

BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
CƠ SỞ TẠI THÀNH PHỐ HỒ CHÍ MINH

KHOA CÔNG NGHỆ THÔNG TIN 2

---o0o---



BÁO CÁO GIỮA KỲ & CUỐI KỲ
TÀI LIỆU HƯỚNG DẪN THIẾT LẬP VÀ TRIỂN KHAI PHẦN MỀM

Môn học: Chuyên đề Hệ thống thông tin

Giảng viên hướng dẫn: Lê Hà Thanh

Lớp: D20CQCNHT01-N

Sinh viên thực hiện:

Hồ Đức Hoàng

N20DCCN018

TP.HCM, ngày 17 tháng 06 năm 2024

MỤC LỤC

I. GIỚI THIỆU TÀI LIỆU	2
II. TỔNG QUAN WEBSITE	2
III. YÊU CẦU HỆ THỐNG	2
IV. HƯỚNG DẪN CÀI ĐẶT VÀ SỬ DỤNG	3
4.1. Clone repository về máy tính cá nhân và cấu hình	3
4.2. Triển khai cài đặt CMS (WordPress)	7
4.3. Triển khai cài đặt E-Commerce (WooCommerce)	12
4.4. Triển khai CI/CD	16
V. TÀI LIỆU ĐÍNH KÈM.....	19

DANH MỤC HÌNH ẢNH

Hình 1. Giao diện phần mềm SourceTree	3
Hình 2. Cấu hình service database.....	4
Hình 3. Cấu hình service WordPress.....	5
Hình 4. Cấu hình service NGINX	6
Hình 5. Cấu hình file nginx.conf	6
Hình 6. Thiết lập ngôn ngữ.....	7
Hình 7. Điền thông tin website.....	8
Hình 8. Thông báo install WordPress thành công.....	9
Hình 9. Giao diện website	9
Hình 10. Giao diện trang login	10
Hình 11. Giao diện trang login	10
Hình 12. Giao diện trang quản lý các post	11
Hình 13. Giao diện một trang blog	11
Hình 14. Cấu hình file workflows CI/CD.....	16
Hình 15. Trường hợp thành công	17
Hình 16. Trường hợp không thành công	18

I. GIỚI THIỆU TÀI LIỆU

- Tài liệu này dùng để:
 - Triển khai cài đặt website sử dụng kỹ thuật Reverse Proxy.
 - Triển khai cài đặt CMS (WordPress blogs) và cài đặt E-commerce (WooCommerce).
 - Triển khai sử dụng CI/CD bằng Github Actions.
- Phạm vi: tài liệu này hướng dẫn triển khai ở môi trường phát triển (Development Environment) và môi trường kiểm thử (Testing Environment).

II. TỔNG QUAN WEBSITE

- Các công nghệ sử dụng:
 - OS-level virtualization: Docker Desktop
 - Reverse proxy server: NGINX
 - Content Management System (CMS): WordPress
 - Database: MySQL

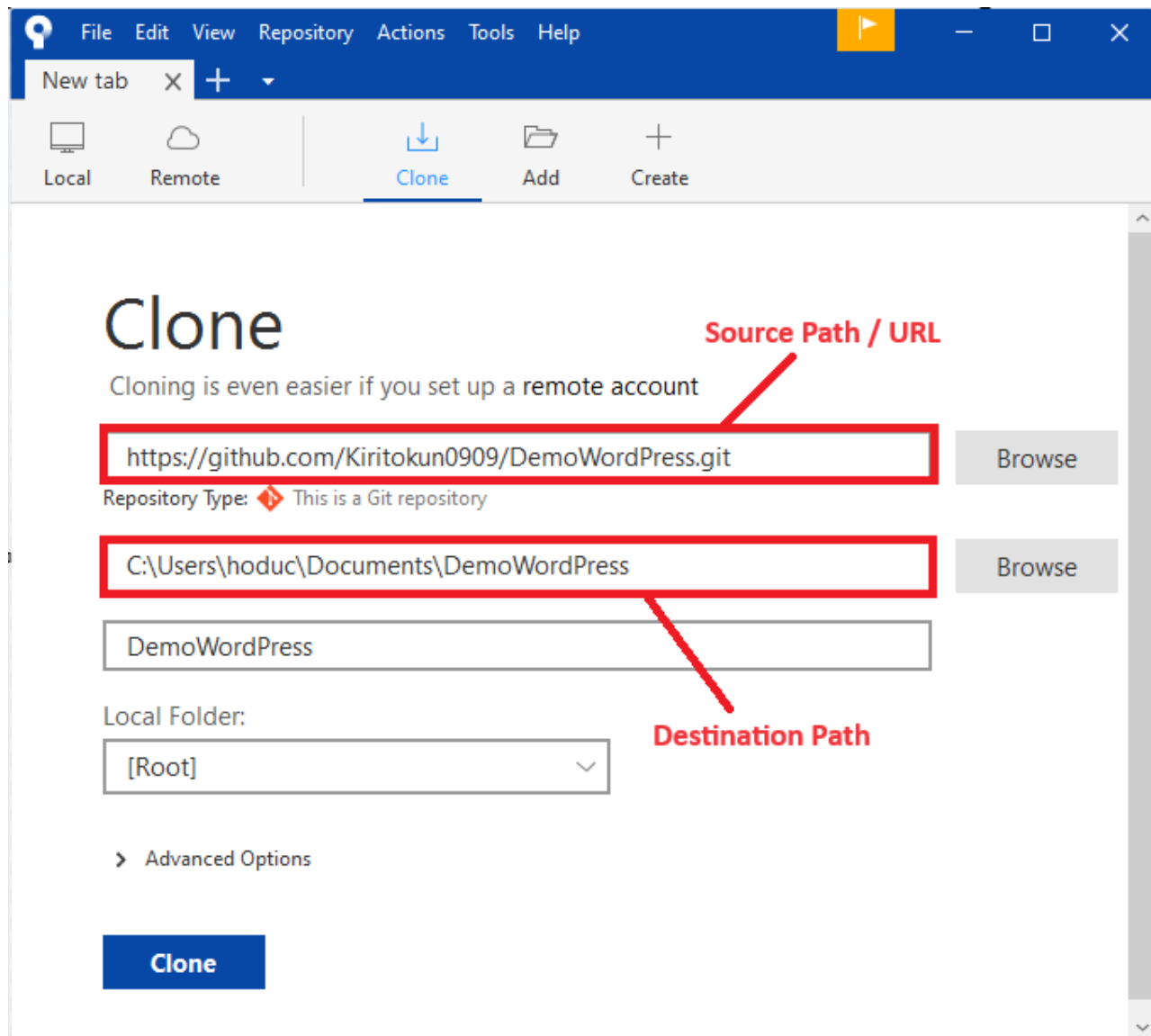
III. YÊU CẦU HỆ THỐNG

- Yêu cầu phần cứng:
 - Hệ điều hành: Windows 10/11 64-bit.
 - Dung lượng RAM: tối thiểu 4GB.
 - Dung lượng ổ cứng: tối thiểu 10GB.
 - Kích hoạt ảo hóa phần cứng (hardware virtualization) trong BIOS.
 - Bật được tính năng WSL 2 trên Windows.
- Yêu cầu phần mềm:
 - Cài đặt sẵn Docker Desktop.
 - Cài đặt sẵn IDE hoặc TextEditor như Visual Studio Code, Notepad++, ...
 - Phần mềm quản lý mã nguồn như Git, SourceTree, ...

IV. HƯỚNG DẪN CÀI ĐẶT VÀ SỬ DỤNG

4.1. Clone repository về máy tính cá nhân và cấu hình

- Mở phần mềm SourceTree đã được cài đặt trong máy tính.
- Source path / URL: đường link mã nguồn github.
- Destination Path: đường dẫn thư mục (folder) muốn dùng để lưu trữ mã nguồn.



