|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI  **KHOA CÔNG NGHỆ THÔNG TIN**    **BÁO CÁO THỰC TẬP TỐT NGHIỆP**   |  |  | | --- | --- | | Giảng viên hướng dẫn | : TS.Hoàng Văn Thông | | Sinh viên thực hiện | : Nguyễn Việt Anh | | Lớp | : CNTT1 | | Mã sinh viên | : 201200016 |   **Hà Nội – 2024** |

LỜI NÓI ĐẦU

**Chương 1. Tìm hiểu về công nghệ**

* 1. **Tìm hiểu về công nghệ thiết kế Web**
     1. **Ngôn ngữ HTML**
* Tổng quan:

HTML (Hypertext Markup Language) là ngôn ngữ đánh dấu sử dụng để tạo cấu trúc và hiển thị nội dung trên trang web. HTML là một phần quan trọng của các trang web và là ngôn ngữ cơ bản cho việc phát triển nội dung web.

* Ưu điểm:
  + Dễ đọc và sử dụng
  + Phổ biến và tiêu chuẩn
  + Dễ duyệt và hiểu
  + Hỗ trợ đa phương tiện
  + Tích hợp và nhiều ngôn ngữ khác: CSS, JS, PHP
* Nhược điểm:
  + Giới hạn trong việc mô tả giao diện
  + Không hỗ trợ lập trình nâng cao
  + Không thể tạo ứng dụng  động phức tạp
  + Phải cập nhật thường xuyên
    1. **Ngôn ngữ CSS**
* Tổng quan:

CSS(Cascading Style Sheets) là một ngôn ngữ đánh dấu sử dụng để định dạng và thiết kế giao diện của các trang web. CSS được sử dụng để điều khiển cách mà các phần tử HTML  hiển thị trên màn hình và cung cấp một cách để thực hiện định dạng, bố trí, màu sắc, kiểu chữ,…

* Ưu điểm:
  + Tách biệt cấu trúc và giao diện
  + Tích hợp dễ dàng
  + Hiệu suất
  + Khả năng kế thừa
  + Kiểm soát tùy chỉnh chi tiết
  + Khả năng đáp ứng(Responsive)
* Nhược điểm:
  + Phải quản lý mã CSS lớn
  + Khả năng xung đột
  + Một số chức năng khá phức tạp
    1. **Bootstrap**
* Tổng quan:

Bootstrap là một framework front-end phổ biến được sử dụng để phát triển trang web và ứng dụng web. Nó được tạo ra bởi Twitter và sau đó được phát triển và duy trì bởi cộng đồng mã nguồn mở. Bootstrap giúp lập trình viên thiết kế và xây dựng giao diện người dùng một cách nhanh chóng và dễ dàng bằng cách cung cấp một bộ phận các thành phần, lớp CSS và mã JS đã được định dạng trước.

* Ưu điểm:
  + Phát triển web nhanh
  + Đáp ứng( Responsive)
  + Tương thích trình duyệt
  + Cộng đồng lớn: nhiều tài liệu
* Nhược điểm:
  + Dung lượng lớn
  + Khó tùy chỉnh
  + Tích hợp phức tạp
  1. **Tìm hiểu về công nghệ lập trình Web**
     1. **Javascript**
* Tổng quan:

JavaScript (được viết tắt JS): là một ngôn ngữ lập trình thông dịch được sử dụng chủ yếu cho phía máy khách(client-side) trong lĩnh vực phát triển web. Nó được sử dụng để tạo ra các tính năng tương tác và động trên các trang web, giúp cải thiện trải nghiệm người dùng trực tuyến.

* Ưu điểm:
  + Phổ biến và năng động
  + Đa năng
  + Tính tương tác với người dùng
  + Cộng đồng lớn và hỗ trợ mạnh mẽ
  + Cú pháp dễ đọc và học
  + Tích hợp với HTML và CSS
  + Khả năng xử lý bất đồng bộ
* Nhược điểm:
  + Phụ thuộc vào trình duyệt
  + Bảo mật kém: nếu không kém soát và xử lý đầu vào
  + Độ phức tạp
  + Hiệu suất : do chạy trực tiếp trên trình duyệt vì vậy phụ thuộc vào tốc độ xử lý của trình duyệt
  + Khó khăn trong việc gỡ lỗi
    1. **Nodejs**
* Tổng quan:

NodeJS là một mã nguồn mở, được xây dựng trên “V8 Javascript engine” được viết bằng c++ và Javascript. Nền tảng này được phát triển bởi Ryan Dahl vào năm 2009.

