

EXPIRIMENT-6:

Aim: Installation and configuration of common software frame works such as Laravel.
(Student should acquire the capability to install and configure a modern framework)

Solution :-

Laravel

Laravel is an open-source PHP framework, which is robust and easy to understand. It follows a model-view-controller design pattern. Laravel reuses the existing components of different frameworks which helps in creating a web application. The web application thus designed is more structured and pragmatic.

Laravel offers a rich set of functionalities which incorporates the basic features of PHP frameworks like CodeIgniter, Yii and other programming languages like Ruby on Rails. Laravel has a very rich set of features which will boost the speed of web development.

If you are familiar with Core PHP and Advanced PHP, Laravel will make your task easier. It saves a lot time if you are planning to develop a website from scratch. Moreover, a website built in Laravel is secure and prevents several web attacks.

Composer

Composer is a tool which includes all the dependencies and libraries. It allows a user to create a project with respect to the mentioned framework (for example, those used in Laravel installation). Third party libraries can be installed easily with help of composer.

All the dependencies are noted in composer.json file which is placed in the source folder.

PHP

PHP is a server-side scripting language. that is used to develop Static websites or Dynamic websites or Web applications. PHP stands for Hypertext Pre-processor, that earlier stood for Personal Home Pages.

PHP scripts can only be interpreted on a server that has PHP installed.

The client computers accessing the PHP scripts require a web browser only.

A PHP file contains PHP tags and ends with the extension “.php”.

Apache

Apache HTTP Server is a free and open-source web server that delivers web content through the internet. It is commonly referred to as Apache and after development, it quickly became the most popular HTTP client on the web. It's widely thought that Apache gets its name from

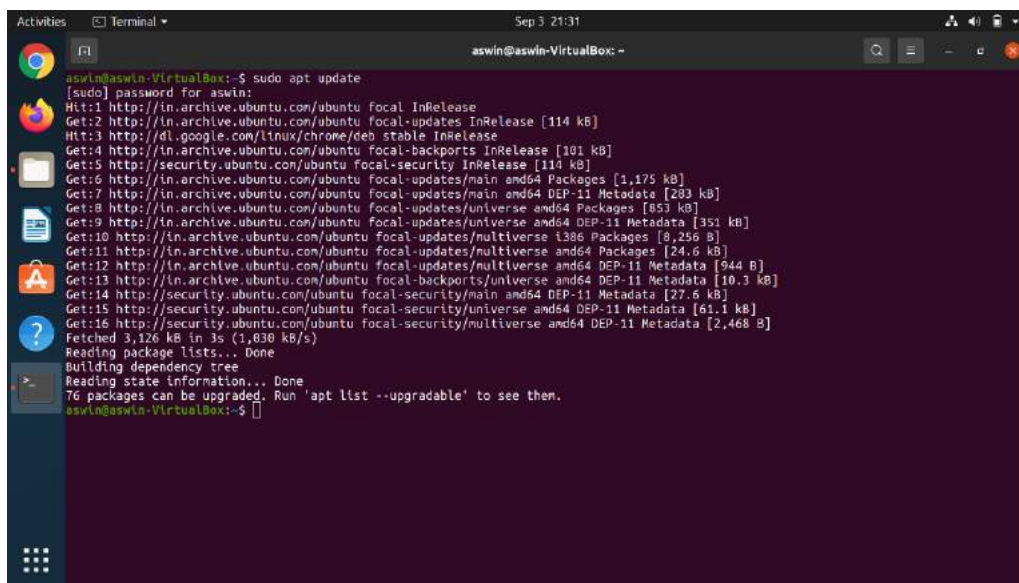
its development history and process of improvement through applied patches and modules but that was corrected back in 2000. It was revealed that the name originated from the respect of the Native American tribe for its resiliency and durability.

Now, before we get too in depth on Apache, we should first go over what a web application is and the standard architecture usually found in web apps.

SETUP LARAVEL ON UBUNTU WITH APACHE

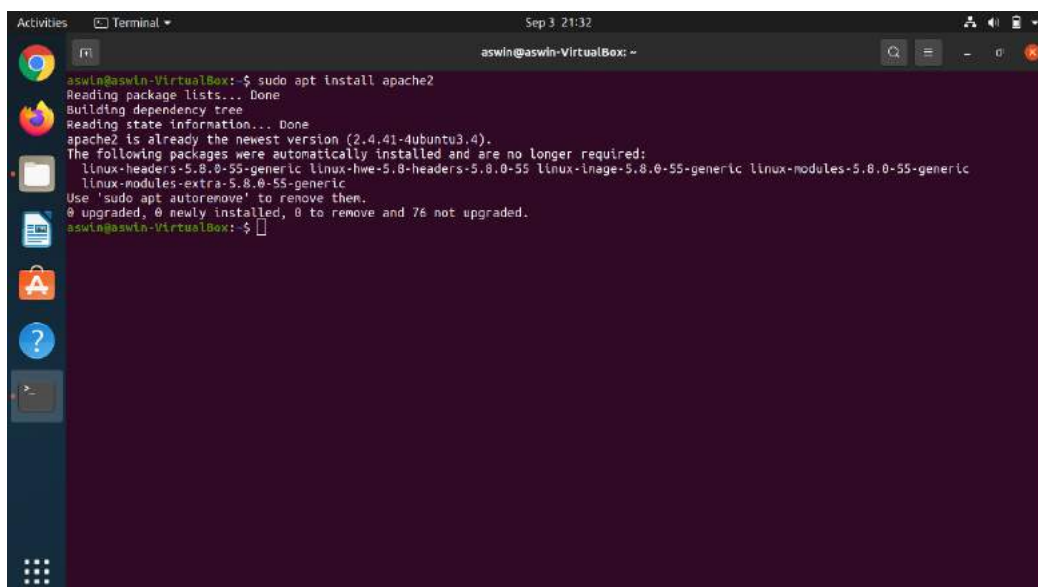
Step-1: Install Apache Web Server

- Let's open up a Terminal and do first thing first update your package list using Sudo apt update command.



```
aswin@aswin-VirtualBox:~$ sudo apt update
[sudo] password for aswin:
Hit:1 http://in.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://in.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Hit:3 http://dl.google.com/linux/chrome/deb stable InRelease
Get:4 http://in.archive.ubuntu.com/ubuntu focal-backports InRelease [101 kB]
Get:5 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [1,175 kB]
Get:7 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 DEP-11 Metadata [283 kB]
Get:8 http://in.archive.ubuntu.com/ubuntu focal-updates/universe amd64 Packages [853 kB]
Get:9 http://in.archive.ubuntu.com/ubuntu focal-updates/universe amd64 DEP-11 Metadata [351 kB]
Get:10 http://in.archive.ubuntu.com/ubuntu focal-updates/multiverse i386 Packages [8,256 B]
Get:11 http://in.archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 Packages [24.6 kB]
Get:12 http://in.archive.ubuntu.com/ubuntu focal-updates/multiverse amd64 DEP-11 Metadata [944 B]
Get:13 http://in.archive.ubuntu.com/ubuntu focal-backports/universe amd64 DEP-11 Metadata [10.3 kB]
Get:14 http://security.ubuntu.com/ubuntu focal-security/main amd64 DEP-11 Metadata [27.6 kB]
Get:15 http://security.ubuntu.com/ubuntu focal-security/universe amd64 DEP-11 Metadata [61.1 kB]
Get:16 http://security.ubuntu.com/ubuntu focal-security/multiverse amd64 DEP-11 Metadata [2,468 B]
Fetched 3,126 kB in 3s (1,030 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
76 packages can be upgraded. Run 'apt list --upgradable' to see them.
aswin@aswin-VirtualBox:~$
```

- After updating your package list install apache webservice. So, go ahead and type sudo apt install apache2 then hit the enter key. Press y key to proceed. You can also setup laravel with Nginx instead of the apache web server.