Hình 1. Giao diện phần mềm SourceTree

Clone repository [DemoWordPress](#) về máy tính cá nhân

Cấu hình các service tại file docker-compose.yml

- Service database:

```
services:
  db:
    image: mysql:5.7
    container_name: wordpress_db
    restart: always
    environment:
      MYSQL_DATABASE: wordpress
      MYSQL_USER: wpuser
      MYSQL_PASSWORD: password
      MYSQL_ROOT_PASSWORD: root_password
    volumes:
      - db_data:/var/lib/mysql
```

Hình 2. Cấu hình service database

Trong đó, cấu hình bao gồm:

- image: sử dụng phiên bản MySQL 5.7 từ Docker Hub.
- container_name: tên cho container.
- restart: always: tự động khởi động lại container nếu nó dừng hoặc bị lỗi.
- environment: Thiết lập các biến môi trường cho MySQL.
 - MYSQL_DATABASE: tên cơ sở dữ liệu.
 - MYSQL_USER: tên người dùng (user) trong cơ sở dữ liệu (database).
 - MYSQL_PASSWORD: mật khẩu của user.
 - MYSQL_ROOT_PASSWORD: mật khẩu cho user root của MySQL.
- volumes:
 - db_data: Liên kết volume "db_data" với thư mục /var/lib/mysql trong container để lưu trữ dữ liệu của MySQL.

- Service WordPress:

```
services:
  wordpress:
    image: wordpress:latest
    container_name: wordpress
    restart: always
    ports:
      - "8080:80"
    environment:
      WORDPRESS_DB_HOST: db
      WORDPRESS_DB_USER: wpuser
      WORDPRESS_DB_PASSWORD: password
      WORDPRESS_DB_NAME: wordpress
    volumes:
      - wordpress_data:/var/www/html
```

Hình 3. Cấu hình service WordPress

Trong đó, cấu hình bao gồm:

- image: phiên bản wordpress (ở đây sử dụng bản mới nhất từ Docker Hub).
- container_name: tên container.
- restart: always: tự động khởi động lại container nếu nó dừng hoặc lỗi.
- ports: liên kết cổng 8080 của máy chủ với cổng 80 của container. Sau khi chạy thành công, có thể truy cập WordPress thông qua <http://localhost:8080>.
- environment: các biến môi trường cần thiết cho WordPress để kết nối với cơ sở dữ liệu
 - WORDPRESS_DB_HOST: tên service database.
 - WORDPRESS_DB_USER: tên user đăng nhập database (lưu ý: phải trùng với tên user đã đặt ở service database).
 - WORDPRESS_DB_PASSWORD: mật khẩu của user đăng nhập database (lưu ý: phải trùng với mật khẩu đã đặt ở service database).
 - WORDPRESS_DB_NAME: tên database (lưu ý: phải trùng với tên database đã đặt ở service database).
- volumes:
 - wordpress_data: liên kết volume "wordpress_data" với thư mục /var/www/html trong container để lưu trữ dữ liệu của WordPress.

- Service NGINX

```
services:
  nginx:
    image: nginx:1.27.0
    container_name: nginx
    restart: always
    ports:
      - "80:80"
    volumes:
      - ./nginx.conf:/etc/nginx/nginx.conf
      - wordpress_data:/var/www/html
```

Hình 4. Cấu hình service NGINX

Trong đó, cấu hình bao gồm:

- image: phiên bản NGINX.
- container_name: tên container.
- restart: always: Tự động khởi động lại container nếu nó dừng hoặc bị lỗi.
- ports: Liên kết cổng 80 của máy chủ với cổng 80 của container.
- volumes:
 - ./nginx.conf: liên kết file cấu hình NGINX từ máy chủ với file cấu hình trong container.
 - wordpress_data: liên kết volume "wordpress_data" với thư mục /var/www/html trong container NGINX để chia sẻ dữ liệu của WordPress.

- Cấu hình file nginx.conf

```
nginx.conf
1  events {}
2
3  http {
4      include    mime.types;
5      default_type  application/octet-stream;
6
7      sendfile    on;
8      keepalive_timeout  65;
9
10     server {
11         listen      80;
12         server_name localhost;
13
14         location / {
15             proxy_pass http://wordpress:80;
16             proxy_set_header Host $host;
17             proxy_set_header X-Real-IP $remote_addr;
18             proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
19             proxy_set_header X-Forwarded-Proto $scheme;
20         }
21     }
22 }
23
```

Hình 5. Cấu hình file nginx.conf

Sau khi đã hoàn thành các bước cấu hình trên, khởi động Docker Desktop.

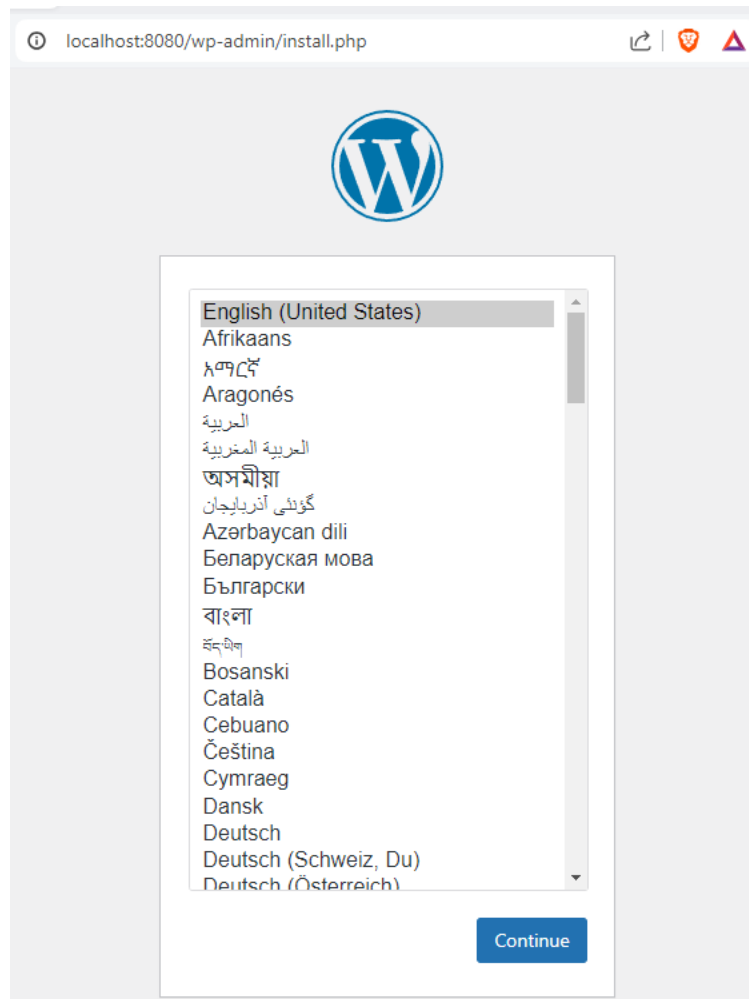
Mở Command Prompt hoặc PowerShell và điều hướng đến thư mục chứa file docker-compose.yml, sau đó chạy lệnh sau:

```
PS C:\Users\hoduc\Documents\DemoWordPress> docker-compose up
```

Sau khi chạy thành công, truy cập vào [website](#) để tiến hành thiết lập website.

