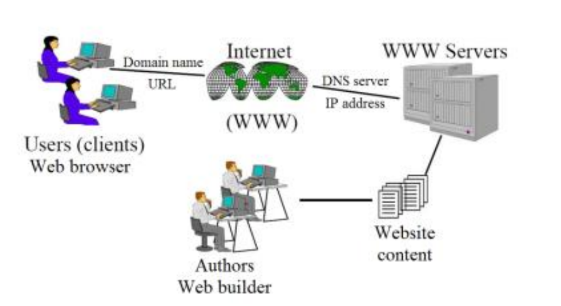
**TỔNG ÔN LẬP TRÌNH WEB VÀ ỨNG DỤNG**

# **Giải thích và mô tả các hình ảnh cho sẵn**

### **Mô hình hoạt động cơ bản của website**

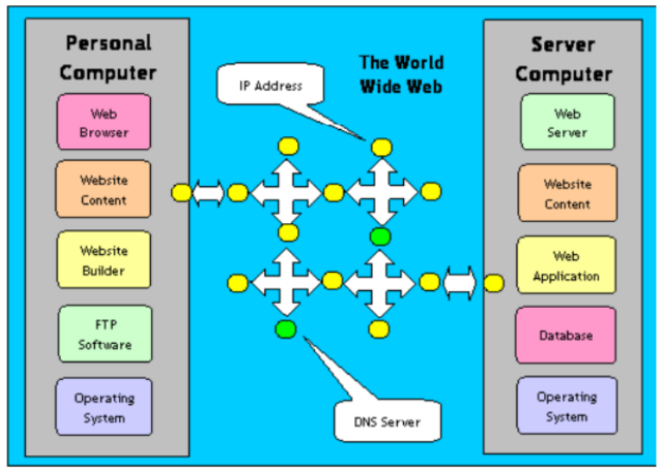


Dưới đây là mô tả các khái niệm và thuật ngữ cũng như giải thích về mô hình hoạt động cơ bản của website:

* **Tên miền (Domain name):**
  + **Khái niệm:** Là một địa chỉ dễ nhớ thay thế cho địa chỉ IP của máy chủ. Thường được sử dụng để truy cập vào các trang web.
  + **Mô hình hoạt động:** Người dùng nhập tên miền vào trình duyệt web, sau đó DNS chuyển đổi nó thành địa chỉ IP của máy chủ tương ứng.
* **URL (Uniform Resource Locator):**
  + **Khái niệm:** Là một chuỗi ký tự mô tả vị trí của một tài nguyên trên Internet.
  + **Mô hình hoạt động:** Người dùng sử dụng URL để xác định vị trí cụ thể của một trang web hoặc tài nguyên trực tuyến.
* **Người dùng (Users):**
  + **Khái niệm:** Những người sử dụng Internet để truy cập thông tin, dịch vụ, và tương tác trực tuyến.
  + **Mô hình hoạt động:** Người dùng sử dụng trình duyệt web để truy cập vào các trang web và tương tác với nội dung.
* **Tác giả (Authors):**
  + **Khái niệm:** Những người tạo ra và duy trì nội dung trên các trang web, bao gồm văn bản, hình ảnh, video, và ứng dụng web.
  + **Mô hình hoạt động:** Tác giả sử dụng các công cụ xây dựng web và ngôn ngữ lập trình để tạo và quản lý nội dung trên trang web.
* **Mạng Toàn Cầu (The World Wide Web):**
  + **Khái niệm:** Là một hệ thống liên kết toàn cầu của các trang web và tài nguyên trên Internet.
  + **Mô hình hoạt động:** Người dùng có thể truy cập và tương tác với các trang web trên toàn thế giới thông qua trình duyệt web.
* **Máy chủ DNS (DNS Server):**
  + **Khái niệm:** Là máy chủ có nhiệm vụ chuyển đổi tên miền thành địa chỉ IP và ngược lại.
  + **Mô hình hoạt động:** Khi người dùng nhập tên miền, DNS chuyển đổi nó thành địa chỉ IP để trình duyệt có thể tìm kiếm máy chủ tương ứng.
* **Địa chỉ IP (IP Address):**
  + **Khái niệm:** Là một dãy số duy nhất xác định vị trí của một máy tính trên mạng Internet.
  + **Mô hình hoạt động:** Địa chỉ IP được sử dụng để định danh máy chủ và các thiết bị kết nối vào Internet.
* **Máy chủ WWW (WWW Servers):**
  + **Khái niệm:** Là máy chủ lưu trữ và cung cấp nội dung website cho người dùng khi được yêu cầu.
  + **Mô hình hoạt động:** Máy chủ WWW xử lý yêu cầu từ trình duyệt và trả về các tài liệu HTML, CSS, JS, hình ảnh, và dữ liệu khác.
* **Nội dung website (Website Content):**
  + **Khái niệm:** Bao gồm mọi thông tin và tài nguyên trên một trang web, bao gồm văn bản, hình ảnh, video, và ứng dụng web.
  + **Mô hình hoạt động:** Nội dung website được tạo, duy trì, và cung cấp cho người dùng thông qua máy chủ WWW và trình duyệt web.
* **Quá trình hoạt động của website có thể được tóm tắt như sau:**

1. Khi người dùng nhập một tên miền hoặc URL vào trình duyệt, trình duyệt sẽ gửi một yêu cầu HTTP đến máy chủ DNS để tìm địa chỉ IP của máy chủ WWW chứa website đó.
2. Sau khi nhận được địa chỉ IP, trình duyệt sẽ gửi một yêu cầu HTTP đến máy chủ WWW và yêu cầu nội dung của website.
3. Máy chủ WWW sẽ tìm kiếm nội dung website trong bộ nhớ hoặc cơ sở dữ liệu và gửi lại cho trình duyệt dưới dạng các tài liệu HTML, CSS, JS, hình ảnh, v.v.
4. Trình duyệt sẽ phân tích và hiển thị nội dung website lên màn hình cho người dùng.

### **Mô hình hoạt động cơ bản của website**



Hình dưới đây minh họa khái niệm và các thuật ngữ về cách một máy tính cá nhân tương tác với Mạng Toàn Cầu và một máy tính máy chủ thông qua địa chỉ IP và máy chủ DNS.

