

## **ĐỒ ÁN LẬP TRÌNH MẠNG**

**Đề tài:**

# **Xây dựng Blog cá nhân chia sẻ kiến thức Lập trình mạng với Java & JavaScript sử dụng SSG Hugo và GitHub Pages**

**Ngành: Công Nghệ Thông Tin**

**Sinh viên thực hiện:**

**2280600643**

**Trần Ngọc Đạt**

*TP. Hồ Chí Minh, tháng 12 năm 2025*

## LỜI NÓI ĐẦU

Trong kỷ nguyên công nghệ số 4.0, Internet đã trở thành nền tảng không thể thiếu, kết nối hàng tỷ thiết bị và con người trên toàn cầu. Sự phát triển mạnh mẽ của các ứng dụng mạng (Network Applications) và các công nghệ lập trình web hiện đại đã thay đổi hoàn toàn cách thức chúng ta giao tiếp, làm việc và chia sẻ thông tin.

Trong bối cảnh đó, việc nắm vững kiến thức về Lập trình mạng, đặc biệt là sự kết hợp giữa ngôn ngữ **Java** (mạnh mẽ ở phía Backend) và **JavaScript** (linh hoạt ở phía Frontend) là vô cùng quan trọng đối với sinh viên Công nghệ thông tin. Tuy nhiên, việc học tập không chỉ dừng lại ở việc tiếp thu kiến thức mà còn cần quá trình lưu trữ, tổng hợp và chia sẻ lại.

Phương pháp học tập "Learning by Sharing" (Học thông qua chia sẻ) đã được chứng minh là một trong những cách hiệu quả nhất để khắc sâu kiến thức. Một Blog cá nhân không chỉ là nơi lưu trữ tài liệu học tập mà còn là hồ sơ năng lực (Portfolio) sống động, thể hiện tư duy kỹ thuật và khả năng thực hành của sinh viên.

Xuất phát từ nhu cầu thực tế đó, em lựa chọn thực hiện đề tài: **"Xây dựng Blog cá nhân chia sẻ kiến thức Lập trình mạng với Java & JavaScript sử dụng SSG Hugo và GitHub Pages"**. Báo cáo này sẽ trình bày chi tiết về quá trình xây dựng hệ thống website tĩnh (Static Site), đồng thời tổng hợp các kiến thức cốt lõi về lập trình mạng mà em đã tích lũy được trong quá trình học tập.

# Mục lục

LỜI NÓI ĐẦU .....	2
DANH MỤC HÌNH ẢNH .....	5
<b>1.1. Lý do chọn đề tài</b> .....	6
<b>1.1.1. Sự phổ biến của Internet và các ứng dụng mạng</b> .....	6
<b>1.2. Yêu cầu đề tài</b> .....	7
<b>1.3. Nội dung đề tài</b> .....	7
CHƯƠNG 2: TỔNG QUAN VỀ MẠNG MÁY TÍNH .....	8
2.1 Khái niệm cơ bản về mạng máy tính .....	8
2.1.1 Sự hình thành và phát triển của mạng máy tính .....	8
2.1.2 Thế nào là mạng máy tính? .....	8
2.1.3 Phân loại mạng máy tính .....	8
2.1.4 Kết nối mạng máy tính .....	9
2.2 Tổng quan về mạng cục bộ (LAN) .....	9
2.2.1 . Tại sao phải kết nối mạng .....	9
2.2.2 . Khái niệm mạng LAN .....	9
2.2.3 . Đặc trưng của mạng LAN .....	9
CHƯƠNG 3: KHẢO SÁT VÀ THIẾT KẾ HỆ THỐNG .....	11
3.1 Tổng quan về công cụ sử dụng .....	11
3.1.1. GitHub .....	11
3.2 Thiết kế Blog cá nhân .....	11
CHƯƠNG 4: XÂY DỰNG BLOG CHIA SẺ KIẾN THỨC .....	13
4.1. Xây dựng nội dung Blog .....	13
4.1.1. Định hướng và phân loại nội dung .....	13
4.1.2. Quy trình biên soạn và công cụ hỗ trợ .....	13

4.1.3. Các nội dung đã hoàn thiện.....	13
4.2. Kết quả giao diện.....	14
4.2.2. Giao diện chi tiết bài viết.....	15
4.2.3. Giao diện trang Profile .....	16
CHƯƠNG 5: KẾT LUẬN VÀ KIẾN NGHỊ .....	17
5.1. Kết Luận .....	17
5.2. Kiến Nghị và Hướng phát triển .....	17
TÀI LIỆU THAM KHẢO .....	18
PHỤ LỤC .....	19

## **DANH MỤC HÌNH ẢNH**

Hình.1:	Giao diện Trang chủ .....	14
Hình.2:	Giao diện chi tiết bài viết.....	15
Hình.3:	Giao diện trang Profile .....	16

# CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

## 1.1. Lý do chọn đề tài

### 1.1.1. Sự phổ biến của Internet và các ứng dụng mạng

*Sự phổ biến của Internet và các ứng dụng mạng* Internet hiện đại không chỉ là môi trường truyền tải thông tin mà còn là nền tảng tính toán khổng lồ. Nhu cầu nhân lực am hiểu về nguyên lý hoạt động của mạng máy tính, giao thức truyền thông và cách xây dựng ứng dụng mạng ngày càng tăng cao. Để đáp ứng yêu cầu này, việc tự học và tổng hợp kiến thức là bắt buộc.

