**Môn Lập trình Web và Ứng dụng**

1. **Nguyên tắc hoạt động website**
   1. **Khái niệm và thuật ngữ**

- Một số thuật ngữ thông dụng:

+ Internet: Mạng máy tính toàn cầu kết nối các mạng máy tính khắp nơi trên thế giới. Tập các giao thức được dùng gọi chung là TCP/IP.

+ Intranet: Mạng cục bộ có kiến trúc tương tự mạng Internet.

+ Website: Tập hợp các trang web. Website của các tổ chức hay cá nhân trên mạng bao gồm tập hợp các trang web liên quan đến tổ chức này.

+ Webpage: Là trang web. Có thể hiển thị các thông tin dưới dạng văn bản, hình ảnh, âm thanh...

+ Web browser: Trình duyệt web, dùng để hiển thị các trang web. Các web browser thông dụng hiện nay là IE và Firefox.

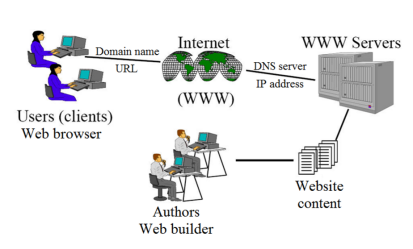
+ Homepage: Trang chủ. Thường là trang đầu tiên (mặc định) khi truy cập một website.

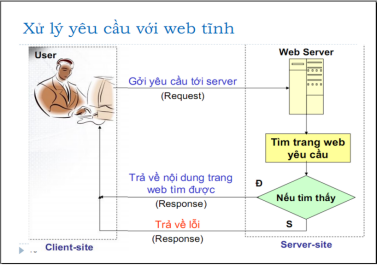
+ Hyperlink: siêu liên kết. Dùng để liên kết các trang web và dịch vụ của các website trên Internet.

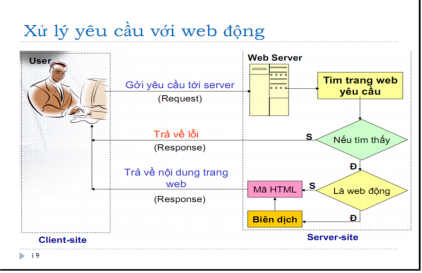
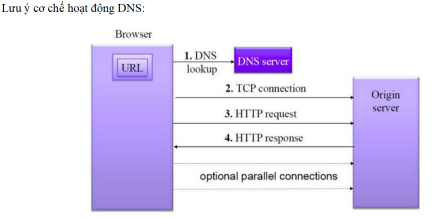
+ ISP (Internet Service Provider): Nhà cung cấp dịch vụ Internet.. Một số ISP hiện nay ở Việt Nam: VDC, FPT, SaigonNet, VNPT, Viettel.

+ Search engines: Máy tìm kiếm. Các công cụ tìm kiếm hiện nay: Google, Wiki,…

+ HTTP, FTP, SMTP, POP3: Đây là các giao thức được dùng cho các dịch vụ web ftp, email trên Internet...

* 1. **Nguyên tắc hoạt động website dựa trên hình:**





* 1. **Web hosting service**

Web hosting service:

Dịch vụ thuê chỗ website (web hosting) là khái niệm dùng để chỉ quá trình làm một website có thể truy cập được từ Internet. Để một website có thể truy cập được từ Internet, website phải được đặt trên một máy có cấu hình mạnh và có kết nối Internet thường xuyên. Vì chi phí để có được một máy như vậy không nhỏ nên mới xuất hiện các công ty cung cấp dịch vụ thuê chỗ phục vụ cho các công ty có nhu cầu đưa website của mình lên Internet.

Có 2 hình thức là miễn phí và trả tiền.

Có 2 dạng thuê chỗ là shared hosting (virtual) và 1 dạng khác bao gồm đeicated hosting và colocated hosting.

****-** [Web Hosting](https://hostingviet.vn/host-la-gi" \t "https://hostingviet.vn/_blank) và Domain** có mối quan hệ mật thiết gần như không thể tách rời nhau. Nếu không có hai hệ thống đó, website sẽ không thể hoạt động được. Đơn giản hơn là nếu không có tên miền Domain thì không tìm ra trang web, còn không có Hosting thì không tạo web được.

**- Tên miền (Domain):**  trong mối tương quan với Web hosting thì tên miền nhằm giúp website dễ dàng hiển thị trên Internet hơn là thông qua giao thức mã hóa IP truyền thống đồng thời tên miền cũng là tên gọi đại diện cho IP.

- Nhà cung cấp các dịch vụ lưu trữ web thường có cả dịch vụ đăng kí tên miền.

**-Dung lượng đĩa cứng (Space)** là dung lượng tối đa cho phép dữ liệu website của bạn. Đơn vị tính được liệt kê trong các bảng thông tin dịch vụ thường là megabytes. Các tập tin chiếm dung lượng đĩa nhiều thông thường là các tập tin nhạc, video như .mp3, .avi, .mpeg, …

**-Băng thông (Transfer/Bandwidth)** là tổng số những gì mà người dùng đã tải từ website của bạn khi truy cập. Thông thường nếu vượt quá giới hạn cho phép, bạn phải trả thêm tiền cho nhà cung cấp dịch vụ. Ví dụ: nếu một khách vào website của bạn xem 3 trang HTML mỗi trang có kích thước trung bình khoảng 20KB và 8 tấm hình nhúng trong đó mỗi tấm hình 10 KB thì với 500 khách truy cập một ngày, bạn sẽ cần ít nhất (3 \* 20 + 8 \* 10) \* 500 \* 30 = 2,100,000 KB = 2.1 GB băng thông mỗi tháng.

**-Uptime** là phần trăm thời gian một server làm việc. Ví dụ nếu bạn đặt website tại một server có uptime trung bình là 99.86%, điều này có nghĩa là website của bạn sẽ bị down khoảng 1h mỗi tháng. Hầu như không có dịch vụ thuê chỗ dạng shared hosting nào có thể đảm bảo uptime 100%.

**- Tên miền con (Sub domain):** Khi muốn phát triển thêm một lĩnh vực hay chuyên mục mới trên website. Vd: tên miền chính: sgu.edu.vn thì sub domain là fit.sgu.edu.vn

**- Tài khoản email:** là số tài khoản email khả dụng theo tên miền chạy trực tiếp trên hosting. Vì đây là email theo hosting nên chúng không được đánh giá cao, thường có dung lượng thấp, dễ bị vào black list và kém ổn định.

**- Ngôn ngữ lập trình web và cơ sở dữ liệu được web server hỗ trợ:** PHP version - hosting hỗ trợ tốt nhất cho website với nhiều phiên bản. Ngoài ra còn có: HTML, .NET, JS, Java, …

- Các bước cơ bản để lựa chọn phương án thuê chỗ thích hợp:

+ Lựa chọn hệ điều hành mà server sử dụng

+ Nhu cầu về băng thông

+ Nhu cầu về không gian đĩa

+ Các ngôn ngữ lập trình được hỗ trợ

+ Hệ quản trị cơ sở dữ liệu hỗ trợ

+ Hỗ trợ kĩ thuật

**WebSite**: tập hợp các trang Web của một tổ chức hay cá nhân, có một chủ đề nào đó.

**WebPage**: là trang Web, có thể hiển thị thông tin dưới dạng: text, image, video …

**HomePage**: là trang Web đầu tiên được hiển thị trong một Website. (Thường có tên: index, default …)

**HyperLink**: các mối liên kết giữa các trang Web, trang web nên có nhiều liên kết, trang web không được là ngỏ cụt.

**Protocol**: giao thức, tập các qui tắc thống nhất giữa các máy tính trên mạng để trao đổi thông tin chính xác. Một số giao thức: HTTP (port 80), FTP (20:data transfer, 21: command), SMTP (25), POP3.