**Một số khái niệm và thuật ngữ quan trọng**

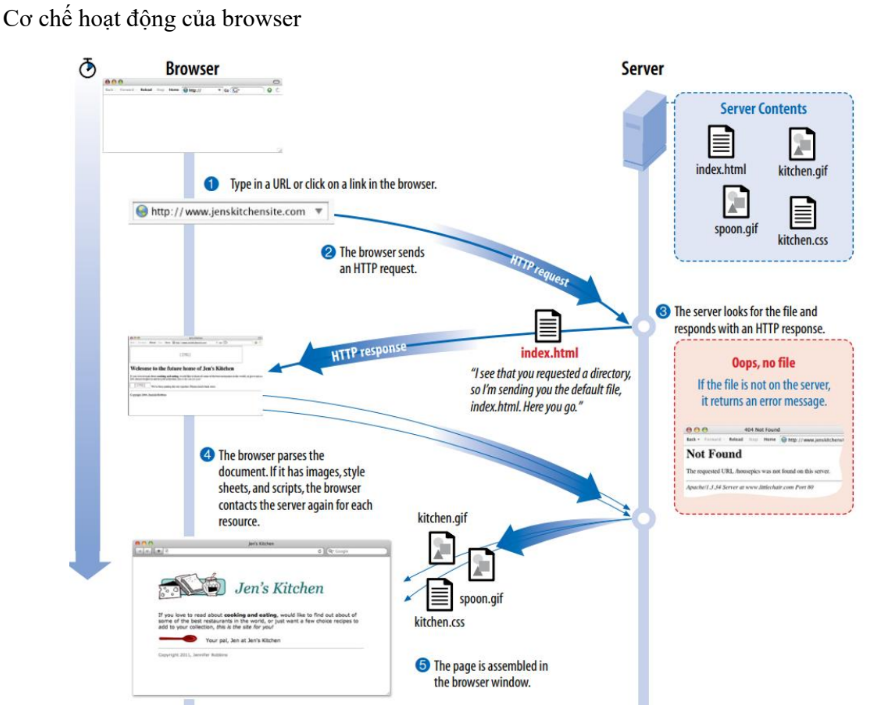
* **Máy Tính Cá Nhân (Personal Computer):**
  + Định nghĩa: Là thiết bị máy tính được sử dụng cá nhân, bao gồm các thành phần như bộ xử lý, bộ nhớ, ổ cứng, và màn hình. Máy tính cá nhân là nơi người dùng tương tác với các ứng dụng và truy cập Internet.
* **Trình Duyệt Web (Web Browser):**
  + Định nghĩa: Là phần mềm cho phép người dùng duyệt và tương tác với các trang web trên Internet. Các trình duyệt phổ biến bao gồm Chrome, Firefox, Safari, và Edge.
* **Phần Mềm Xây Dựng Website (Website Builder):**
  + Định nghĩa: Là các công cụ hoặc ứng dụng giúp người dùng tạo và thiết kế trang web mà không cần kiến thức lập trình sâu. Cung cấp giao diện đồ họa để tạo và quản lý nội dung trang web.
* **Phần Mềm FTP (FTP Software):**
  + Định nghĩa: Là phần mềm sử dụng giao thức FTP (File Transfer Protocol) để truyền tải dữ liệu giữa máy tính cá nhân và máy chủ. Thường được sử dụng để tải lên nội dung website lên máy chủ.
* **Hệ Điều Hành (Operating System):**
  + Định nghĩa: Là phần mềm quản lý và điều khiển các hoạt động của máy tính cá nhân. Bao gồm các chức năng như quản lý tài nguyên, giao tiếp với phần cứng, và cung cấp giao diện người dùng.
* **Địa Chỉ IP (IP Address):**
  + Định nghĩa: Là dãy số duy nhất định danh một máy tính trên mạng Internet. Có hai loại địa chỉ IP: IPv4 (ví dụ: 192.168.0.1) và IPv6.
* **Mạng Toàn Cầu (The World Wide Web):**
  + Định nghĩa: Là một hệ thống liên kết các trang web và tài nguyên trên Internet. Người dùng có thể truy cập và tương tác với thông tin trên toàn thế giới thông qua [WWW](http://www/).
* **Máy Chủ DNS (DNS Server):**
  + Định nghĩa: Là máy chủ có nhiệm vụ chuyển đổi tên miền thành địa chỉ IP và ngược lại. DNS giúp người dùng truy cập website bằng cách sử dụng tên miền dễ nhớ thay vì địa chỉ IP.
* **Máy Tính Máy Chủ (Server Computer):**
  + Định nghĩa: Là máy tính được cấu hình để cung cấp dịch vụ cho các máy tính khác trong mạng, thường là máy chủ web, máy chủ email, hoặc máy chủ cơ sở dữ liệu.
* **Máy Chủ Web (Web Server):**
  + Định nghĩa: Là máy chủ chứa và phục vụ các trang web cho người dùng qua Internet. Xử lý các yêu cầu HTTP từ trình duyệt và trả về nội dung trang web.
* **Ứng Dụng Web (Web Application):**
  + Định nghĩa: Là phần mềm được thiết kế để hoạt động qua trình duyệt web. Cung cấp các chức năng tương tác và thường liên quan đến cơ sở dữ liệu để lưu trữ thông tin.
* **Cơ Sở Dữ Liệu (Database):**
  + Định nghĩa: Là nơi lưu trữ và quản lý dữ liệu. Cơ sở dữ liệu được sử dụng trong ứng dụng web để lưu trữ thông tin như thông tin người dùng, nội dung trang web, và dữ liệu khác.

**Nguyên tắc hoạt động của một website có thể được mô tả qua các bước và thuật ngữ sau**

1. **Người Dùng và Máy Tính Cá Nhân:**
   * Người dùng sử dụng máy tính cá nhân của mình để truy cập và tương tác với các trang web.
2. **Trình Duyệt Web (Web Browser):**
   * Người dùng sử dụng trình duyệt web để nhập địa chỉ URL hoặc tên miền của trang web mà họ muốn truy cập.
3. **Mạng Toàn Cầu (The World Wide Web):**
   * Internet kết nối máy tính cá nhân của người dùng với các máy chủ trên toàn thế giới thông qua các giao thức truyền thông như TCP/IP, HTTP, HTTPS.
4. **Địa Chỉ IP (IP Address):**
   * Khi nhập tên miền vào trình duyệt, trình duyệt gửi yêu cầu DNS để chuyển đổi tên miền thành địa chỉ IP của máy chủ chứa trang web.
5. **Máy Chủ DNS (DNS Server):**
   * Máy chủ DNS giúp chuyển đổi tên miền thành địa chỉ IP và ngược lại, giúp trình duyệt xác định vị trí của máy chủ chứa trang web.
6. **Máy Chủ Web (Web Server):**
   * Sau khi có địa chỉ IP, trình duyệt kết nối đến máy chủ web, nơi chứa nội dung của trang web.
7. **Ứng Dụng Web (Web Application):**
   * Máy chủ web có thể chứa các ứng dụng web, cung cấp chức năng nâng cao như đăng nhập, đăng ký, tìm kiếm, và giao tiếp với cơ sở dữ liệu.
8. **Cơ Sở Dữ Liệu (Database):**
   * Thông tin cần thiết như dữ liệu người dùng, nội dung trang web được lưu trữ và quản lý trong cơ sở dữ liệu.
9. **Phần Mềm FTP (FTP Software):**
   * Nếu có sửa đổi nội dung, người quản trị có thể sử dụng phần mềm FTP để tải lên nội dung mới lên máy chủ web.
10. **Hệ Điều Hành (Operating System):**
    * Máy chủ web chạy trên một hệ điều hành để quản lý tài nguyên và xử lý các yêu cầu từ trình duyệt.
11. **Trả Về Nội Dung (Response):**
    * Máy chủ web trả về trình duyệt các tài liệu HTML, CSS, JS, hình ảnh, và dữ liệu khác, mà trình duyệt sẽ hiển thị cho người dùng.