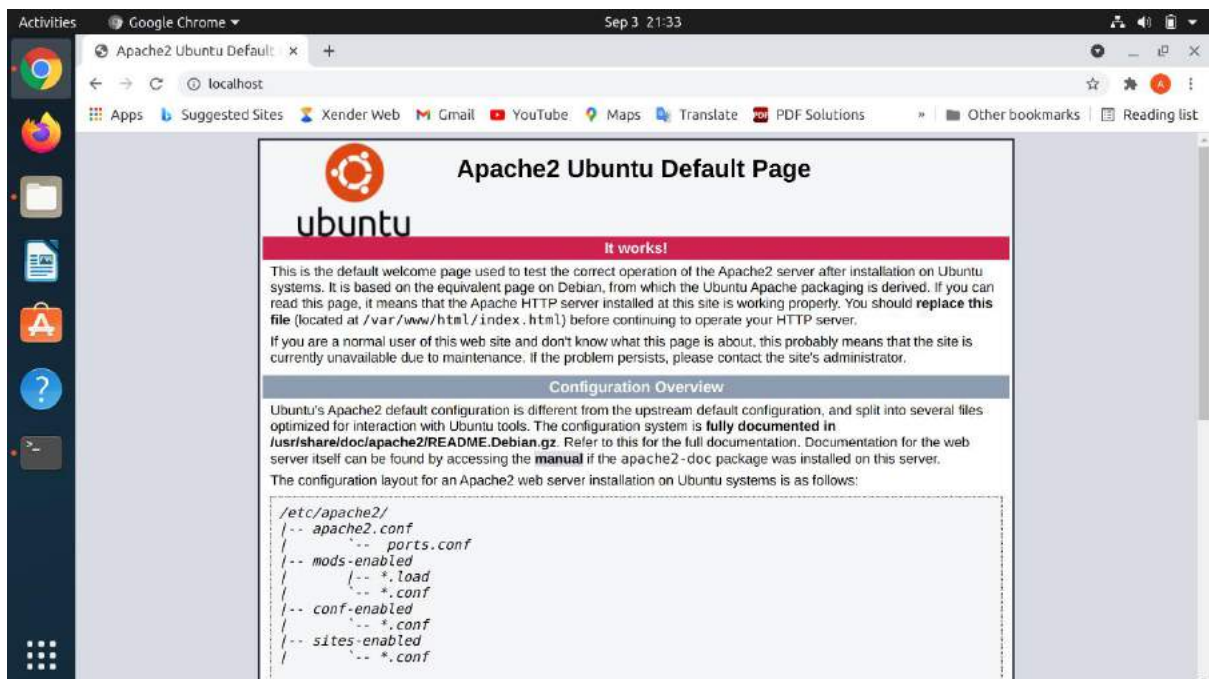


```
aswin@aswin-VirtualBox:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
apache2 is already the newest version (2.4.41-4ubuntu3.4).
The following packages were automatically installed and are no longer required:
  linux-headers-5.8.0-55-generic linux-hwe-5.8.0-headers-5.8.0-55 linux-image-5.8.0-55-generic linux-modules-extra-5.8.0-55-generic
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 76 not upgraded.
aswin@aswin-VirtualBox:~$
```

```
Activities Terminal Sep 3 21:32 aswin@aswin-VirtualBox: -
aswin@aswin-VirtualBox:~$ systemctl status apache2
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2021-09-03 21:18:31 IST; 14min ago
     Docs: https://httpd.apache.org/docs/2.4/
   Process: 750 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
    Main PID: 845 (apache2)
      Tasks: 11 (limit: 3517)
     Memory: 56.4M
    CGroup: /system.slice/apache2.service
            └─ 845 /usr/sbin/apache2 -k start
               868 /usr/sbin/apache2 -k start
               869 /usr/sbin/apache2 -k start
               870 /usr/sbin/apache2 -k start
               871 /usr/sbin/apache2 -k start
               872 /usr/sbin/apache2 -k start
              2828 /usr/sbin/apache2 -k start
              2850 /usr/sbin/apache2 -k start
              2851 /usr/sbin/apache2 -k start
              2852 /usr/sbin/apache2 -k start
              2853 /usr/sbin/apache2 -k start

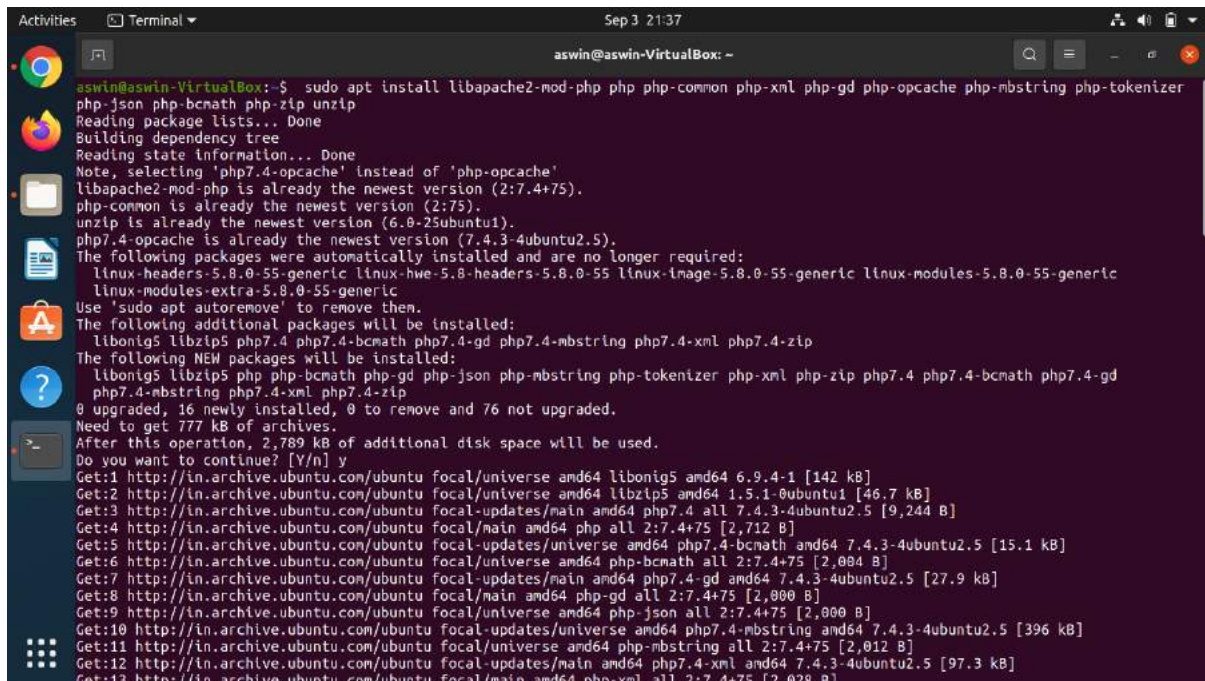
Sep 03 21:18:25 aswin-VirtualBox systemd[1]: Starting The Apache HTTP Server...
Sep 03 21:18:31 aswin-VirtualBox apachectl[773]: AH00558: apache2: Could not reliably determine the server's fully qualified domain
Sep 03 21:18:31 aswin-VirtualBox systemd[1]: Started The Apache HTTP Server.
lines 1-24/24 (END)
```