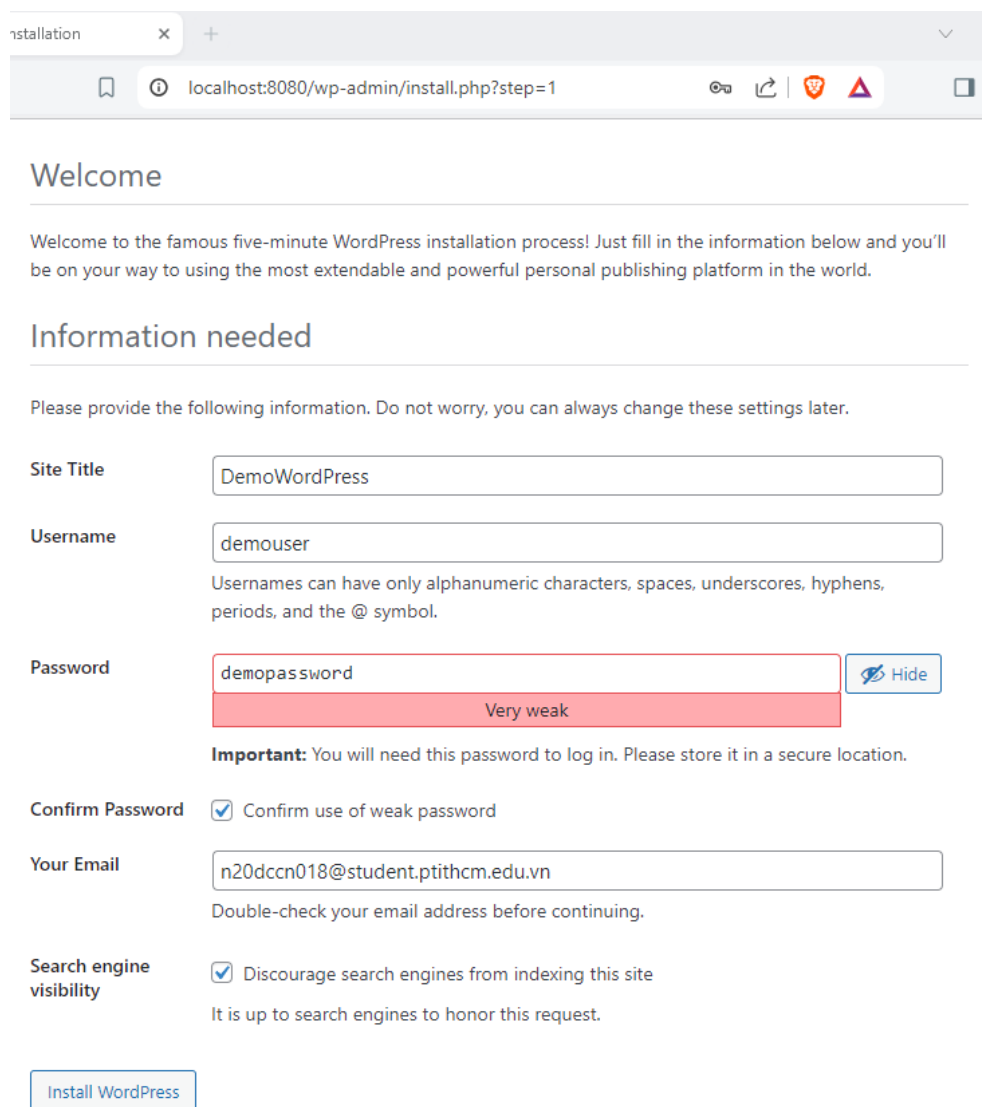
4.2. Triển khai cài đặt CMS (WordPress)

- Chọn một ngôn ngữ cho website và nhấn Continue.



Hình 6. Thiết lập ngôn ngữ

- Điền tên website, username, password và email. Sau khi điền xong thông tin, nhấn Install WordPress.



The screenshot shows the WordPress installation process at step 1. The browser address bar indicates the URL is localhost:8080/wp-admin/install.php?step=1. The page has a 'Welcome' section followed by an 'Information needed' section. Below this, a message states: 'Please provide the following information. Do not worry, you can always change these settings later.'

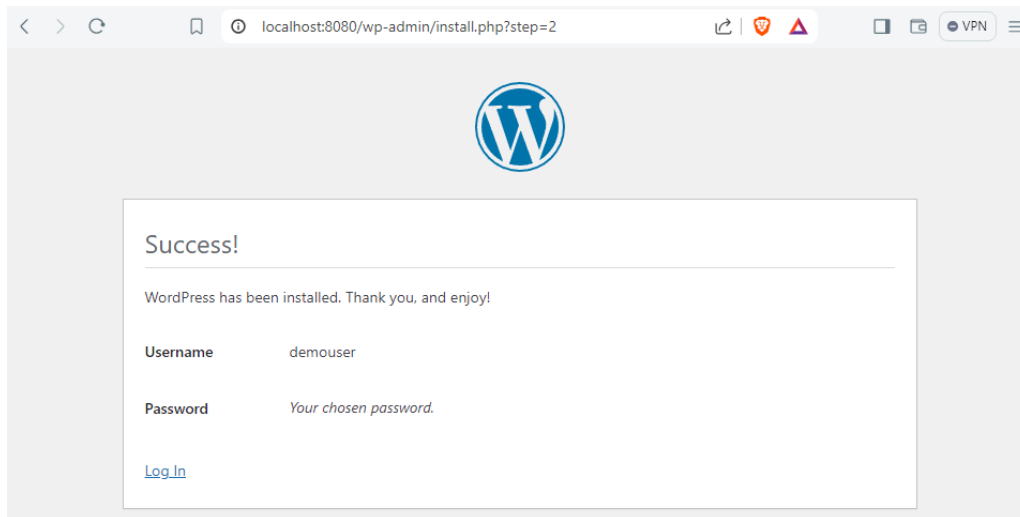
The form contains the following fields and options:

- Site Title:** A text input field containing 'DemoWordPress'.
- Username:** A text input field containing 'demouser'. Below it, a note states: 'Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.'
- Password:** A text input field containing 'demopassword'. To its right is a 'Hide' button. Below the input field is a red bar with the text 'Very weak'. Below this, an **Important:** note states: 'You will need this password to log in. Please store it in a secure location.'
- Confirm Password:** A checkbox labeled 'Confirm use of weak password' which is checked.
- Your Email:** A text input field containing 'n20dccn018@student.ptithcm.edu.vn'. Below it, a note states: 'Double-check your email address before continuing.'
- Search engine visibility:** A checkbox labeled 'Discourage search engines from indexing this site' which is checked. Below it, a note states: 'It is up to search engines to honor this request.'

At the bottom of the form is a button labeled 'Install WordPress'.