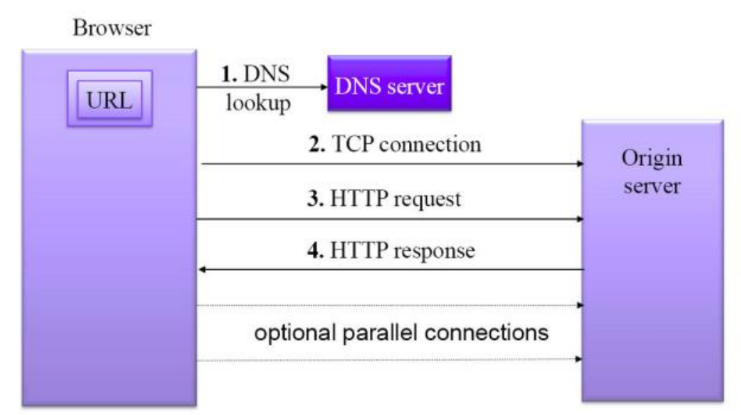
* Tóm lại, nguyên tắc hoạt động của website dựa vào sự tương tác giữa máy tính cá nhân, trình duyệt web, máy chủ DNS, máy chủ web, cơ sở dữ liệu và các ứng dụng web, đảm bảo việc truyền tải và hiển thị thông tin một cách hiệu quả và tương tác.

### **Cơ chế hoạt động của browser**



* + Bước 1: Nhập URL vào trình duyệt.
  + Bước 2: Gửi yêu cầu HTTP đến máy chủ DNS.
  + Bước 3: Nhận địa chỉ IP của máy chủ web từ máy chủ DNS.
  + Bước 4: Gửi yêu cầu HTTP đến máy chủ web.
  + Bước 5: Nhận các tài liệu HTML, CSS, JS, hình ảnh, v.v. từ máy chủ web.
  + Bước 6: Phân tích cú pháp tài liệu HTML và tạo cây nội dung (content tree).
  + Bước 7: Phân tích cú pháp các tệp CSS và tạo cây dựng hình (render tree).
  + Bước 8: Tính toán layout và painting cho các nút DOM.
  + Bước 9: Phân tích và thực thi các tệp JavaScript và xử lý các sự kiện người dùng.
* **Máy Tính Cá Nhân:**
  + **Trình Duyệt Web (Web Browser):** Người dùng sử dụng trình duyệt web để truy cập các website. Trình duyệt này cho phép họ nhập tên miền hoặc URL của trang web mà họ muốn xem.
  + **Công Cụ Xây Dựng Web và Ngôn Ngữ Lập Trình Web:** Người dùng có thể tạo nội dung website bằng cách sử dụng công cụ xây dựng web hoặc lập trình bằng các ngôn ngữ như HTML, CSS, JavaScript.
  + **Phần Mềm FTP (FTP Software):** Để tải lên nội dung của trang web lên máy chủ, người dùng sử dụng phần mềm FTP để truyền tải tệp tin giữa máy tính cá nhân và máy chủ.
  + **Hệ Điều Hành (Operating System):** Máy tính cá nhân cần có hệ điều hành để quản lý các hoạt động của nó.
* **Internet (WWW):**
  + **Mạng Lưới Toàn Cầu:** Internet là một mạng lưới kết nối các máy tính trên toàn thế giới, sử dụng các giao thức như TCP/IP, HTTP, HTTPS, FTP. Điều này cho phép người dùng gửi và nhận thông tin từ các website khác nhau.
* **Máy Chủ DNS (DNS Server):**
  + **Chuyển Đổi Tên Miền thành Địa Chỉ IP:** Máy chủ DNS chuyển đổi tên miền thành địa chỉ IP và ngược lại. Điều này giúp trình duyệt xác định địa chỉ IP của máy chủ WWW chứa trang web cần truy cập.
* **Máy Chủ WWW (WWW Server):**
  + **Lưu Trữ và Cung Cấp Nội Dung:** Máy chủ WWW lưu trữ và cung cấp nội dung của trang web cho người dùng khi họ yêu cầu.
  + **Phần Mềm Máy Chủ:** Sử dụng các phần mềm như Apache, Nginx, IIS để xử lý yêu cầu HTTP từ trình duyệt và trả về các tài liệu HTML, CSS, JS, hình ảnh.
  + **Ứng Dụng Web và Kết Nối Cơ Sở Dữ Liệu:** Máy chủ WWW có thể chạy các ứng dụng web để xử lý các chức năng như đăng nhập, đăng ký, tìm kiếm. Nó cũng có thể kết nối với cơ sở dữ liệu để lưu trữ và truy xuất dữ liệu liên quan đến trang web.
* Tất cả các thành phần này tương tác với nhau để tạo ra một trải nghiệm web toàn diện từ khi người dùng nhập URL cho đến khi họ nhận được và hiển thị nội dung trên trình duyệt web của họ.

**4. Cơ chế hoạt động của DNS**



* Sơ đồ minh họa quá trình hoạt động của DNS (Domain Name System).
  + - DNS là một hệ thống giúp chuyển đổi tên miền thành địa chỉ IP và ngược lại.
    - DNS cho phép bạn truy cập vào các website bằng cách nhập tên miền dễ nhớ thay vì phải ghi nhớ một dãy số khó nhớ.
    - DNS cũng giúp quản lý các tài nguyên trên mạng Internet.
* Quá trình hoạt động của DNS có thể được mô tả như sau:
  + Bước 1: Khi bạn nhập một URL hoặc nhấp vào một liên kết trong trình duyệt, trình duyệt sẽ gửi một yêu cầu DNS đến máy chủ DNS gần nhất, gọi là DNS Recursor. Yêu cầu DNS là một thông điệp dùng để hỏi địa chỉ IP của một tên miền.
  + Bước 2: DNS Recursor sẽ tìm kiếm địa chỉ IP của tên miền trong bộ nhớ đệm của nó. Nếu không có, nó sẽ tiếp tục gửi yêu cầu đến các máy chủ DNS khác theo một cấu trúc phân cấp, gồm có Root Name Server, TLD Name Server, Authoritative Name Server và Secondary Name Server. Các máy chủ DNS này sẽ trả về địa chỉ IP của máy chủ DNS có thẩm quyền hoặc có trách nhiệm lưu trữ thông tin về tên miền đó.
  + Bước 3: Sau khi nhận được địa chỉ IP từ Authoritative Name Server, DNS Recursor sẽ gửi lại cho trình duyệt và lưu vào bộ nhớ đệm để sử dụng cho lần truy vấn sau. Trình duyệt sẽ sử dụng địa chỉ IP để thiết lập kết nối với máy chủ web và yêu cầu nội dung của website.