- Check Apache server is working



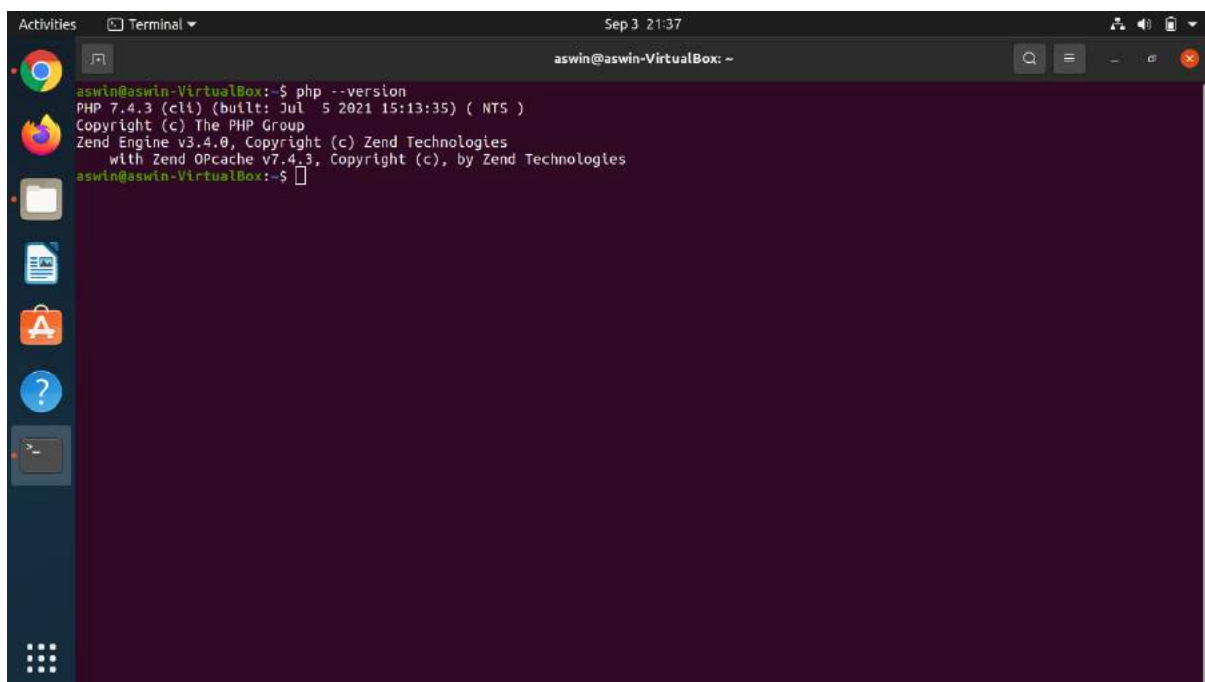
Step-2: Install and Configure PHP 7.4

- To install Laravel 8.x, at least you must have PHP ≥ 7.3 on your system. And by default, the official Ubuntu 20.04 repository provides PHP 7.4 packages. Install PHP 7.4 packages using the apt command



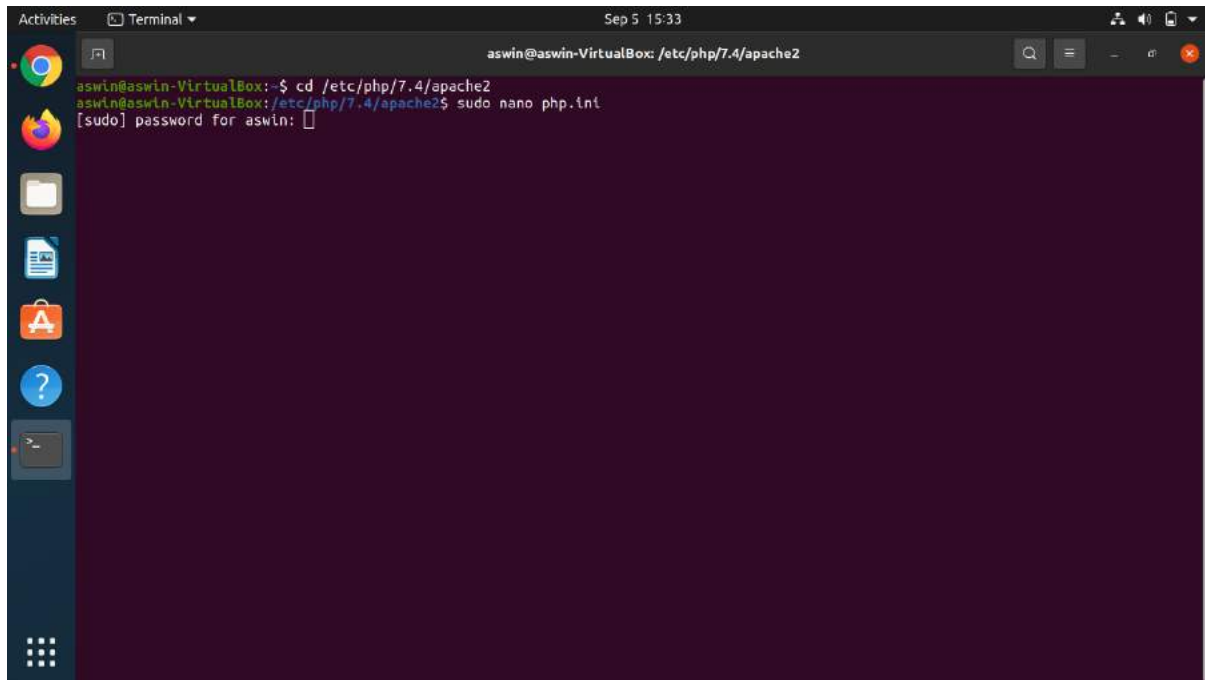
```
aswin@aswin-VirtualBox:~$ sudo apt install libapache2-mod-php php php-common php-xml php-gd php-opcache php-mbstring php-tokenizer
php-json php-bcmath php-zip unzip
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'php7.4-opcache' instead of 'php-opcache'
libapache2-mod-php is already the newest version (2:7.4+75).
php-common is already the newest version (2:75).
unzip is already the newest version (6.0-25ubuntu1).
php7.4-opcache is already the newest version (7.4.3-4ubuntu2.5).
The following packages were automatically installed and are no longer required:
  linux-headers-5.8.0-55-generic linux-hwe-5.8-headers-5.8.0-55 linux-image-5.8.0-55-generic linux-modules-5.8.0-55-generic
  linux-modules-extra-5.8.0-55-generic
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  libonig5 libzip5 php7.4 php7.4-bcmath php7.4-gd php7.4-mbstring php7.4-xml php7.4-zip
The following NEW packages will be installed:
  libonig5 libzip5 php php-bcmath php-gd php-json php-mbstring php-tokenizer php-xml php-zip php7.4 php7.4-bcmath php7.4-gd
  php7.4-mbstring php7.4-xml php7.4-zip
0 upgraded, 16 newly installed, 0 to remove and 76 not upgraded.
Need to get 777 kB of archives.
After this operation, 2,789 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://in.archive.ubuntu.com/ubuntu focal/universe amd64 libonig5 amd64 6.9.4-1 [142 kB]
Get:2 http://in.archive.ubuntu.com/ubuntu focal/universe amd64 libzip5 amd64 1.5.1-0ubuntu1 [46.7 kB]
Get:3 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 php7.4 all 7.4.3-4ubuntu2.5 [9,244 B]
Get:4 http://in.archive.ubuntu.com/ubuntu focal/main amd64 php all 2:7.4+75 [2,712 B]
Get:5 http://in.archive.ubuntu.com/ubuntu focal-updates/universe amd64 php7.4-bcmath amd64 7.4.3-4ubuntu2.5 [15.1 kB]
Get:6 http://in.archive.ubuntu.com/ubuntu focal/universe amd64 php-bcmath all 2:7.4+75 [2,004 B]
Get:7 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 php7.4-gd amd64 7.4.3-4ubuntu2.5 [27.9 kB]
Get:8 http://in.archive.ubuntu.com/ubuntu focal/main amd64 php-gd all 2:7.4+75 [2,000 B]
Get:9 http://in.archive.ubuntu.com/ubuntu focal/universe amd64 php-json all 2:7.4+75 [2,000 B]
Get:10 http://in.archive.ubuntu.com/ubuntu focal-updates/universe amd64 php7.4-mbstring amd64 7.4.3-4ubuntu2.5 [396 kB]
Get:11 http://in.archive.ubuntu.com/ubuntu focal/universe amd64 php-mbstring all 2:7.4+75 [2,012 B]
Get:12 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 php7.4-xml amd64 7.4.3-4ubuntu2.5 [97.3 kB]
Get:13 http://in.archive.ubuntu.com/ubuntu focal/main amd64 php-xml all 2:7.4+75 [2,028 B]
```