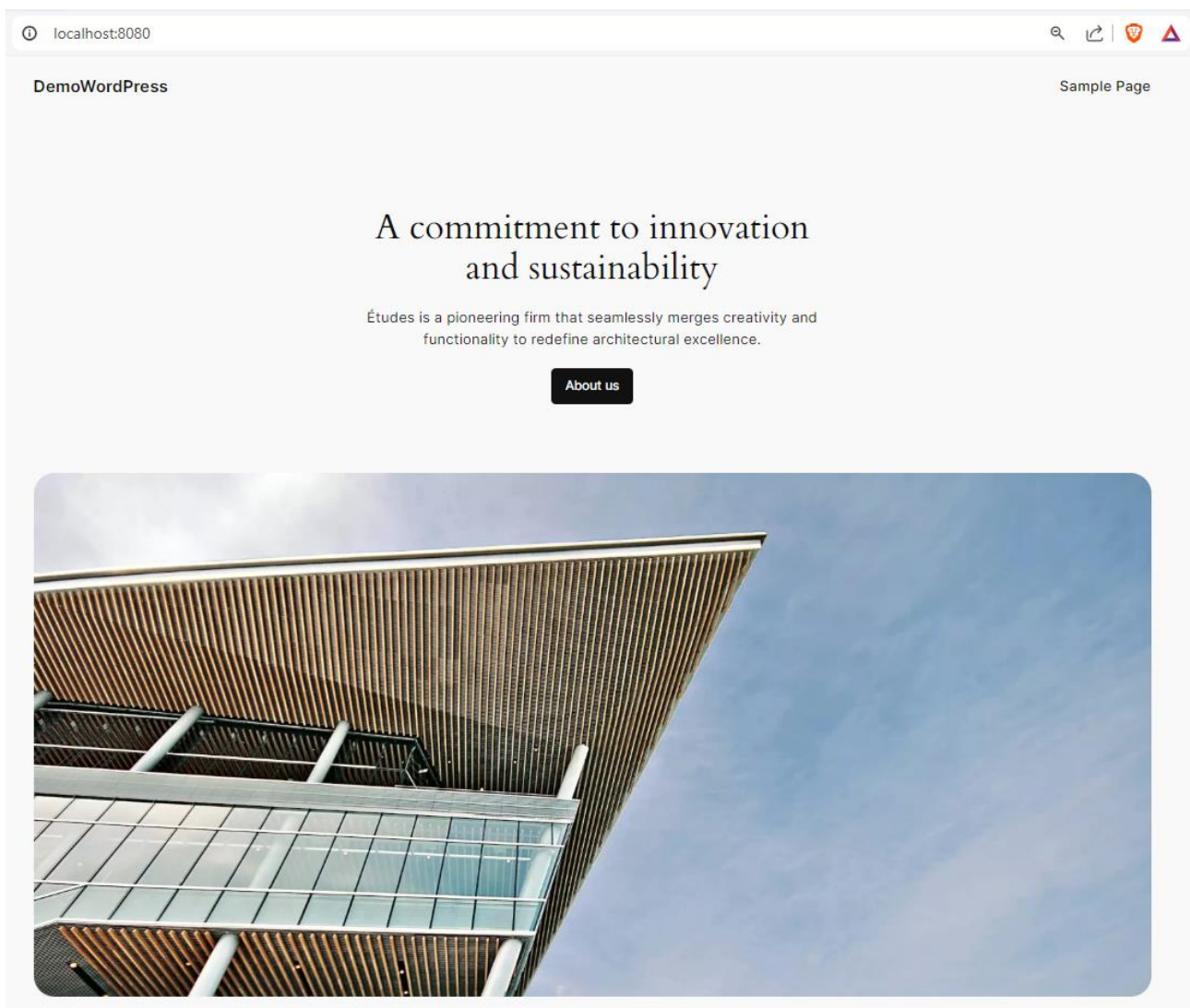
Hình 7. Điền thông tin website

- Thông báo install WordPress thành công



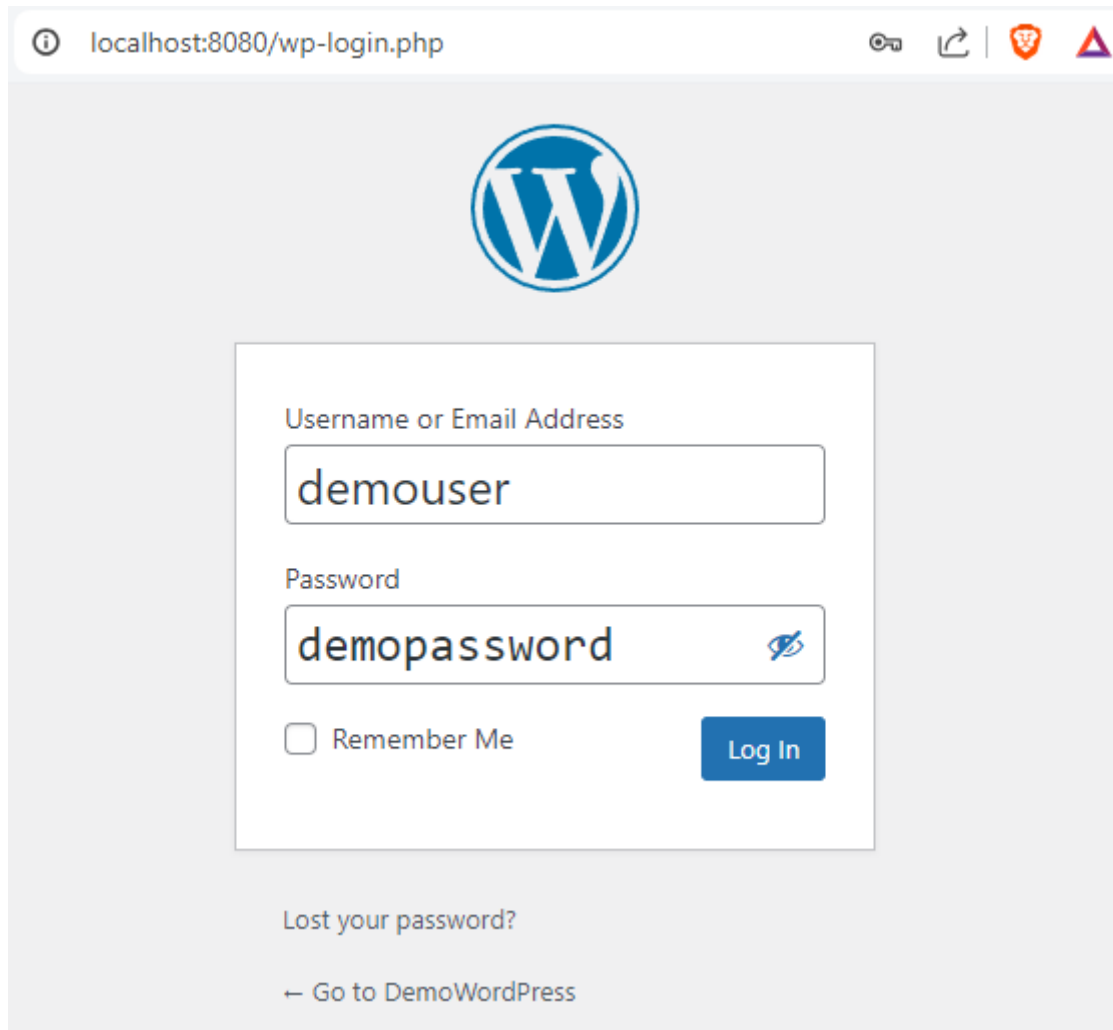
Hình 8. Thông báo install WordPress thành công

- Giao diện website khi truy cập [trang web](#).