**IP Address**: trong hệ thống mạng để các máy tính có thể liên lạc với nhau được thì mỗi máy tính cần có một địa chỉ IP (Internet Protocol Address) tồn tại duy nhất. Ví dụ: 192.168.11.100

**Domain name**: là tên ở dạng chuỗi ký tự được “gắn” với 1 địa chỉ IP. Ví dụ: www.sgu.edu.vn

**DNS** (Domain Name Service): là dịch vụ chuyển đổi từ tên miền sang địa chỉ IP và ngược lại.

**URL** (Uniform Resource Locator): đường dẫn chỉ tới một tập tin trong một máy chủ trên Internet

* Protocol
* Domain name
* File name: tên file và đường dẫn nếu có.

Ví dụ: <http://www.yahoo.com/email/beta.htm>

**Server**: là máy chủ - máy phục vụ, máy tính chuyên cung cấp tài nguyên, dịch vụ cho máy tính khác. Một máy chủ có thể dùng cho một hay nhiều mục đích. Ví dụ: File server, Mail server, Web server.

**Client**: là máy khách, máy khai thác dịch vụ của máy chủ. Một máy tính có thể vừa là client vừa là server.

**Web Server**: máy lưu trữ và cung cấp thông tin dạng Web

Các phần mềm Web Server:

* Internet Information Server, Xampp, Wamp
* Apache
* Netscape Enterprise server
* ….

**Web Client**: máy truy xuất và hiển thị thông tin dạng Web

Để truy xuất các thông tin trên Web Server, các Web Client phải sử dụng một chương trình để duyệt các thông tin này gọi là Web Browser (trình duyệt Web)

Các trình duyệt Web

* Internet Explorer
* Netscape Navigator
* Mozilla Firefox
* Chrome
* Opera
* …

**Trang web tĩnh** (static web pages): dùng ngôn ngữ HTML, dễ phát triển, không có khả năng tương tác với người sử dụng.

**Trang web động** (dynamic web pages): dùng nhiều ngôn ngữ khác nhau, có khả năng tương tác với người truy cập trang Web đó. (PHP,JSP, ASP, NodeJS…)

1. **HTML**

**Cú pháp** tổng quát: thẻ/tag

<TagName Attribute1="ValueA" Attribute2="ValueB" …>

Thẻ HTML gồm hai phần:

* **TagName**: tên thẻ được đặt giữa hai dấu ngoặc nhọn ( < > ) để báo cho trình duyệt biết cách thể hiện văn bản. (có thể hiểu tên thẻ như là lệnh, mỗi thẻ có một công dụng khác nhau)
* **Attribute**: thuộc tính của thẻ, cung cấp các tuỳ chọn cho thẻ. Tuỳ theo thẻ có thể không có, có một hay nhiều thuộc tính. Các thuộc tính cách nhau bởi khoảng trắng. Thuộc tính thường có trị (value) xác định cách tác động của thẻ. Nên đặt trị trong dấu nháy kép.

Không phân biệt chữ hoa chữ thường.

**Thẻ chứa và thẻ rỗng**

a. Thẻ chứa (Container tag)

Thẻ mở (Opening thẻ): <TagName [Attribute]>

Thẻ đóng (Closing thẻ): </TagName>

Các thành phần đặt giữa thẻ mở và thẻ đóng chịu tác động của thẻ đó.

Ví dụ:

<B> Xin chào bạn </B> Nguyễn An

b. Thẻ rỗng (Empty tag)

Chỉ có thẻ mở không có thẻ đóng dùng để hiển thị yêu cầu 1 lần.

Ví dụ: xuống dòng <br>, chèn đường kẻ ngang <hr>

c. Các thẻ có thể lồng nhau, nội dung nào nằm trong nhiều thẻ sẽ chịu tác động của các thẻ đó.

***Lưu ý***: mở trước đóng sau/mở sau đóng trước.

**Cấu trúc cơ bản**

Một trang HTML chuẩn cần có cấu trúc, phải được bắt đầu và kết thúc bằng thẻ <html> và </html>.

Trong thẻ đó gồm 2 phần:

* Phần đầu <head> </head> chứa những thông tin về trang HTML, không được hiển thị trong nội dung trang Web.
* Phần thân <body> </body> chứa nội dung cần trình bày trong trang Web.

Trang Web có thể hiển thị tốt trên hầu hết các máy không cần quan tâm đến các thẻ này. Tuy nhiên khi có sử dụng chúng, trang Web hoàn toàn tương thích với chuẩn HTML quốc tế và tất cả các trình duyệt Web.

**[V](C:\\Users\\sang\\Downloads\\ViDu\\Vd1_1.htm)****[í dụ](C:\\Users\\sang\\Downloads\\ViDu\\Vd1_1.htm)**

<html>

<head>

<title>nguyen</title>

</head>

<body>

<!- - Nội dung - ->

Chào bạn đến với WebSite trung tâm ABC

</body>

</html>

### Các thẻ cơ bản

1.**<!DOCTYPE>** là dòng đầu tiên của trang HTML, ghi thông tin về version HTML dùng trong trang HTML này.

Ví dụ: <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 transitional//en">

2.**<HTML> … </HTML>** định nghĩa phạm vi của văn bản HTML.

3.**<HEAD> … </HEAD>** định nghĩa các mô tả về trang HTML. Các thông tin tin này sẽ không hiển thị trong nội dung trang web.

4.**<TITLE> … </TITLE>** mô tả tiêu đề của trang, thường được hiển thị trên thanh tiêu đề của cửa sổ hiển thị trang web. Tiêu đề được dùng khi bookmark trang Web này và làm chỉ mục khi tìm kiếm trang Web (search engines index).

5.**< BODY> … </BODY>** xác định phần "thân" của trang web.

6.**<!- - nội dung chú thích … - ->**  trình duyệt web bỏ qua không đọc, không hiển thị trên trang Web.

***Lưu ý***: Không được có khoảng trắng giữa < và !. Nội dung chú thích có thể viết trên nhiều dòng.

**<COMMENT> … </COMMENT>** tương tự <!- - … - ->

7.**<BASE>**

HREF: khai báo URL gốc của tài liệu (dùng để tạo URL tương đối).

TARGET: quy định đích đến mặc định cho các link trong trang Web.

8.**<META>** đặt ở giữa <head>…</head>, thường dùng quy định thuộc tính cho trang web

NAME: nhúng thêm thông tin.

HTTP-EQUIV: tự động chuyển đến trang Web khác và ấn định ngôn ngữ viết kịch bản mặc định.

Ví dụ:

<META name="GENERATOR" content="Microsoft FrontPage 5.0">

<META name="description" content="ABC - Nam Viet Company., Ltd" >

<META http-equiv="Refresh" content=" 3; url=http://domain/directory/file.html">

<META http-equiv=”Content-Type” content="text/html; charset=utf-8">

1. **<HR>** (horizontal rule) thêm đường kẻ ngang.
2. **<BR>** ngắt xuống dòng mới nhưng vẫn thuộc cùng đoạn. Nội dung sau thẻ này sẽ hiển thị từ đầu lề trái của dòng mới. Đây là thẻ rỗng.