### 1.1.2. Nhu cầu học tập và tổng hợp kiến thức

Lập trình mạng là một lĩnh vực rộng lớn, bao gồm nhiều mảng kiến thức từ lý thuyết mạng (OSI model, TCP/IP) đến các kỹ thuật lập trình cụ thể. Do đó, sinh viên cần tổng hợp kiến thức về:

**Lập trình mạng:** Hiểu về Socket, luồng dữ liệu, các tầng trong mô hình mạng.

**Java:** Ngôn ngữ mạnh mẽ, bảo mật, hỗ trợ tốt cho lập trình Socket và các hệ thống phân tán.

**JavaScript:** Ngôn ngữ kịch bản thiết yếu cho môi trường Web, giúp xử lý các yêu cầu HTTP (Fetch API, AJAX) một cách linh hoạt.

### 1.1.3. Ưu điểm của Blog cá nhân

Thay vì lưu trữ kiến thức trong các tệp tin Word hay ghi chú rời rạc, việc xây dựng Blog mang lại nhiều lợi ích:

**Để chia sẻ:** Kiến thức được công khai (public) giúp cộng đồng cùng tham khảo và góp ý.

**Để cập nhật:** Nội dung có thể được chỉnh sửa, bổ sung theo thời gian thực.

**Phù hợp học tập lâu dài:** Blog đóng vai trò như một "bộ nhớ ngoài", giúp tra cứu lại các đoạn code, các khái niệm đã học một cách nhanh chóng.

#### 1.1.4. Lý do chọn Hugo (SSG) và GitHub Pages

Thay vì sử dụng các CMS công kênh như WordPress hay tốn kém chi phí thuê Hosting/VPS, đề tài lựa chọn giải pháp **Static Site Generator (Hugo)** kết hợp với **GitHub Pages** vì: tốc độ tải trang cực nhanh, tính bảo mật cao (do không có cơ sở dữ liệu động), chi phí vận hành bằng 0, và quan trọng nhất là rèn luyện kỹ năng sử dụng Git và quản lý mã nguồn.

### 1.2. Yêu cầu đề tài

Để đảm bảo tính thực tiễn và chất lượng chuyên môn, đề tài đặt ra các yêu cầu cụ thể sau:

**Mục tiêu chính:** Xây dựng thành công một Blog cá nhân chia sẻ kiến thức lập trình mạng.

**Ngôn ngữ:** Nội dung bài viết được trình bày bằng tiếng Việt.

**Cấu trúc Website:** Phải có tối thiểu các trang chức năng:

Trang Chủ (Home): Tổng quan và bài viết mới.

Trang Blog: Danh sách bài viết lưu trữ.

Trang Profile cá nhân: Giới thiệu tác giả.

**Số lượng bài viết:** Có ít nhất **09 bài viết** chất lượng cao liên quan đến chủ đề Java và JavaScript.

**Giao diện:** Thiết kế theo phong cách tối giản (Minimalism), dễ đọc, tập trung vào trải nghiệm người dùng.

**Công nghệ:** Sử dụng GitHub Repository để quản lý mã nguồn và Hugo để sinh trang tĩnh.

### 1.3. Nội dung đề tài

Đề tài được chia thành các giai đoạn nghiên cứu và thực hiện chính:

1. Tìm hiểu cơ sở lý thuyết về mạng máy tính, các mô hình kết nối và giao thức.
2. Nghiên cứu vai trò và kỹ thuật của Java & JavaScript trong lập trình mạng.
3. Thiết kế và xây dựng Blog cá nhân: Cài đặt môi trường, tùy biến giao diện.
4. Đưa Blog lên Internet thông qua dịch vụ GitHub Pages và kiểm thử hoạt động.

## CHƯƠNG 2: TỔNG QUAN VỀ MẠNG MÁY TÍNH

### 2.1 Khái niệm cơ bản về mạng máy tính

#### 2.1.1 Sự hình thành và phát triển của mạng máy tính

Lịch sử mạng máy tính bắt đầu từ cuối thập niên 1960 với sự ra đời của mạng **ARPANET** do Bộ Quốc phòng Mỹ tài trợ. Đây là mạng đầu tiên sử dụng kỹ thuật chuyên mạch gói (packet switching), tiền thân của Internet ngày nay. Trải qua nhiều giai đoạn phát triển, **Internet hiện đại** đã trở thành một hệ thống thông tin toàn cầu, kết nối hàng tỷ thiết bị dựa trên bộ giao thức chuẩn TCP/IP. Trong bối cảnh đó, **Web và lập trình mạng** đóng vai trò then chốt. World Wide Web (WWW) là dịch vụ chạy trên nền Internet, sử dụng giao thức HTTP để truyền tải siêu văn bản, cho phép người dùng truy cập thông tin dễ dàng.