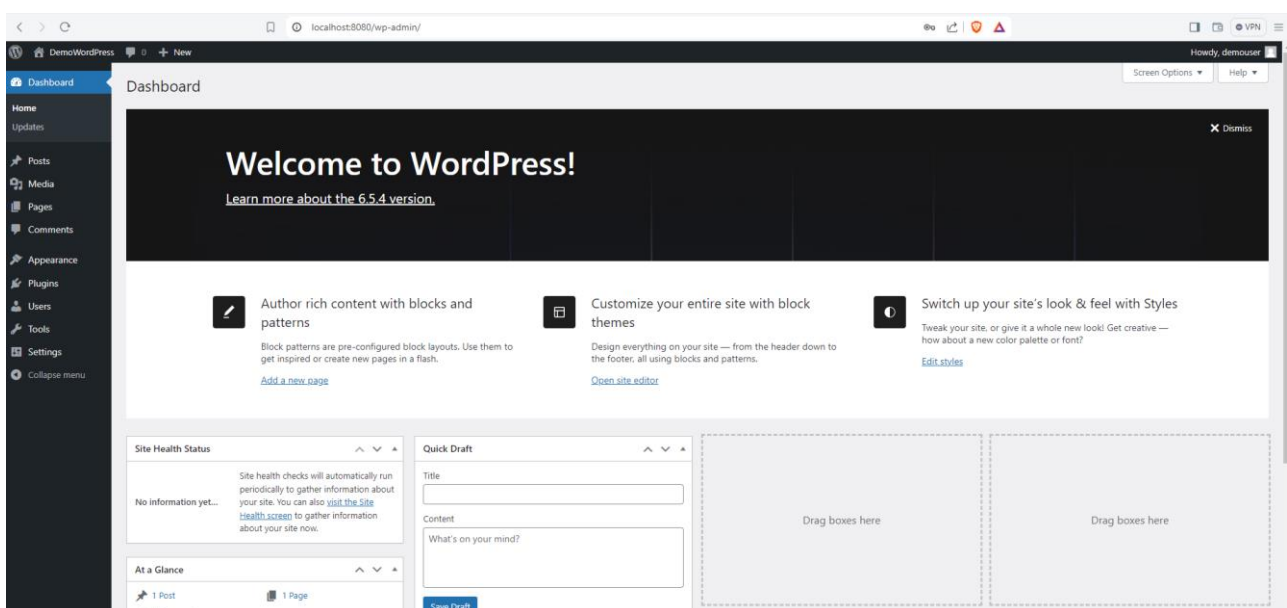
Hình 9. Giao diện website

- Truy cập vào [trang login](#) để đăng nhập



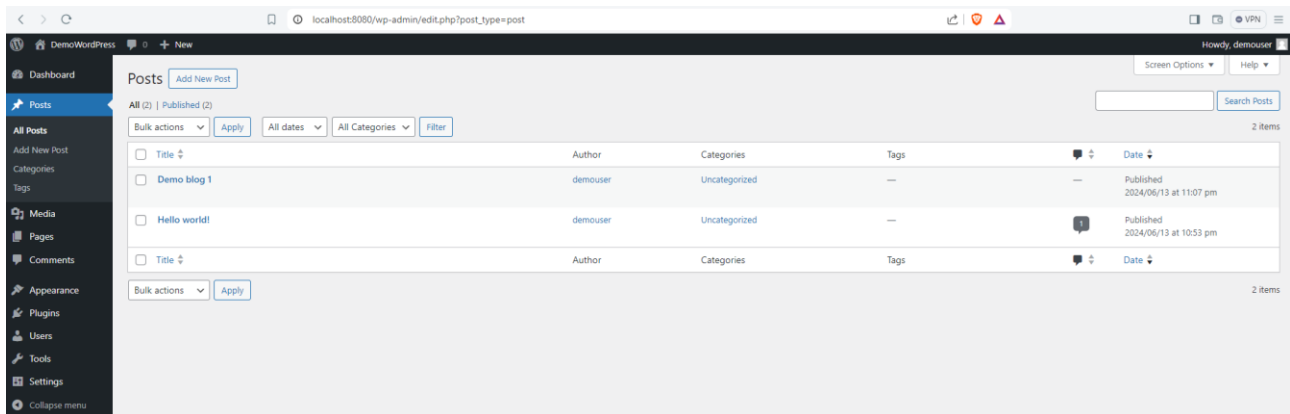
Hình 10. Giao diện trang login

- Giao diện trang admin



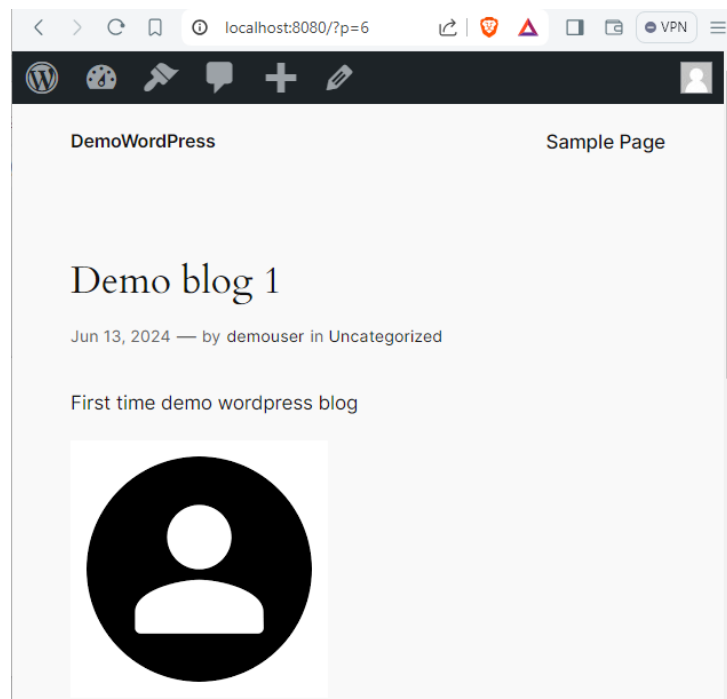
Hình 11. Giao diện trang login

- Giao diện trang quản lý các post



Hình 12. Giao diện trang quản lý các post

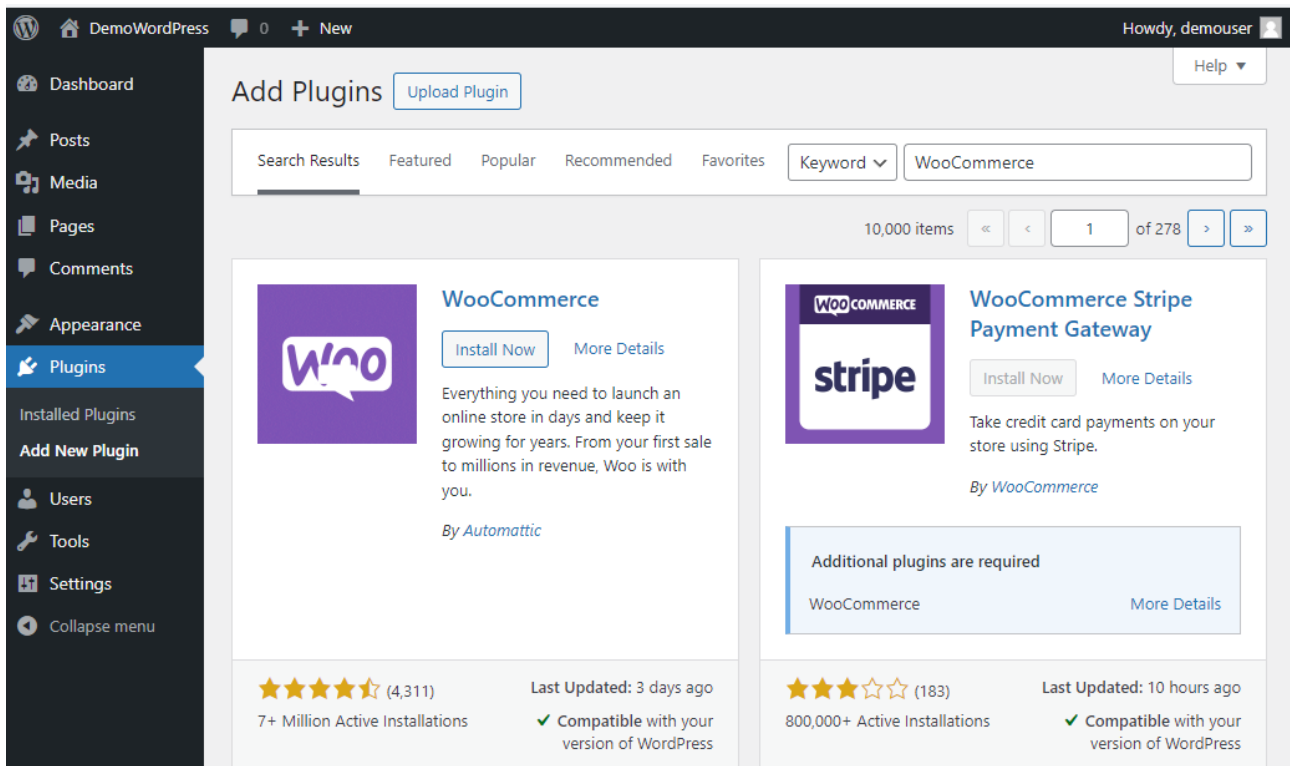
- Giao diện một trang blog