**Block là gì?**

Phần tử khối (Block Elements) là thuật ngữ chỉ chung các thẻ HTML có chức năng tạo một khu vực hay là một khối. Khối này có nghĩa là một thẻ khi mà ta khai báo thì nó sẽ được hiển thị ở mỗi dòng riêng biệt bao gồm các nội dung nằm bên trong. Một số thẻ block cơ bản như <p>, <ul>, <ol>, <h1>, <div>,…

**Inline là gì?**

Phần tử nội dòng (Inline Elements) là thuật ngữ chỉ chung các thẻ HTML khi mà khai báo vào nội dung thì nó vẫn sẽ nằm chung một dòng với các văn bản khác. Một số thẻ inline rất hay dùng đó là <b>, <strong>, <i>, <u>,…và đặc biệt là <span> nếu ta cần gộp nhóm các phần tử nào đó mà không ảnh hưởng đến các văn bản chung một hàng, thẻ <span> này có ý nghĩa và cách sử dụng giống như <div> nhưng nó được dùng trong inline.

|  |  |  |
| --- | --- | --- |
| [<a></a>](https://hocwebchuan.com/reference/tag/tag_a.php)  [<abbr></abbr>](https://hocwebchuan.com/reference/tag/tag_abbr.php)  [<acronym></acronym>](https://hocwebchuan.com/reference/tag/tag_acronym.php)  [<b></b>](https://hocwebchuan.com/reference/tag/tag_font_style.php)  [<basefont></basefont>](https://hocwebchuan.com/reference/tag/tag_basefont.php)  [<bdo></bdo>](https://hocwebchuan.com/reference/tag/tag_bdo.php)  [<big></big>](https://hocwebchuan.com/reference/tag/tag_font_style.php)  [<br />](https://hocwebchuan.com/reference/tag/tag_br.php)  [<cite></cite>](https://hocwebchuan.com/reference/tag/tag_font_style.php)  [<code></code>](https://hocwebchuan.com/reference/tag/tag_font_style.php)  [<var></var>](https://hocwebchuan.com/reference/tag/tag_font_style.php) | [<dfn></dfn>](https://hocwebchuan.com/reference/tag/tag_font_style.php)  [<em></em>](https://hocwebchuan.com/reference/tag/tag_font_style.php)  [<font></font>](https://hocwebchuan.com/reference/tag/tag_font.php)  [<i></i>](https://hocwebchuan.com/reference/tag/tag_font_style.php)  [<input></input>](https://hocwebchuan.com/reference/tag/tag_input.php)  [<kbd></kbd>](https://hocwebchuan.com/reference/tag/tag_font_style.php)  [<label></label>](https://hocwebchuan.com/reference/tag/tag_label.php)  [<q></q>](https://hocwebchuan.com/reference/tag/tag_q.php)  [<s></s>](https://hocwebchuan.com/reference/tag/tag_s.php)  [<samp></samp>](https://hocwebchuan.com/reference/tag/tag_font_style.php) | [<select></select>](https://hocwebchuan.com/reference/tag/tag_select.php)  [<small></small>](https://hocwebchuan.com/reference/tag/tag_font_style.php)  [<span></span>](https://hocwebchuan.com/reference/tag/tag_span.php)  [<strike></strike>](https://hocwebchuan.com/reference/tag/tag_s.php)  [<strong></strong>](https://hocwebchuan.com/reference/tag/tag_font_style.php)  [<sub></sub>](https://hocwebchuan.com/reference/tag/tag_sub.php)  [<sup></sup>](https://hocwebchuan.com/reference/tag/tag_sub.php)  [<textarea></textarea>](https://hocwebchuan.com/reference/tag/tag_textarea.php)  [<tt></tt>](https://hocwebchuan.com/reference/tag/tag_font_style.php)  [<u></u>](https://hocwebchuan.com/reference/tag/tag_u.php) |

Một số ký tự đặc biệt: bắt đầu: & kết thúc: ; giữa là tên mã

|  |  |  |
| --- | --- | --- |
| &copy; | © | Copyright (bản quyền) |
| &trade; | ™ | Trademark (nhãn quyền) |
| &reg; | ® | Registered trademark (đăng ký nhãn hiệu) |
| &gt; | > | Greater than (dấu lớn hơn) |
| &lt; | < | Less than (dấu nhỏ hơn) |
| &quot; | " | Double quote (dấu trích) |
| &amp; | & | Ampersand (dấu và) |
| &nbsp; |  | Non breaking space (chèn khoảng trắng) |

Biểu mẫu (Form) là một kỹ thuật cho phép trang web giao tiếp với người truy cập web. Qua đó, người dùng có thể nêu các thắc mắc, đóng góp ý kiến, chọn lựa mục trong danh sách định sẵn, thay đổi các thông tin trên trang web … gởi cho trang Web tiếp nhận và xử lý những thông tin đó.

Form bao gồm các textbox (văn bản), label (nhãn), control (nút) …

### I. Tạo Form

Cú pháp Form tổng quát:

**<form name=”” method=”post/get” action=”url script” > nội\_dung </form>**

name: tên phân biệt khi có nhiều form

method : chỉ định phương thức gửi thông tin là post hay get.

action: chỉ định chương trình tiếp nhận và xử lý thông tin, trong đó url script là địa chỉ xử lý thông tin trong Form.

**1. TextBox (văn bản)**

**<input type=”text” name=”TextName” value=”DefaultValue” size=”n” maxlength=”n” placeholder=”abc”>**

**input là empty thẻ.** Có thể nhập thêm nhãn để hướng dẫn nhập liệu (nhãn thường bên trái).

|  |  |
| --- | --- |
| type=”text” | ghi chính xác để khai báo kiểu TextBox |
| name | khai báo tên nhận diện TextBox |
| value | dữ liệu hiển thị trong TextBox, người dùng có thể nhập dữ liệu mới, nếu không ghi thuộc tính này thì TextBox rỗng. |
| size | chỉ định kích thước textbox tính bằng ký tự. Mặc định là 20 |
| maxlength | chỉ định kích thước tối đa của chuỗi ký tự có thể nhập vào TextBox |

**2. PasswordBox (mật mã)**

**<input type=”password” name=”PassName” value=”DefaultValue” size=”n” maxlength=”n”>**

Cũng gần giống một TextBox nhưng khi người dùng nhập nội dung thì thông tin chỉ hiển thị là các dấu hoa thị \* hoặc dấu chấm.

**[Ví](C:\\Users\\sang\\Downloads\\ViDu\\Vd5_1.htm)** **[d](C:\\Users\\sang\\Downloads\\ViDu\\Vd5_1.htm)****[ụ](C:\\Users\\sang\\Downloads\\ViDu\\Vd5_1.htm)**

<body>

<form>

<h3>ĐĂNG NHẬP</h3>

<pre>

Tên đăng nhập: <input type="text" name="username" maxlength="20">

Mật khẩu: <input type="password" name="password" size="20" maxlength="20" >

</pre>

</form>

</body>

**3. TextArea**

Là hộp văn bản lớn cho phép nhập nhiều dòng văn bản.

