### 1.1.5       Bài tập và thực hành

Bài tập 1. Sử dụng hosting và tên miền miễn phí

– Tạo một website đơn giản, gồm mã HTML, CSS, JavaScript.

– Lựa chọn một tên miền của riêng bạn. Ví dụ: *trangwebcuateo*

– Sử dụng ít nhất 2 dịch vụ lưu trữ web miễn phí, thực hiện đưa website lên Internet theo kiểu lưu trữ chia sẻ (shared hosting).

Ví dụ, có thể sử dụng dịch vụ miễn phí của các nhà cung cấp sau:

|  |  |  |  |
| --- | --- | --- | --- |
| pages.github.com | 000webhost.com | wordpress.com | blogspot.com |
| my.noip.com | my.freenom.com | somee.com | biz.nf |

Bài tập 2. Thiết lập và cấu hình shared hosting trên máy cục bộ.

Yêu cầu:

 – Thực hiện trên phần mềm Apache

–  Tạo trang web index.html với nội dung bất kì, để trang web ở vị trí phù hợp để có thể truy cập trang web dưới dạng http://127.0.0.1/, http://localhost/, http://127.0.0.1/index.html, http://localhost/index.html.  Tại sao không cần gõ tên tập tin index.html trong trường hợp (*http://127.0.0.1/,*http://localhost/*)*mà web server vẫn trả về trang index.html?

– Tạo website có tên là mywebsite, trong website tạo tập tin index.html, với nội dung bất kì.

– Cấu hình để có thể truy cập website dưới dạng http://127.0.0.1/mywebsite, http://localhost/ mywebsite

– Cấu hình để có thể truy cập website dưới dạng http://mywebsite.local (đặt tên là .local để báo là website đang chạy ở máy local, bạn có thể đặt tên là mywebsite.com, hoặc một tên miền khác tùy thích)

– Cấu hình để có thể truy cập website dưới dạng http://mywebsite.local:8888

– Để website ở ngoài thư mục htdocs của web server. Thực hiện cấu hình web server để có thể truy cập được trang web dưới dạng http://mywebsite.local

 – Tạo thêm website có tên là teowebsite, trong website tạo tập tin index.html, với nội dung bất kì.

– Cấu hình để có thể truy cập website dưới dạng http://teowebsite.local

(b) Thực hiện các yêu cầu như trong phần (a), tuy nhiên triển khai trên Nodejs hoặc Nginx

Bài tập 3. Tìm hiểu các thông số liên quan đến một web hosting?

– Dung lượng lưu trữ (disk space)

– Băng thông (bandwidth)

– Cơ chế bảo mật, sao lưu và phục hồi dữ liệu

– Phần mềm hỗ trợ thao tác với web hosting (upload/download)

– Khả năng tương thích với các: CMS, framework, ngôn ngữ lập trình (loại ngôn ngữ, phiên bản)

Bài tập 4. (tùy chọn). Thực hành, hoặc xem trên youtube cách đưa website lên Internet bằng các công cụ chuyên dùng, ví dụ: cPanel, Rsync, DirectAdmin, FTP (FileZilla, SmartFTP, CoreFTP).

[Gợi ý làm các bài tập]

Bài tập 1 (a). Tạo một website đơn giản, ví dụ tạo thư mục có tên là *websitecuteo*, tạo tập tin *index.html* trong thư mục *websitecuteo*, thêm nội dung sau cho tập tin *index.html*,

*<html>*

*<head> <style> h1 { color: red; } </style> </head>*

*<body>*

*<h1>It works! hi cu teo</h1>*

*<script> alert('hi cu Ti'); </script>*

*</body></html>*

Đưa website *websitecuteo* lên Internet, sử dụng dịch vụ lưu trữ web miễn phí của github:

(bạn có thể tham khảo các video trên youtube, hoặc đọc thêm tại đây: https://hackernoon.com/use-custom-domain-with-github-pages-2-straightforward-steps-cf561eee244f)

– Vào trang github.com, nếu bạn chưa có tài khoản trên hệ thống của github thì bấm vào Sign up để tạo một tài khoản, nếu có tài khoản rồi thì bấm vào Sign in để đăng nhập hệ thống.