# **Các bước thiết lập website**

### **Hosting và domain**

* Hosting là dịch vụ cung cấp không gian lưu trữ trên máy chủ (server) để chứa các tập tin, dữ liệu, mã nguồn của website. Hosting giúp website có thể hoạt động liên tục và truy cập được từ bất kỳ thiết bị nào có kết nối internet.
* Domain là tên định danh duy nhất của một website trên internet, ví dụ như “bing.com” hay “google.com”. Domain giúp người dùng dễ dàng nhớ và tìm kiếm website mà không cần phải ghi nhớ địa chỉ IP của máy chủ.

### **Các khái niệm liên quan**

* Dung lượng lưu trữ (disk space) là sức chứa tối đa của hosting, thường được tính bằng megabyte (MB) hoặc gigabyte (GB). Dung lượng lưu trữ phụ thuộc vào số lượng và kích thước của các tập tin, dữ liệu, mã nguồn của website. Bạn cần chọn dung lượng lưu trữ phù hợp với nhu cầu của website để đảm bảo hoạt động ổn định và hiệu quả.
* Băng thông (bandwidth) là lượng dữ liệu có thể truyền tải qua lại giữa hosting và người dùng trong một khoảng thời gian nhất định, thường là một tháng. Băng thông được tính bằng megabit (Mb) hoặc gigabit (Gb) mỗi giây. Băng thông ảnh hưởng đến tốc độ tải và hiển thị của website. Bạn cần chọn băng thông đủ lớn để đáp ứng lượng truy cập của website và tránh bị giới hạn hoặc phát sinh chi phí phụ.
* Tên miền con (sub-domain) là phần mở rộng của tên miền chính, có thể được sử dụng để tạo ra các website riêng biệt với địa chỉ URL khác nhau, ví dụ như “blog.bing.com” hay “maps.google.com”. Tên miền con giúp bạn phân loại và quản lý các nội dung khác nhau trên website một cách hiệu quả và linh hoạt.
* Email là dịch vụ cho phép bạn gửi và nhận thư điện tử trên internet. Email có thể được tạo ra với tên miền của bạn, ví dụ như “info@bing.com” hay “support@google.com”. Email giúp bạn giao tiếp và xây dựng uy tín với khách hàng và đối tác một cách chuyên nghiệp và an toàn.
* Ngôn ngữ lập trình web là ngôn ngữ được sử dụng để viết mã nguồn cho website, ví dụ như HTML, CSS, JavaScript, PHP, Python, v.v. Ngôn ngữ lập trình web giúp bạn tạo ra các nội dung, chức năng, giao diện, tương tác cho website theo ý muốn và nhu cầu của bạn.
* Cơ sở dữ liệu (database) là nơi lưu trữ và quản lý các dữ liệu liên quan đến website, ví dụ như bài viết, hình ảnh, thông tin khách hàng, đơn hàng, v.v. Cơ sở dữ liệu giúp bạn lấy, thêm, sửa, xóa các dữ liệu một cách nhanh chóng và dễ dàng. Các cơ sở dữ liệu phổ biến được web server hỗ trợ là MySQL, PostgreSQL, MongoDB, v.v.

# **Website**

### **Phân loại website: website tĩnh và website động**

Website được phân loại theo dạng cấu trúc có thể hiểu là dạng dữ liệu và cách vận hành của trang web. Chúng gồm 2 loại như sau:

* **Website tĩnh (static website)**

Website tĩnh là dữ liệu web không được thay đổi thường xuyên. Website tĩnh được lập trình trên nền tảng HTML, CSS và Javascript. Nếu bạn muốn thay đổi nội dung trên web quản trị viên phải thực hiện sửa đổi trực tiếp trên mã lệnh và phải am hiểu về ngôn ngữ lập trình. Vì thế nên website tĩnh không được sử dụng phổ biến nữa.

* Trang web không kết nối cơ sở dữ liệu
* Được thiết kế bằng thẻ HTML và client script (kịch bản trên trình khách)
* Mọi ng sdụng nhận được kquả giống nhau.
* Trang web được viết = HTML, chỉ thđổi khi có sự thđổi của ng xdựng.
* Khả năng tương tác yếu
* Webserver hoạt động giống 1 file server.
* **Website động (dynamic website)**

Hầu hết các website hiện nay đều là website động. Đây là loại web dùng nền tảng HTML, CSS, Javascript, PHP hoặc ASP.NET. Quản trị viên của web khi muốn cập nhật thông tin hoăc thêm bớt module sẽ thao tác đơn giản, nhanh chóng hơn.

* Ứng dụng web bao hàm qtrình kết nối và truy cập cơ sở dữ liệu(điều này có nghĩa là khi bạn refresh lại trang web, -dữ liệu trên trag web được cập nhật bằng cách đọc từ cơ sở dữ liệu)
* Trang web động được thiết kế bằng server script (kịch bản trên trình chủ)
* Mỗi ng sdcó thể nhận được nd #nhau pthuộc vào kquả chạy ch trình.
* Trang web viết = HTML + Ngôn ngữ lập trình phía server. Có thể được thay đổi bởi ng sd
* Khả năng tương tác mạnh

### **Các thành phần của 1 trang web: header, footer, navigator,...**

Các thành phần cơ bản của một trang web thông thường bao gồm:

* Header:
  + Tiêu đề (Title): Thường là logo hoặc tên của trang web.
  + Menu điều hướng (Navigation Menu): Các liên kết đến các trang con hoặc các phần khác của trang web.
* Footer:
  + Thông tin liên hệ: Địa chỉ, số điện thoại, email, v.v.
  + Liên kết bổ sung: Các liên kết đến các trang con, điều khoản sử dụng, chính sách bảo mật, v.v.
  + Mạng xã hội: Các biểu tượng hoặc liên kết đến các trang mạng xã hội.
* Navigator/Menu:
  + Navigation Bar: Thường được đặt ở phía trên hoặc bên cạnh, cung cấp các liên kết chính để điều hướng trong trang web.
* Content Area:
  + Nội dung chính (Main Content): Phần chứa thông tin chính của trang web, như bài viết, sản phẩm, dịch vụ, v.v.
  + Sidebar: Phần bên cạnh nội dung chính thường chứa các liên kết phụ, danh sách, hoặc quảng cáo.
* Additional Components:
  + Form: Để người dùng nhập thông tin.
  + Search Bar: Cho phép tìm kiếm nhanh trên trang web.
  + Social Media Links: Các liên kết đến trang mạng xã hội của trang web.
  + Comments/Interactivity: Phần cho phép người dùng tương tác, bình luận.
* Mỗi thành phần này có thể được xây dựng bằng HTML và được trình bày bằng CSS để tạo ra giao diện hấp dẫn và dễ sử dụng cho người dùng. Sử dụng các phần tử này cùng với cấu trúc và thiết kế hợp lý sẽ tạo ra một trang web có cấu trúc rõ ràng và thu hút người dùng.

### **Các khái niệm khác: Sitemap, Homepage, giao diện người dùng, giao diện người quản trị.**

* **Sitemap**

Sitemap (sơ đồ website) là một file liệt kê các trang và tệp tin trên website. Danh sách liệt kê được sắp xếp theo dạng sơ đồ phân tầng (giảm dần sự quan trọng) giúp các công cụ tìm kiếm: Thu thập dữ liệu trên trang web của bạn hiệu quả hơnBiết những URL nào bạn muốn ưu tiên xuất hiệnHiển thị kết quả trên trang tìm kiếm thông minh hơn. A diagram of a website

Description automatically generated

Ví dụ về site map

*Các loại sitemap:*

* HTML Sitemap

HTML Sitemap là sơ đồ website xây dựng bằng mã HTML giúp cho người dùng dễ tiếp cận mục họ đang tìm hơn

*HTML Sitemap nên được đặt ở phần Footer để người dùng dễ tìm thấy nhất.*

* XML Sitemap

XML Sitemap được tạo nên với mục đích giúp bot của các công cụ tìm kiếm định hướng và thu nhập thông tin trên website dễ dàng, nhanh chóng hơn.

* Các loại Sitemap khác

Sitemap Index: Tập hợp các Sitemap được đính kèm và được dùng để đặt trong file robots.txt

Sitemap-category.xml: Tập hợp cấu trúc của các danh mục trên website.

Sitemap-products.xml: Sitemap dành cho các link chi tiết về các sản phẩm trên trang.

Sitemap-articles.xml: Sitemap dành cho các link chi tiết của từng bài viết trên website.

Sitemap-tags.xml: Sitemap dành cho các thẻ trên website.

Sitemap-video.xml: Sitemap dành riêng cho video trên các page, website.

Sitemap-image.xml: Sitemap dành cho các link về hình ảnh.

* **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
  + Lưu ý khi sử dụng đồ hoạ
* **Hệ thống liên kế**

- Link là một liên kết nối trang web này với một trang kia, các kết nối này có thể trên cùng một trang web (một domain, gọi là internal) hoặc liên kết với một tranh khác bên ngoài (khác domain, gọi là external).

- Hệ thống link là một mạng lưới gồm tất cả các internal và external link đan xem lại với nhau một cách logic nhằm mục đích nâng cao giá trị cung cấp thông tin cho người đọc cũng như là tương thích với các thuật toán của Google, nâng trang web lên những thứ hạng cao khi có người dùng tìm kiếm.

* External Links là gì?
  + Liên kết ngoài (External Link) được chia thành Inbound Link và Outbound Link.Trong đó Inbound Link các liên kết trỏ đến trang web của bạn từ các trang web khác (còn được gọi là backlink). Còn Outbound Link là các liên kết trỏ đến các trang web khác từ trang web của bạn.
* Internal Links là gì?
  + Internal Link (Liên kết nội bộ) là một liên kết từ trang này sang trang khác của cùng 1 tên miền (Domain). Tất nhiên, điều hướng trang web, menu website (Website navigation) của bạn là một ví dụ về liên kết nội bộ. Nhưng Internal Link thường chỉ các liên kết trong nội dung trên các trang.
* Backlinks là gì?
  + Backlink là những liên kết trả về từ các blog, diễn đàn, mạng xã hội, website khác tới website của bạn.
  + Backlink được xem là một trong những ưu tiên hàng đầu khi đánh giá xếp hạng của công cụ tìm kiếm. Một website có càng nhiều backlink chất lượng thì khả năng cạnh tranh thứ hạng càng cao.
* **Giao diện người dùng và giao diện người quản trí**

Giao diện người dùng cuối và giao diện người quản trị web (hay còn gọi là bảng điều khiển quản trị) là hai phần của một trang web hoặc ứng dụng có mục tiêu và chức năng khác nhau, được thiết kế để phục vụ các đối tượng người dùng khác nhau.

* Giao diện người dùng cuối:

***Mục đích:*** Được thiết kế để cung cấp trải nghiệm tốt nhất cho người dùng cuối khi họ truy cập và sử dụng trang web hoặc ứng dụng.

***Chức năng:*** Hiển thị nội dung chính, sản phẩm, dịch vụ, và tương tác với người dùng (như tạo tài khoản, mua hàng, đọc bài viết, v.v.).

***Đặc điểm:***

* Thiết kế thân thiện với người dùng: Dễ dàng sử dụng, thân thiện, tập trung vào trải nghiệm người dùng.
* Tối ưu hóa cho đa dạng thiết bị: Responsive, thích ứng với các kích thước màn hình khác nhau (desktop, mobile, tablet).
* Trực quan và hấp dẫn: Sử dụng hình ảnh, màu sắc, và giao diện thân thiện để thu hút người dùng.
* Giao diện người quản trị web (Bảng điều khiển quản trị):

***Mục đích:*** Dành cho người quản trị hoặc nhóm quản lý để quản lý và điều khiển nội dung, thông tin, và các chức năng của trang web hoặc ứng dụng.

***Chức năng:*** Quản lý người dùng, quản lý nội dung (thêm, sửa, xóa), theo dõi dữ liệu, cấu hình cài đặt, v.v.

***Đặc điểm:***

* Giao diện dễ quản lý: Cung cấp các công cụ và giao diện dễ sử dụng để quản lý nội dung và các chức năng.
* Bảo mật và quản lý tài khoản: Cung cấp các công cụ để quản lý quyền truy cập, tài khoản người dùng và các tính năng bảo mật.
* Tính linh hoạt và tùy chỉnh: Cho phép quản trị viên điều chỉnh cài đặt và quản lý nội dung theo nhu cầu cụ thể.

