,kiểu xếp,giá thành,các thiết bị,khu trung tâm thương mại,khu liên hiệp thể thao có sân golf,sân tennis,bể bơi,câu lạc bộ billiard gồm giá cho khách lẻ,cho hội viên và các dịch vụ đi kèm,thời gian mở cửa. Bên cạnh đó là các nhà hàng&bar,phòng karaoke,massage&sauna gồm có giá dịch vụ,tiện nghi,thời gian mở cửa...

Khách cũng có thể thông qua trang web gửi các thông tin yêu cầu khác về khách sạn qua trang Liên Hệ.Thông tin này được truyền vào database cho phép người quản trị quản lý chúng

3.1. MÔ TẢ BÀI TOÁN

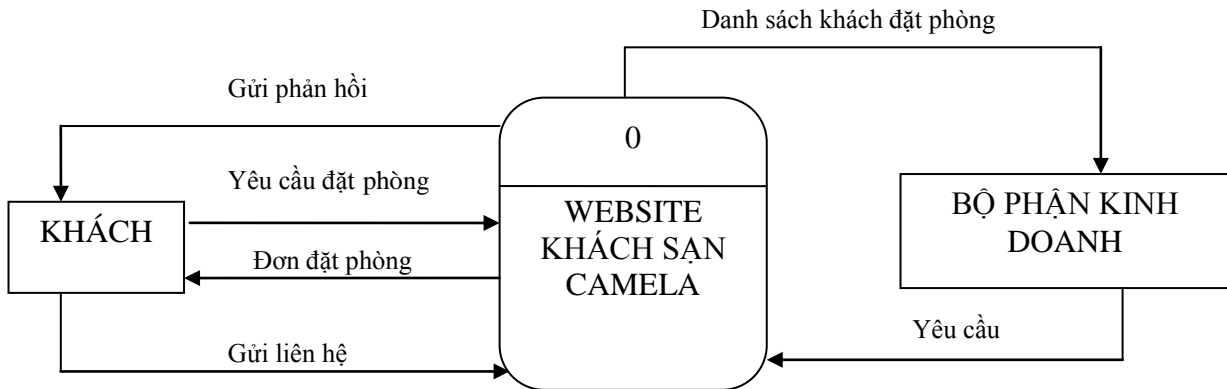
Khách hàng có thể truy cập vào website để tra cứu các thông tin về khách sạn các loại phòng của khách sạn. Thông tin mỗi loại phòng bao gồm: giá thành,diện tích,tiện nghi,tiện ích.Khi khách hàng muốn đặt phòng thì khách hàng phải tiến hành lập đơn đặt phòng.Thông tin đăng phòng gồm: họ tên, điện thoại, email. số phòng, ngày đến, ngày đi, số người ..

Khách hàng có thể chọn phòng mà mình mong muốn bằng cách kích chuột vào nút đặt phòng ngay bên cạnh mỗi loại phòng.Trong khi vào hệ thống website, khách hàng có thể đưa