**<textarea name=”TextAreaName ” rows=”n1” cols=”n2” wrap=” off/physical/virtual” >**

**..............**

**</textarea>**

|  |  |
| --- | --- |
| rows=”n1” , cols=”n2” | khai báo kích thước hiển thị của hộp thoại, rows là dòng và cols là cột, đơn vị tính là ký tự. Mặc định n1=2, n2=20. |
| wrap | cuộn văn bản xuống dòng khi đến lề phải, có 3 giá trị: off (tắt)/ physical(xuống dòng ký tự khi đụng lề)/ virtual(xuống dòng cả từ khi đụng lề) |
| ............. | văn bản hiển thị trong TexArea. Nhập tối đa 32.700 ký tự. |

**4. Radio button**

Gồm tập hợp ít nhất hai nút, tại một thời điểm chỉ có một nút được chọn.

**<input type=”radio” name=”RadioName” value=”Data” checked>**

|  |  |
| --- | --- |
| name | các nút trong bộ radio thì có cùng tên |
| value | giá trị được gửi đến Server nếu nút được chọn. Nếu không ghi thì trị mặc định được gửi là ON như không ý nghĩa vì không phân biệt nút chọn. |
| checked | chỉ định nút chọn mặc định |

Có thể nhập nhãn để hướng dẫn nhập liệu (nhãn thường bên phải).

**5. CheckBox**

Gồm 1 hay nhiều ô, tại một thời điểm có một hoặc nhiều ô được chọn hay không có ô nào được chọn.

**<input type=”checkbox” name=”CheckboxName” value=”Data” checked>**

**6. Button**

**<input type=”” name=”ButtonName” value=”Data” >**

|  |  |
| --- | --- |
| type=”button” | Thông thường |
| type=”reset” | Cho phép người dùng bắt đầu biểu mẫu mới. Tất cả các thành phần trên form về trạng thái ban đầu |
| type=“submit” | Thi hành url script trong thuộc tính action, chuyển toàn bộ dữ liệu trong form đến nơi nhận. |
| value | Khai báo nhãn của nút (chuỗi hiển thị trên nút). |

**7. Image button**

Giống như Submit button nhưng thay chữ bằng ảnh. Ảnh được dùng có dạng gif và jpg.

**<input type=”image” name=”ImageName” src=”url” border=”n” width=”n” height=”n”>**

**8. Drop down menu**

**<select name=”MenuName” size=”n” multiple>**

**<option value=”data\_1” selected> text\_1**

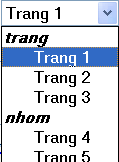
**…………..**

**<option value=”data\_2” > text\_2**

**</select>**

|  |  |
| --- | --- |
| size | chiều cao menu tính bằng dòng, n không vượt quá số mục menu nếu không người dùng có thể nhập vào vùng trống. (số option hiển thị cùng lúc) |
| multiple | cho phép chọn nhiều mục cùng lúc (nhấn ctrl/ shift) |
| value | dữ liệu chuyển đi nếu option này được chọn |
| selected | mục tương ứng được chọn mặc định. Nếu không ghi thì option đầu được hiển thị |
| text\_x | nội dung hiển thị của mỗi mục (nhãn) |

**9. Menu có menu con**

<select name=”MenuName” size=”n” multiple>

<optgroup label=”title\_1”>

<option value=”data\_1” > text\_1

…………..

</optgroup>

<optgroup label=”title\_2”>

<option value=”data\_2” > text\_2

…………..

</optgroup>

</select>

### 

### II. Nâng cấp Form

**1. Tạo button có image trên nền button chỉ có IE hỗ trợ**

**<button type=”submit/reset/normal” name=”SubmitName” value=”data”>**

**chuỗi ký tự hiển thị bên trái hình ảnh**

**<img src=”hinh.gif”>**

**chuỗi ký tự hiển thị bên phải hình ảnh**

**</button>**

**2. Vô hiệu hoá thành phần FORM**

Ngăn người dùng tác dụng đến thành phần đó bằng cách thêm thuộc tính disabled vào thẻ mở của thành phần đó.

Muốn thành phần đó có hiệu lực trở lại, cách duy nhất là thực hiện thông qua Script.

Netscape không hỗ trợ thuộc tính này.

**3. Không cho thay đổi nội dung**

Để bảo vệ nội dung thành phần được định trước và không cho người dùng thay đổi thì thêm vào thuộc tính thuộc tính readonly.

Netscape không hỗ trợ thuộc tính này.

**4. Dùng thành phần ẩn**

Đây là trường hợp nội dung thành phần được định trước và không hiển thị. Thành phần này dùng để xử lý Script.

<input type=”hidden” name=”tên” value=”dữ\_liệu ”>

**5. Lập thứ tự tab**

Để thiết lập thứ tự TAB thêm thuộc tính tabindex=n vào thẻ tạo các thành phần

Trị hợp lệ của n là từ 0 đến 32767.

**6. Tạo phím tắt**

Thêm thuộc tính accesskey=”ký\_tự\_phím” vào thẻ tạo thành phần.

Ví dụ: accesskey=”ALT-a”

Dùng cẩn thận vì có thể vô hiệu hoá các phím tắt của trình duyệt.

Netscape không hỗ trợ thuộc tính này.

**7. Tạo nút upload file**

**<input type=”file” name=”FileName”>**

**[Ví](C:\\Users\\sang\\Downloads\\ViDu\\Vd5_2.htm)** **[d](C:\\Users\\sang\\Downloads\\ViDu\\Vd5_2.htm)****[ụ](C:\\Users\\sang\\Downloads\\ViDu\\Vd5_2.htm)**

<form enctype="multipart/form.data" action=" " method="post" >

Please choose a picture to upload.

<br><input name="FName" type="file" accept="image/\*">

<br><input type="submit" value="submit me!">

</form>

enctype: đặc tả kiểu mã hoá thông tin lấy từ form khi gửi đi. Loại thông thường là URL\_encoded (thay spaces bằng + …) chỉ tốt cho thông tin dạng text. Để truyền file hiệu quả nên dùng multipart/form.data.

accept="image/\*">: giới hạn loại file sẽ xuất hiện trong hộp thoại.

***Lưu ý***: Không thể định dạng chữ trên button hoặc trong textbox.Trong 1 trang HTML có thể có nhiều form nhưng không được lồng nhau.

1. **CSS - Cascading Style Sheet**

- Mục đích: Hỗ trợ việc thiết kế và hiệu chỉnh các định dạng trong trang web dễ dàng, thống nhất trong toàn WebSite.

Mô tả cách hiển thị các thành phần trong trang Web.

**- Phân loại: Inline, Embedded, External.**

Inline style sheet: dùng thuộc tính style cho từng thẻ HTML

Embedded style sheet: định nghĩa các định dạng trong thẻ <style> trong phần <head> của webpage

External style sheet: định nghĩa các định dạng trong file .css và các webpage link tới file .css (trong phần <head>)

**Inline Style:** <TênThẻ style=”property:value; property:value….”> </ThẻName>

**Embedded Style:** Chèn thẻ style trong phần <head></head>

<style type="text/css">

<!--

Selector { property: value; property: value; ... }

………..