Mục tiêu của mỗi giao diện này là đáp ứng nhu cầu cụ thể của người dùng mà nó được thiết kế cho. Giao diện người dùng cuối tập trung vào trải nghiệm người dùng và tiếp cận nội dung, trong khi giao diện người quản trị web hỗ trợ quản lý và điều khiển các phần khác nhau của trang web hoặc ứng dụng.

### **Quan điểm khi thiết kế giao diện website**

***4.1. Thiết kế hướng người sử dụng:***

* Tập trung vào trải nghiệm người dùng: Đặt người dùng vào trung tâm thiết kế để đảm bảo giao diện dễ sử dụng và tối ưu hóa trải nghiệm của họ.
* Tính khả dụng và truy cập: Đảm bảo trang web dễ dàng tiếp cận và sử dụng cho người dùng với mọi trình độ kỹ thuật.

***4.2. Các giúp đỡ định hướng rõ ràng:***

* Hướng dẫn và điều hướng: Cung cấp các chỉ dẫn và liên kết rõ ràng để người dùng dễ dàng điều hướng trên trang web.

***4.3. Không có trang cuối (dead-end):***

* Liên kết hợp lý: Đảm bảo mọi trang đều có liên kết hoặc đường dẫn để người dùng có thể tiếp tục tìm kiếm thông tin.

***4.4. Băng thông và các yếu tố phần cứng:***

* Tối ưu hóa băng thông: Sử dụng hình ảnh tối giản, tối ưu hóa mã nguồn để giảm tải cho máy chủ và tối ưu hóa tốc độ tải trang.
* Tính tương thích với yếu tố phần cứng: Thiết kế sao cho trang web hoạt động tốt trên các thiết bị có cấu hình khác nhau.

***4.5. Đơn giản, nhất quán và tính ổn định:***

* Thiết kế đơn giản và nhất quán: Tránh quá nhiều chi tiết không cần thiết, duy trì một phong cách nhất quán trên toàn bộ trang web.
* Tính ổn định: Đảm bảo trang web hoạt động một cách ổn định trên mọi trình duyệt và thiết bị.

***4.6. Phản hồi đối thoại:***

* Tính tương tác: Cung cấp cơ hội cho người dùng tương tác thông qua các biểu mẫu, nút bấm, hoặc tính năng bình luận để họ có thể tương tác với trang web.

***4.7. Tính tương thích trên các trình duyệt:***

* Tối ưu hóa đa trình duyệt: Đảm bảo trang web hoạt động một cách tốt trên các trình duyệt phổ biến như Chrome, Firefox, Safari, và Edge.

***4.8. Tính năng Thiết kế cho quản trị và cập nhật nội dung:***

* Giao diện quản trị thân thiện: Cung cấp một bảng điều khiển dễ sử dụng cho người quản trị để có thể dễ dàng cập nhật nội dung, thêm mới thông tin và quản lý trang web.
* Những quan điểm này hỗ trợ việc thiết kế giao diện web để đáp ứng nhu cầu của người dùng và người quản trị, cung cấp trải nghiệm tốt nhất và tối ưu hóa hiệu suất của trang web.

***4.9 Tính tiện dụng trong thiết kế giao diện:***

* ***Nguyên tắc thiết kế tiện dụng:***
* Tổ chức website chặt chẽ và dễ sử dụng: Sắp xếp cấu trúc, đường dẫn và nội dung sao cho người dùng dễ dàng tìm kiếm và điều hướng.
* Sử dụng từ ngữ dễ hiểu: Tránh sử dụng ngôn ngữ phức tạp, sử dụng từ ngữ rõ ràng và dễ hiểu.
* Dễ dàng khám phá các đường link: Đặt liên kết một cách rõ ràng và dễ nhận biết để người dùng có thể điều hướng dễ dàng.
* Thời gian tải về nhanh: Tối ưu hóa website để giảm thời gian tải trang.
* Nội dung không có hình ảnh: Đảm bảo nội dung văn bản có ý nghĩa và dễ đọc, tránh sự cản trở của hình ảnh quá nhiều.
* Dễ theo dõi "quá trình bán hàng": Cung cấp thông tin rõ ràng và dễ hiểu về quá trình mua hàng hoặc các bước liên quan.
* Tương thích với đa số trình duyệt web: Đảm bảo website hoạt động mượt mà trên các trình duyệt phổ biến.
* ***Các vấn đề cần tránh:***
* Tránh các lỗi chính tả: Đảm bảo nội dung viết đúng chính tả để tạo ấn tượng chuyên nghiệp.
* Tránh những lời mời chào thái quá: Tạo lời mời chào hấp dẫn nhưng tránh sự quá mức và quá mạnh mẽ.
* Sử dụng những trạng thái thông báo: Cung cấp thông báo rõ ràng khi có các thao tác hoặc thông tin quan trọng.
* ***Một số sai lầm thường gặp khi thiết kế giao diện website***
* Kích cỡ website quá lớn: Tạo ảnh hưởng đến thời gian tải trang.
* Màu sắc và độ tương phản: Sử dụng màu sắc phù hợp và tạo độ tương phản để nội dung dễ đọc.
* Không biết mua hàng thế nào: Đảm bảo quá trình mua hàng hoặc hành động cần thực hiện rõ ràng và dễ thực hiện.
* Điều hướng kém, bố cục rối rắm: Tạo cấu trúc và bố cục hợp lý để người dùng dễ điều hướng và tìm kiếm thông tin.
* Không phân loại nội dung: Tổ chức nội dung thành các danh mục rõ ràng và dễ tiếp cận.
* Không tương tác với người truy cập: Cung cấp các tính năng tương tác để tạo sự hứng thú và tương tác với người dùng.
* Website có thiết kế quá mức cần thiết: Tránh thiết kế phức tạp, tập trung vào tối ưu hóa chức năng.
* Nhồi nhét quá nhiều thứ vào 1 website: Giữ giao diện đơn giản và tập trung vào thông tin quan trọng.

### **Nguyên tắc C.R.A.P**

Nguyên tắc C.R.A.P là một hệ thống các nguyên tắc thiết kế cơ bản để tạo ra giao diện hấp dẫn và dễ hiểu cho người dùng. Đây là một phương pháp giúp cải thiện cấu trúc, tổ chức và trình bày thông tin trên giao diện.

***5.1. Contrast (Sự Tương Phản):***

* Mục tiêu: Tạo sự khác biệt giữa các yếu tố trên trang web để nổi bật và dễ nhận biết.
* Cách áp dụng: Sử dụng màu sắc, kích thước, font chữ khác nhau để tạo ra sự tương phản giữa các yếu tố. Ví dụ, sử dụng màu sáng và tối, hoặc font chữ lớn và nhỏ để làm nổi bật tiêu đề, nút bấm, hoặc thông tin quan trọng.

