Web Lab 1: Apache Web Server Installation & Maintenance

Checkpoint 1: Apache web server installation and configuration
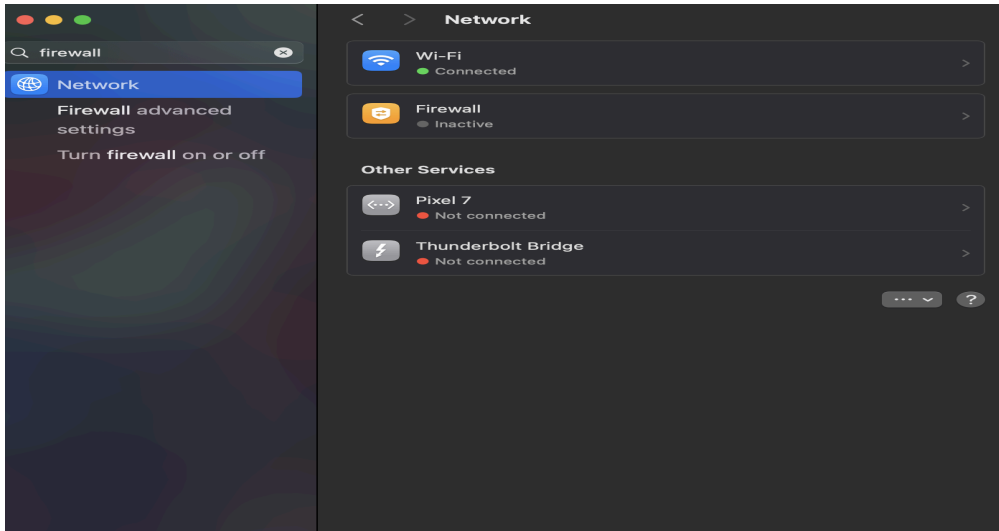
Installing Apache on my System:

      As i am using mac, it has apache installed within the homebrew package.

      I ran this command: which httpd so that i can find if it is installed or not.

      I got this output: /opt/homebrew/bin/httpd
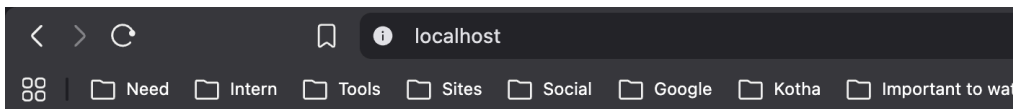
Adjusting the firewall:

      I have adjusted the firewall to restrict usage

Checking my webserver:

      I have run this command : sudo apachectl start

      And I opened `http:// localhost` in my browser.

It works!

Checkpoint 2: Setting Up a Virtual Hosts

Creating the directory for example.com :
>sudo mkdir -p /opt/homebrew/var/www/example.com

Assigning ownership of the directory to self
>sudo chown -R $(whoami):staff /opt/homebrew/var/www/

Creating a sample index.html page using nano:
>nano /opt/homebrew/var/www/example.com/index.html

I have added the following HTML code in index.html:

>\<html>
>\<head>
>\<title>Welcome to Example.com!\</title>
>\</head>
>\<body>
>\<p>This is the Apache test page for example.com.\</p>
>\<p>Tested by shakera jannat.\</p>
>\</body>
>\</html>