- You can check your PHP version using it.



```
aswin@aswin-VirtualBox:~$ php --version
PHP 7.4.3 (cli) (built: Jul 5 2021 15:13:35) ( NTS )
Copyright (c) The PHP Group
Zend Engine v3.4.0, Copyright (c) Zend Technologies
with Zend OPcache v7.4.3, Copyright (c), by Zend Technologies
aswin@aswin-VirtualBox:~$
```


- Now go ahead and make tweak changes in PHP ini file and set `cgi.fix_pathinfo` set to be 0. If this number is kept as a 1, the php interpreter will do its best to process the file that is as near to the requested file as possible. This is a possible security risk. If this number is set to 0, conversely, the interpreter will only process the exact file path—a much safer alternative.

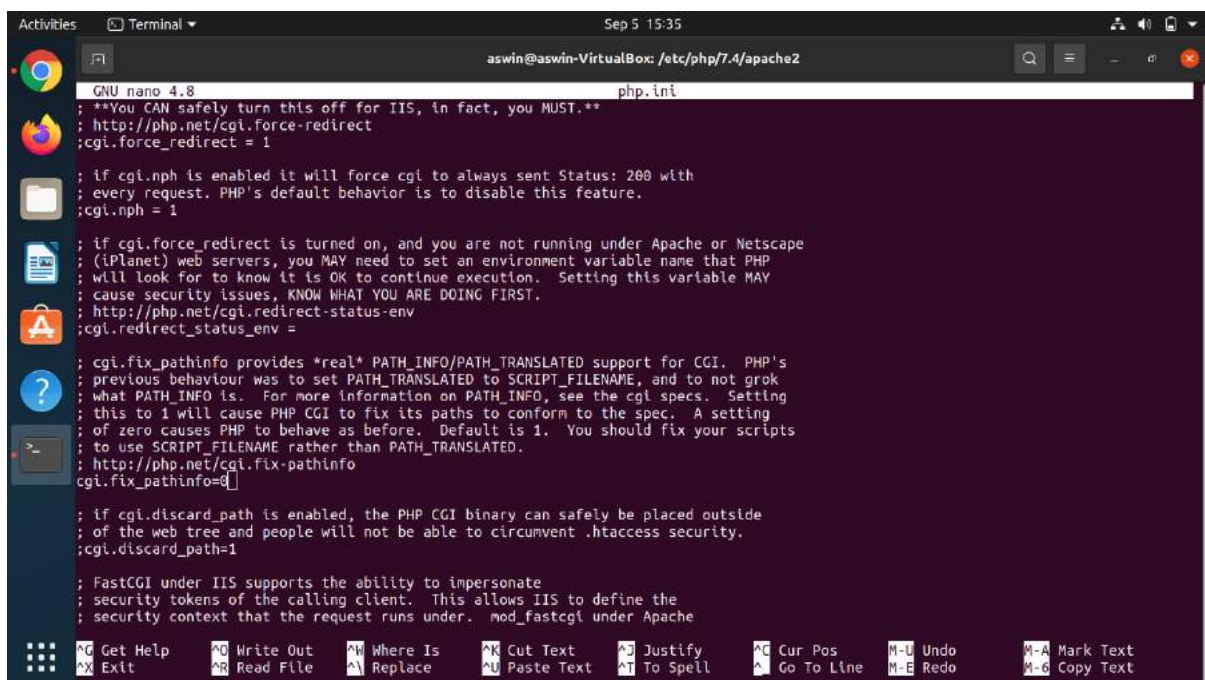


```

Activities  Terminal  Sep 5 15:33
aswin@aswin-VirtualBox: /etc/php/7.4/apache2

aswin@aswin-VirtualBox:~$ cd /etc/php/7.4/apache2
aswin@aswin-VirtualBox:/etc/php/7.4/apache2$ sudo nano php.ini
[sudo] password for aswin: 