Năm 2009, Ryan Dahl tạo NodeJS để xây dựng các server có khả năng mở rộng và ổn định cao. Ryan Dahl sử dụng Google’s V8 JavaScript engine( V8 là engine của Google Chrome).

Với sự phát triển của NodeJS, JavaScript trở thành một phẩn quan trọng trong các ngôn ngữ xây dựng ứng dụng ở phía Server.

Microsoft, Linkedin,… sử dụng Node trong các dự án.

Node.js sử dụng mô hình lập trình không đồng bộ (non-blocking), điều này có nghĩa là các hoạt động I/O như đọc/ghi file hoặc truy vấn cơ sở dữ liệu không chặn quy trình chính, cho phép server xử lý hàng nghìn kết nối cùng một lúc mà không cần sử dụng nhiều luồng (threads).

Node Package Manager (NPM) là một trong những công cụ quan trọng nhất trong việc phát triển ứng dụng Node.js. Nó cho phép bạn quản lý các gói (packages) của Node.js, tải về và cài đặt các thư viện, framework, và công cụ mà bạn cần cho dự án của mình

* Ứng dụng:
  + Tạo ứng dụng phía Server và ứng dụng đóng vai trò WebAPI.
  + Tạo ứng dụng Cơ sở dữ liệu
  + Tạo công cụ xây dựng và phát triển(Build tools)
* Mô hình MVC
  + MVC (Model-View-Controller) là một mô hình kiến trúc phổ biến được sử dụng trong phát triển phần mềm để tách biệt logic dữ liệu, giao diện người dùng và luồng điều khiển trong một ứng dụng. Khi áp dụng MVC trong Node.js, có thể sắp xếp mã của mình một cách có tổ chức và dễ bảo trì.
  + Model (Mô hình): Đây là nơi bạn đặt logic dữ liệu của ứng dụng. Model là trách nhiệm của việc tương tác với cơ sở dữ liệu hoặc bất kỳ nguồn dữ liệu nào khác. Trong Node.js, bạn thường sử dụng các thư viện như Mongoose cho MongoDB hoặc Sequelize cho SQL để định nghĩa các mô hình dữ liệu và thực hiện các thao tác CRUD (Create, Read, Update, Delete).
  + View (Giao diện người dùng): View là phần giao diện của ứng dụng, nơi mà dữ liệu được hiển thị cho người dùng. Trong Node.js, views thường được tạo bằng cách sử dụng các công cụ như HTML, CSS và JavaScript cùng với các định dạng mẫu như Pug, EJS hoặc Handlebars để tạo ra các trang HTML động.
  + Controller (Bộ điều khiển): Controller là nơi chứa logic điều khiển của ứng dụng. Nó nhận yêu cầu từ người dùng thông qua router, tương tác với các model để lấy hoặc cập nhật dữ liệu, sau đó chuyển dữ liệu này đến view để hiển thị cho người dùng. Trong Node.js, bạn có thể sử dụng các framework như Express.js để xác định các tuyến đường (routes) và xử lý logic điều khiển.
    1. **Express.js**
* Tổng quan:

Express.js là một framework web mạnh mẽ và phổ biến được xây dựng trên nền tảng Node.js. Nó cung cấp một cách tiếp cận linh hoạt để phát triển ứng dụng web và API trong Node.js.

* Routing

Express cung cấp một cách tiện lợi để xác định các tuyến đường (routes) trong ứng dụng của bạn. Bạn có thể xác định các tuyến đường cho các phương thức HTTP như GET, POST, PUT, DELETE, vv., và xử lý các yêu cầu từ người dùng dựa trên URL và phương thức.

* Middleware

Middleware là một tính năng quan trọng trong Express, cho phép bạn thêm các chức năng xử lý trung gian vào quy trình xử lý yêu cầu của ứng dụng. Điều này bao gồm các chức năng như xác thực, ghi nhật ký, xử lý lỗi, nén và nhiều hơn nữa. Middleware có thể được sử dụng cho toàn bộ ứng dụng hoặc chỉ định cho một tuyến đường cụ thể.

* Template Engines

Express không bao gồm một template engine như Pug (formerly Jade), EJS, Handlebars, vv., nhưng nó cho phép bạn tích hợp dễ dàng các template engine này vào ứng dụng của mình để tạo ra các trang HTML động.

* 1. **Sản phẩm đã thực hiện**

**KẾT LUẬN**

1. **Kết quả đạt được**
2. **Hướng phát triển của bản thân**