Enabling virtual host in Apache

sudo nano /opt/homebrew/etc/httpd/httpd.conf

Uncomment :#Include /opt/homebrew/etc/httpd/extra/httpd-vhosts.conf

Configuring Virtual Host

sudo nano /opt/homebrew/etc/httpd/extra/httpd-vhosts.conf

Setting up the domain for example.com

>\<VirtualHost *:80>
>   ServerName example.com
>   DocumentRoot "/opt/homebrew/var/www/example.com"
>   ErrorLog "/opt/homebrew/var/logs/example-error_log"
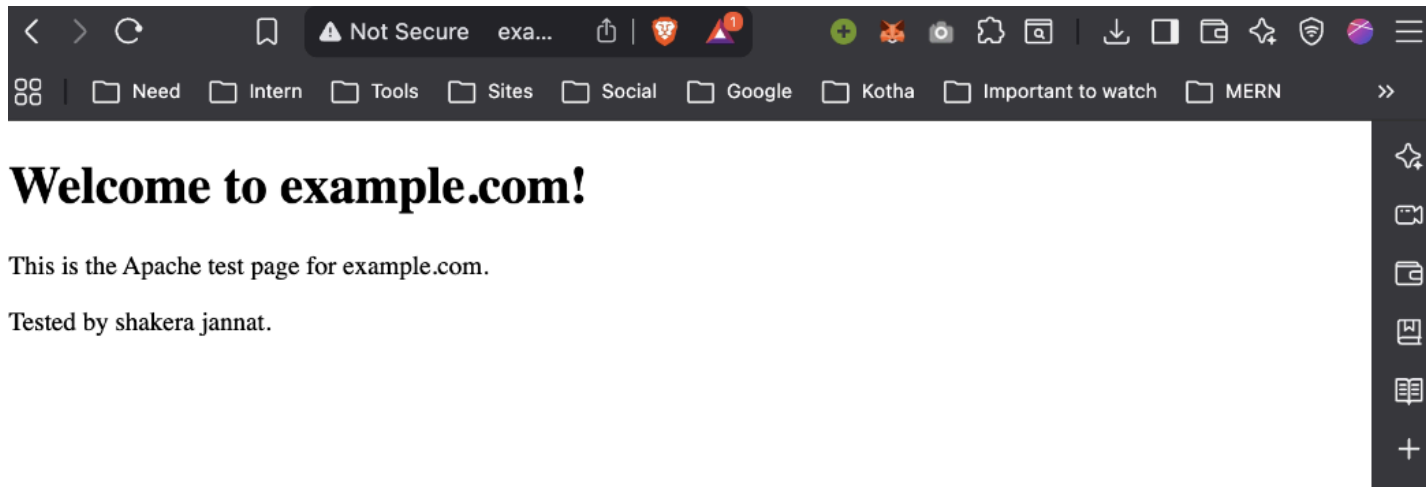>   CustomLog "/opt/homebrew/var/logs/example-access_log" common
>\</VirtualHost>

Adding host

sudo nano /etc/hosts

127.0.0.1 example.com

Restarting the Apache Server
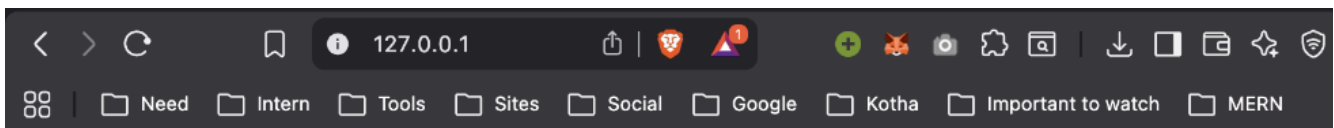
sudo apachectl restart

Welcome to example.com!

This is the Apache test page for example.com.

Tested by shakera jannat.

Checkpoint 3: Try navigating to http://webserverlab.com, observe what happens. Think about what is happening. Try to navigate to http://127.0.0.1. What happened and why?

When I navigate to http://webserver.com it checks my local DNS list to check if the url is listed by any ip address. Since i have listed this domain using 127.0.0.1 ip so it will quickly access this ip then the request goes to the Apache (port: 80). Apache checks it config files and matches the server name and serves it