```



```

Activities  Terminal  Sep 5 15:35
aswin@aswin-VirtualBox: /etc/php/7.4/apache2

GNU nano 4.8 php.ini
; **You CAN safely turn this off for IIS, in fact, you MUST.**
; http://php.net/cgi.force-redirect
;cgi.force_redirect = 1

; if cgi.nph is enabled it will force cgi to always sent Status: 200 with
; every request. PHP's default behavior is to disable this feature.
;cgi.nph = 1

; if cgi.force_redirect is turned on, and you are not running under Apache or Netscape
; (iPlanet) web servers, you MAY need to set an environment variable name that PHP
; will look for to know it is OK to continue execution.  Setting this variable MAY
; cause security issues, KNOW WHAT YOU ARE DOING FIRST.
; http://php.net/cgi.redirect-status-env
;cgi.redirect_status_env =

; cgi.fix_pathinfo provides *real* PATH_INFO/PATH_TRANSLATED support for CGI.  PHP's
; previous behaviour was to set PATH_TRANSLATED to SCRIPT_FILENAME, and to not grok
; what PATH_INFO is.  For more information on PATH_INFO, see the cgi specs.  Setting
; this to 1 will cause PHP CGI to fix its paths to conform to the spec.  A setting
; of zero causes PHP to behave as before.  Default is 1.  You should fix your scripts
; to use SCRIPT_FILENAME rather than PATH_TRANSLATED.
; http://php.net/cgi.fix-pathinfo
cgi.fix_pathinfo=0

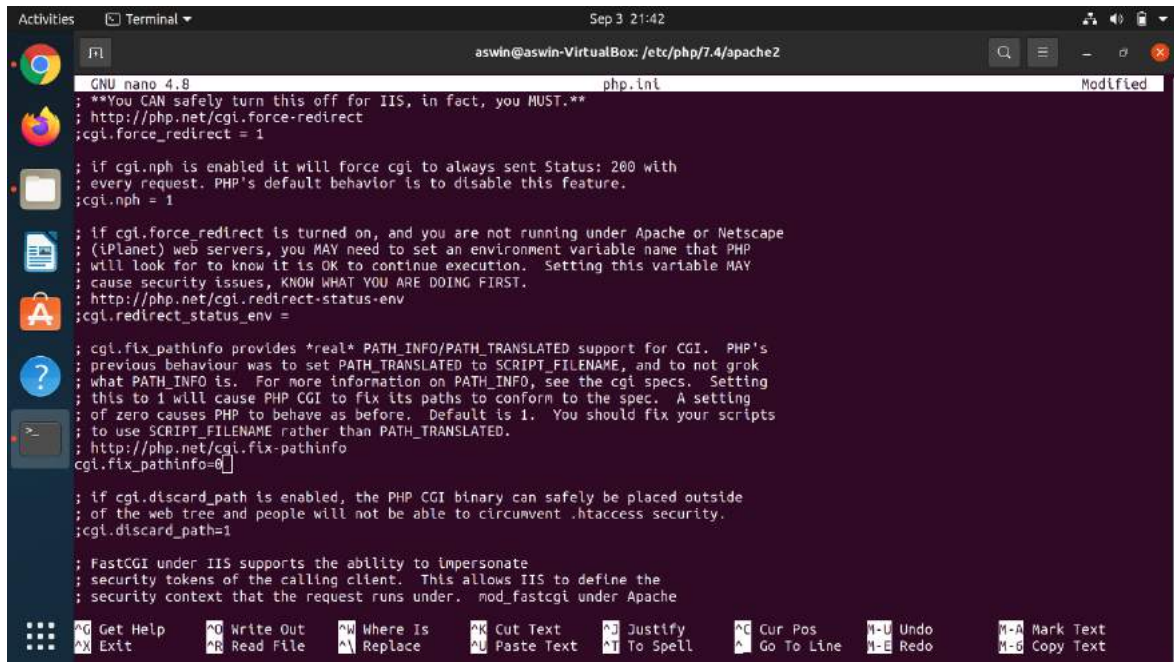
; if cgi.discard_path is enabled, the PHP CGI binary can safely be placed outside
; of the web tree and people will not be able to circumvent .htaccess security.
;cgi.discard_path=1

; FastCGI under IIS supports the ability to impersonate
; security tokens of the calling client.  This allows IIS to define the
; security context that the request runs under.  mod_fastcgi under Apache

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text   ^J Justify    ^C Cur Pos   M-U Undo     M-A Mark Text
^X Exit      ^R Read File  ^M Replace   ^U Paste Text ^T To Spell   ^G Go To Line M-E Redo     M-G Copy Text

```

- Press ctrl+w and search for the word “cgi.fix” the uncomment the line and set it to 0.



```
GNU nano 4.8 php.ini Modified
; **You CAN safely turn this off for IIS, in fact, you MUST.**
; http://php.net/cgi.force-redirect
;cgi.force_redirect = 1

; if cgi.nph is enabled it will force cgi to always sent Status: 200 with
; every request. PHP's default behavior is to disable this feature.
;cgi.nph = 1

; if cgi.force_redirect is turned on, and you are not running under Apache or Netscape
; (iPlanet) web servers, you MAY need to set an environment variable name that PHP
; will look for to know it is OK to continue execution.  Setting this variable MAY
; cause security issues, KNOW WHAT YOU ARE DOING FIRST.
; http://php.net/cgi.redirect-status-env
;cgi.redirect_status_env =

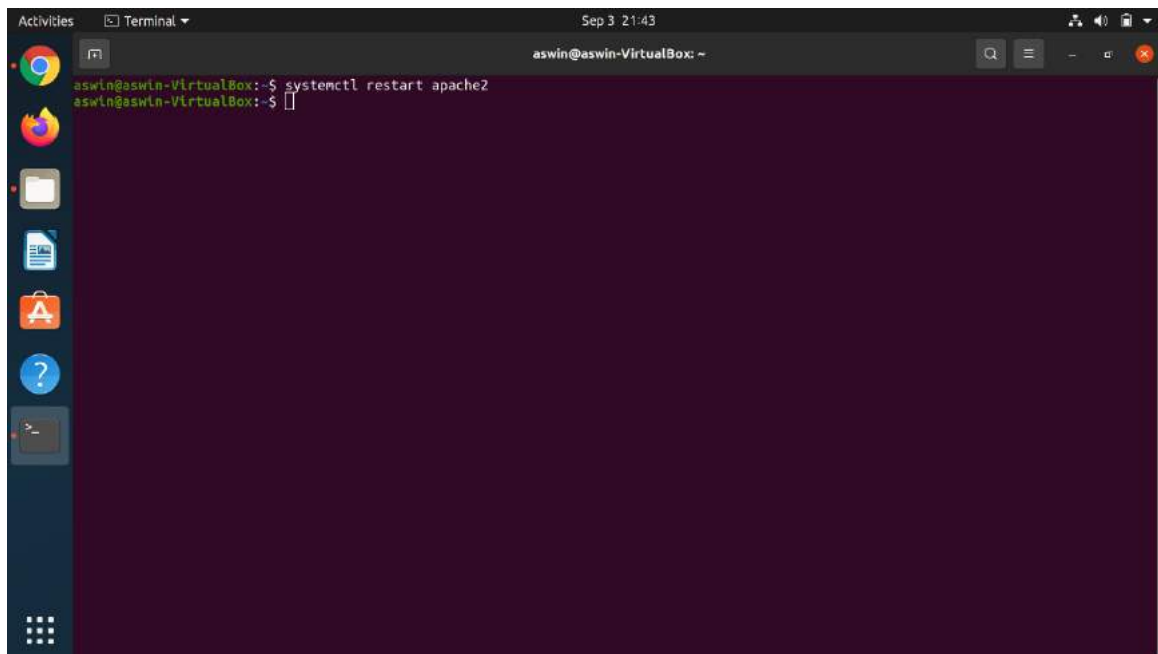
; cgi.fix_pathinfo provides *real* PATH_INFO/PATH_TRANSLATED support for CGI.  PHP's
; previous behaviour was to set PATH_TRANSLATED to SCRIPT_FILENAME, and to not grok
; what PATH_INFO is.  For more information on PATH_INFO, see the cgi specs.  Setting
; this to 1 will cause PHP CGI to fix its paths to conform to the spec.  A setting
; of zero causes PHP to behave as before.  Default is 1.  You should fix your scripts
; to use SCRIPT_FILENAME rather than PATH_TRANSLATED.
; http://php.net/cgi.fix-pathinfo
cgi.fix_pathinfo=0

; if cgi.discard_path is enabled, the PHP CGI binary can safely be placed outside
; of the web tree and people will not be able to circumvent .htaccess security.
;cgi.discard_path=1

; FastCGI under IIS supports the ability to impersonate
; security tokens of the calling client.  This allows IIS to define the
; security context that the request runs under.  mod_fastcgi under Apache
```