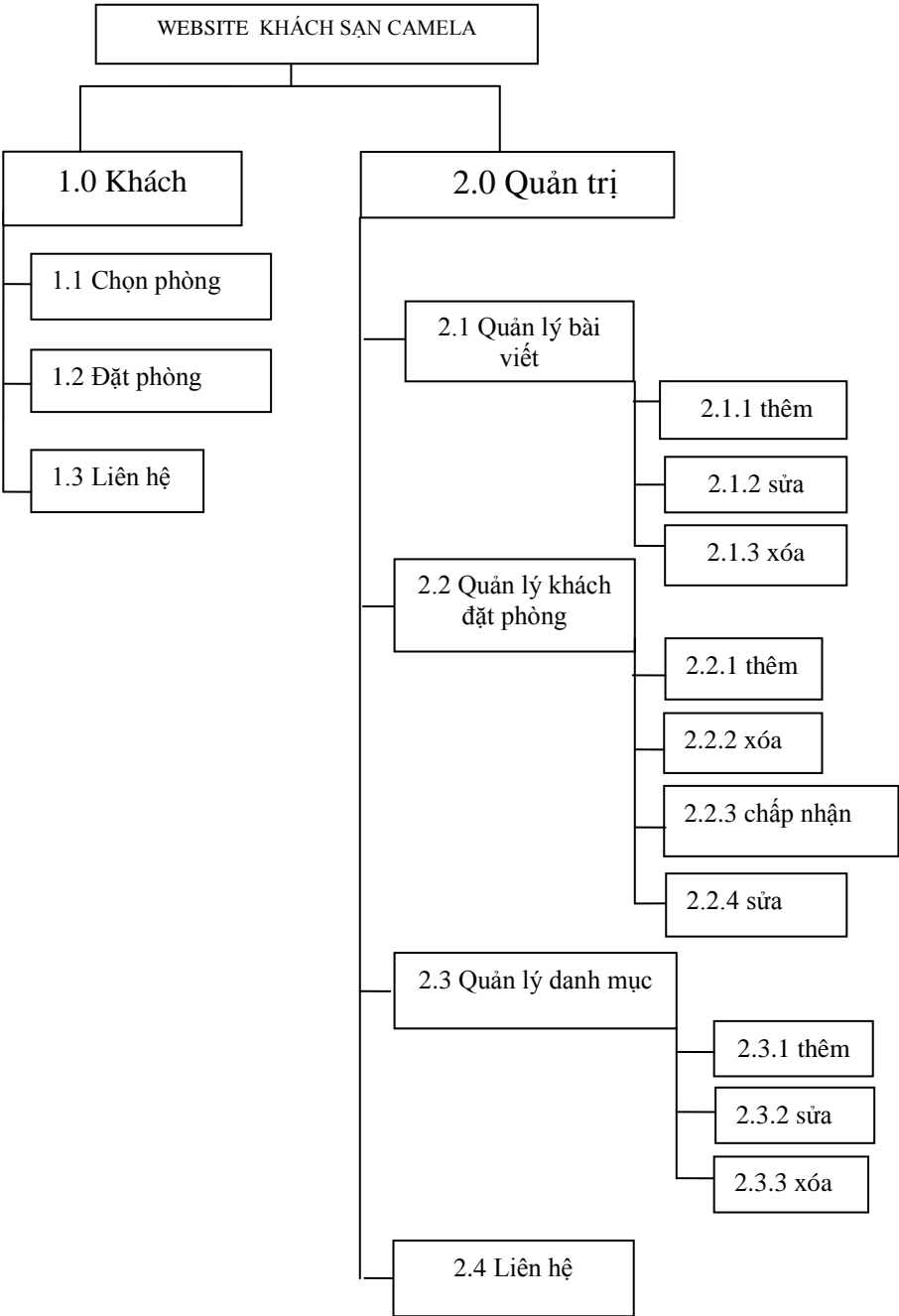
ra những câu hỏi,góp ý ở trang liên hệ. Thông tin liên hệ bao gồm: Họ tên, email, tiêu đề, nội dung liên hệ. Admin sẽ trả lời các câu hỏi của khách hàng sau khi cập nhật.Admin sẽ tiến hành cập nhật thông tin,cập nhật hình ảnh, cập nhật đơn đặt phòng, cập nhật các câu hỏi của khách hàng. Đồng thời theo dõi danh sách khách hàng đặt phòng để gửi lại cho bộ phận kinh doanh.

2.3 MÔ HÌNH NGHIỆP VỤ CỦA HỆ THỐNG

2.3.1 Mô hình ngữ cảnh của hệ thống



2.3.2. Biểu đồ phân rã chức năng



2.3.3. Mô tả chi tiết các chức năng

- Quản lý khách đặt phòng: Lưu trữ tất cả các thông tin của khách hàng đặt phòng. Admin có thể xem và chấp nhận các đơn hợp lệ và xóa các đơn hàng không hợp lệ.
- Quản lý bài viết: Lưu trữ các bài viết trên website. Admin có thể thêm mới, xóa, sửa những bài viết trên trang website.
- Quản lý danh mục: Admin có thể xóa, sửa, thêm các danh mục trên website, để hợp với thị hiếu.
- Liên hệ (admin): Hàng ngày admin phải kiểm tra các câu hỏi của khách hàng và trả lời các câu hỏi đó.
- Chọn phòng: Khách hàng có thể chọn các phòng mà mình muốn đặt tại website.
- Đặt phòng: Khách hàng có thể đặt các phòng mà mình ưng ý trên website của khách sạn.
- Liên hệ (khách hàng): Khách hàng có thể đưa ra các câu hỏi và sẽ được admin của website trả lời.