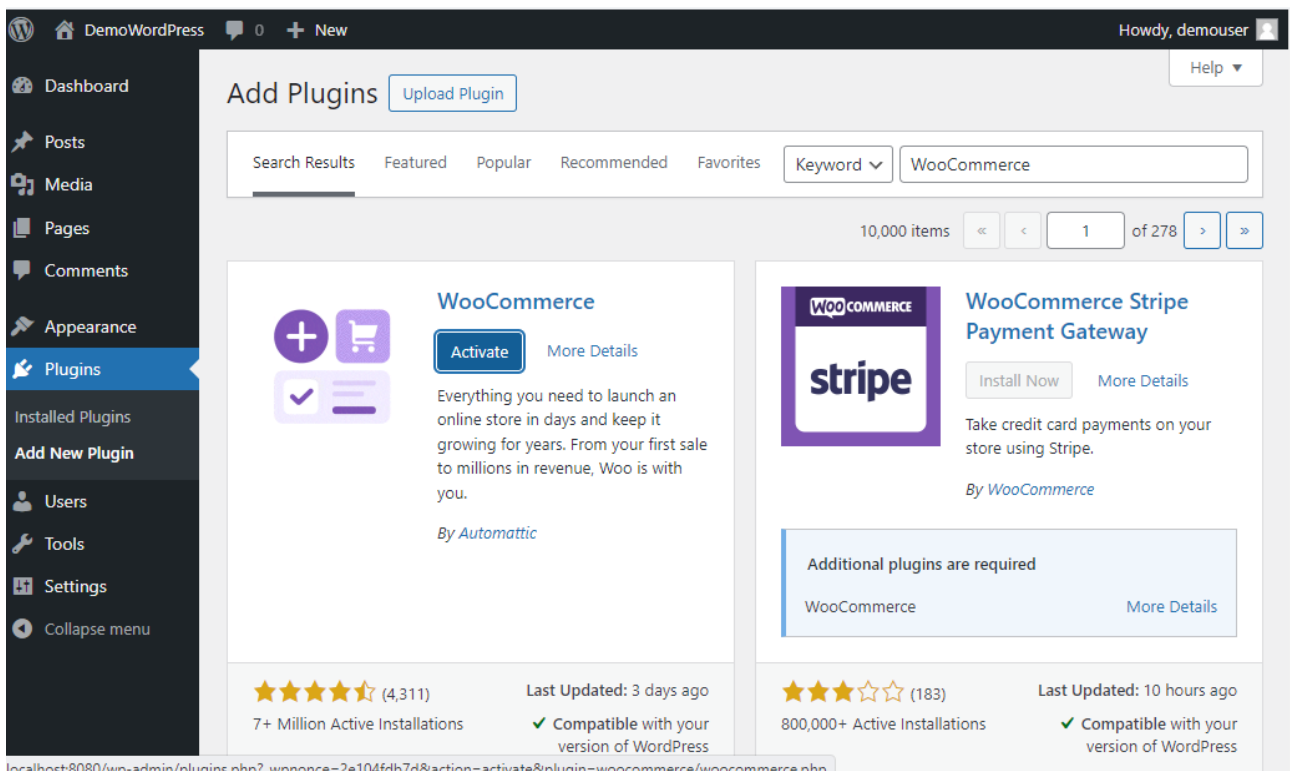
Hình 13. Giao diện một trang blog

4.3. Triển khai cài đặt E-Commerce (WooCommerce)

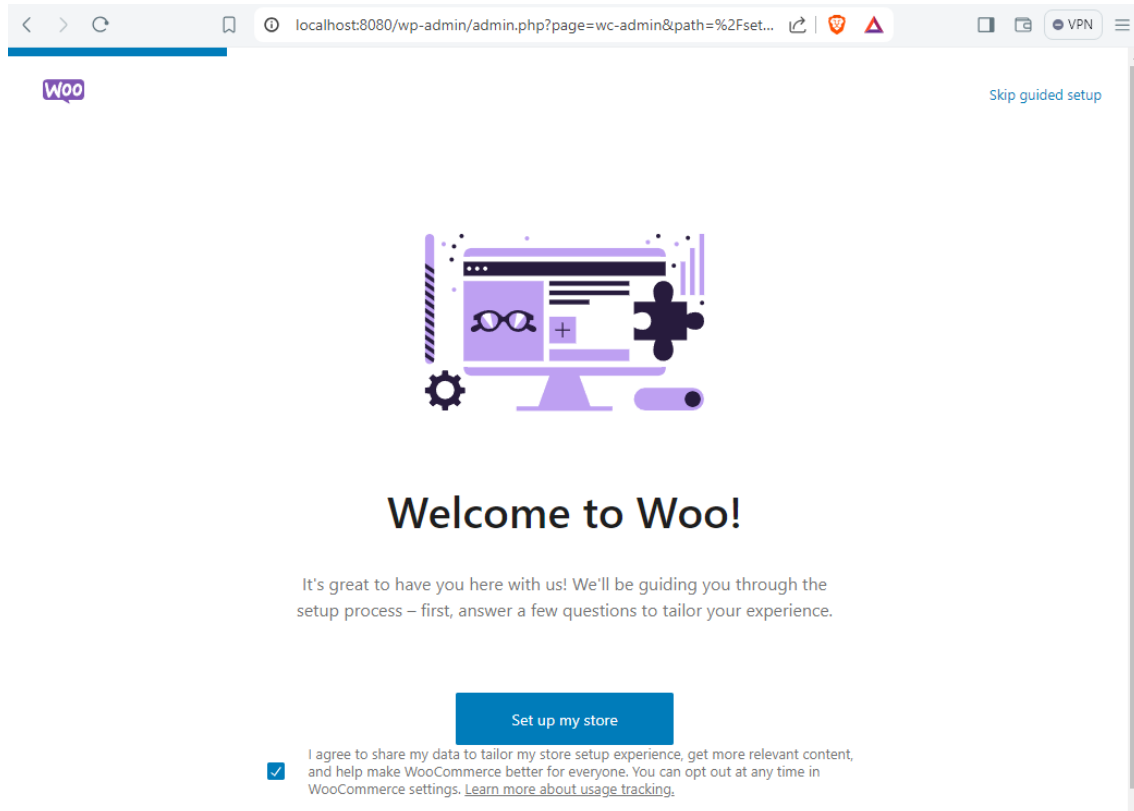
- Tại trang admin, chọn Add New Plugin ở mục Plugins. Sau đó gõ tìm kiếm WooCommerce và nhấn Install Now.



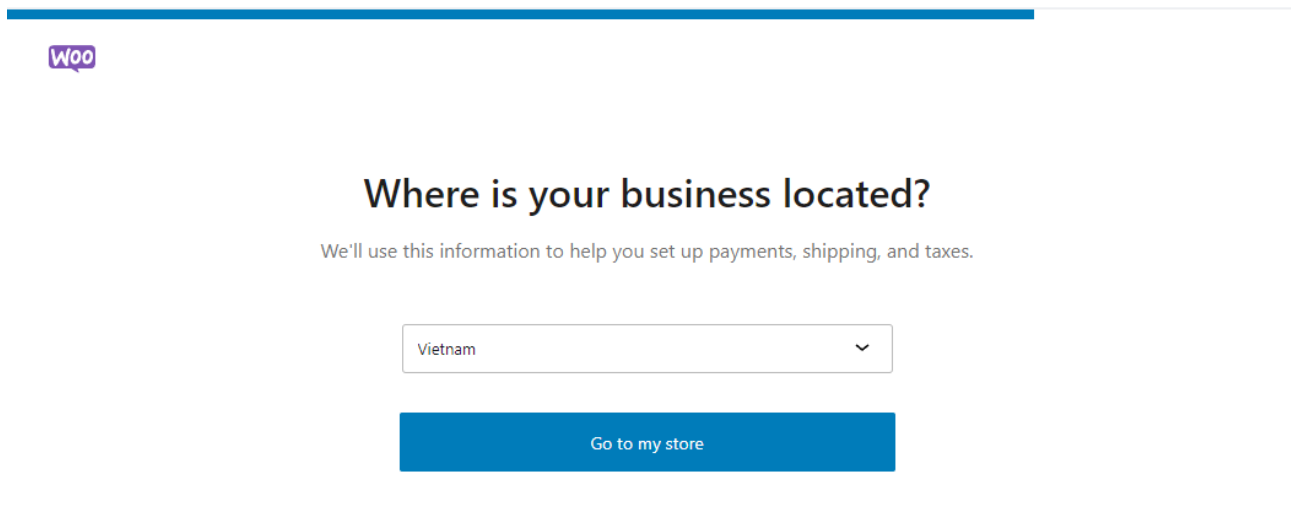
- Sau khi cài đặt thành công nhấn Active.



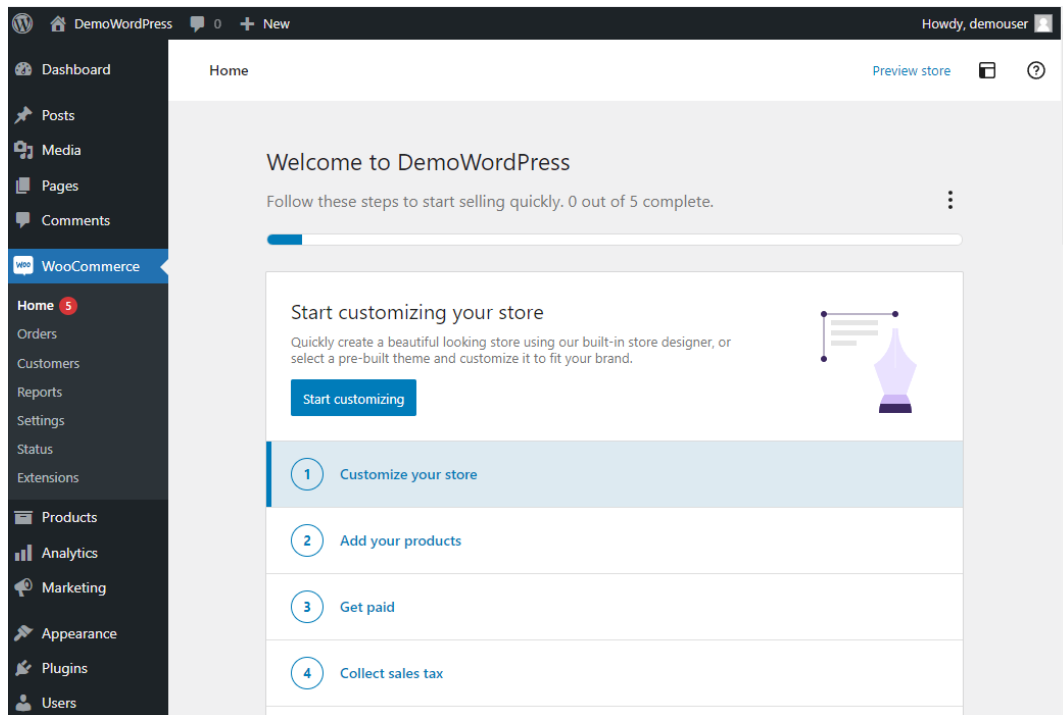
- Tiến hành setup WooCommerce, nhấn Set up my store hoặc skip guide setup.