– Sau khi đăng nhập vào hệ thống, bấm vào mục New (ở góc trên, bên trái) để tạo một thư mục (repository), chứa các tập tin mã nguồn của website. Trong github, repository được gọi là kho chứa. Ví dụ đặt tên cho repository là *websitecuteo.github.io* (phần tên miền github.io là không bắt buộc, bạn có thể bỏ đi, hoặc thay bằng tên bất kì). Tên thư mục này sẽ được dùng làm tên miền của website, hệ thống sẽ kiểm tra xem tên miền này đã có ai dùng chưa, nếu có người dùng rồi thì bạn cần đổi tên miền khác, ví dụ websitecuteo1.github.io.

– Sau khi nhập tên cho repository, ví dụ websitecuteo.github.io, bấm nút Create repository để tạo

– Bước tiếp theo sẽ tạo trang index.html cho website. Trong repository *websitecuteo.github.io*, chọn tab Code, bấm vào mục “creating a new file”. Đặt tên cho tập tin là index.html. Chép nội dung tập tin index.html ở phía trên vào cửa sổ “Edit new file”. Bấm Commit new file, để lưu lại nội dung tập tin index.html.

– Xuất bản website ra bên ngoài. Trong repository *websitecuteo.github.io*, chọn tab Settings, trong mục GitHub Pages, có dòng thông báo “Pages settings now has its own dedicated tab! Check it out here”, bạn bấm vào nút Check it out here để mở trang cấu hình mới. Trong mục Source, mục Branch, chọn Main, bấm nút Save. Bạn sẽ nhận được thông báo website đã xuất bản thành công ví dụ: “Your site is published at https://legiacong.github.io/websitecuteo.github.io/”.

– Bấm vào đường dẫn của website vừa được xuất bản để quan sát kết quả.

Xem thêm video: https://www.youtube.com/watch?v=WvgAsqZyevg

Bài tập 1 (b). Đưa website *websitecuteo* lên Internet, sử dụng dịch vụ lưu trữ web miễn phí của 000webhost.com

– Tạo thư mục websitecuteo

– Lên mạng tìm và tải tập tin hình ảnh bất kỳ (ví dụ con-vit.png), lưu vào trong thư mục websitecuteo

– Tạo tập tin index.html trong thư mục websitecuteo vừa tạo với nội dung sau:

*<html>*

*<head> <style> h1 { color: red; } </style> </head>*

*<body>*

*<h1>It works! hi cu teo</h1>*

*<img src="con-vit.png" alt="Con vịt">*

*<script> alert('hi cu Ti'); </script>*

*</body></html>*

– Tạo tài khoản và đăng nhập vào hệ thống 000webhost.com

– Nhập tên website, ví dụ: websitecuteo, nhớ lưu lại mật khẩu đăng nhập (ví dụ: vn@6JLZCAmUAktC2uYFK)

– Lựa chọn kiểu website sẽ tạo (Upload your site)

– Tải tập tin index.html vào thư mục public\_html (tải luôn tập tin con-vit.png vào thư mục này)

– Đăng nhập vào trang web để xem kết quả: https://websitecuteo.000webhostapp.com/

Bài tập 2. Thiết lập và cấu hình shared hosting trên máy cục bộ.

***Thực hiện trên Apache***

*Tạo trang web index.html với nội dung bất kì, để trang web ở vị trí phù hợp để có thể truy cập trang web dưới dạng http://127.0.0.1/, http://localhost/, http://127.0.0.1/index.html, http://localhost/index.html*

– Tải, cài đặt và chạy phần mềm Apache

– Tạo trang web index.html với nội dung sau,

[index.html]

<!DOCTYPE html>

<html>

            <head>

                        <title>Trang web cua Teo</title>

            </head>

            <body>

                        <h1>Trang mywebsite</h1>

            </body>

</html>

– Chép tập tin index.html vào thư mục htdocs của Apache

– Mở trình duyệt, nhập vào các đường dẫn *http://127.0.0.1/, http://localhost/, http://127.0.0.1/index.html, http://localhost/index.html,*nếu xuất hiện nội dung trang web trên trình duyệt là được.