Press Ctrl + x then y to Save and Exit.

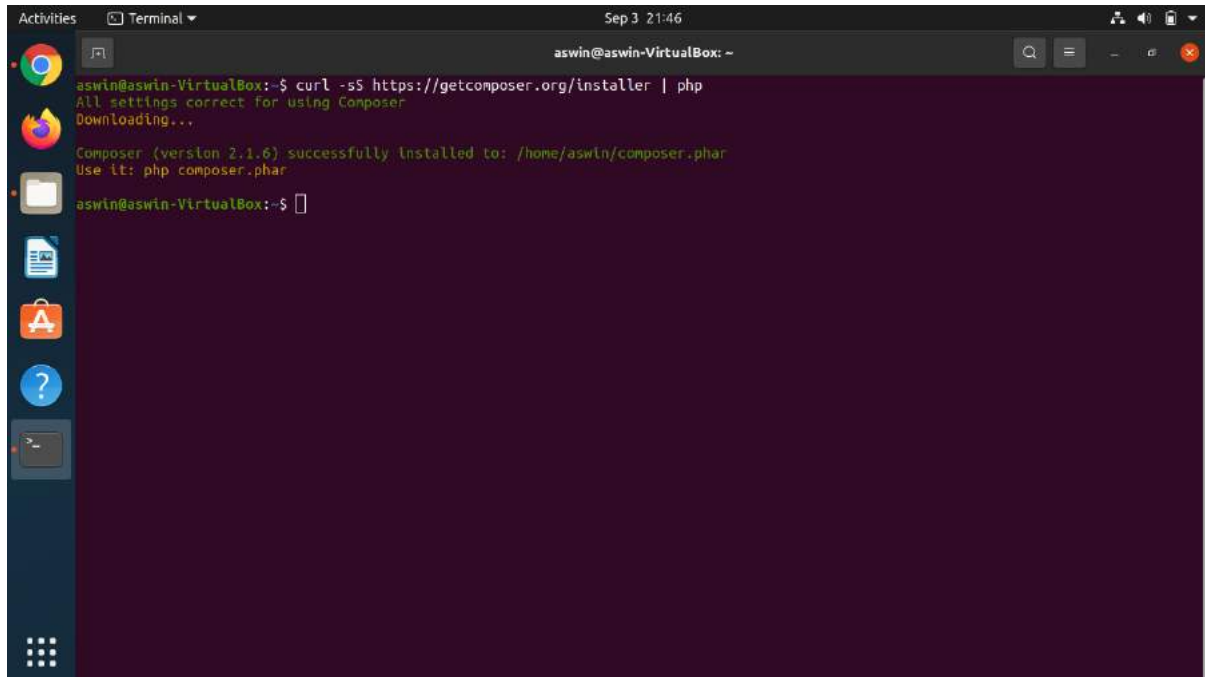
- Now Restart The apache service.



```
aswin@aswin-VirtualBox:~$ systemctl restart apache2
aswin@aswin-VirtualBox:~$
```

Step-3: Install Composer PHP Packages Management

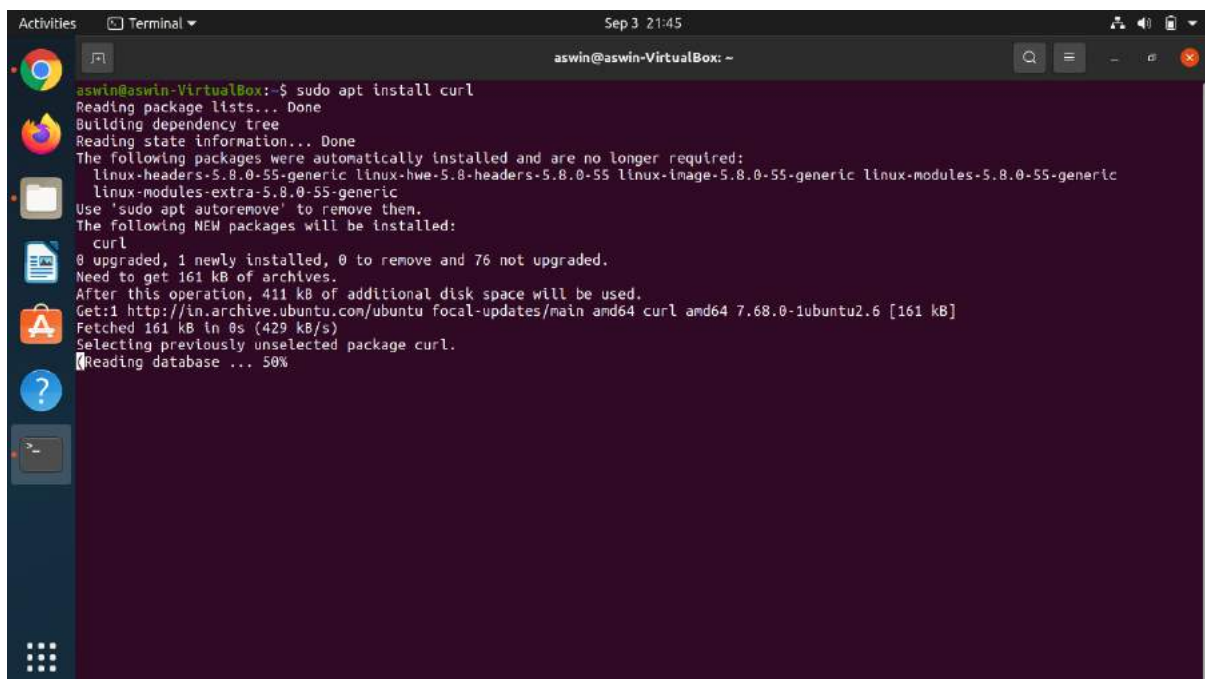
- install the composer package manager go ahead and download and install Composer. and move the composer .phar file to usr/local/bin/composer directory.