#### 2.1.2 Thế nào là mạng máy tính?

- **Định nghĩa:** Mạng máy tính là tập hợp các máy tính hoặc thiết bị độc lập được kết nối với nhau thông qua các đường truyền vật lý và tuân theo các quy tắc giao tiếp (giao thức) để chia sẻ tài nguyên.
- **Thành phần cơ bản của mạng:** Bao gồm thiết bị đầu cuối (máy tính, điện thoại), thiết bị mạng (Router, Switch), môi trường truyền dẫn (cáp, sóng) và giao thức mạng (Protocol).

#### 2.1.3 Phân loại mạng máy tính

Dựa trên phạm vi địa lý, mạng máy tính được phân loại thành:

- **LAN (Local Area Network):** Mạng cục bộ, phạm vi hẹp (văn phòng, tòa nhà).
- **WAN (Wide Area Network):** Mạng diện rộng, kết nối xuyên quốc gia/châu lục.
- **MAN (Metropolitan Area Network):** Mạng đô thị, quy mô thành phố.
- **Internet:** Mạng của các mạng, phạm vi toàn cầu.



## 2.1.4 Kết nối mạng máy tính

Có hai phương thức kết nối chính:

- **Có dây (Wired):** Sử dụng cáp xoắn đôi, cáp đồng trục, cáp quang. Ưu điểm là ổn định, tốc độ cao.
- **Không dây (Wireless):** Sử dụng sóng vô tuyến (Wi-Fi), vi sóng, hồng ngoại. Ưu điểm là linh hoạt, thẩm mỹ.

## 2.2 Tổng quan về mạng cục bộ (LAN)

### 2.2.1. Tại sao phải kết nối mạng

Mạng LAN ra đời nhằm giải quyết các nhu cầu thiết yếu:

- **Chia sẻ tài nguyên:** Dùng chung máy in, máy scan, ổ cứng lưu trữ.
- **Truyền thông tin:** Gửi nhận tệp tin, email, chat nội bộ.
- **Ứng dụng thực tế:** Quản lý dữ liệu tập trung, triển khai các phần mềm quản lý doanh nghiệp (ERP, CRM).

### 2.2.2. Khái niệm mạng LAN

Mạng LAN (Local Area Network) là hệ thống mạng dùng để kết nối các máy tính trong một phạm vi nhỏ (thường dưới 1km). Các máy tính trong mạng LAN có thể chia sẻ tài nguyên và trao đổi dữ liệu với tốc độ cao.

### 2.2.3. Đặc trưng của mạng LAN

Mạng LAN có các đặc trưng kỹ thuật chính: Tốc độ truyền dữ liệu cao (từ 100 Mbps đến 10 Gbps), phạm vi địa lý giới hạn, tỷ lệ lỗi bit thấp và thường thuộc quyền sở hữu quản lý của một tổ chức đơn lẻ.

### 2.2.4. Ưu và nhược điểm của mạng LAN

- **Ưu điểm:** Chi phí triển khai thấp, quản trị đơn giản, tốc độ cao, bảo mật nội bộ tốt.
- **Nhược điểm:** Phạm vi kết nối hạn chế, số lượng máy trạm bị giới hạn bởi thiết bị phần cứng.

### 2.2.5. Các kiểu mạng LAN (Topology)

Hình thái mạng (Topology) mô tả cách bố trí hình học của các thiết bị. Các kiểu phổ biến bao gồm:

- Mạng hình sao (Star).
- Mạng tuyến tính (Bus).
- Mạng vòng (Ring).
- Mạng lưới (Mesh).

### 2.2.6 Hệ thống cáp mạng dùng cho LAN

Hệ thống cáp đóng vai trò là môi trường truyền dẫn tín hiệu:

- **Cáp xoắn đôi (Twisted Pair):** Phổ biến nhất (Cat5e, Cat6), giá rẻ, dễ lắp đặt.
- **Cáp quang (Fiber Optic):** Dùng cho đường trục (Backbone) hoặc kết nối xa, tốc độ cực cao, không bị nhiễu điện từ.
- **Cáp đồng trục (Coaxial):** Ít được sử dụng trong LAN hiện đại.

### 2.2.7 Đường truyền không dây

Mạng LAN không dây (WLAN) sử dụng sóng vô tuyến để kết nối theo chuẩn IEEE 802.11 (Wi-Fi). Các băng tần phổ biến là 2.4GHz và 5GHz.