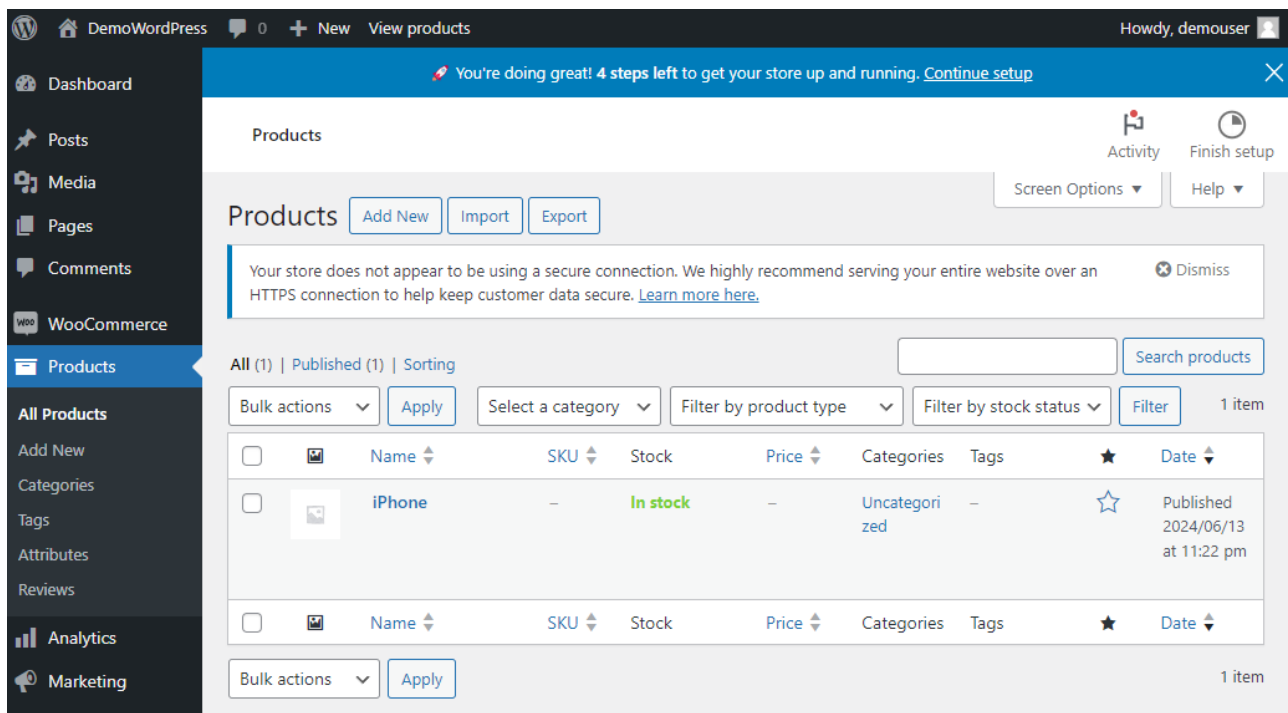
- Cuối cùng chọn vùng và nhấn Go to my store.



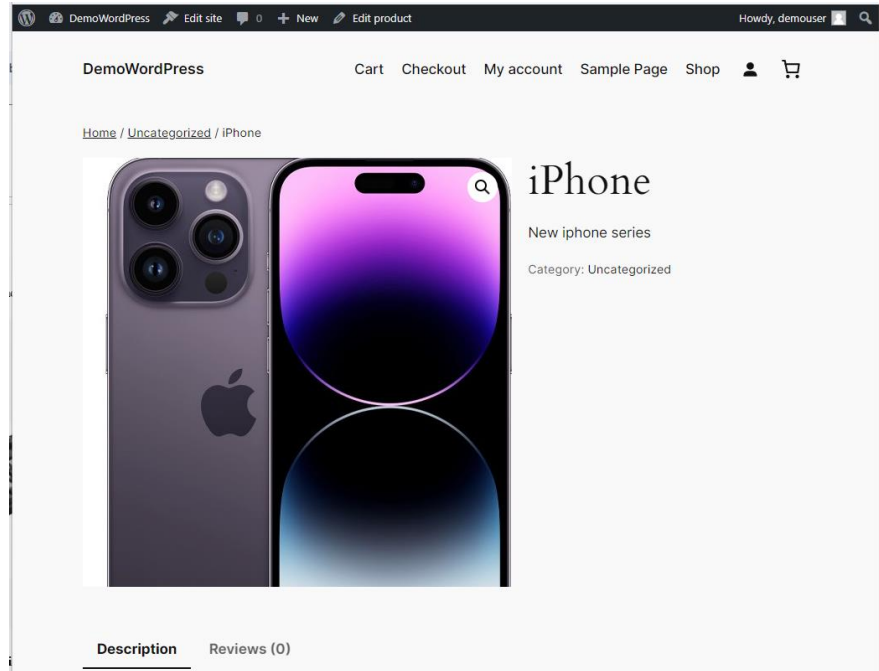
- Tại đây, có thể cá nhân hoá store theo sở thích



- Trang quản lý các sản phẩm trên website



- Giao diện hiển thị sản phẩm trên website



4.4. Triển khai CI/CD

- Clone repository [DemoBasicCICD](#) về máy tính cá nhân.
- Cấu hình file workflows tại `./.github/workflows/cicd.yml`

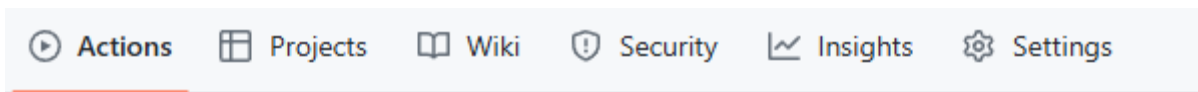
```
.github > workflows > ! cicd.yml
1  name: CI
2
3  on: [push, pull_request]
4
5  jobs:
6    build:
7      runs-on: ubuntu-latest
8
9      steps:
10     - name: Checkout repository
11       uses: actions/checkout@v2
12
13     - name: Set up Python
14       uses: actions/setup-python@v2
15       with:
16         python-version: '3.x'
17
18     - name: Install dependencies
19       run: |
20         python -m pip install --upgrade pip
21         if [ -f requirements.txt ]; then pip install -r requirements.txt; fi
22
23     - name: Run all Python script tests
24       run: |
25         python -m unittest discover -s tests
26
```