-->

</style>

**External Style:** thêm style cho nhiều trang web <link href="filename.css" rel="stylesheet" >

- Thứ tự ưu tiên áp dụng định dạng khi sử dụng các loại CSS (độ ưu tiên giảm dần)

Inline Style Sheet

Embedding Style Sheet

External Style Sheet

Browser Default

- Đơn vị đo trong CSS:

+ Đơn vị kích thước tuyệt đối ( Absolute Lengths)

Giá trị độc lập, không liên quan đến các phần tử thành phần khác

|  |  |
| --- | --- |
| Ký hiệu Đơn vị | Diễn giải |
| cm | centimet (dài bằng 1/100 m) |
| mm | milimét (dài bằng 1/1000 m) |
| in | Inche: 1in = 2.54cm = 96px |
| px | Điểm ảnh (10px là 10 điểm ảnh màn hình) |
| pt | Điểm (point), đây là đơn vị phổ biến dùng biểu diễn kích thước font chữ (72pt = 1inch = 25.4mm) hay 1pt = (1/72)in |

+ Đơn vị kích thước tương đối (Relative Lengths)

Có độ lớn thực tế phụ thuộc vào thành phần khác, thường là phụ thuộc vào thuộc tính phần tử cha trong HTML hoặc viewport.

|  |  |
| --- | --- |
| Ký hiệu Đơn vị | Diễn giải |
| em | Bằng cỡ font của phần tử cha. Có nghĩa phần tử cha có cỡ (font-size) là 14px, thì 1em là 14px - nếu cỡ font phần tử cha là 10pt thì 2em là 20px.  <div style="font-size: 20px">  font-size phần tử cha 20px  <div style="font-size: 0.5em">  Cỡ font thiết lập 0.5em (bằng nửa phần tử cha)  </div>  </div>  font-size phần tử cha 20px  Cỡ font thiết lập 0.5em (bằng nửa phần tử cha) |
| rem | Bằng cỡ font của phần tử gốc - root, trong CSS để thiết lập các thuộc tính của root thì dùng ký hiệu :root, rồi định nghĩa các thuộc tính giống class:  :root {  font-size: 16px;  color: #333;  } |
| lh | Bằng chiều cao dòng phần tử cha |
| vw | Bằng 1% chiều rộng cửa sổ |
| vh | Bằng 1% chiều cao cửa sổ |

Các thuộc tính có trong CSS:

- Background

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [background](http://www.w3schools.com/css/pr_background.asp) | A shorthand property for setting all background properties in one declaration | *background-color background-image background-repeat background-attachment background-position* |
| [background-attachment](http://www.w3schools.com/css/pr_background-attachment.asp) | Sets whether a background image is fixed or scrolls with the rest of the page | scroll fixed |
| [background-color](http://www.w3schools.com/css/pr_background-color.asp) | Sets the background color of an element | *color-rgb color-hex color-name* transparent |
| [background-image](http://www.w3schools.com/css/pr_background-image.asp) | Sets an image as the background | url(*URL*) none |
| [background-position](http://www.w3schools.com/css/pr_background-position.asp) | Sets the starting position of a background image | top left top center top right center left center center center right bottom left bottom center bottom right *x% y%* *xpos ypos* |
| [background-repeat](http://www.w3schools.com/css/pr_background-repeat.asp) | Sets if/how a background image will be repeated | repeat repeat-x repeat-y no-repeat |

- Border:

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [border](http://www.w3schools.com/css/pr_border.asp) | A shorthand property for setting all of the properties for the four borders in one declaration | *border-width border-style border-color* |
| [border-bottom](http://www.w3schools.com/css/pr_border-bottom.asp) | A shorthand property for setting all of the properties for the bottom border in one declaration | *border-bottom-width border-style border-color* |
| [border-bottom-color](http://www.w3schools.com/css/pr_border-bottom_color.asp) | Sets the color of the bottom border | *border-color* |
| [border-bottom-style](http://www.w3schools.com/css/pr_border-bottom_style.asp) | Sets the style of the bottom border | *border-style* |
| [border-bottom-width](http://www.w3schools.com/css/pr_border-bottom_width.asp) | Sets the width of the bottom border | thin medium thick *length* |
| [border-color](http://www.w3schools.com/css/pr_border-color.asp) | Sets the color of the four borders, can have from one to four colors | *color* |
| [border-left](http://www.w3schools.com/css/pr_border-left.asp) | A shorthand property for setting all of the properties for the left border in one declaration | *border-left-width border-style border-color* |
| [border-left-color](http://www.w3schools.com/css/pr_border-left_color.asp) | Sets the color of the left border | *border-color* |
| [border-left-style](http://www.w3schools.com/css/pr_border-left_style.asp) | Sets the style of the left border | *border-style* |
| [border-left-width](http://www.w3schools.com/css/pr_border-left_width.asp) | Sets the width of the left border | thin medium thick *length* |
| [border-right](http://www.w3schools.com/css/pr_border-right.asp) | A shorthand property for setting all of the properties for the right border in one declaration | *border-right-width border-style border-color* |
| [border-right-color](http://www.w3schools.com/css/pr_border-right_color.asp) | Sets the color of the right border | *border-color* |
| [border-right-style](http://www.w3schools.com/css/pr_border-right_style.asp) | Sets the style of the right border | *border-style* |
| [border-right-width](http://www.w3schools.com/css/pr_border-right_width.asp) | Sets the width of the right border | thin medium thick *length* |
| [border-style](http://www.w3schools.com/css/pr_border-style.asp) | Sets the style of the four borders, can have from one to four styles | none hidden dotted dashed solid double groove ridge inset outset |
| [border-top](http://www.w3schools.com/css/pr_border-top.asp) | A shorthand property for setting all of the properties for the top border in one declaration | *border-top-width border-style border-color* |
| [border-top-color](http://www.w3schools.com/css/pr_border-top_color.asp) | Sets the color of the top border | *border-color* |
| [border-top-style](http://www.w3schools.com/css/pr_border-top_style.asp) | Sets the style of the top border | *border-style* |
| [border-top-width](http://www.w3schools.com/css/pr_border-top_width.asp) | Sets the width of the top border | thin medium thick *length* |
| [border-width](http://www.w3schools.com/css/pr_border-width.asp) | A shorthand property for setting the width of the four borders in one declaration, can have from one to four values | thin medium thick *length* |

- Classification:

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [clear](http://www.w3schools.com/css/pr_class_clear.asp) | Sets the sides of an element where other floating elements are not allowed | left right both none |
| [cursor](http://www.w3schools.com/css/pr_class_cursor.asp) | Specifies the type of cursor to be displayed | *url* auto crosshair default pointer move e-resize ne-resize nw-resize n-resize se-resize sw-resize s-resize w-resize text wait help |
| [display](http://www.w3schools.com/css/pr_class_display.asp) | Sets how/if an element is displayed | none inline block list-item run-in compact marker table inline-table table-row-group table-header-group table-footer-group table-row table-column-group table-column table-cell table-caption |
| [float](http://www.w3schools.com/css/pr_class_float.asp) | Sets where an image or a text will appear in another element | left right none |
| [position](http://www.w3schools.com/css/pr_class_position.asp) | Places an element in a static, relative, absolute or fixed position | static relative absolute fixed |
| [visibility](http://www.w3schools.com/css/pr_class_visibility.asp) | Sets if an element should be visible or invisible | visible hidden collapse |

- Dimension:

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [height](http://www.w3schools.com/css/pr_dim_height.asp) | Sets the height of an element | auto *length %* |
| [line-height](http://www.w3schools.com/css/pr_dim_line-height.asp) | Sets the distance between lines | normal *number length %* |
| [max-height](http://www.w3schools.com/css/pr_dim_max-height.asp) | Sets the maximum height of an element | none *length %* |
| [max-width](http://www.w3schools.com/css/pr_dim_max-width.asp) | Sets the maximum width of an element | none *length %* |
| [min-height](http://www.w3schools.com/css/pr_dim_min-height.asp) | Sets the minimum height of an element | *length %* |
| [min-width](http://www.w3schools.com/css/pr_dim_min-width.asp) | Sets the minimum width of an element | *length %* |
| [width](http://www.w3schools.com/css/pr_dim_width.asp) | Sets the width of an element | auto *% length* |

- Font:

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [font](http://www.w3schools.com/css/pr_font_font.asp) | A shorthand property for setting all of the properties for a font in one declaration | *font-style font-variant font-weight font-size/line-height font-family* caption icon menu message-box small-caption status-bar |
| [font-family](http://www.w3schools.com/css/pr_font_font-family.asp) | A prioritized list of font family names and/or generic family names for an element | *family-name generic-family* |
| [font-size](http://www.w3schools.com/css/pr_font_font-size.asp) | Sets the size of a font | xx-small x-small small medium large x-large xx-large smaller larger *length %* |
| [font-size-adjust](http://www.w3schools.com/css/pr_font_font-size-adjust.asp) | Specifies an aspect value for an element that will preserve the x-height of the first-choice font | none *number* |
| [font-stretch](http://www.w3schools.com/css/pr_font_font-stretch.asp) | Condenses or expands the current font-family | normal wider narrower ultra-condensed extra-condensed condensed semi-condensed semi-expanded expanded extra-expanded ultra-expanded |
| [font-style](http://www.w3schools.com/css/pr_font_font-style.asp) | Sets the style of the font | normal italic oblique |
| [font-variant](http://www.w3schools.com/css/pr_font_font-variant.asp) | Displays text in a small-caps font or a normal font | normal small-caps |
| [font-weight](http://www.w3schools.com/css/pr_font_weight.asp) | Sets the weight of a font | normal bold bolder lighter 100 200 300 400 500 600 700 800 900 |

**Generated Content**

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [content](http://www.w3schools.com/css/pr_gen_content.asp) | Generates content in a document. Used with the :before and :after pseudo-elements | *string* *url* counter(*name*) counter(*name, list-style-type*) counters(*name, string*) counters(*name, string, list-style-type*) attr(X) open-quote close-quote no-open-quote no-close-quote |
| [counter-increment](http://www.w3schools.com/css/pr_gen_counter-increment.asp) | Sets how much the counter increments on each occurrence of a selector | none *identifier number* |
| [counter-reset](http://www.w3schools.com/css/pr_gen_counter-reset.asp) | Sets the value the counter is set to on each occurrence of a selector | none *identifier number* |
| [quotes](http://www.w3schools.com/css/pr_gen_quotes.asp) | Sets the type of quotation marks | none *string string* |

**List and Marker**

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [list-style](http://www.w3schools.com/css/pr_list-style.asp) | A shorthand property for setting all of the properties for a list in one declaration | *list-style-type list-style-position list-style-image* |
| [list-style-image](http://www.w3schools.com/css/pr_list-style-image.asp) | Sets an image as the list-item marker | none *url* |
| [list-style-position](http://www.w3schools.com/css/pr_list-style-position.asp) | Sets where the list-item marker is placed in the list | inside outside |
| [list-style-type](http://www.w3schools.com/css/pr_list-style-type.asp) | Sets the type of the list-item marker | none disc circle square decimal decimal-leading-zero lower-roman upper-roman lower-alpha upper-alpha lower-greek lower-latin upper-latin hebrew armenian georgian cjk-ideographic hiragana katakana hiragana-iroha katakana-iroha |
| marker-offset |  | auto *length* |

**Margin**

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [margin](http://www.w3schools.com/css/pr_margin.asp) | A shorthand property for setting the margin properties in one declaration | *margin-top margin-right margin-bottom margin-left* |
| [margin-bottom](http://www.w3schools.com/css/pr_margin-bottom.asp) | Sets the bottom margin of an element | auto *length %* |
| [margin-left](http://www.w3schools.com/css/pr_margin-left.asp) | Sets the left margin of an element | auto *length %* |
| [margin-right](http://www.w3schools.com/css/pr_margin-right.asp) | Sets the right margin of an element | auto *length %* |
| [margin-top](http://www.w3schools.com/css/pr_margin-top.asp) | Sets the top margin of an element | auto *length %* |

**Outlines**

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [outline](http://www.w3schools.com/css/pr_outline.asp) | A shorthand property for setting all the outline properties in one declaration | *outline-color outline-style outline-width* |
| [outline-color](http://www.w3schools.com/css/pr_outline-color.asp) | Sets the color of the outline around an element | *color* invert |
| [outline-style](http://www.w3schools.com/css/pr_outline-style.asp) | Sets the style of the outline around an element | none dotted dashed solid double groove ridge inset outset |
| [outline-width](http://www.w3schools.com/css/pr_outline-width.asp) | Sets the width of the outline around an element | thin medium thick *length* |

**Padding**

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [padding](http://www.w3schools.com/css/pr_padding.asp) | A shorthand property for setting all of  the padding properties in one declaration | *padding-top padding-right padding-bottom padding-left* |
| [padding-bottom](http://www.w3schools.com/css/pr_padding-bottom.asp) | Sets the bottom padding of an element | *length %* |
| [padding-left](http://www.w3schools.com/css/pr_padding-left.asp) | Sets the left padding of an element | *length %* |
| [padding-right](http://www.w3schools.com/css/pr_padding-right.asp) | Sets the right padding of an element | *length %* |
| [padding-top](http://www.w3schools.com/css/pr_padding-top.asp) | Sets the top padding of an element | *length %* |

**Positioning**

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [bottom](http://www.w3schools.com/css/pr_pos_bottom.asp) | Sets how far the bottom edge of an element is above/below the bottom edge of the parent element | auto *% length* |
| [clip](http://www.w3schools.com/css/pr_pos_clip.asp) | Sets the shape of an element. The element is clipped into this shape, and displayed | *shape* auto |
| [left](http://www.w3schools.com/css/pr_pos_left.asp) | Sets how far the left edge of an element is to the right/left of the left edge of the parent element | auto *% length* |
| [overflow](http://www.w3schools.com/css/pr_pos_overflow.asp) | Sets what happens if the content of an element overflow its area | visible hidden scroll auto |
| [position](http://www.w3schools.com/css/pr_class_position.asp) | Places an element in a static, relative, absolute or fixed position | static relative absolute fixed |
| [right](http://www.w3schools.com/css/pr_pos_right.asp) | Sets how far the right edge of an element is to the left/right of the right edge of the parent element | auto *% length* |
| [top](http://www.w3schools.com/css/pr_pos_top.asp) | Sets how far the top edge of an element is above/below the top edge of the parent element | auto *% length* |
| [vertical-align](http://www.w3schools.com/css/pr_pos_vertical-align.asp) | Sets the vertical alignment of an element | baseline sub super top text-top middle bottom text-bottom *length* *%* |
| [z-index](http://www.w3schools.com/css/pr_pos_z-index.asp) | Sets the stack order of an element | auto *number* |