*Tạo website có tên là mywebsite, trong website tạo tập tin index.html, với nội dung bất kì.*

– Trong ổ đĩa tùy ý, tạo thư mục mywebsite, trong thư mục mywebsite, tạo tập tin index.html, với nội dung là:

 [mywebsite/index.html]

*<!DOCTYPE html>*

*<html>*

*<head>*

*<title>Trang web cua Teo</title>*

*</head>*

*<body>*

*<h1>Trang mywebsite</h1>*

*</body>*

*</html>*

*Cấu hình để có thể truy cập website dưới dạng http://127.0.0.1/mywebsite, http://localhost/ mywebsite*

– Chép thư mục mywebsite vào trong thư mục htdocs. Chạy Apache, mở trình duyệt để kiểm tra các đường dẫn http://127.0.0.1/mywebsite, http://localhost/ mywebsite, nếu xuất hiện nội dung trang web là được.

*Cấu hình để có thể truy cập website dưới dạng http://mywebsite.local (đặt tên là .local để ngầm hiểu website đang chạy ở máy local, bạn có thể đặt tên là mywebsite.com, hoặc một tên miền khác tùy thích)*

– Để có thể truy cập trang web dưới dạng một tên miền thực thụ (mywebsite.local) trên máy cục bộ, bạn cần phải cấu hình máy tính cục bộ đóng 2 vai trò, như một web server và như một máy Name server của hệ thống DNS.

– Tập tin hosts trong Windows đóng vai trò như một bảng lưu các mẩu tin “tên miền – IP” của Name server, mở tập tin hosts, thêm dòng dữ liệu sau vào cuối tập tin, lưu lại và đóng tập tin. Đường dẫn của tập tin hosts: C:\Windows\System32\drivers\etc. Trong tập tin hosts dấu # ở đầu mỗi dòng là dấu chú thích, nó sẽ vô hiệu hóa khai báo tại dòng tương ứng.

*127.0.0.1 mywebsite.local*

– Sau khi khai báo dòng dữ liệu “*127.0.0.1 mywebsite.local*” trong tập tin hosts, nếu người dùng vào trình duyệt gõ URL http://mywebsite.local, máy cục bộ sẽ chuyển yêu cầu truy cập web của trình duyệt tới thư mục web root (document root) của phần mềm web server đang chạy trên máy cục bộ (cổng 80). Nếu trong web root đang có tập tin index.html thì trình duyệt sẽ nhận được nội dung trang web này.

– Để có thể cấu hình máy cục bộ trở thành một máy web server chạy ở chế độ shared hosting (cho phép để nhiều website cùng một máy server), thì bạn cần cấu hình trong tập tin *C:\Apache24\conf\extra\httpd-vhost.conf* hoặc trong tập tin *C:\Apache24\conf\httpd.conf.* Vì tập tin *httpd.conf* sẽ tham chiếu tới tập tin *httpd-vhost.conf*, nên bạn chỉ cần cấu hình ở 1 trong 2 tập tin. Nên cấu hình trong *httpd-vhost.conf* để dễ theo dõi, dễ quản lý.

– Mở tập tin *httpd-vhost.conf* bằng một text editor bất kỳ (Notepad, Sublime text, VS code), thêm các dòng khai báo sau, và lưu lại tập tin *httpd-vhost.conf*:

<VirtualHost \*:80>

    DocumentRoot "C:/Apache24/htdocs/mywebsite"

    ServerName mywebsite.local

</VirtualHost>

– Các dòng khai báo trên sẽ tạo ra một chỗ lưu trữ web trên một máy web server giả lập (ảo) (virtual host), lắng nghe trên cổng 80, thư mục chứa mã nguồn trang web là “mywebiste” và tên miền của máy server (cũng là tên miền của trang web) là mywebsite.local.