If I try to navigate to http://127.0.0.1 the browser directly sends a request to my loopback addresses. Apache receives the request. Since this time no domain name is explicitly mentioned, Apache will serve the first matching which is my webserverlab.com. This is usually whichever <VirtualHost> block appears first in my vhosts.conf file.
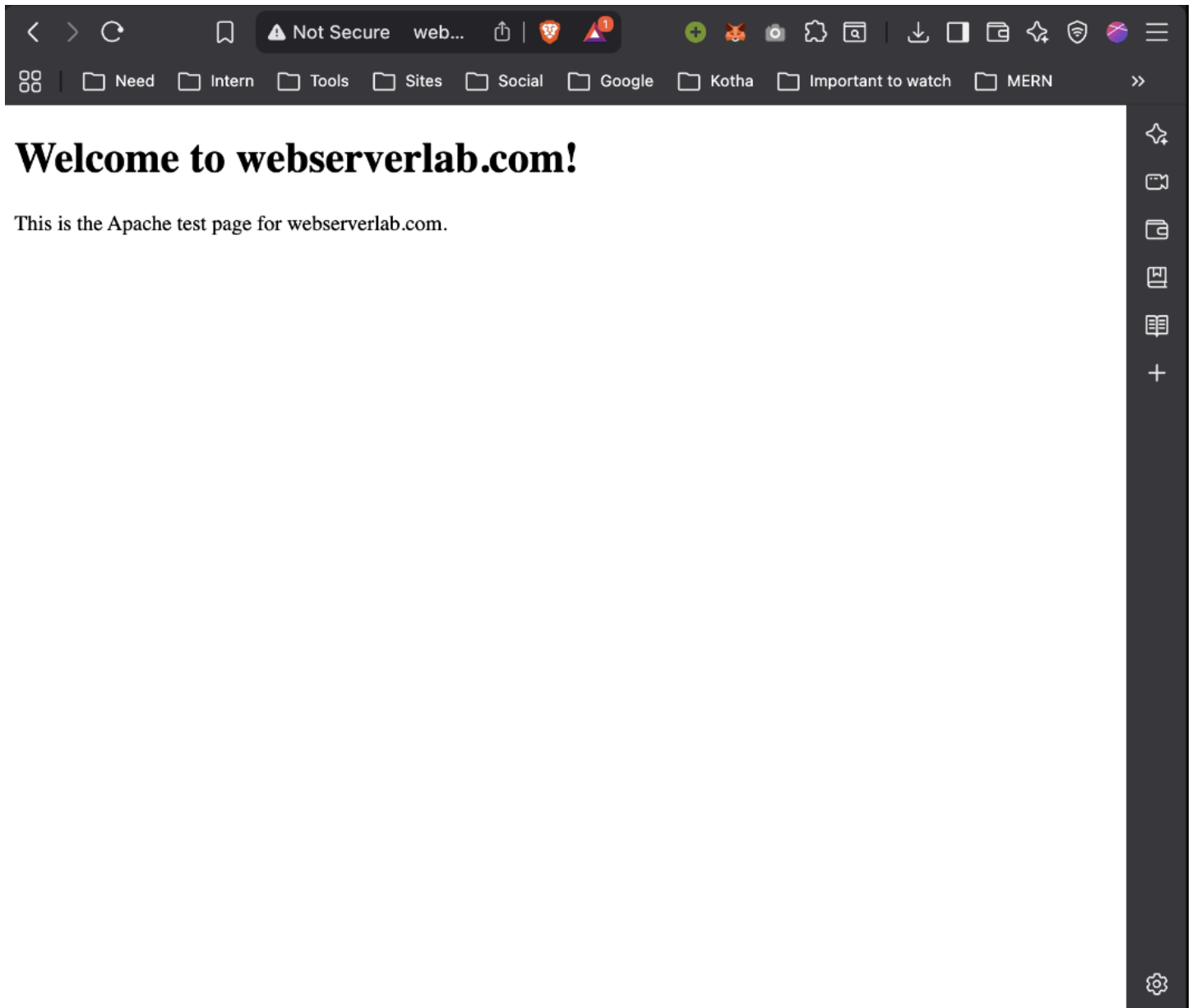

Welcome to webserverlab.com!