Hình 14. Cấu hình file workflows CI/CD

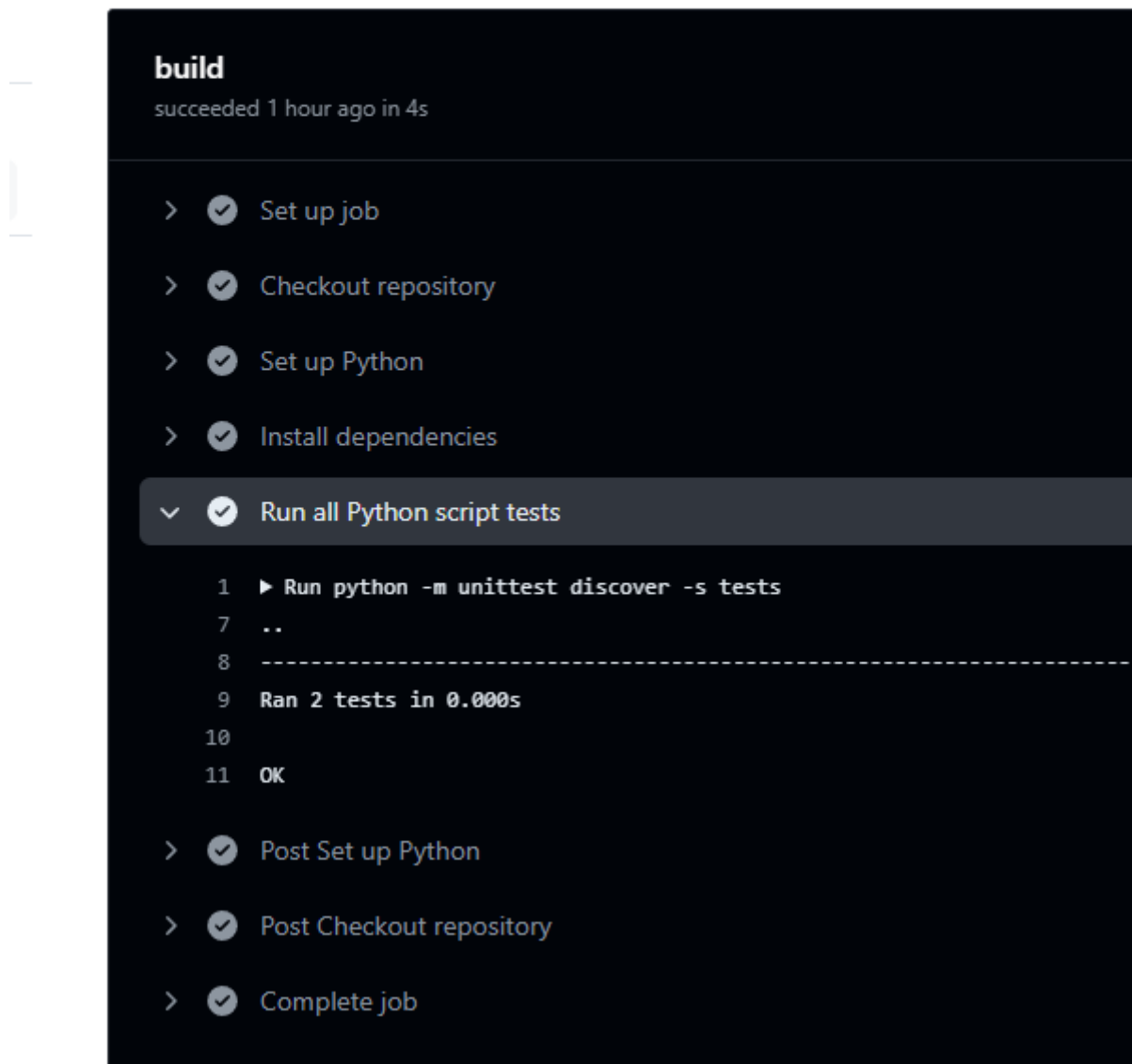
Trong đó, cấu hình bao gồm:

- name: tên của workflow
- on: sự kiện kích hoạt workflows
- jobs: định nghĩa các công việc cần thực hiện trong workflow
- build: tên của jobs
- runs-on: môi trường chạy jobs ubuntu-latest
- steps: định nghĩa các bước cần thực hiện trong job
- name: tên của bước hiện tại
- uses: sử dụng action để lấy mã nguồn từ repository về môi trường làm việc
- run: Các lệnh shell để cài đặt các dependencies (phụ thuộc) hoặc lệnh để kiểm tra mã nguồn

- Sau khi config thành công, mỗi lần push hoặc pull request sẽ tự động chạy workflows để kiểm tra.
- Nếu thành công vượt qua các bài test thì mục Actions trên repository sẽ hiển thị như sau:

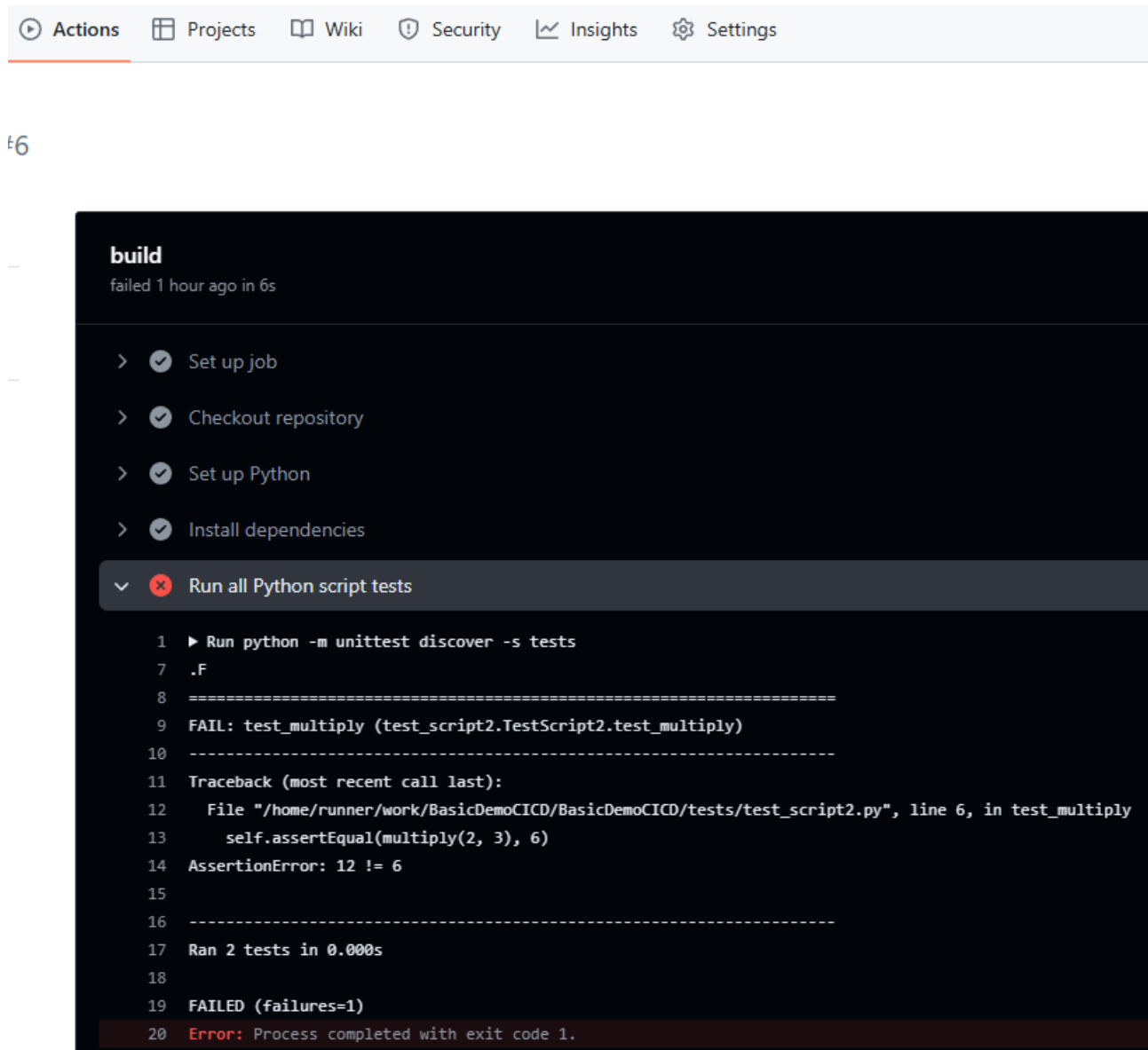


ript2, update CI to run tests #5



Hình 15. Trường hợp thành công

- Nếu không thành công vượt qua các bài test sẽ hiển thị như sau:



Hình 16. Trường hợp không thành công

V. TÀI LIỆU ĐÍNH KÈM

- Link github demo CMS & E-Commerce: [DemoWordPress](#)
- Link github demo CI/CD: [DemoBasicCICD](#)