### 2.2.8 Các thiết bị dùng trong mạng LAN

- **NIC (Network Interface Card):** Card mạng giúp máy tính giao tiếp với môi trường mạng.
- **Switch (Bộ chuyển mạch):** Thiết bị trung tâm trong mạng hình sao, chuyển tiếp dữ liệu đến đúng đích.
- **Router (Bộ định tuyến):** Kết nối LAN với Internet hoặc các mạng WAN khác.
- **Access Point:** Điểm truy cập phát sóng Wi-Fi.

## CHƯƠNG 3: KHẢO SÁT VÀ THIẾT KẾ HỆ THỐNG

### 3.1 Tổng quan về công cụ sử dụng

#### 3.1.1. GitHub

Giới thiệu GitHub: GitHub là nền tảng lưu trữ mã nguồn dựa trên hệ thống quản lý phiên bản Git lớn nhất thế giới. Nó cung cấp môi trường cộng tác cho các lập trình viên.

Vai trò trong lưu trữ mã nguồn: Trong đề tài này, GitHub đóng vai trò là "kho chứa" (Repository) lưu trữ toàn bộ mã nguồn, bài viết (Markdown) và tài nguyên hình ảnh của Blog.

GitHub Pages: Là dịch vụ hosting miễn phí của GitHub, cho phép xuất bản các website tĩnh trực tiếp từ repository. Đây là nơi Blog sẽ được "live" trên Internet.

#### 3.1.2. Static Site Generator – Hugo

Khái niệm SSG: Trình tạo trang tĩnh (Static Site Generator) là công cụ biên dịch nội dung (thường là Markdown) và khuôn mẫu (Templates) thành các file HTML, CSS, JS thuần túy.

Ưu điểm của Hugo: Hugo được viết bằng ngôn ngữ Go (Golang), nổi tiếng với tốc độ "build" trang cực nhanh (mili-giây), cộng đồng hỗ trợ lớn và kho giao diện (Themes) phong phú.

So sánh Hugo với CMS truyền thống: Khác với WordPress (cần Database, PHP), Hugo không cần cơ sở dữ liệu động, do đó an toàn hơn trước các cuộc tấn công mạng và không tốn tài nguyên server xử lý mỗi khi có người truy cập.

### 3.2 Thiết kế Blog cá nhân

**3.2.1. Cấu trúc website:** Hệ thống được thiết kế với cây sitemap đơn giản:

- **Home:** Hiện thị lời chào, danh sách các bài viết mới nhất và liên kết mạng xã hội.
- **Blog:** Danh sách toàn bộ bài viết, phân loại theo Tags (Thẻ) hoặc Categories (Danh mục).
- **Profile:** Trang tĩnh giới thiệu về tác giả, kỹ năng và dự án.

### **3.2.2. Cấu trúc thư mục Hugo:** Một dự án Hugo chuẩn bao gồm các thư mục chính:

- ☐ content/: Chứa các file .md (nội dung bài viết).
- ☐ layouts/: Chứa các file HTML template quy định bố cục trang.
- ☐ static/: Chứa hình ảnh, file CSS, JS tùy chỉnh.
- ☐ themes/: Chứa giao diện đã tải về (ví dụ: PaperMod, Ananke).
- ☐ config.toml: File cấu hình chính của toàn bộ website.

### **3.2.3. Thiết kế giao diện**

- Tiêu chí: Thiết kế theo hướng "Content-first" (Nội dung là trên hết), tối giản, sử dụng font chữ không chân (Sans-serif) để dễ đọc.
- Responsive: Giao diện tự động co giãn, tương thích tốt với mọi kích thước màn hình từ Desktop đến Mobile.

## CHƯƠNG 4: XÂY DỰNG BLOG CHIA SẺ KIẾN THỨC

### 4.1. Xây dựng nội dung Blog

#### 4.1.1. Định hướng và phân loại nội dung

Nội dung của Blog được xây dựng dựa trên phương pháp "Learning by Sharing" (Học thông qua chia sẻ). Đối tượng độc giả hướng đến là các sinh viên chuyên ngành Công nghệ thông tin và những người mới bắt đầu tìm hiểu về lập trình Web/Backend. Hệ thống bài viết được phân loại thành hai chuyên mục (Categories) chính để người đọc dễ dàng tiếp cận:

- **Chuyên mục JavaScript & Web:** Tập trung vào các kiến thức nền tảng của ngôn ngữ JavaScript, các kỹ thuật xử lý bất đồng bộ và thao tác với dữ liệu trên trình duyệt.
- **Chuyên mục Java & Lập trình mạng:** Tập trung vào kiến thức về giao thức mạng, Socket và cách xây dựng ứng dụng mạng với ngôn ngữ Java.