A terminal window titled 'aswin@aswin-VirtualBox: ~' showing the installation of Composer. The user runs a curl command to download the installer. The output shows that Composer (version 2.1.6) was successfully installed to /home/aswin/composer.phar and provides instructions to use it via php composer.phar.

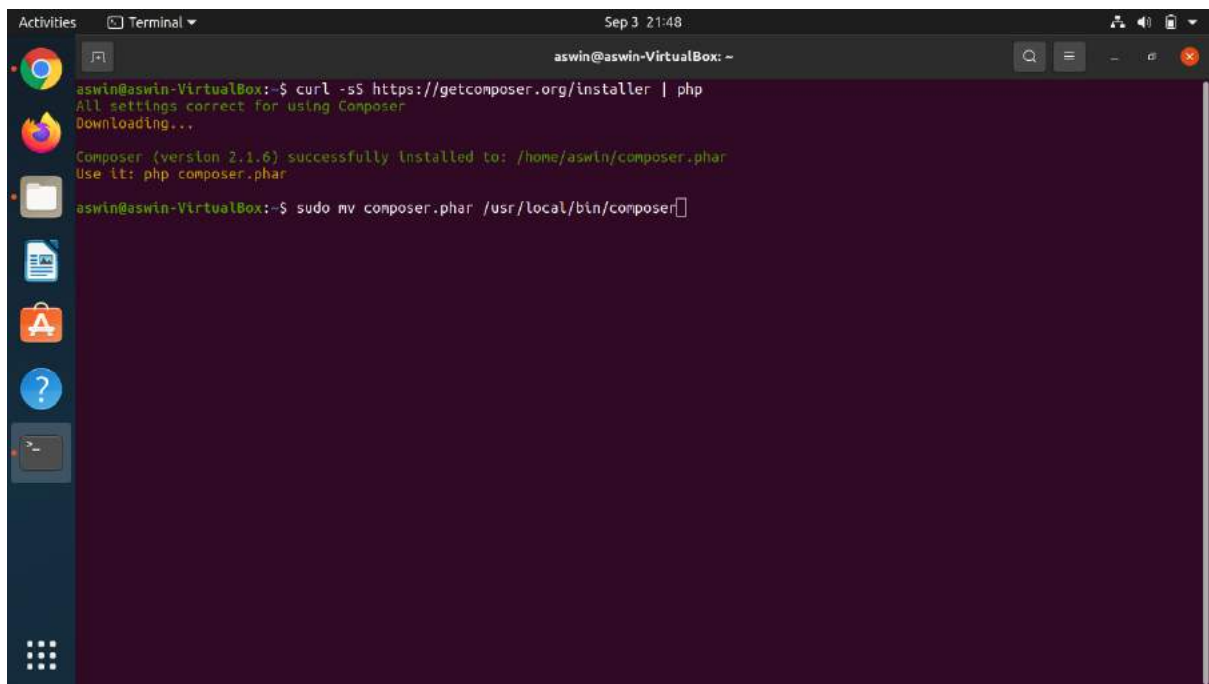
```
aswin@aswin-VirtualBox:~$ curl -sS https://getcomposer.org/installer | php
All settings correct for using Composer
Downloading...

Composer (version 2.1.6) successfully installed to: /home/aswin/composer.phar
Use it: php composer.phar

aswin@aswin-VirtualBox:~$
```

A terminal window titled 'aswin@aswin-VirtualBox: ~' showing the installation of the curl package using apt. The output details the package lists, dependency tree, and the successful installation of curl, including disk space requirements and download progress.

```
aswin@aswin-VirtualBox:~$ sudo apt install curl
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  linux-headers-5.8.0-55-generic linux-hwe-5.8-headers-5.8.0-55 linux-image-5.8.0-55-generic linux-modules-extra-5.8.0-55-generic
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  curl
0 upgraded, 1 newly installed, 0 to remove and 76 not upgraded.
Need to get 161 kB of archives.
After this operation, 411 kB of additional disk space will be used.
Get:1 http://in.archive.ubuntu.com/ubuntu focal-updates/main amd64 curl amd64 7.68.0-1ubuntu2.6 [161 kB]
Fetched 161 kB in 0s (429 kB/s)
Selecting previously unselected package curl.
Reading database ... 50%
```

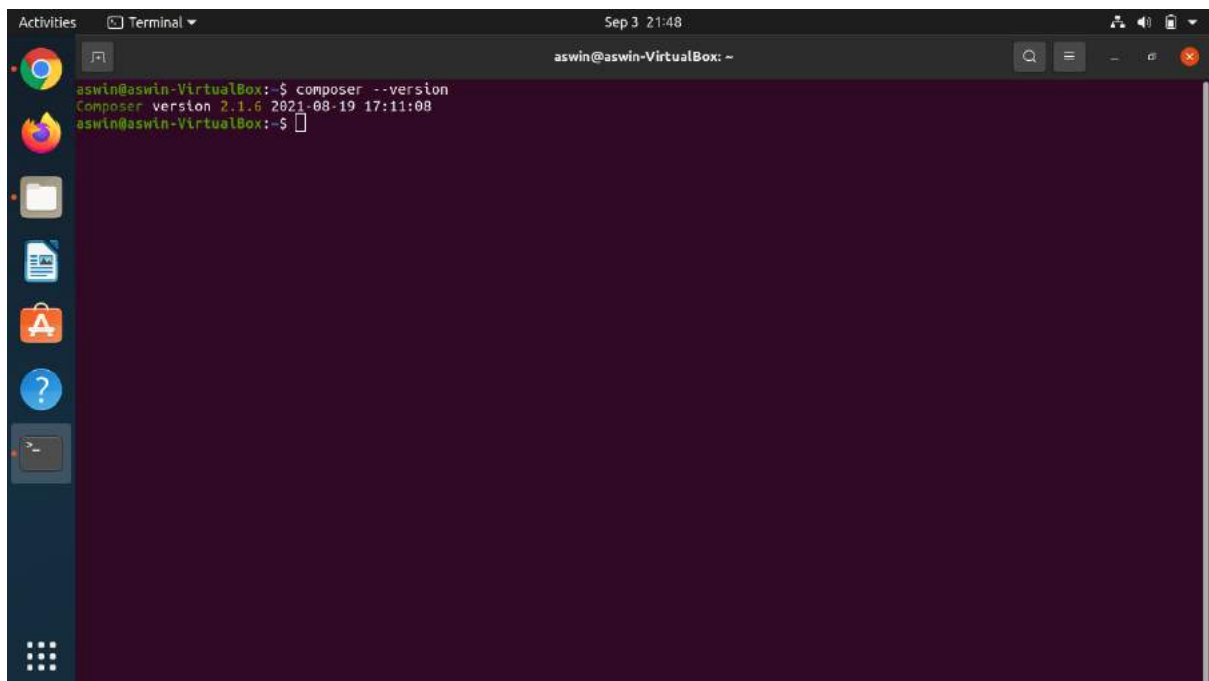
A terminal window titled 'aswin@aswin-VirtualBox: ~' with a dark purple background. The terminal shows the installation of Composer. The first command is `curl -sS https://getcomposer.org/installer | php`, which outputs 'All settings correct for using Composer' and 'Downloading...'. The second command is `sudo mv composer.phar /usr/local/bin/composer`. The terminal window has a sidebar on the left with icons for various applications and a top bar with system information and window controls.

```
aswin@aswin-VirtualBox:~$ curl -sS https://getcomposer.org/installer | php
All settings correct for using Composer
Downloading...

Composer (version 2.1.6) successfully installed to: /home/aswin/composer.phar
Use it: php composer.phar

aswin@aswin-VirtualBox:~$ sudo mv composer.phar /usr/local/bin/composer
```