***5.2. Repetition (Sự Lặp Lại):***

* Mục tiêu: Tạo sự nhất quán và giao diện dễ nhận biết bằng cách lặp lại các yếu tố thiết kế trên trang web.
* Cách áp dụng: Sử dụng các mẫu, các phong cách thiết kế, font chữ, hoặc màu sắc tương tự trên các phần khác nhau của trang web để tạo ra sự nhất quán.

***5.3. Alignment (Sự Sắp Hàng):***

* Mục tiêu: Tạo sự sắp xếp hợp lý và giao diện gọn gàng bằng cách sắp xếp các yếu tố theo một cách hợp lý.
* Cách áp dụng: Đảm bảo các yếu tố trên trang web được sắp xếp thẳng hàng, không lệch lạc. Sử dụng lưới hệ thống để sắp xếp các phần tử và tạo sự cân đối.

***5.4. Proximity (Khoảng Cách Gần):***

* Mục tiêu: Nhóm các yếu tố liên quan lại với nhau để tạo ra một hình ảnh tổng thể rõ ràng và dễ hiểu.
* Cách áp dụng: Đặt các yếu tố liên quan gần nhau để tạo ra một mối liên kết hợp lý. Ví dụ, đặt tiêu đề và nội dung tương ứng gần nhau để người dùng dễ dàng nhận ra mối quan hệ giữa chúng.

Khi sử dụng các nguyên tắc C.R.A.P trong thiết kế giao diện, bạn có thể tạo ra một trang web hấp dẫn, dễ hiểu và thú vị cho người dùng, giúp họ tương tác và tiếp cận thông tin một cách dễ dàng và thuận tiện.

# **Quá trình xây dựng website**

### **Tổng quan**

* Tiếp nhận yêu cầu nguồn thông tin từ khách hàng
* Lập kế hoạch
* Lựa chọn tên miền website và hosting
* Thiết kế giao diện
* Xây dựng tính năng website
* Kiểm tra và chỉnh sửa
* Đào tạo
* Bảo trì

### **Khái niệm cơ bản cho mỗi giai đoạn cụ thể**

***2.1. Tiếp nhận yêu cầu nguồn thông tin từ khách hàng:***

* Thu thập thông tin: Gặp gỡ khách hàng để hiểu rõ yêu cầu, mục tiêu, và nhu cầu của họ đối với website.

***2.2. Lập kế hoạch:***

* Phân tích yêu cầu: Xác định các yêu cầu cụ thể, chức năng, và mục tiêu của dự án.
* Xác định phạm vi và nguồn lực: Đề ra kế hoạch về thời gian, ngân sách, và nguồn lực cần thiết.

***2.3. Lựa chọn tên miền website và hosting:***

* Chọn tên miền: Xác định tên miền phù hợp với thương hiệu và mục tiêu của website.
* Chọn dịch vụ hosting: Lựa chọn nhà cung cấp hosting phù hợp với yêu cầu kỹ thuật và ngân sách.

***2.4. Thiết kế giao diện:***

* Tạo layout và wireframe: Xác định cấu trúc trang, vị trí của các phần tử và chức năng cơ bản.
* Thiết kế UI/UX: Tạo giao diện người dùng bằng công cụ thiết kế hoặc mã nguồn.

***2.5. Xây dựng tính năng website:***

* Phát triển mã nguồn: Xây dựng và lập trình các tính năng, chức năng, và cơ sở dữ liệu theo yêu cầu đã xác định.
* Tối ưu hóa độ phản hồi và hiệu suất: Đảm bảo website hoạt động mượt mà và nhanh chóng trên mọi thiết bị.

***2.6. Kiểm tra và chỉnh sửa:***

* Kiểm tra chất lượng: Kiểm tra tính năng, giao diện và sửa chữa các lỗi.
* Tối ưu hóa SEO: Cải thiện công cụ tìm kiếm và tối ưu hóa website để tăng thứ hạng trên các công cụ tìm kiếm.

***2.7. Đào tạo:***

* Đào tạo người quản lý nội dung: Hướng dẫn cách quản lý, cập nhật và thay đổi nội dung trên website.

***2.8. Bảo trì:***

* Hỗ trợ và bảo trì: Cung cấp hỗ trợ kỹ thuật, cập nhật bảo mật, và thực hiện các cải tiến sau khi website hoạt động.
* Mỗi giai đoạn đều quan trọng và cần được thực hiện cẩn thận để đảm bảo website hoạt động tốt và đáp ứng được yêu cầu của khách hàng và người dùng cuối.

### **Vai trò của các nhóm người (người thiết kế, người lập trình, khách hàng) trong quá trình này.**

Tại các giai đoạn khác nhau trong quá trình xây dựng website, mỗi nhóm người (người thiết kế, người lập trình và khách hàng) đóng vai trò quan trọng và có trách nhiệm cụ thể:

* ***Người Thiết Kế (Designer):***

Giai đoạn thiết kế giao diện (UI/UX):

* Tạo layout, wireframes và mockups dựa trên yêu cầu và thông tin từ khách hàng.
* Xác định trải nghiệm người dùng tốt nhất thông qua việc thiết kế giao diện dễ sử dụng và hấp dẫn.

Công việc:

* Lựa chọn màu sắc, font chữ, hình ảnh, và giao diện phù hợp.
* Xác định cấu trúc và tạo giao diện thân thiện, tương tác và hợp lý với người dùng.
* ***Người Lập Trình (Developer):***

Giai đoạn phát triển (Frontend/Backend):

* Xây dựng website theo các thiết kế được cung cấp bởi người thiết kế, biến các thiết kế thành trang web hoạt động.
* Xây dựng tính năng, chức năng, và tương tác trên website.

Công việc:

* Viết mã nguồn HTML, CSS, JavaScript (frontend) hoặc ngôn ngữ lập trình như PHP, Python, Ruby (backend).
* Kết nối cơ sở dữ liệu và xử lý dữ liệu để tạo ra trải nghiệm tương tác trên website.
* ***Khách Hàng (Client):***

Giai đoạn thu thập yêu cầu và phản hồi:

* Cung cấp thông tin yêu cầu, mục tiêu, và đánh giá để định hình dự án.
* Đưa ra phản hồi về thiết kế, chức năng, và tính năng của website.

Công việc:

* Xác định yêu cầu cụ thể của dự án, từ việc chọn màu sắc, font chữ, đến chức năng và giao diện.
* Đưa ra phản hồi chi tiết và cung cấp thông tin để điều chỉnh và cải thiện dự án.