**Table**

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [border-collapse](http://www.w3schools.com/css/pr_tab_border-collapse.asp) | Sets whether the table borders are collapsed into a single border or detached as in standard HTML | collapse separate |
| [border-spacing](http://www.w3schools.com/css/pr_tab_border-spacing.asp) | Sets the distance that separates cell borders (only for the "separated borders" model) | *length length* |
| [caption-side](http://www.w3schools.com/css/pr_tab_caption-side.asp) | Sets the position of the table caption | top bottom left right |
| [empty-cells](http://www.w3schools.com/css/pr_tab_empty-cells.asp) | Sets whether or not to show empty cells in a table (only for the "separated borders" model) | show hide |
| [table-layout](http://www.w3schools.com/css/pr_tab_table-layout.asp) | Sets the algorithm used to display the table cells, rows, and columns | auto fixed |

**Text**

|  |  |  |
| --- | --- | --- |
| **Property** | **Description** | **Values** |
| [color](http://www.w3schools.com/css/pr_text_color.asp) | Sets the color of a text | *color* |
| [direction](http://www.w3schools.com/css/pr_text_direction.asp) | Sets the text direction | ltr rtl |
| [line-height](http://www.w3schools.com/css/pr_dim_line-height.asp) | Sets the distance between lines | normal *number length %* |
| [letter-spacing](http://www.w3schools.com/css/pr_text_letter-spacing.asp) | Increase or decrease the space between characters | normal *length* |
| [text-align](http://www.w3schools.com/css/pr_text_text-align.asp) | Aligns the text in an element | left right center justify |
| [text-decoration](http://www.w3schools.com/css/pr_text_text-decoration.asp) | Adds decoration to text | none underline overline line-through blink |
| [text-indent](http://www.w3schools.com/css/pr_text_text-indent.asp) | Indents the first line of text in an element | *length %* |
| text-shadow |  | none *color* *length* |
| [text-transform](http://www.w3schools.com/css/pr_text_text-transform.asp) | Controls the letters in an element | none capitalize uppercase lowercase |
| unicode-bidi |  | normal embed bidi-override |
| [white-space](http://www.w3schools.com/css/pr_text_white-space.asp) | Sets how white space inside an element is handled | normal pre nowrap |
| [word-spacing](http://www.w3schools.com/css/pr_text_word-spacing.asp) | Increase or decrease the space between words | normal *length* |

**Pseudo-classes**

|  |  |
| --- | --- |
| **Pseudo-class** | **Purpose** |
| [:active](http://www.w3schools.com/css/pr_pseudo_active.asp) | Adds special style to an activated element |
| [:focus](http://www.w3schools.com/css/pr_pseudo_focus.asp) | Adds special style to an element while the element has focus |
| [:hover](http://www.w3schools.com/css/pr_pseudo_hover.asp) | Adds special style to an element when you mouse over  it |
| [:link](http://www.w3schools.com/css/pr_pseudo_link.asp) | Adds special style to an unvisited link |
| [:visited](http://www.w3schools.com/css/pr_pseudo_visited.asp) | Adds special style to a visited link |
| [:first-child](http://www.w3schools.com/css/pr_pseudo_first-child.asp) | Adds special style to an element that is the first child of some other element |
| [:lang](http://www.w3schools.com/css/pr_pseudo_lang.asp) | Allows the author to specify a language to use in a specified element |

**Pseudo-elements**

|  |  |
| --- | --- |
| **Pseudo-element** | **Purpose** |
| [:first-letter](http://www.w3schools.com/css/pr_pseudo_first-letter.asp) | Adds special style to the first letter of a text |
| [:first-line](http://www.w3schools.com/css/pr_pseudo_first-line.asp) | Adds special style to the first line of a text |
| [:before](http://www.w3schools.com/css/pr_pseudo_before.asp) | Inserts some content before an element |
| [:after](http://www.w3schools.com/css/pr_pseudo_after.asp) | Inserts some content after an element |

1. **Giao diện website**

- Web là các dịch vụ phân tán cung cấp thông tin multimedia dựa trên hypertext

+ Phân tán: Thông tin được đặt trên nhiêu máy chủ khắp thế giới

+ Multimedia: thông tin bao gồm text, graphics, sound, video,…

+ Hypertext: là kỹ thuật được sử dụng để truy cập thông tin

- Web = protocol + language + naming infrastructure

- HTTP - HyperText Transfer Protocol

+ Là giao tiếp giữa WWW client và server

- HTML - HyperText Markup Language

+ Ngôn ngữ biểu diễn các tài liệu WWW

- URL - Uniform Resource Locator

+ Địa chỉ web (xác định duy nhất)

- Client-script: VB script, JavaScript

- Server-script: ASP, PHP, JSP, ASP.NET,…

- Địa chỉ IP - IP Address:

+ Là 1 con số 32bit, chia thành 4 số 8bit, VD: 203.162.33.44 (gồm 2 phần: network adddress, host address).

+ Xác định đối tượng nhận và gửi thông tin trên Internet

+ Để biết IP: ping [www.intel.com](http://www.intel.com)

- Tên miền - Domain name:

+ Là tên giao dịch của công ty hay tổ chức trên Internet

+ VD: [www.fit.sgu.edu.vn](http://www.fit.sgu.edu.vn)

- Ánh xạ giữa tên miền và địa chỉ IP:

+ DNS - Domain Name System server

- Host server: dịch vụ lưu trữ web.

- Site map: hệ thống định hướng ( navigation - site map)

Hệ thống menu, naviation đầy đủ, rõ ràng.

Vị trí nhất quán, logic (logo công ty)

Cần quan tâm đến việc sư dụng hiệu ứng.

- Trang chủ:

Toàn bộ website đều được thiết lập quanh trang chủ.

Trang chủ xác định rõ chủ đề website

Trang chủ xác định phong cách thiết kế của website

- Cấu trúc website

Hệ thống phân cấp

Hệ thống các trang tiếp nối

Ô lưới

Mạng nhện.

- Hệ thống liên kết: các trang liên kết được với nhau…….

- Phân biệt giao diện cho ngdung cuối và giao diện cho ng quản trị web:

- Phân loại: Web tĩnh và Web động

- Các thành phần của website:

+ URL

+ Web browser

+ Web page

- Quan điểm khi thiết kế giao diện website

Thiết kế hướng người sử dụng

Các giúp đỡ định hướng rõ ràng

Không có trang cuối (dead-end)

Băng thông và các yếu tố phần cứng

Đơn giản, nhất quán và tính ổn định

Phản hổi đối thoại

Tính tương thích trên các trình duyệt

TK cho quản trị và cập nhật nội dung.

- Nguyên tắc C.R.A.P:

C: Contrast - sự tương phản

R: Repitition - sự lập lại

A: Alignment - sự canh lề

P: Proximity - sự tương đồng

- Các bước xây dựng website:

+ Phân tích yêu cầu

Web để làm gì?

Ai dùng? Trình độ người dùng?

Nội dung, hình ảnh? (Xác định nguồn tài nguyên về nội dung thông tin, hình ảnh … mà chúng ta cần để tạo website. Ngoài ra cần lưu ý nguồn thông tin sẽ duy trì cho website hoạt động sau này)

Mối liên quan giữa các nội dung?

Thứ tự các nội dung?