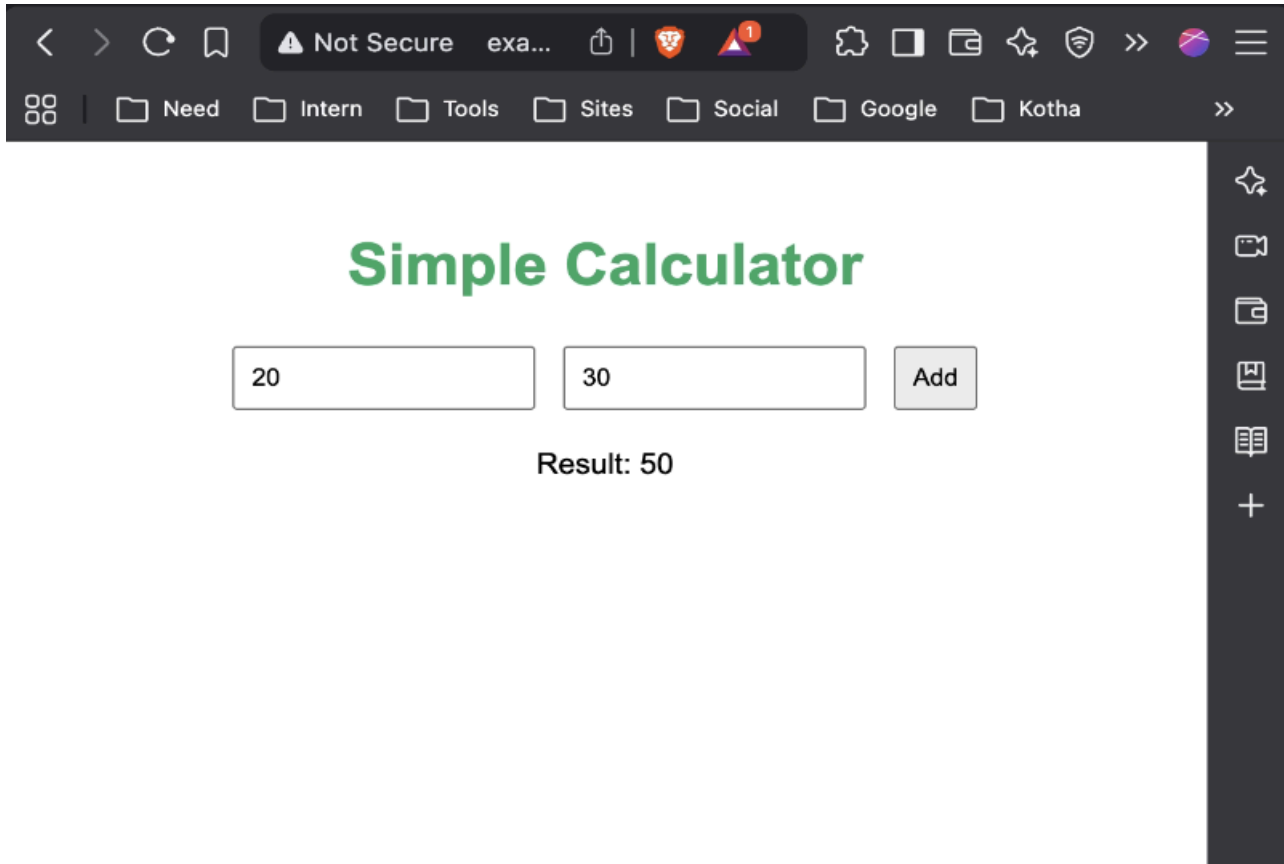This is the Apache test page for webserverlab.com.

Checkpoint 4: Setting Up multiple virtual hosts

I have repeated the same process again to create another host for webserverlab.com , and it runs as expected.
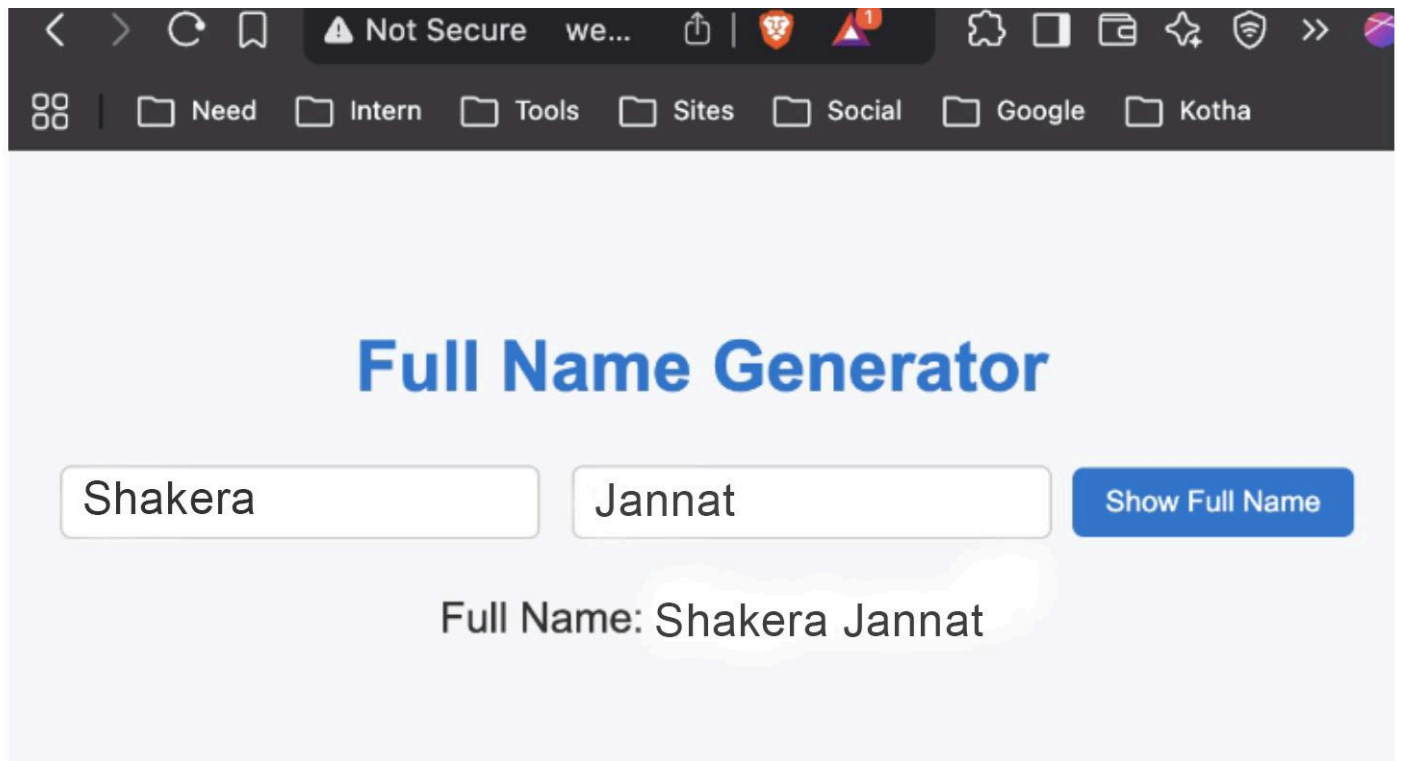
Checkpoint 5: Hosting a Dynamic Website using HTML and Javascript