Mỗi nhóm người có vai trò cụ thể trong việc biến ý tưởng thành hiện thực và đảm bảo rằng website đáp ứng được yêu cầu và mục tiêu đề ra từ đầu. Sự hợp tác hiệu quả giữa các nhóm này quyết định đến thành công của dự án xây dựng website.

# **Lý thuyết Javascript**

### **Kiến thức về ngôn ngữ lập trình thông dịch ở client: Javascript**

* JavaScript là gì?
  + - JavaScript (JS) là một ngôn ngữ lập trình thông dịch (được dịch lúc chạy), JavaScript ngày nay được ứng dụng rộng rãi trên nhiều môi trường khác nhau, nó tạo ra các ứng dụng chạy phía Server (xem như NodeJS), các ứng dụng di động (xem React) ... Javascript được sử dụng rất phổ biến để tích hợp vào các trang web, chạy ngay trong trình duyệt phía client giúp cho Website có được cách tương tác linh hoạt hơn với người dùng.
    - Nhiệm vụ của Javascript là xử lý những đối tượng HTML trên trình duyệt. Nó có thể can thiệp với các hành động như thêm / xóa / sửa các thuộc tính CSS và các thẻ HTML một cách dễ dàng. Hay nói cách khác, Javascript là một ngôn ngữ lập trình trên trình duyệt ở phía client. Tuy nhiên, hiện nay với sự xuất hiện của NodeJS đã giúp cho Javascript có thể làm việc ở backend.
* Javascript dùng để làm gì?
  + - Bên cạnh việc tìm hiểu javascript là ngôn ngữ gì thì chúng được sử dụng để làm gì cũng rất quan trọng. Việc nắm bắt được mục đích của ngôn ngữ đặc biệt này sẽ giúp bạn dễ dàng sử dụng chúng hơn trong công việc. Cụ thể như sau:
    - Thay đổi nội dung HTML: Một trong số nhiều phương thức HTML JavaScript chính là getElementById (). Chúng được sử dụng để tìm một phần tử của HTML với id =”demo” và dùng để thay đổi nội dung của phần từ (Internal HTML) sang thành “Hello JavaScript”
    - Thay đổi giá trị thuộc tính HTML: Tổng quan về javascript còn có thể sử dụng để thay đổi các giá trị của thuộc tính. Ví dụ: thay đổi thuộc tính src (source) của tag<img>.
    - Thay đổi kiểu HTML: Đây chính là một hoạt động biến thể của việc thay đổi thuộc tính của HTML ở trên. Ví dụ: document.getElementById(‘demo’).style.fontSize = ’35px;
    - Ẩn các phần tử HTML: Một hoạt động tiếp theo là Javascript có thể ẩn được các phần tử HTML. Chúng có thể được thực hiện thông qua hoạt động thay đổi kiểu hiển thị các phần tử HTML.
    - Hiển thị các phần tử HTML: Một điểm đặc biệt là JavaScript có thể hiển thị được các yếu tố HTML ẩn. Đồng thời, cũng có thể thực hiện được thông qua cách thay đổi kiểu hiển thị phần tử.
* Ưu điểm
  + Chương trình rất dễ học.
  + Những lỗi Javascript rất dễ để phát hiện, từ đó giúp bạn sửa lỗi một cách nhanh chóng hơn.
  + Những trình duyệt web có thể dịch thông qua HTML mà không cần sử dụng đến một compiler.
  + JS có thể hoạt động ở trên nhiều nền tảng và các trình duyệt web khác nhau.
  + Được các chuyên gia đánh giá là một loại ngôn ngữ lập trình nhẹ và nhanh hơn nhiều so với các ngôn ngữ lập trình khác.
  + JS còn có thể được gắn trên một số các element hoặc những events của các trang web.
  + Những website có sử dụng JS thì chúng sẽ giúp cho trang web đó có sự tương tác cũng như tăng thêm nhiều trải nghiệm mới cho người dùng.
  + Người dùng cũng có thể tận dụng JS với mục đích là để kiểm tra những input thay vì cách kiểm tra thủ công thông qua hoạt động truy xuất database.
  + Giao diện của ứng dụng phong phú với nhiều thành phần như Drag and Drop, Slider để cung cấp đến cho người dùng một Rich Interface (giao diện giàu tính năng).
  + Giúp thao tác với người dùng phía Client và tách biệt giữa các Client với nhau.
* Nhược điểm
  + JS Code Snippet khá lớn.
  + JS dễ bị các hacker và scammer khai thác hơn.
  + JS cũng không có khả năng đa luồng hoặc đa dạng xử lý.
  + Có thể được dùng để thực thi những mã độc ở trên máy tính của người sử dụng.
  + Những thiết bị khác nhau có thể sẽ thực hiện JS khác nhau, từ đó dẫn đến sự không đồng nhất.
  + Vì tính bảo mật và an toàn nên các Client-Side Javascript sẽ không cho phép đọc hoặc ghi các file.
  + JS không được hỗ trợ khi bạn sử dụng ở trong tình trạng thiết bị được kết nối mạng.

### **Kiến thức sử dụng Javascript: cú pháp, kiểu dữ liệu cơ bản, biến, hàm,... (Tự xem trong tài liệu thực hành)**

### **Kiến thức về sự kiện và xử lý sự kiện. mô hình Javascript + HTML DOM (Tự xem trong tài liệu thực hành)**

### **Kiến thức về biểu thức chính quy kiểm tra dữ liệu trên form và xử lý theo yêu cầu.**

* Định nghĩa:
  + Regex là một chuỗi ký tự đặc biệt được sử dụng để mô tả một mẫu chuỗi.
  + Nó giúp định rõ các quy tắc cho việc tìm kiếm, so khớp và xử lý chuỗi.
* Cú pháp cơ bản:
  + Các ký tự thông thường như chữ cái, số thường được so khớp đúng với chính chúng.
  + Các ký tự đặc biệt như . (đại diện cho bất kỳ ký tự nào), \* (đại diện cho 0 hoặc nhiều lần lặp), + (đại diện cho ít nhất một lần lặp) thường được sử dụng.
* Ký tự đặc biệt:
  + \d: So khớp với bất kỳ chữ số nào.
  + \w: So khớp với bất kỳ ký tự chữ cái, số hoặc gạch dưới.
  + \s: So khớp với bất kỳ ký tự khoảng trắng nào (ví dụ, dấu cách, tab).
* Nhóm và Ràng buộc:
  + ( ): Tạo một nhóm của các mẫu.
  + [ ]: Định nghĩa một tập hợp các ký tự có thể khớp.
  + { }: Xác định số lần lặp (ví dụ, \d{3} so khớp với ba chữ số).
* Bắt đầu và Kết thúc:
  + ^: Đánh dấu đầu chuỗi.
  + $: Đánh dấu cuối chuỗi.

VD:

