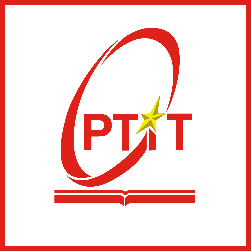
**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG CƠ**

**SỞ HỒ CHÍ MINH**



**BÁO CÁO CUỐI KÌ MÔN CHUYÊN ĐỀ CÔNG NGHỆ PHẦN MỀM**

***Đề tài: Triển khai Website CMS và E-Commerce với CI/CD và Reverse Proxy***

**Giảng viên hướng dẫn: Lê Hà Thanh**

**Sinh viên thực hiện:**

**Chung Khánh Toàn – N20DCCN139**

# Mục Lục

Mục lục

[Mục Lục 2](#_Toc169833004)

[**CHƯƠNG I: GIỚI THIỆU ĐỀ TÀI** 3](#_Toc169833005)

[**1.** **Mô tả** 3](#_Toc169833006)

[**2.** **Mục tiêu của đề tài** 3](#_Toc169833007)

[**CHƯƠNG II: CÔNG NGHỆ SỬ DỤNG** 5](#_Toc169833008)

[**1.** **WordPress** 5](#_Toc169833009)

[**2.** **Nginx** 6](#_Toc169833010)

[**3.** **MySQL Workbench** 7](#_Toc169833011)

[**4.** **WooCommerce** 8](#_Toc169833012)

[**5.** **GitHub** 9](#_Toc169833013)

[**6.** **GitHub Actions** 10](#_Toc169833014)

[**7.** **InfinityFree** 12](#_Toc169833015)

[**CHƯƠNG III: THÔNG SỐ CẤU HÌNH** 13](#_Toc169833016)

[**1.** **Server** 13](#_Toc169833017)

[**2.** **Reverser Proxy** 15](#_Toc169833018)

[**3.** **Ci/Cd** 18](#_Toc169833019)

[**CHƯƠNG IV: PHỤ LỤC** 22](#_Toc169833020)

[**1.** **Mã nguồn** 22](#_Toc169833021)

[**2.** **Tài liệu tham khảo** 22](#_Toc169833022)

# **CHƯƠNG I: GIỚI THIỆU ĐỀ TÀI**

1. **Mô tả**

Đề tài này nhằm nghiên cứu và triển khai một website tích hợp hai thành phần quan trọng là hệ thống quản lý nội dung (CMS) và nền tảng thương mại điện tử (E-Commerce) bằng cách sử dụng các công nghệ hiện đại như Continuous Integration/Continuous Deployment (CI/CD) và cơ chế Reverse Proxy.

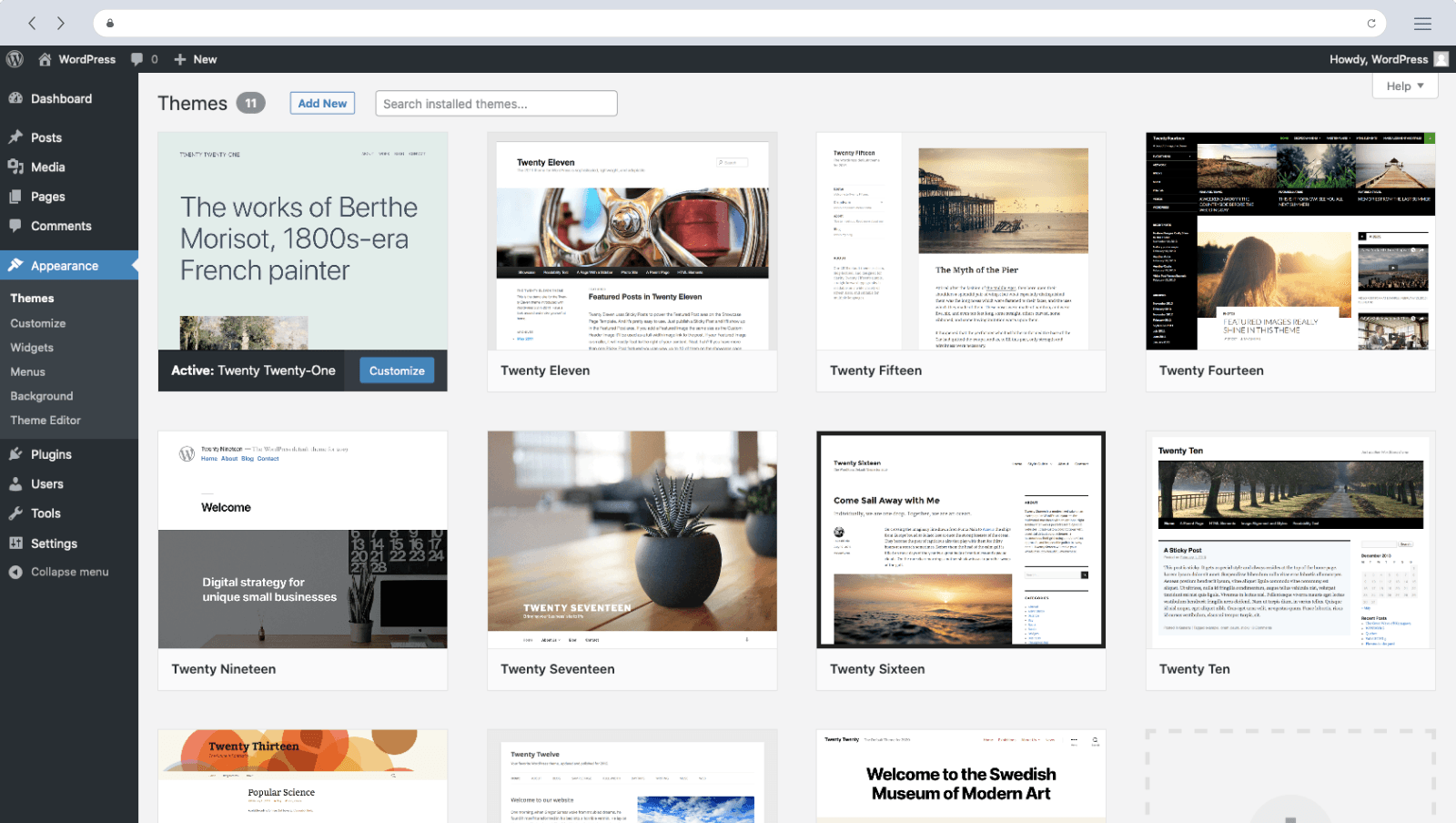
* **Hệ thống quản lý nội dung (CMS):** Sẽ lựa chọn và triển khai một nền tảng CMS phổ biến như WordPress hoặc Drupal để quản lý nội dung của website một cách hiệu quả và linh hoạt.
* **Nền tảng thương mại điện tử (E-Commerce):** Sử dụng một giải pháp E-Commerce phù hợp như WooCommerce (cho WordPress) hoặc Magento để xây dựng và quản lý các hoạt động mua bán trực tuyến trên website.
* **ContinuousIntegration/ContinuousDeployment (CI/CD):** Áp dụng và cấu hình một quy trình CI/CD để tự động hóa việc kiểm tra, xây dựng và triển khai các thay đổi vào hệ thống. Các công cụ có thể sử dụng bao gồm Jenkins, GitLab CI/CD hoặc GitHub Actions để đảm bảo quy trình phát triển và triển khai được thực hiện một cách nhanh chóng và đáng tin cậy.
* **Reverse Proxy:** Sử dụng cơ chế Reverse Proxy như Nginx để cải thiện bảo mật và hiệu suất của website. Reverse Proxy sẽ giúp che giấu địa chỉ IP thực của máy chủ và tối ưu hóa quản lý lưu lượng truy cập đến từ người dùng.

1. **Mục tiêu của đề tài**

* **Triển khai hoàn chỉnh Website CMS và E-Commerce:** Xây dựng và triển khai một website hoàn chỉnh tích hợp cả hai thành phần CMS và E-Commerce, đảm bảo tính linh hoạt và hiệu quả trong quản lý nội dung và giao dịch thương mại điện tử.
* **Tự động hóa quy trình phát triển và triển khai:** Thiết lập một quy trình CI/CD để tự động hóa các bước kiểm tra, xây dựng và triển khai mã nguồn lên môi trường sản xuất.
* **Cải thiện bảo mật và hiệu suất:** Áp dụng cơ chế Reverse Proxy để tăng cường bảo mật hệ thống và cải thiện hiệu suất của website trong việc xử lý lưu lượng truy cập.
* **Thực hành và demo:** Thực hiện thực hành triển khai website theo cách thức CI/CD và cung cấp một bản demo để chứng minh tính hoạt động và tính khả dụng của hệ thống.

# **CHƯƠNG II: CÔNG NGHỆ SỬ DỤNG**

1. **WordPress**



WordPress là một hệ thống quản lý nội dung (CMS) mã nguồn mở phổ biến nhất trên thế giới, được sử dụng rộng rãi cho việc xây dựng và quản lý các trang web và blog. Với hơn 40% trang web trên internet sử dụng WordPress, nó đã khẳng định vị thế là một trong những nền tảng phát triển web mạnh mẽ và đáng tin cậy.

Những đặc điểm nổi bật của WordPress:

* Dễ dàng sử dụng và cài đặt:

WordPress cung cấp một quy trình cài đặt đơn giản và giao diện quản trị thân thiện, cho phép người dùng dễ dàng thêm, chỉnh sửa và quản lý nội dung trang web mà không cần có kiến thức chuyên sâu về lập trình.

* Phong phú về plugin và theme:

Với hơn 58,000+ plugin và nhiều theme miễn phí và trả phí có sẵn, WordPress cho phép người dùng mở rộng chức năng và tùy chỉnh giao diện của trang web một cách linh hoạt và dễ dàng.

* Cộng đồng lớn và hỗ trợ:

WordPress có một cộng đồng sôi động với hàng triệu người dùng và nhà phát triển trên toàn thế giới. Người dùng có thể tìm thấy hỗ trợ thông qua các diễn đàn, tài liệu hướng dẫn và các sự kiện giao lưu.

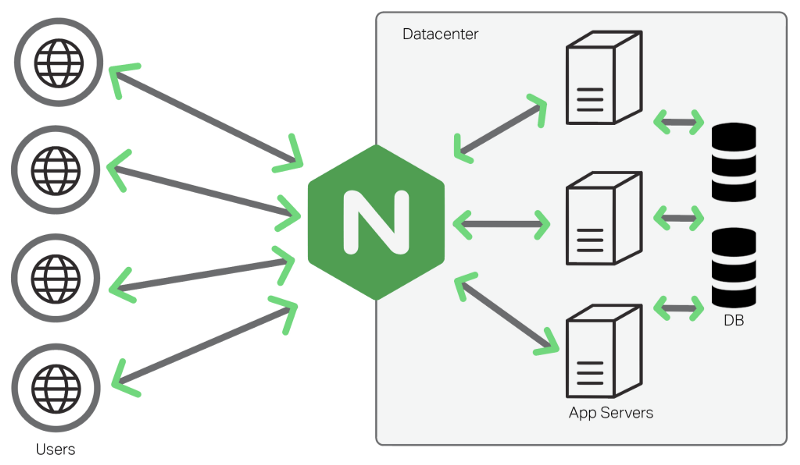
* Tối ưu hóa SEO và hiệu suất:

WordPress được xây dựng với các chuẩn tối ưu hóa SEO, giúp website dễ dàng được tìm thấy trên các công cụ tìm kiếm như Google. Ngoài ra, nó cũng có các plugin hỗ trợ tối ưu hóa hiệu suất và tăng tốc độ tải trang.

* Bảo mật và cập nhật thường xuyên:

WordPress có cộng đồng phát triển mạnh mẽ và luôn cập nhật bảo mật thường xuyên, giúp bảo vệ trang web của bạn khỏi các lỗ hổng bảo mật tiềm ẩn.

1. **Nginx**



Nginx là một phần mềm máy chủ web mã nguồn mở rất mạnh mẽ và phổ biến, được sử dụng rộng rãi để làm máy chủ web và cũng là một reverse proxy (proxy chuyển hướng ngược). Dưới đây là một số điểm nổi bật về việc sử dụng Nginx trong vai trò máy chủ và reverse proxy:

* Làm máy chủ web:

Hiệu suất cao: Nginx được thiết kế để xử lý hàng ngàn kết nối đồng thời một cách hiệu quả, với khả năng xử lý tải cao.

Tính ổn định và tin cậy: Nginx là máy chủ web có khả năng chịu tải tốt, đồng thời có khả năng khôi phục nhanh sau sự cố.

* Reverse Proxy:

Điều hướng traffic: Nginx có thể được cấu hình như một reverse proxy để điều hướng lưu lượng truy cập từ người dùng đến các ứng dụng web hoặc máy chủ phía sau một cách hiệu quả.

Cải thiện bảo mật: Reverse Proxy giúp che giấu địa chỉ IP thực của máy chủ và tăng cường bảo mật bằng cách kiểm soát và lọc lưu lượng truy cập.

* Cấu hình và quản lý:

Dễ dàng cấu hình: Nginx sử dụng cú pháp cấu hình đơn giản và mạnh mẽ, cho phép người quản trị dễ dàng điều chỉnh các tùy chọn để phù hợp với yêu cầu cụ thể của hệ thống.

Quản lý tài nguyên: Nginx sử dụng ít tài nguyên hệ thống so với một số máy chủ web khác, làm tăng hiệu quả và tiết kiệm chi phí vận hành.

* Tích hợp và mở rộng:

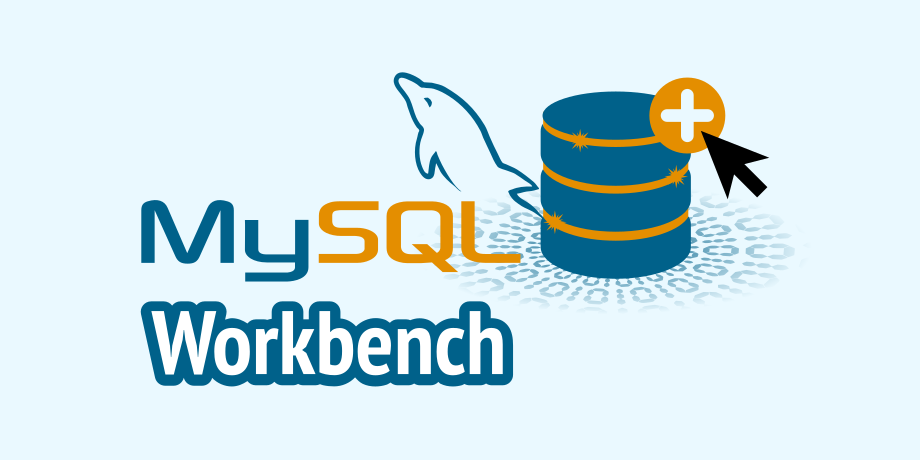
Hỗ trợ module và plugin: Nginx hỗ trợ nhiều module mở rộng và có thể tích hợp với các dịch vụ bổ sung như cache, SSL/TLS, load balancing, ... để mở rộng chức năng và hiệu suất của hệ thống.

* Sử dụng trong các kịch bản phức tạp:

Load balancing: Nginx có thể được sử dụng làm máy chủ cân bằng tải (load balancer) để phân phối công việc và tải trọng giữa các máy chủ backend.

Cache server: Nginx có thể làm proxy cache, giúp cải thiện thời gian phản hồi của các ứng dụng web bằng cách lưu trữ nội dung đã xử lý trước.

1. **MySQL Workbench**



MySQL Workbench là một công cụ quản lý cơ sở dữ liệu (DBMS - Database Management System) phổ biến được sử dụng để thiết kế cơ sở dữ liệu, thực thi truy vấn SQL, quản lý kết nối và thực hiện các tác vụ quản lý cơ sở dữ liệu khác.

Dưới đây là một số tính năng chính của MySQL Workbench:

* Thiết kế và quản lý cơ sở dữ liệu: MySQL Workbench cung cấp các công cụ để thiết kế cơ sở dữ liệu ER (Entity-Relationship), tạo bảng, quản lý ràng buộc, chỉ mục và các đối tượng cơ sở dữ liệu khác.
* Thực thi truy vấn SQL: Bạn có thể thực hiện các truy vấn SQL trực tiếp từ giao diện của MySQL Workbench, xem kết quả và thậm chí lưu các truy vấn phổ biến vào tệp tin.
* Xem và quản lý dữ liệu: MySQL Workbench cho phép bạn xem và chỉnh sửa dữ liệu trong bảng, cũng như thực hiện các thao tác như chèn, cập nhật và xóa dữ liệu.
* Quản lý kết nối và máy chủ: Bạn có thể quản lý kết nối đến các máy chủ MySQL khác nhau từ giao diện của MySQL Workbench, cho phép bạn thực hiện các thao tác như kết nối, ngắt kết nối, và kiểm tra trạng thái kết nối.
* Tối ưu hóa và xem hồ sơ truy vấn: MySQL Workbench cung cấp các công cụ để tối ưu hóa các truy vấn SQL và xem hồ sơ thời gian thực hiện của chúng để cải thiện hiệu suất của cơ sở dữ liệu.
* Tích hợp với các công cụ khác: MySQL Workbench có thể tích hợp với các công cụ khác như môi trường phát triển phần mềm (IDE) và các công cụ quản lý phiên bản, giúp tăng cường hiệu suất làm việc của người phát triển.

1. **WooCommerce**



WooCommerce là một plugin thương mại điện tử mạnh mẽ được phát triển cho nền tảng WordPress. Được ra mắt lần đầu vào năm 2011, WooCommerce nhanh chóng trở thành một trong những giải pháp thương mại điện tử phổ biến nhất trên thế giới, được sử dụng rộng rãi bởi các doanh nghiệp nhỏ và lớn để xây dựng các cửa hàng trực tuyến.

Đặc điểm nổi bật của WooCommerce:

* Tích hợp mượt mà với WordPress:
* WooCommerce được tích hợp chặt chẽ và hoàn toàn miễn phí với WordPress. Nó mở rộng khả năng của WordPress từ một nền tảng blog thành một nền tảng thương mại điện tử đầy đủ chức năng.
* Dễ dàng sử dụng và cài đặt:
* Nhờ vào giao diện người dùng thân thiện và quản lý sản phẩm đơn giản, người dùng có thể dễ dàng cài đặt và vận hành WooCommerce mà không cần có kỹ năng lập trình sâu.
* Tính tùy biến cao:
* WooCommerce cung cấp nhiều tính năng mở rộng và plugin bổ sung giúp người dùng tùy chỉnh cửa hàng của mình theo nhu cầu riêng. Người dùng có thể thêm các tính năng như thanh toán trực tiếp, lựa chọn vận chuyển, ưu đãi khách hàng, và nhiều hơn nữa.

Bảo mật và cộng đồng hỗ trợ:

* WooCommerce luôn được cập nhật để đảm bảo tính bảo mật cao và tuân thủ các tiêu chuẩn an ninh. Ngoài ra, có một cộng đồng lớn hỗ trợ và sẵn sàng chia sẻ kiến thức và kinh nghiệm liên quan đến việc sử dụng WooCommerce.
* Thân thiện với SEO:

WooCommerce được tối ưu hóa để hỗ trợ SEO, giúp cửa hàng trực tuyến dễ dàng được tìm kiếm và xếp hạng cao trên các công cụ tìm kiếm như Google.

* Phù hợp cho nhiều loại doanh nghiệp:

Từ các cửa hàng nhỏ với một vài sản phẩm đến các doanh nghiệp lớn với hàng ngàn sản phẩm, WooCommerce phù hợp với nhiều loại hình kinh doanh và mô hình bán hàng.

* Dịch vụ bổ sung và tích hợp mở rộng:

Ngoài các tính năng cơ bản, WooCommerce còn có thể tích hợp với các dịch vụ bổ sung như thanh toán trực tuyến, vận chuyển, quản lý kho hàng và hệ thống quản lý mối quan hệ khách hàng (CRM).

1. **GitHub**



GitHub là một dịch vụ lưu trữ mã nguồn và quản lý dự án phổ biến, được sử dụng rộng rãi trong cộng đồng phát triển phần mềm để hỗ trợ quản lý và theo dõi các phiên bản mã nguồn, hợp tác và kiểm soát các thay đổi trong dự án phát triển phần mềm. Dưới đây là các đặc điểm chính của GitHub:

Đặc điểm nổi bật của GitHub:

* Lưu trữ mã nguồn:

GitHub cung cấp không gian lưu trữ trực tuyến cho các dự án phát triển phần mềm, cho phép các nhà phát triển lưu trữ mã nguồn của họ trên các kho chứa (repositories). Mỗi kho chứa có thể chứa mã nguồn, tài liệu, hình ảnh và các tài nguyên khác liên quan đến dự án.

* Quản lý phiên bản:

GitHub hỗ trợ quản lý các phiên bản của mã nguồn (version control) thông qua hệ thống Git. Người dùng có thể quản lý lịch sử thay đổi, so sánh các phiên bản khác nhau của mã nguồn và điều chỉnh các thay đổi một cách dễ dàng.

* Hợp tác và phân phối công việc:

GitHub cung cấp các công cụ hỗ trợ hợp tác nhóm như theo dõi vấn đề (issue tracking), pull request và code review. Điều này giúp cho các thành viên trong dự án có thể làm việc cùng nhau một cách hiệu quả, đánh giá và chấp nhận các thay đổi trong mã nguồn.

* Quản lý dự án:

GitHub cung cấp các tính năng quản lý dự án như lập lịch, theo dõi tiến độ công việc và phân phối nhiệm vụ giữa các thành viên trong dự án. Điều này giúp cho quản lý dự án và điều phối công việc trở nên hiệu quả hơn.

* Tích hợp và mở rộng:

GitHub cho phép tích hợp với các công cụ và dịch vụ phát triển phần mềm khác như CI/CD pipelines, công cụ kiểm thử tự động và các dịch vụ hỗ trợ phát triển và triển khai ứng dụng.

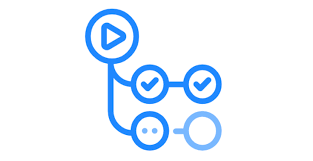
* Cộng đồng lớn và hỗ trợ mở:

GitHub có một cộng đồng người dùng rộng lớn và tích cực, cung cấp các diễn đàn thảo luận, hướng dẫn sử dụng và chia sẻ tài liệu để hỗ trợ nhau trong việc phát triển phần mềm.

* Dịch vụ đám mây:

GitHub cũng cung cấp các dịch vụ đám mây để lưu trữ và quản lý dữ liệu, bảo mật và sao lưu thông tin dự án một cách an toàn.

1. **GitHub Actions**



GitHub Actions là một dịch vụ của GitHub cho phép tự động hóa các quy trình phát triển phần mềm, từ kiểm thử, xây dựng (build), đóng gói (package), đến triển khai (deployment). Được tích hợp trực tiếp vào GitHub, GitHub Actions giúp các nhà phát triển và các nhóm phát triển phần mềm có thể tự động hóa các công việc phổ biến mà trước đây thường phải thực hiện bằng tay hoặc phải sử dụng các công cụ bên ngoài.

Đặc điểm nổi bật của GitHub Actions:

* Tích hợp hoàn hảo với GitHub:

GitHub Actions tích hợp sâu vào GitHub, cho phép người dùng cấu hình và quản lý các workflow (quy trình làm việc) trực tiếp từ giao diện GitHub.

* Tự động hóa quy trình CI/CD:

GitHub Actions cho phép người dùng xây dựng (build), kiểm thử, và triển khai ứng dụng một cách tự động. Bằng cách sử dụng các workflow được định nghĩa trong file YAML, người dùng có thể tự động hóa các bước như kiểm tra mã nguồn, xây dựng ứng dụng, đóng gói, và triển khai lên môi trường sản phẩm.

* Hỗ trợ nhiều nền tảng và ngôn ngữ:

GitHub Actions hỗ trợ nhiều nền tảng và ngôn ngữ lập trình, cho phép người dùng xây dựng và triển khai các ứng dụng trên nhiều môi trường khác nhau như Linux, macOS, Windows.

* Các sự kiện kích hoạt (trigger events):

Người dùng có thể cấu hình các workflow để phản ứng với các sự kiện nhất định trên GitHub như khi có commit mới vào repository, khi tạo hoặc đóng pull request, hoặc các sự kiện tùy chỉnh.

* Thư viện Actions và Community Marketplace:

GitHub Actions có một thư viện Actions phong phú, cho phép người dùng tái sử dụng các hành động (actions) đã được định nghĩa sẵn. Ngoài ra, còn có Community Marketplace để người dùng chia sẻ và tìm kiếm các workflow và hành động mở rộng từ cộng đồng.

* Khả năng mở rộng và tùy chỉnh:

GitHub Actions cho phép người dùng tạo các hành động tùy chỉnh (custom actions) để phù hợp với các yêu cầu cụ thể của dự án phần mềm.

1. **InfinityFree**

**A screenshot of a computer program

Description automatically generated**

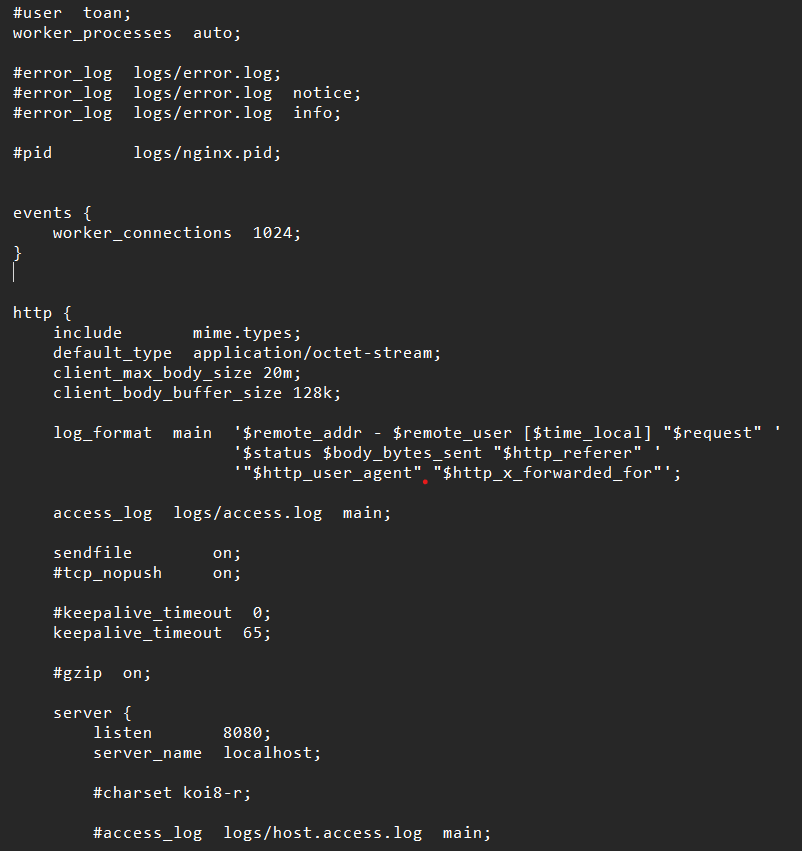
InfinityFree là một nhà cung cấp dịch vụ lưu trữ web miễn phí. Đây là một lựa chọn phổ biến cho những người dùng muốn tạo trang web mà không cần phải trả tiền cho dịch vụ lưu trữ.

Đặc điểm:

* Miễn phí: InfinityFree cung cấp dịch vụ lưu trữ web hoàn toàn miễn phí, không yêu cầu phí đăng ký hay duy trì dịch vụ.
* Dung lượng lưu trữ không giới hạn: Mặc dù miễn phí, InfinityFree cung cấp dung lượng lưu trữ không giới hạn cho các tệp trang web.
* Băng thông không giới hạn: Dịch vụ này cũng cung cấp băng thông không giới hạn, cho phép lượng lớn lưu lượng truy cập mà không gặp phải các vấn đề về băng thông.
* Tên miền miễn phí: InfinityFree cung cấp các tên miền phụ miễn phí (subdomains).
* Hỗ trợ PHP và MySQL: InfinityFree hỗ trợ các công nghệ phổ biến như PHP và MySQL, giúp dễ dàng triển khai các trang web động và cơ sở dữ liệu.
* SSL miễn phí: Có thể cài đặt chứng chỉ SSL miễn phí để bảo mật trang web.
* Công cụ quản lý: InfinityFree cung cấp bảng điều khiển (control panel) dễ sử dụng, bao gồm các công cụ như trình quản lý tệp và trình cài đặt ứng dụng tự động.

# **CHƯƠNG III: THÔNG SỐ CẤU HÌNH**

1. **Server**



A computer screen shot of a black screen

Description automatically generated

* worker\_processes: Số lượng tiến trình làm việc (worker processes) mà Nginx sẽ khởi chạy.

Giá trị cụ thể: auto (Nginx sẽ tự động chọn số lượng tiến trình dựa trên số lượng CPU có sẵn)

* worker\_connections: Số lượng kết nối tối đa mỗi worker process có thể xử lý.

Giá trị cụ thể: 1024

* include: Bao gồm các loại MIME từ một tệp cấu hình bên ngoài.

Giá trị cụ thể: mime.types

* default\_type: Định nghĩa kiểu mặc định cho các nội dung không có loại MIME cụ thể.

Giá trị cụ thể: application/octet-stream

* client\_max\_body\_size: Kích thước tối đa của thân yêu cầu (request body) mà máy chủ có thể xử lý.

Giá trị cụ thể: 20m (20 megabytes)

* client\_body\_buffer\_size: Kích thước bộ đệm cho thân yêu cầu.

Giá trị cụ thể: 128k (128 kilobytes)

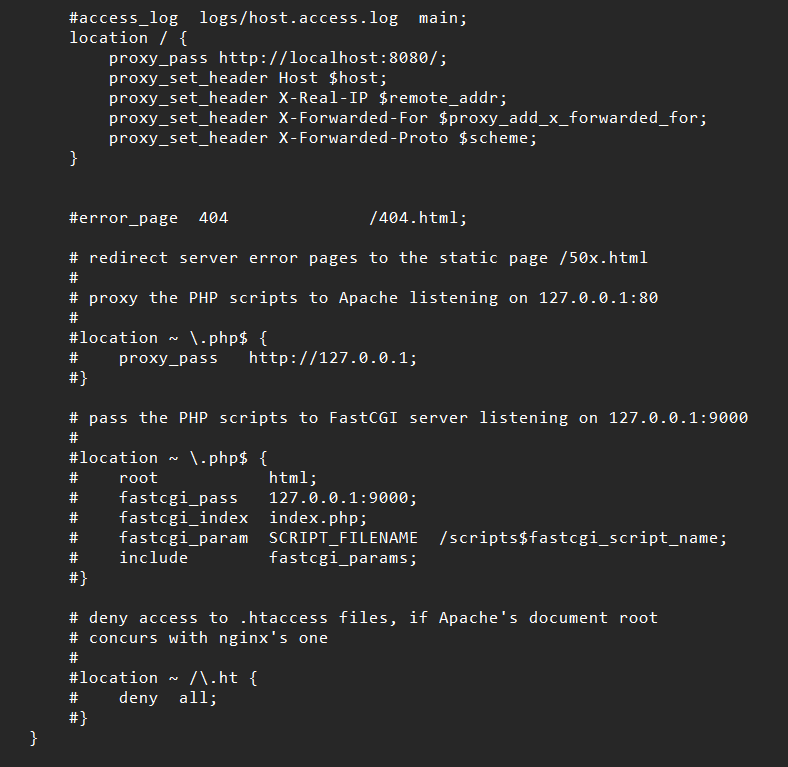
* keepalive\_timeout: Thời gian giữ kết nối mở.

Giá trị cụ thể: 65 (giây)

1. **Reverser Proxy**

A screen shot of a computer

Description automatically generated



* listen 80;: Câu lệnh này đặt Nginx chờ đợi các yêu cầu đến trên cổng 80. Cổng 80 là cổng tiêu chuẩn cho HTTP và là cổng mà các trình duyệt web sẽ sử dụng để gửi yêu cầu HTTP tới máy chủ web.
* server\_name localhost;: Đây là chỉ định máy chủ ảo (virtual host) trong Nginx. Trong trường hợp này, Nginx sẽ xử lý các yêu cầu đến từ máy chủ có tên là localhost. Máy chủ ảo này có thể có nhiều hơn một và được sử dụng để phân định các trang web trên cùng một máy chủ vật lý.
* proxy\_pass:

Định nghĩa: Địa chỉ của máy chủ backend mà Nginx sẽ chuyển tiếp yêu cầu đến.

Ví dụ: proxy\_pass <http://localhost:8080/>;

* proxy\_set\_header:

Định nghĩa: Thiết lập các tiêu đề HTTP được chuyển tiếp đến máy chủ backend.

* Các giá trị cụ thể:
* Host $host: Chuyển tiếp tiêu đề Host từ client.
* X-Real-IP $remote\_addr: Chuyển tiếp địa chỉ IP thực của client.
* X-Forwarded-For $proxy\_add\_x\_forwarded\_for: Thêm địa chỉ IP của client vào chuỗi các địa chỉ IP đã đi qua proxy.
* X-Forwarded-Proto $scheme: Chuyển tiếp giao thức (HTTP hoặc HTTPS) mà client sử dụng.
* proxy\_buffering:

Định nghĩa: Bật hoặc tắt bộ đệm proxy.

Giá trị cụ thể: proxy\_buffering on; hoặc proxy\_buffering off;

* proxy\_buffers:

Định nghĩa: Định nghĩa số lượng và kích thước của các bộ đệm cho proxy.

Ví dụ: proxy\_buffers 8 16k;

* proxy\_buffer\_size:

Định nghĩa: Kích thước của bộ đệm đầu tiên nhận từ máy chủ backend.

Ví dụ: proxy\_buffer\_size 32k;

* proxy\_busy\_buffers\_size:

Định nghĩa: Kích thước của bộ đệm bận, sử dụng khi các bộ đệm thông thường đã đầy.

Ví dụ: proxy\_busy\_buffers\_size 64k;

* proxy\_temp\_file\_write\_size:

Định nghĩa: Kích thước của tệp tạm cho proxy.

Ví dụ: proxy\_temp\_file\_write\_size 64k;

* proxy\_cache:

Định nghĩa: Tên của bộ đệm proxy.

Ví dụ: proxy\_cache my\_cache;

* proxy\_cache\_valid:

Định nghĩa: Thời gian lưu trữ các phản hồi trong bộ đệm proxy.

Ví dụ: proxy\_cache\_valid 200 302 10m; (lưu trữ phản hồi với mã trạng thái 200 và 302 trong 10 phút)

* proxy\_cache\_key:

Định nghĩa: Khóa duy nhất cho mỗi yêu cầu được lưu trữ trong bộ đệm proxy.

Ví dụ: proxy\_cache\_key "$scheme$proxy\_host$request\_uri";

* proxy\_redirect:

Định nghĩa: Điều chỉnh URL trong các tiêu đề Location và Refresh trong các phản hồi từ máy chủ backend.

Ví dụ: proxy\_redirect http://localhost:8080/ /;

* proxy\_read\_timeout:

Định nghĩa: Thời gian chờ để đọc phản hồi từ máy chủ backend.

Ví dụ: proxy\_read\_timeout 90s;

* proxy\_connect\_timeout:

Định nghĩa: Thời gian chờ để kết nối tới máy chủ backend.

Ví dụ: proxy\_connect\_timeout 30s;

* proxy\_send\_timeout:

Định nghĩa: Thời gian chờ để gửi yêu cầu đến máy chủ backend.

Ví dụ: proxy\_send\_timeout 30s;

* proxy\_intercept\_errors:

Định nghĩa: Bật hoặc tắt việc chặn lỗi từ máy chủ backend và hiển thị trang lỗi do Nginx định nghĩa.

Ví dụ: proxy\_intercept\_errors on; hoặc proxy\_intercept\_errors off;

1. **Ci/Cd**

**A screenshot of a computer program

Description automatically generated**

**A computer screen shot of a program

Description automatically generated**

* File cấu hình:

Tên file: .github/workflows/deploy.yml

Cấu hình sự kiện kích hoạt:

* Sự kiện: push

Nhánh: main

Ý nghĩa: Khi có sự thay đổi được đẩy lên nhánh main, workflow sẽ được kích hoạt.

* Môi trường chạy:

Hệ điều hành: windows-latest

Ý nghĩa: Workflow sẽ chạy trên môi trường Windows để thực hiện các bước xây dựng.

* Các bước thực hiện (steps):

Bước 1: Checkout mã nguồn từ repository

- name: Checkout repository

uses: actions/checkout@v2

Bước 2: Thiết lập môi trường PHP

- name: Set up PHP

uses: shivammathur/setup-php@v2

with:

php-version: '7.4' # Chọn phiên bản PHP phù hợp với WordPress

Bước 3: Cài đặt các phụ thuộc của Composer

- name: Install Composer dependencies

run: |

composer install --no-dev --optimize-autoloader

Bước 4: Cài đặt Node.js

- name: Install Node.js

uses: actions/setup-node@v2

with:

node-version: '22' # Chọn phiên bản Node.js phù hợp

Bước 5: Cài đặt các phụ thuộc của npm

- name: Install npm dependencies

run: |

npm install

Bước 6: Xây dựng chủ đề hoặc plugin của WordPress

- name: Build WordPress theme or plugin

run: |

npm run build

Bước 7: Triển khai lên máy chủ FTP

- name: Deploy to FTP server

uses: SamKirkland/FTP-Deploy-Action@4.0.0

with:

server: ${{ secrets.FTP\_SERVER }}

username: ${{ secrets.FTP\_USERNAME }}

password: ${{ secrets.FTP\_PASSWORD }}

port: 21

protocol: ftp

local-dir: ./dist/

server-dir: ./htdocs/

Giải thích các thông số:

* push: Sự kiện kích hoạt workflow khi có sự thay đổi được đẩy lên nhánh main.
* runs-on: windows-latest: Workflow chạy trên môi trường Windows để xây dựng và triển khai.
* uses: Sử dụng các action có sẵn từ GitHub Marketplace để thực hiện các bước cài đặt và triển khai.
* with: Các tham số cài đặt như phiên bản PHP, Node.js, tài khoản FTP, thư mục local và server để triển khai.

Đây là một cấu hình CI/CD đơn giản để xây dựng và triển khai WordPress sử dụng GitHub Actions và FTP-Deploy-Action.

# **CHƯƠNG IV: PHỤ LỤC**

1. **Mã nguồn**

**Link dự án:** <https://github.com/Ckt2002/wordpress-nginx-cicd.git>

1. **Tài liệu tham khảo**

[1] **Lê Hà Thanh**, ‘Ci/Cd’