I have added a simple calculator on example.com that takes two inputs from the user and performs addition and shows the result.

I have added a simple html website on webserverlab.com that takes two inputs (first name and last name) from the user and generates the full name and shows the result.

Full Name Generator

Shakera

Jannat

Show Full Name

Full Name: Shakera Jannat

Snapshots:

```
UW PICO 5.09                                          File: /opt/homebrew/etc/httpd/extra/httpd-vhosts.conf

# Virtual Hosts
#
# Required modules: mod_log_config

# If you want to maintain multiple domains/hostnames on your
# machine you can setup VirtualHost containers for them. Most configurations
# use only name-based virtual hosts so the server doesn't need to worry about
# IP addresses. This is indicated by the asterisks in the directives below.
#
# Please see the documentation at
# <URL:http://httpd.apache.org/docs/2.4/vhosts/>
# for further details before you try to setup virtual hosts.
#
# You may use the command line option '-S' to verify your virtual host
# configuration.

#
# VirtualHost example:
# Almost any Apache directive may go into a VirtualHost container.
# The first VirtualHost section is used for all requests that do not
# match a ServerName or ServerAlias in any <VirtualHost> block.
#
<VirtualHost *:8080>
    ServerAdmin webmaster@dummy-host.example.com
    DocumentRoot "/opt/homebrew/opt/httpd/docs/dummy-host.example.com"
    ServerName dummy-host.example.com
    ServerAlias www.dummy-host.example.com
    ErrorLog "/opt/homebrew/var/log/httpd/dummy-host.example.com-error_log"
    CustomLog "/opt/homebrew/var/log/httpd/dummy-host.example.com-access_log" common
</VirtualHost>

<VirtualHost *:8080>
    ServerAdmin webmaster@dummy-host2.example.com
    DocumentRoot "/opt/homebrew/opt/httpd/docs/dummy-host2.example.com"
    ServerName dummy-host2.example.com
    ErrorLog "/opt/homebrew/var/log/httpd/dummy-host2.example.com-error_log"
    CustomLog "/opt/homebrew/var/log/httpd/dummy-host2.example.com-access_log" common
</VirtualHost>
<VirtualHost *:8080>
    ServerAdmin webmaster@example.com
    DocumentRoot "/usr/local/var/www/example.com/html"
    ServerName example.com
    ErrorLog "/private/var/log/apache2/example.com-error_log"
    CustomLog "/private/var/log/apache2/example.com-access_log" common
</VirtualHost>
```