2.3.4. Các hồ sơ dữ liệu sử dụng

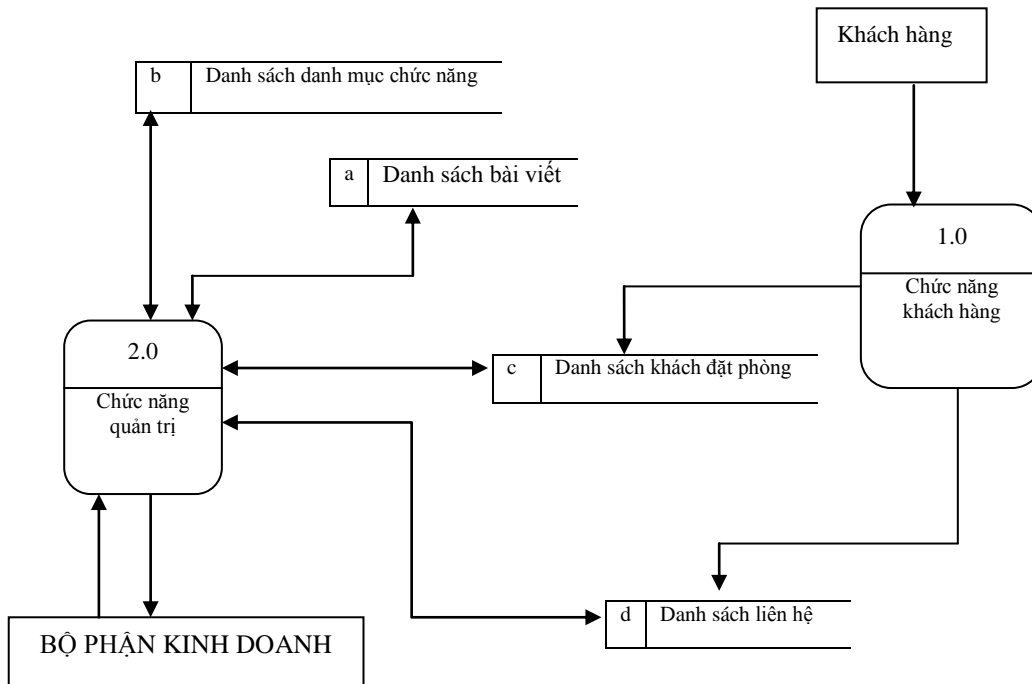
Với các hoạt động nghiệp vụ như trên, hệ thống đã sử dụng các hồ sơ dữ liệu sau đây:

- Danh sách bài viết
- Danh sách danh mục chức năng
- Danh sách khách đặt phòng
- Danh sách liên hệ