+ Tạo site map, giao diện

Giao diện, nội dung webpage

Sơ đồ cấu trúc website: liên kết các trang

Cấu trúc thư mục

+ Cài đặt

+ Thử nghiệm

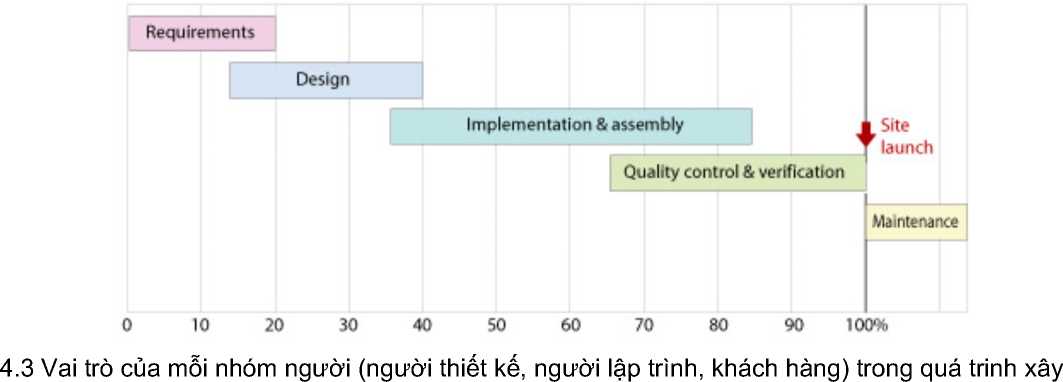
Kiểm tra trên nhiều trình duyệt, loại màn hình

Kiểm tra trên nhiều loại mạng

Kiểm tra tốc độ

Kiểm tra các liên kết

Thử các lỗi bảo mật



* 1. **Kiến thức về NNLT thông dịch client-script: Javascript**

Javascript không phải là Java, Javascript là một ngôn ngữ kịch bản (script) phía server (server side) và phía client (client side) (thông dụng hơn).

Ngôn ngữ kịch bản là ngôn ngữ lập trình kiểu thông dịch. Các đoạn chương trình viết bằng ngôn ngữ này được nhúng trong các trang web sẽ được các trình duyệt thông dịch để thực hiện. Các ngôn ngữ kịch bản cho phép phát triển các chương trình đơn giản nhanh và dễ dàng hơn là các ngôn ngữ lập trình dạng biên dịch C, C++…Tuy nhiên lại khó phát triển các ứng dụng lớn, phức tạp.

Lập trình phía client

* Đoạn chương trình được thi hành tại trình duyệt (browser). Chương trình viết theo dạng này sẽ được load vào trình duyệt và được thi hành một cách tự động.
* Ưu điểm của ngôn ngữ này là trình duyệt sẽ làm tất cả mọi việc, giảm tải gánh nặng cho Server và có thể thi hành nhanh hơn ngôn ngữ phía Server. Khi có yêu cầu từ người dùng, chương trình phía client sẽ tiếp nhận, xử lý và phản hồi ngay thay vì đưa lên Server.
* Thường dùng tính toán, kiểm tra định dạng khi nhập liệu trước khi gởi về Server, trình bày động các đối tượng: thay đổi màu sắc…các yêu cầu này thường không liên quan đến nguồn cơ sở dữ liệu trên server.
  1. **Kiến thức về sử dụng Javascript. Vd: cú pháp, kiểu dữ liệu, biến, hàm...**

Cách đặt mã script trong trang web: có 2 cách

**Cách 1: Chèn trực tiếp đoạn mã lệnh Javascript vào trang web.**

Đặt các dòng mã lệnh của Javascript giữa cặp thẻ <script>…</script> vào bất kỳ vị trí nào trong trang web: phần đầu (<head>…</head>) hoặc phần thân (<body>…</body>). Tuy nhiên nên đặt trong phần đầu để dễ kiểm soát và sửa đổi chương trình. Có thể viết nhiều đoạn mã lệnh Javascript trong cùng một trang web.

Lưu ý phải nhớ đóng thẻ <script> vì nếu không chương trình không hiển thị nội dung trang web.

**[Ví](C:\\Users\\sang\\Downloads\\ViDu\\ViDu_JavaScript\\vd1_1.htm)** **[dụ](C:\\Users\\sang\\Downloads\\ViDu\\ViDu_JavaScript\\vd1_1.htm)**

<html>

<head>

<script language="javascript">

<!--

document.write("Chao ban a");

-->

</script>

</head>

<body>

<script language="javascript">

<!--

document.write("Chao ban b");

-->

</script>

</body>

**Cách 2: Có thể viết một tập tin Javascript riêng và sau đó kết nối với một hoặc nhiều tập tin trang web khác nhau.**

**[Ví d](C:\\Users\\sang\\Downloads\\ViDu\\ViDu_JavaScript\\vd1_2.htm)****[ụ](C:\\Users\\sang\\Downloads\\ViDu\\ViDu_JavaScript\\vd1_2.htm)**

Bước 1:

Soạn thảo tập tin chứa đoạn chương trình Javascript như sau:

document.write(“Chao ban a”);

Lưu lại với tên vd1\_2.js (lưu ý: trong tập tin này không chứa bất kỳ một thẻ nào của HTML).

Bước 2:

Soạn thảo trang web liên kết với tập tin trên

<html>

<body>

<script language="javascript" src="vd1\_2.js">

</script>

</body>

</html>

1. **Kiểu dữ liệu**

Kiểu chuỗi

Kiểu số

Kiểu luận lí

1. **Khai báo biến:** dùng từ khóa var để khai báo
2. **Toán tử**

Toán tử số học: +, -, \*, /, %, ++, --

Toán tử gán: =, +=, -=, \*=, /=, %=

Toán tử so sánh: ==, !=, >, <, >=, <=

Toán tử logic: &&, ||, !

Toán tử chuỗi: +: là phép toán nối các chuỗi với nhau

Toán tử điều kiện: (Điều kiện) ? giá trị 1: giá trị 2

1. **Hàm**

Xây dụng hàm

Cú pháp

function <tên hàm>(tham số 1, tham số 2,….)

{ //đoạn mã lệnh

}

Ví dụ

function cong(a,b)

{

c=a+b

return c

}

Hàm có sẵn

eval(s): trả về giá trị số của s.

isNaN(s): trả về true nếu s không phải là một số, false trong trường hợp ngược lại.

parseInt(s): trả về một số nguyên từ s. Nếu như s theo sau là các ký tự chữ thì các ký tự này sẽ bị bỏ qua. Nếu s bắt đầu bằng ký tự chữ thì hàm trả về giá trị NaN

parseFloat(s): trả về một số thực từ s. Nếu như s theo sau là các ký tự chữ thì các ký tự này sẽ bị bỏ qua. Nếu s bắt đầu bằng ký tự chữ thì hàm trả về giá trị NaN.

alert(): đưa ra thông báo

confirm(): đưa ra thông báo và chờ xác nhận

prompt(): cho ng dung nhập dữ liệu

<script language="javascript">

alert("Xin chao ban!");

var traloi = confirm("Ban co muon minh tinh tuoi cho ban khong?");

if (traloi == true) {

var namsinh = prompt("Nhap vao nam sinh cua ban!");

document.write("Tuoi ban la: "+(2022-namsinh)+" tuoi");

}

else {

alert("Tam biet ban!");

}

</script>

**Tạo form**

<form action="">

<fieldset>

<legend>Form dang nhap</legend>

<label for="name">Ten dang nhap: </label><input id="name" type="text"><br>

<label for="pass">Mat khau: </label><input id="pass" type="password"><br>

<input type="submit" id="btn-submit"> <!--cái này hoặc cái button đều được -->

<button>Submit</button>

</fieldset>

</form>