– Khi người dùng gõ http://mywebsite.local, trình duyệt sẽ thực hiện chuyển chuỗi trên thành địa chỉ IP. Trình duyệt sẽ tìm trong DNS cached trên máy cụ bộ và tìm ra được IP tương ứng là 127.0.0.1 (đọc lại phần về DNS để biết tại sao). Trình duyệt sẽ gửi request tới máy 127.0.0.1, web server nhận được yêu cầu, tham chiếu tới tập tin *httpd.conf > httpd-vhosts.conf* và trả về trang web tại thư mục htdocs/mywebsite.

– Mở cửa sổ Task manager, chọn tab Services, chuột phải vào dịch vụ Apache, chọn restart để khởi động lại Apache (giúp cập nhật các nội dung vừa thay đổi).

– Mở trình duyệt, nhập đường dẫn trang web http://mywebsite.local, để xem kết quả. Trang web trong website mywebsite sẽ được hiển thị trên trình duyệt.

*Cấu hình để có thể truy cập website dưới dạng http://mywebsite.local:8888*

– Thay đổi cổng lắng nghe mặc định của Apache từ 80 sang 8888. Mở tập tin *httpd.conf*, tìm tới mục có chữ Listen 80, đổi giá trị 80 thành 8888, lưu tập ti httpd.conf, restart lại dịch vụ Apache.

–  Mở trình duyệt, nhập đường dẫn http://localhost:8888 để hiển thị trang mặc định trong web root. Trang index.html trong web root sẽ được hiển thị trên trình duyệt.

– Tuy nhiên, nếu truy cập trang *http://mywebsite.local:8888*sẽ không hiển thị được, lý do là trong phần cấu hình shared hosting, vẫn để cổng mặc định là 80. Bạn cần sửa lại như sau:

<VirtualHost \*:8888>

    DocumentRoot "C:/Apache24/htdocs/mywebsite"

    ServerName mywebsite.local

</VirtualHost>

– Lưu lại tập tin *httpd-vhosts.conf,*restart lại dịch vụ Apache, mở lại trang [*http://mywebsite.local:8888*](http://mywebsite.local:8888/)trên trình duyệt sẽ thấy trang web trong mywebsite được hiển thị.

*Để website ở ngoài thư mục htdocs của web server. Thực hiện cấu hình web server để có thể truy cập được trang web dưới dạng http://mywebsite.local*

– Cắt website mywebsite ra khỏi htdocs, để ở ổ đĩa D:\ chẳng hạn

– Sửa lại trong <VirtualHost ….> từ

<VirtualHost \*:8888>

    DocumentRoot " C:/Apache24/htdocs/mywebsite "

    ServerName mywebsite.local

</VirtualHost>

Thành,

<VirtualHost \*:8888>

    DocumentRoot "D:/mywebsite"

    ServerName mywebsite.local

</VirtualHost>

– Lưu lại tập tin *httpd-vhosts.conf,*restart lại dịch vụ Apache, mở lại trang *http://mywebsite.local:8888*trên trình duyệt sẽ thấy trang web trong mywebsite không được hiển thị mà có thông báo lỗi, hoặc hiển thị trang trong web root.

*Forbidden*

*You don't have permission to …..*

– Nếu web server hiển thị trang trong web root, thì bạn mở tập tin httpd.conf, bỏ dấu chú thích ở dòng có nội dung: *Include….*

*# Virtual hosts*

Include conf/extra/httpd-vhosts.conf

– Vì thư mục D:/mywebsite chưa được phân quyền, nên có thể gây ra lỗi truy cập. Thêm dòng lệnh sau để phân quyền truy cập vào thư mục D:/mywebsite:

Đoạn mã phân quyền,

<Directory "D:/mywebsite">

        Options Indexes FollowSymLinks MultiViews

        AllowOverride all

        Require all granted

    </Directory>

Kết quả là,

<VirtualHost \*:8888>

    DocumentRoot "D:/mywebsite"

    ServerName mywebsite.local

    <Directory "D:/mywebsite">

        Options Indexes FollowSymLinks MultiViews

        AllowOverride all

        Require all granted

    </Directory>

</VirtualHost>

– Lưu lại tập tin *httpd-vhosts.conf,*restart lại dịch vụ Apache, mở lại trang *http://mywebsite.local:8888*trên trình duyệt sẽ thấy nội dung trang web trong mywebsite.