#### 4.1.2. Quy trình biên soạn và công cụ hỗ trợ

Quy trình để tạo ra một bài viết hoàn chỉnh trên Blog Hugo được thực hiện qua các bước:

1. **Nghiên cứu từ khóa và tài liệu:** Tham khảo các tài liệu chính thống (như MDN Web Docs, Oracle Docs) và giáo trình của Cisco Networking Academy để đảm bảo tính chính xác của kiến thức.
2. **Soạn thảo bằng Markdown:** Các bài viết được viết dưới định dạng .md (Markdown). Đây là ngôn ngữ đánh dấu nhẹ, giúp trình bày tiêu đề, danh sách, in đậm/ngheêng một cách nhanh chóng mà không cần mã HTML phức tạp.
3. **Chèn mã nguồn (Code Snippets):** Các đoạn code minh họa được đặt trong khối lệnh (Code block) với cú pháp ````java` hoặc ````javascript`. Việc này giúp Hugo tự động nhận diện và tô màu cú pháp (Syntax Highlighting), giúp người đọc dễ dàng phân biệt từ khóa, biến và chuỗi.
4. **Cấu hình Metadata (Front Matter):** Đầu mỗi file bài viết đều được khai báo các thông tin như: Tiêu đề (title), Ngày đăng (date), Tác giả (author), và Thẻ (tags) để phục vụ cho việc tìm kiếm và sắp xếp.

#### 4.1.3. Các nội dung đã hoàn thiện

Trong khuôn khổ của đồ án, tác giả đã hoàn thành biên soạn và đăng tải chuỗi bài viết chất lượng cao, bao gồm:

#### Nhóm bài viết về Kỹ thuật Lập trình Web & JavaScript:

- *Biến và Kiểu dữ liệu trong JavaScript:* Phân biệt var, let, const và các kiểu dữ liệu nguyên thủy.

- *JavaScript là gì?:* Tổng quan về vai trò của JS trong mô hình Client-Server.
- *Async/Await trong JavaScript:* Hướng dẫn xử lý bất đồng bộ hiện đại, giúp code gọn gàng và dễ bảo trì hơn so với Callback hell.
- *Fetch API trong JavaScript:* Kỹ thuật gửi yêu cầu HTTP (GET, POST) đến Server để lấy dữ liệu mà không cần tải lại trang.
- *Gọi API thực tế bằng JavaScript:* Bài hướng dẫn thực hành lấy dữ liệu từ một Public API và hiển thị lên giao diện.
- *LocalStorage trong JavaScript:* Kỹ thuật lưu trữ dữ liệu cục bộ trên trình duyệt người dùng, ứng dụng cho chức năng lưu trạng thái đăng nhập hoặc giao diện Sáng/Tối.

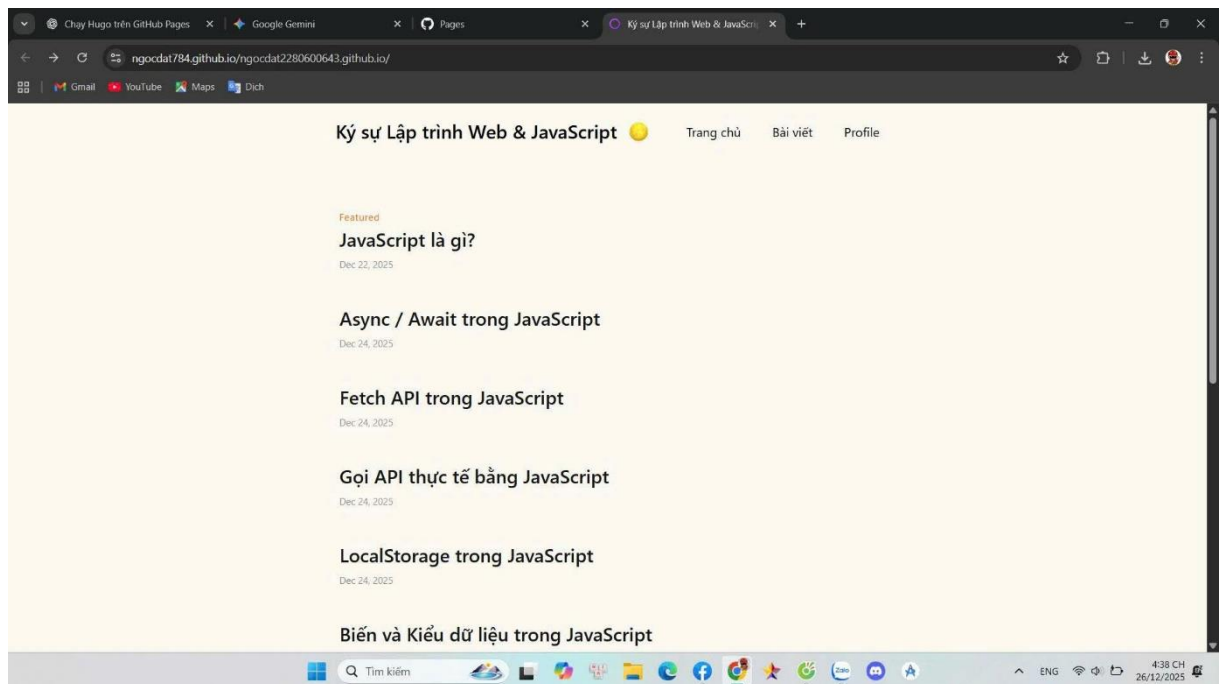
## Nhóm bài viết về Java & Mạng máy tính:

- *Tổng quan về Lập trình mạng với Java:* Giới thiệu gói thư viện `java.net`.
- *Mô hình OSI và TCP/IP:* Kiến thức lý thuyết nền tảng về cách dữ liệu di chuyển trên mạng.

### 4.2. Kết quả giao diện

Sau quá trình triển khai mã nguồn Hugo lên GitHub Pages, hệ thống hoạt động ổn định với giao diện người dùng (UI) hoàn chỉnh, đảm bảo tính thẩm mỹ và trải nghiệm người dùng (UX) tối giản. Dưới đây là hình ảnh thực tế của website:

#### 4.2.1. Giao diện Trang chủ (Homepage)

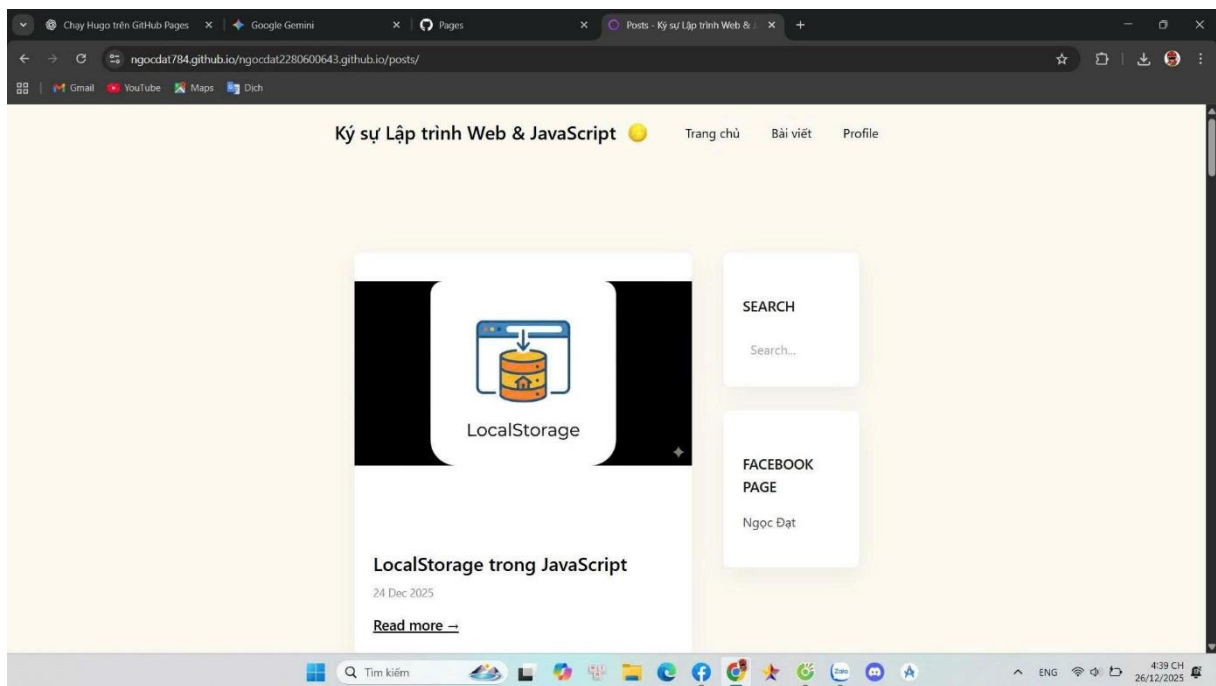


Hình.1: Giao diện Trang chủ

Trang chủ là trung tâm điều hướng của toàn bộ Blog, được thiết kế với tông màu nền kem nhạt dịu mắt, tạo cảm giác dễ chịu cho người đọc.

