WEB BACK END

1. **Web, trang web, website, url**

Web: mạng lưới thông tin toàn cầu, nhiều trang web kết nối lại với nhau, đa dạng thông tin

Trang web: một trang tài liệu trên hệ thống web, chứa các thông tin

Website: tập hợp nhiều trang web, có một địa chỉ url

Ứng dụng web: một loại ứng dụng mà người dùng có thể truy cập và sử dụng thông qua Internet hoặc mạng nội bộ.

1. **CMS, Framework**

CMS: dễ dàng tạo, chỉnh sửa trên web, dễ quản lý, dễ dàng tùy chỉnh, tối ưu hóa seo, chi phí thấp. **Vấn đề về bảo mật, dễ bị tấn công, sẽ bị chậm nếu kh được tối ưu đúng cách, mã nguồn lớn, phụ thuộc hạ tầng, chức năng vào nhà cung cấp.**

Framework: tiết kiệm thời gian, tái sử dụng mã nguồn, tích hợp các tính năng, cấu trúc mã nguồn rõ rang, giảm thiểu lỗi, tăng cường bảo mật không dễ bị tấn công từ bên ngoài,dễ bảo trì. **Phải dành thời gian học, làm quen, hạn chế tùy chỉnh ở một số framework, phụ thuộc quy mô, môi trường, yêu cầu, chi phí cao**

1. **Web server**

Web server: máy tính(phần mềm) dung lưu trữ và cung cấp toàn bộ nội dung cho một website, ứng dụng web.

Quá trình xử lý: nhận request, phân tích, đọc dữ liệu, đổ dữ liệu vào HTML, gửi trang HTML về trình duyệt người dung, trình duyệt người dùng hiển thị nội dung response

1. **Nodejs** là một môi trường để chạy mã nguồn JavaScript đa nền tảng, mã nguồn mở, không phải là một framework, không phải là một ngôn ngữ lập trình mà là một nền tảng (platform), một môi trường (environment), một phần mềm gồm rất nhiều thành phần bên trong.

V8 để dịch và thực thi mã JavaScript

- HTTP module để tạo web server (hay còn gọi là HTTP server)

- Chương trình để quản lý gói và các thư viện có tên là NPM

- Các module, thư viện để lập trình ứng dụng web

**NPM** chính là công cụ để quản lý các thư viện mà chúng ta sử dụng khi làm ứng dụng web

**Kiểm tra đã cài đặt xem phiên bản của git:** cmd -> git -v

**Dự án và nhúng git**

Open terminal của thư mục đó (vd: thư mục Teo ở ổ D) -> git init ( khởi tạo)

Thành công sẽ xuất hiện: “Initialized empty Git repository in D:/Teo/.git/

*git config* : **Sử dụng để đọc, ghi hoặc cập nhật các thiết lập cấu hình của Git**

**Package.json**

*PS D:\TeoShop> npm init -y (tạo dự án mới với file package.json)*

**Cài express (pnpm)**

*D:\TeoShop>npm install -g pnpm@latest-10*

Kiểm tra lại: *D:\TeoShop>pnpm i -s express (i=install, s=save)*

*Tạo tập index.js khai báo để tạo web*

*PS D:\TeoShop> node index.js*

*server dang chay tren cong 9000*

**Tắt web server:** *ctrl C*

**Nodemon:** *npm install –g nodemon (tự khởi động lại web server khi cập nhật)*

**8. Hàm ẩn danh và các vòng lặp**

**function {}**

**Console.log()**

Hàm ẩn danh thông thường:

function() {

// thân hàm

}

Hàm ẩn danh kiểu mũi tên:

() => {

// thân hàm

Cú pháp hàm ẩn danh: (function() { ... })()

(function() {

    console.log("Hàm chạy ngay sau khi định nghĩa, không cần thao tác gọi hàm!");

})();

**9.*****Hàm mũi tên***

*- [1] Không tham số: () => { ... }*

*- [2] Một tham số: x => { ... } (không cần dấu ngoặc cho một tham số)*

**10. Hàm CallBack**

Hàm callback là một hàm được truyền vào một hàm khác như một tham số, và sẽ được thực thi sau khi hàm "cha" hoàn thành một tác vụ nào đó.

setTimeout(() => {

    console.log("Đã hết 5 giây! (sau 5 giây)");

}, 5000);

console.log("Đang chờ");

**11. Giao diện**

*Tạo tập tin* ***.gitignore*** *trong thư mục dự án*

***git status:*** Hiển thị trạng thái hiện tại của Kho lưu trữ và Thư mục làm việc

***git add .:*** Đưa tất cả các thay đổi từ Thư mục làm việc vào Khu tạm

***git commit -m "Initialize TeoShop project with Express and Nodemon":*** đưa các thay đổi vào kho chứa

***git log:*** xem lịch sử

**12. Cú pháp của một URL:**

*protocol://domain:port/path?query#fragment*

*protocol (giao thức): xác định phương thức giao tiếp mà trình duyệt sử dụng để giao tiếp với server.*

*domain (tên miền): tên của máy server.*

*port (cổng) (tùy chọn): số cổng mà server sử dụng để lắng nghe yêu cầu (request).*

*Path (đường dẫn): đường dẫn đến một vị trí (thư mục, tập tin, API) trên server.*

*query (truy vấn) (tùy chọn): chuỗi các tham số được sử dụng để truyền dữ liệu đến server.*

*fragment (phân mảnh) (tùy chọn): một phần cụ thể của tài nguyên, thường là một phần của trang HTML.*

**13. Gửi request tới web server**

*[1] nhận request*

*[2] phân tích request, gửi tới ứng dụng web để xử lý*

*[3] đọc dữ liệu từ cơ sở dữ liệu*

*[4] đổ dữ liệu vào trang HTML (các template HTML)*

*[5] gửi trang HTML (đã bao gồm dữ liệu) (response) về trình duyệt người dùng*

*[6] trình duyệt người dùng hiển thị nội dung response*

*[7] trình duyệt gửi riêng một request khác để lấy các nội dung tĩnh của ứng dụng web (HTML, CSS, JavaScript, hình ảnh, và các tài nguyên khác)*

**14. EventEmitter: bộ phận quản lý sự kiện**

*on(eventName, listener): khai báo tên sự kiện*

*emit(eventName) phát ra một sự kiện kèm theo tham số*

*Event: sự kiện*

*Event Loop: vòng lặp sự kiện, cơ chế cốt lõi của nodejs, kiểm tra hàng đợi sự kiện, thực thi các hàm callback*

*Event Handler: hàm sự kiện*