2.3.5. Ma trận thực thể chức năng

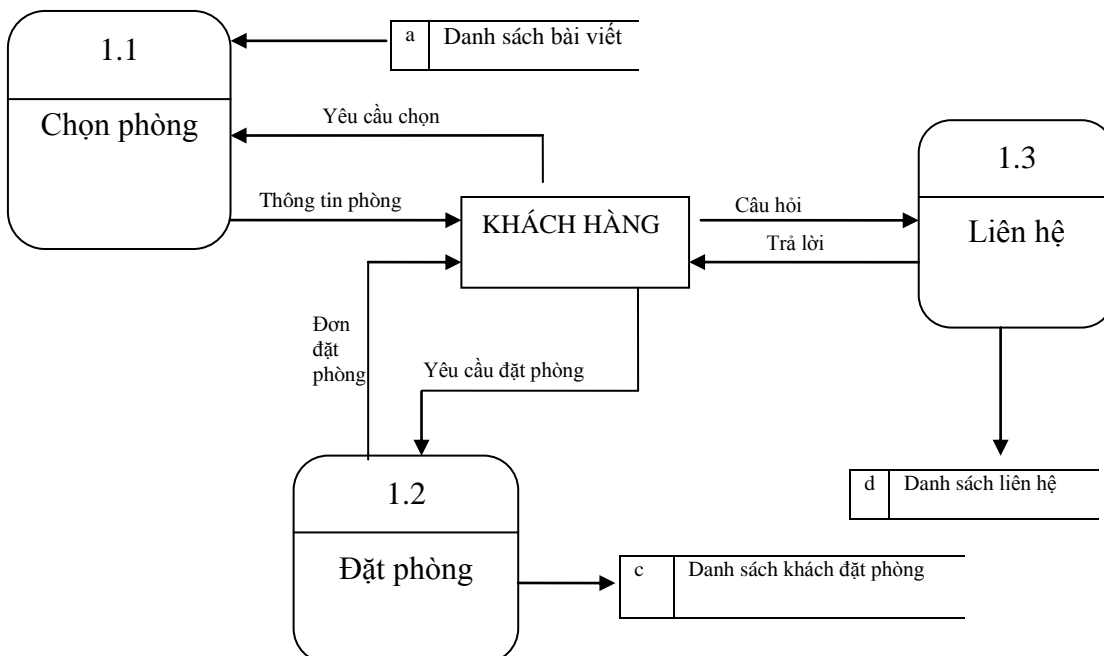
Các thực thể				
a. Danh sách bài viết				
b. Danh sách danh mục chức năng				
c. Danh sách khách đặt phòng				
d. Danh sách liên hệ				
Các chức năng nghiệp vụ	a	b	c	d
1. Chức năng khách hàng	R	R	C	C
2. Chức năng quản trị	U	U	U	U

2.3.6. Biểu đồ luồng dữ liệu mức 0

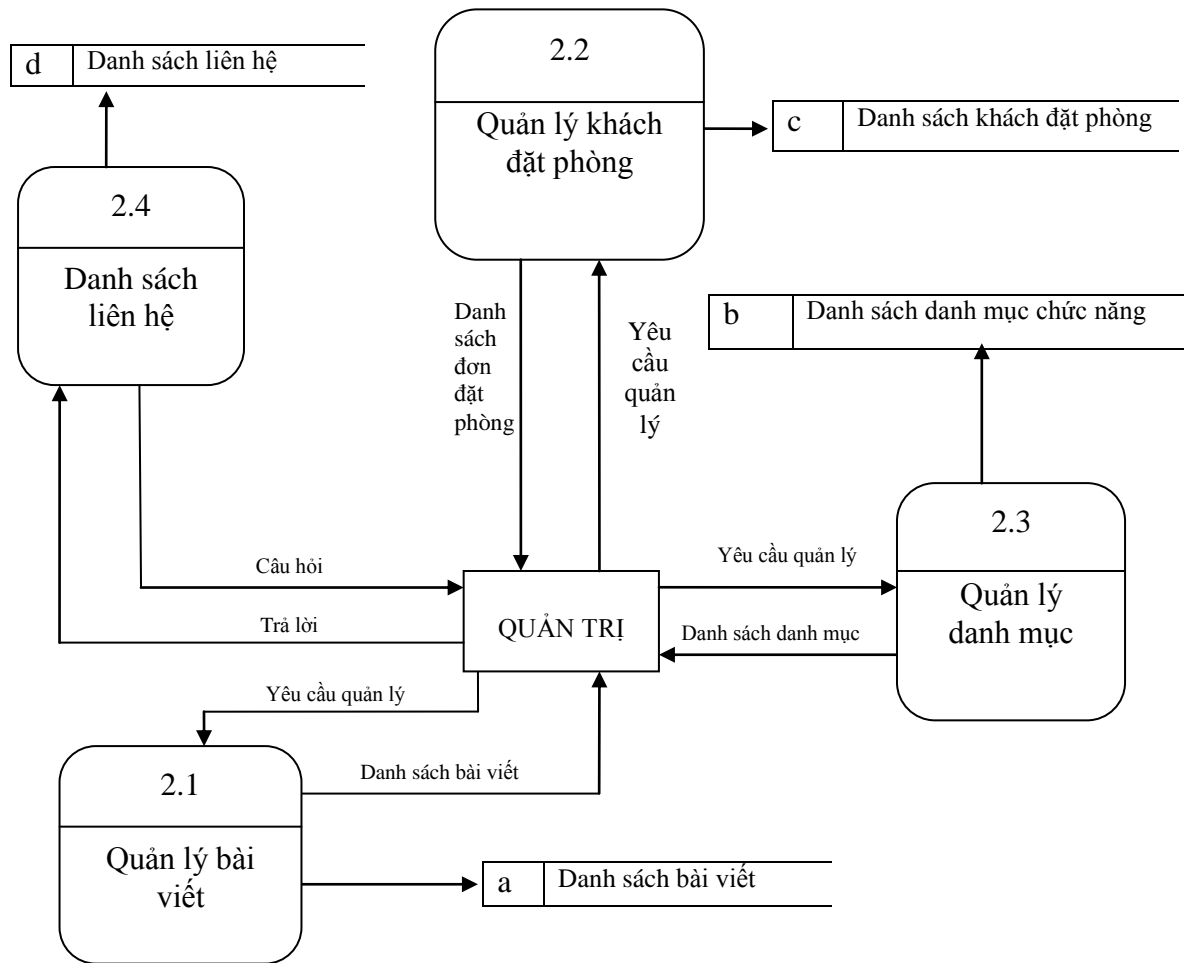


2.3.7. Biểu đồ luồng dữ liệu mức 1

a) Biểu đồ luồng dữ liệu tiến trình "1.0 Khách hàng"



b) Biểu đồ luồng dữ liệu tiến trình "2.0 Quản trị"



2.4 THIẾT KẾ CÁC BẢNG DỮ LIỆU

2.4.1 Bảng *khach_dat_phong*

Diễn giải	Thuộc tính	Kiểu dữ liệu	Khóa
Mã KH	CustomerID	Int	Khóa chính
Tên KH	CustomerName	nvarchar(25)	
Ngày đến	OrderDateTime	Datetime	
Ngày đi	OrderDateTime	Datetime	
Địa chỉ	Address	nvarchar(MAX)	
Điện thoại	Phone	varchar(15)	
Thư điện tử	Email	varchar(30)	
Thông tin phòng	Status	nvarchar(255)	
Số người	Quantity	Int	
Trạng thái	infor	Nvarchar(15)	
Giá thuê	Price	money	

2.4.2 Bảng *lien_he_hoi_dap*

Diễn giải	code	Kiểu dữ liệu	Khóa
Mã LHHĐ	feedbackID	int	Khóa chính
Câu hỏi	Questions	nvarchar(MAX)	
Câu trả lời	Anwer	nvarchar(MAX)	
Ngày hỏi	dateQ	Datetime	
Ngày trả lời	dateA	Datetime	
Thư điện tử	email	Varchar(30)	

Chương 3: XÂY DỰNG VÀ TRIỂN KHAI CHƯƠNG TRÌNH

Website được hoàn thành bằng ngôn ngữ lập trình PHP và hệ quản trị cơ sở dữ liệu MySQL. Website xây dựng lại dựa trên một số component mã mở của Joomla và một số module khác.

3.1 CÀI ĐẶT CHƯƠNG TRÌNH

3.1.1 Môi trường cài đặt

Mô hình client/server.

Ngôn ngữ cài đặt: PHP.

Hệ quản trị cơ sở dữ liệu: MySQL.

3.1.2 Công cụ hỗ trợ

a. PHP Webserver

Cài đặt phần mềm ‘*xampp-win32-1.7.3*’ và phiên bản Joomla 1.5

b. Phần mềm hỗ trợ

Adobe Dreamweaver.

Photoshop.

Demo chương trình

3.2 Trang chủ