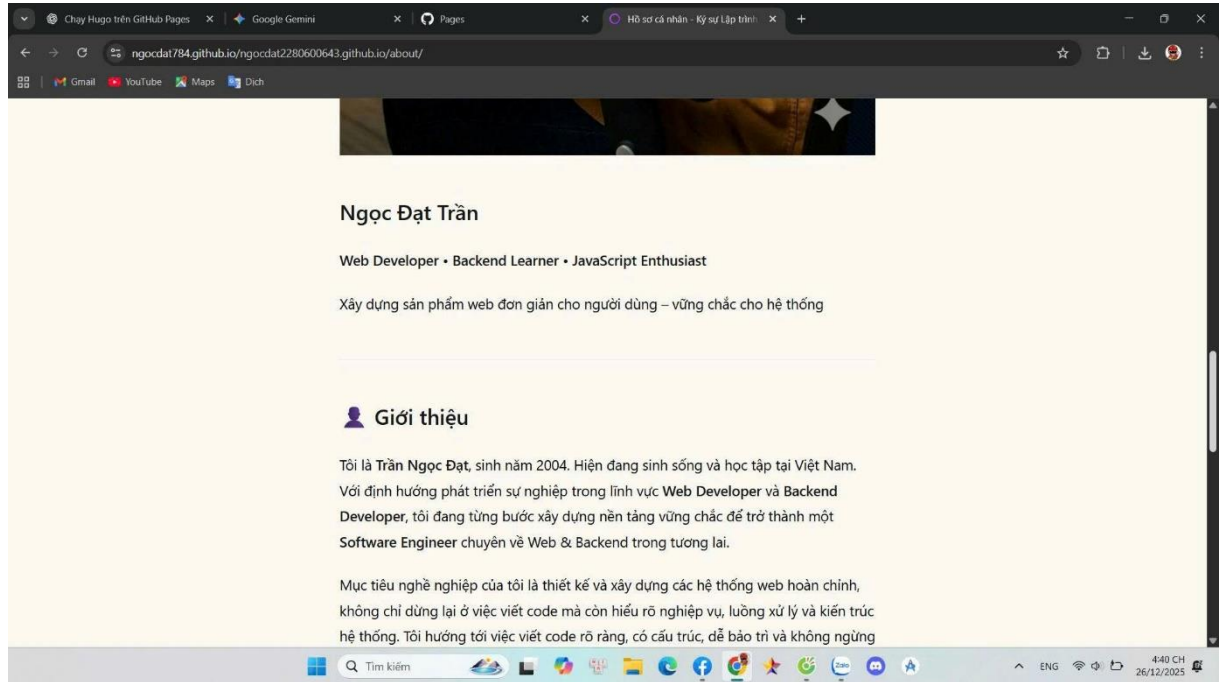
- **Header (Đầu trang):** Nổi bật với tiêu đề Blog "**Kỹ sư Lập trình Web & JavaScript**" cùng biểu tượng nhận diện thương hiệu. Thanh điều hướng (Navigation bar) bên phải giúp truy cập nhanh vào các mục: *Trang chủ*, *Bài viết*, *Profile*.
- **Danh sách bài viết (Post List):** Các bài viết mới nhất được liệt kê theo trình tự thời gian đảo ngược (mới nhất lên đầu). Trên giao diện hiển thị các bài viết nổi bật như: *JavaScript là gì?*, *Async/Await trong JavaScript*, *Fetch API*, và *LocalStorage trong JavaScript*. Mỗi bài viết đi kèm ngày đăng (ví dụ: Dec 24, 2025), giúp độc giả nhanh chóng nắm bắt các chủ đề mới nhất.
- **Thiết kế:** Sử dụng phong cách tối giản (Minimalism), loại bỏ các thành phần rườm rà để người dùng tập trung hoàn toàn vào tiêu đề bài viết.

#### 4.2.2. Giao diện chi tiết bài viết



Hình.2: Giao diện chi tiết bài viết

### 4.2.3. Giao diện trang Profile



Hình.3: Giao diện trang Profile

Trang Profile (Hồ sơ) được thiết kế như một Portfolio online chuyên nghiệp để giới thiệu về tác giả **Trần Ngọc Đạt**.

- **Thông tin định danh:** Hiện thị ảnh đại diện cá nhân và Tên tác giả "**Ngọc Đạt Trần**" được in đậm nổi bật ở trung tâm.
- **Tagline:** Dòng mô tả định hướng nghề nghiệp: "*Web Developer • Backend Learner • JavaScript Enthusiast*" thể hiện rõ đam mê và lĩnh vực chuyên môn đang theo đuổi.
- **Giới thiệu chi tiết:** Phần nội dung văn bản giới thiệu về năm sinh (2004), quá trình học tập tại Việt Nam và mục tiêu nghề nghiệp dài hạn là trở thành một *Software Engineer* chuyên về Web & Backend. Tác giả cũng nhấn mạnh triết lý làm việc: "*Xây dựng sản phẩm web đơn giản cho người dùng – vững chắc cho hệ thống*".



## CHƯƠNG 5: KẾT LUẬN VÀ KIẾN NGHỊ

### 5.1. Kết Luận

Sau thời gian nghiên cứu và thực hiện đề tài "**Xây dựng Blog cá nhân chia sẻ kiến thức Lập trình mạng với Java & JavaScript sử dụng SSG Hugo và GitHub Pages**", em đã hoàn thành các mục tiêu đề ra và đạt được những kết quả cụ thể sau:

- **Về sản phẩm ứng dụng:** Đã xây dựng và triển khai thành công website Blog cá nhân lên môi trường Internet thông qua nền tảng GitHub Pages. Website hoạt động ổn định, tốc độ truy cập nhanh, giao diện tối giản và thân thiện với người dùng trên cả máy tính và thiết bị di động. Hệ thống quản lý mã nguồn và bài viết được tổ chức khoa học trên GitHub Repository.
- **Về kiến thức chuyên môn:**
  - **Mạng máy tính:** Củng cố vững chắc kiến thức nền tảng về mạng máy tính, bao gồm mô hình OSI, bộ giao thức TCP/IP, phương thức hoạt động của HTTP và cách thức Client-Server giao tiếp.
  - **Lập trình:** Nâng cao kỹ năng lập trình Java (xử lý đa luồng, socket) và JavaScript (xử lý bất đồng bộ, Fetch API, DOM) thông qua quá trình biên soạn các bài viết hướng dẫn chuyên sâu.
  - **Công cụ phát triển:** Thành thạo quy trình làm việc với Git (các lệnh commit, push, branch), ngôn ngữ đánh dấu Markdown và hiểu rõ cơ chế biên dịch trang tĩnh của Hugo.
- **Về kỹ năng mềm:** Hình thành tư duy "Học thông qua chia sẻ" (Learning by Sharing), rèn luyện kỹ năng viết tài liệu kỹ thuật và khả năng tự nghiên cứu, giải quyết vấn đề độc lập.

### 5.2. Kiến Nghị và Hướng phát triển

Mặc dù Blog đã đáp ứng được các yêu cầu cơ bản của một trang chia sẻ kiến thức, nhưng để hệ thống trở nên hoàn thiện và chuyên nghiệp hơn, em xin đề xuất các hướng phát triển trong tương lai như sau:

- **Tích hợp hệ thống bình luận (Comment System):** Hiện tại Blog là website tĩnh nên chưa hỗ trợ tương tác trực tiếp. Em dự kiến sẽ tích hợp các dịch vụ bình luận bên thứ ba như **Disqus** hoặc **Utterances** (dựa trên GitHub Issues) để người đọc có thể thảo luận và góp ý ngay dưới mỗi bài viết.
- **Mở rộng nội dung chuyên sâu:** Bên cạnh các kiến thức cơ bản, Blog sẽ bổ sung thêm các bài viết nâng cao về **Bảo mật mạng (Network Security)** và xây dựng ứng dụng Backend phức tạp với **Java Spring Boot**, nhằm đa dạng hóa nội dung và phục vụ đối tượng độc giả có trình độ cao hơn.
- **Tối ưu hóa công cụ tìm kiếm (SEO):** Cải thiện cấu trúc Metadata, thẻ mô tả (Description) và Open Graph để các bài viết dễ dàng tiếp cận người đọc thông qua Google Search.

- **Nâng cấp giao diện người dùng:** Tùy biến sâu hơn về CSS/JS để tích hợp thêm các tính năng như: thanh tiến trình đọc bài (Reading Progress Bar), chế độ xem ảnh Lightbox và chức năng tìm kiếm nâng cao (Advanced Search).

Việc tiếp tục phát triển Blog sẽ không chỉ giúp duy trì thói quen học tập của bản thân mà còn đóng góp giá trị hữu ích cho cộng đồng lập trình viên.

## TÀI LIỆU THAM KHẢO

- [1] **GitHub.** (2024). *GitHub Pages Documentation*. Truy cập tại: <https://docs.github.com/en/pages>
- [2] **GoHugo.** (2024). *Hugo Documentation*. Truy cập tại: <https://gohugo.io/documentation/>
- [3] **Kurose, J.F. & Ross, K.W.** (2017). *Computer Networking: A Top-Down Approach (7th Edition)*. Pearson Education.
- [4] **Mozilla Developer Network (MDN).** (2024). *JavaScript Reference & Guide*. Truy cập tại: <https://developer.mozilla.org/en-US/docs/Web/JavaScript>
- [5] **Oracle.** (2023). *Java Platform, Standard Edition Networking Guide*. Truy cập tại: <https://docs.oracle.com/en/java/javase/>
- [6] **Tanenbaum, A.S. & Wetherall, D.J.** (2011). *Computer Networks (5th Edition)*. Prentice Hall.

## PHỤ LỤC

### Phụ lục 1: Chứng chỉ Networking Basics



*Hình.1. Chứng chỉ hoàn thành khóa học Networking Basics.*

**Phụ lục 2: Chứng chỉ JavaScript Essentials 1.**



*Hình.2. Chứng chỉ hoàn thành khóa học JavaScript Essentials 1.*

**Phụ lục 3: Chứng chỉ JavaScript Essentials 2.**



*Hình.3. Chứng chỉ hoàn thành khóa học JavaScript Essentials 2.*