*Tạo thêm website có tên là teowebsite, trong website tạo tập tin index.html, với nội dung bất kì. Cấu hình để có thể truy cập website dưới dạng http://teowebsite.local*

– Cách làm tương tự như website mywebsite

– Thêm một khai báo *127.0.0.1 teowebsite.local*trong tập tin hosts

– Thêm một khai báo trong tập tin *httpd-vhost.conf*,

<VirtualHost \*:8888>

    DocumentRoot "D:/teowebsite"

    ServerName teowebsite.local

    <Directory "D:/teowebsite">

        Options Indexes FollowSymLinks MultiViews

        AllowOverride all

        Require all granted

    </Directory>

</VirtualHost>

***Các lỗi thường xảy ra với Apache và cách sửa***

(1) Lỗi do cồng 80 (http) và cổng 443 (https) đã có dịch vụ hoặc chương trình khác sử dụng. Cách sửa: đổi cổng lắng nghe mặc định của Apache thành các cổng khác 80 và 443 hoặc vào mục Services của Task manager tắt các dịch vụ/chương trình đang chạy mà có sử dụng 2 cổng này.

Một số dịch vụ/chương trình hay tranh chấp cổng với Apache:

– World Wide Web Publishing Service (WWW-Publishing Service)

– Web Deployment Agent Service

– Skype

– SQL Server Reporting Service(MSSQLSERVER)

– VMware Workstation

– IIS

(2) Lỗi do chưa đăng ký mục tin trong tập tin hosts.

(3) Lỗi do chưa khai báo tên miền trong phần cấu hình virtual host (httpd-vhost.conf).

### 1.1.6       Câu hỏi ôn tập

Câu 1. Cái nào dưới đây không phải là loại dịch vụ lưu trữ website?

A. Lưu trữ chia sẻ (shared hosting), lưu trữ trên đám mây (cloud hosting)

B. Lưu trữ trên máy chủ ảo dùng riêng (virtual private server hosting)

C. Lưu trữ trên máy chủ dùng riêng (dedicated server hosting)

D. Lưu trữ trên máy chủ của VNNIC (VNNIC server hosting)

Câu 2. Để có một website trên không gian web, cần làm một số việc, trong các việc dưới đây, cái nào không cần thiết?

A. Lập trình website

B. Đăng ký tên miền

C. Cấu hình tập tin hosts trên máy cục bộ (Windows)

D. Đưa website lên web server

Câu 3. What kind of web hosting service this description refered?

 “One's website is placed on the same server as many other sites, ranging from a few sites to hundreds of websites. Typically, all domains may share a common pool of server resources, such as RAM and the CPU”.

A. cloud hosting

B. VPS hosting

C. dedicated server hosting

D. shared hosting

Câu 4. Your \_\_\_\_\_\_ is the amount of data transfer when your website is acceded. You use \_\_\_\_\_\_  through traffic generation, email, uploads, and downloads. Some hosting plans offer unlimited \_\_\_\_\_\_, but you should choose the highest amount necessary for your requirements.

A. download

B. upload

C. bandwidth

D. up time

Câu 5. With\_\_\_\_\_\_\_, you have a global network of servers all working together. That means if one server goes down, the other servers can pick up the load. It also allows you to pay for only the resources you're using, and you can scale up or down immediately.

A. cloud hosting

B. VPS hosting

C. dedicated server hosting

D. shared hosting