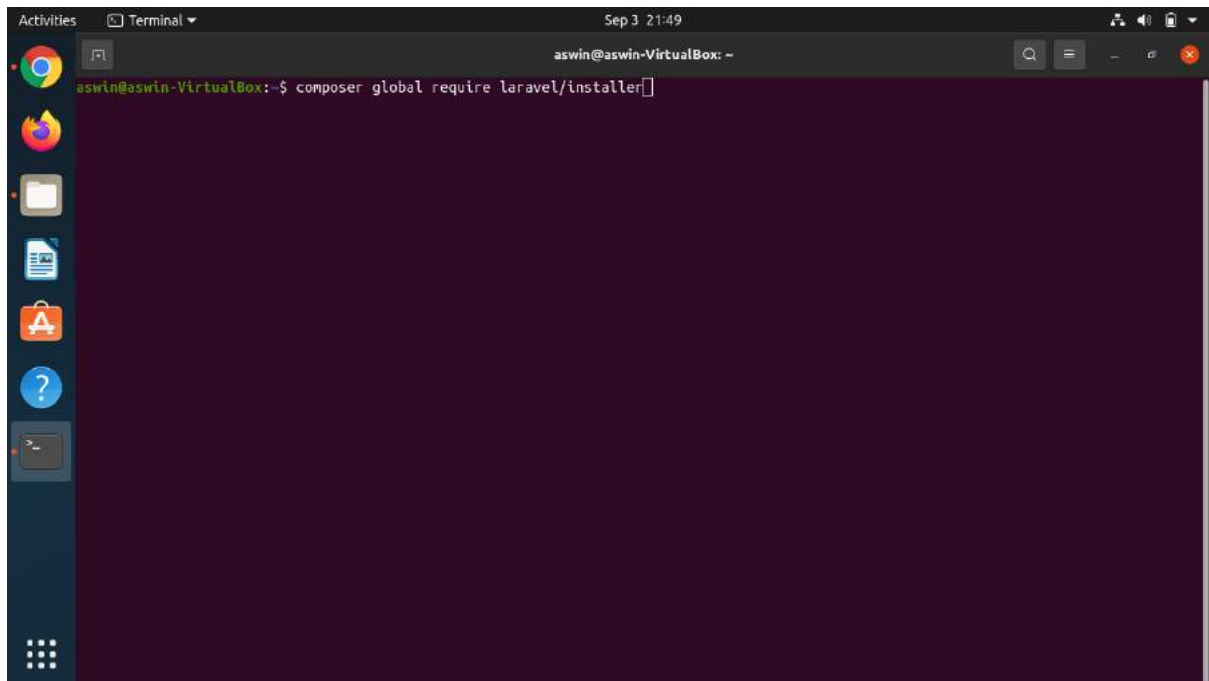
- You can check your installed composer version by typing the `composer --version`.

A terminal window titled 'aswin@aswin-VirtualBox: ~' with a dark purple background. The terminal shows the command `composer --version` being executed, which outputs 'Composer version 2.1.6 2021-08-19 17:11:08'. The terminal window has a sidebar on the left with icons for various applications and a top bar with system information and window controls.

```
aswin@aswin-VirtualBox:~$ composer --version
Composer version 2.1.6 2021-08-19 17:11:08
aswin@aswin-VirtualBox:~$
```

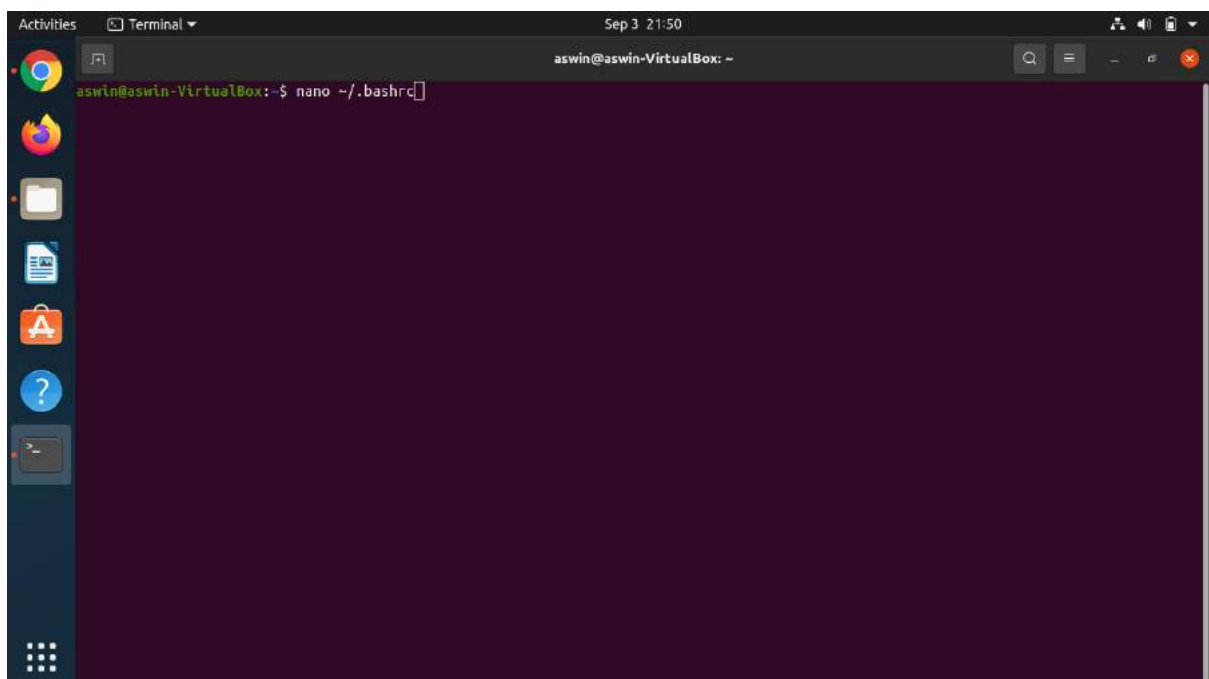

Step-4: Install Laravel 8.x on Ubuntu 20.04

- Now install Laravel Framework using composer, just type composer global require Laravel/installer It will take a while to complete download its dependencies.



A screenshot of a terminal window in a virtual machine. The window title is 'aswin@aswin-VirtualBox: ~'. The prompt is 'aswin@aswin-VirtualBox:~\$' and the command 'composer global require laravel/installer' is being typed. The terminal has a dark purple background. On the left side of the screen, there is a vertical dock with icons for various applications including a web browser, a file manager, and a terminal. The top of the window shows system information like 'Sep 3 21:49' and window control buttons.

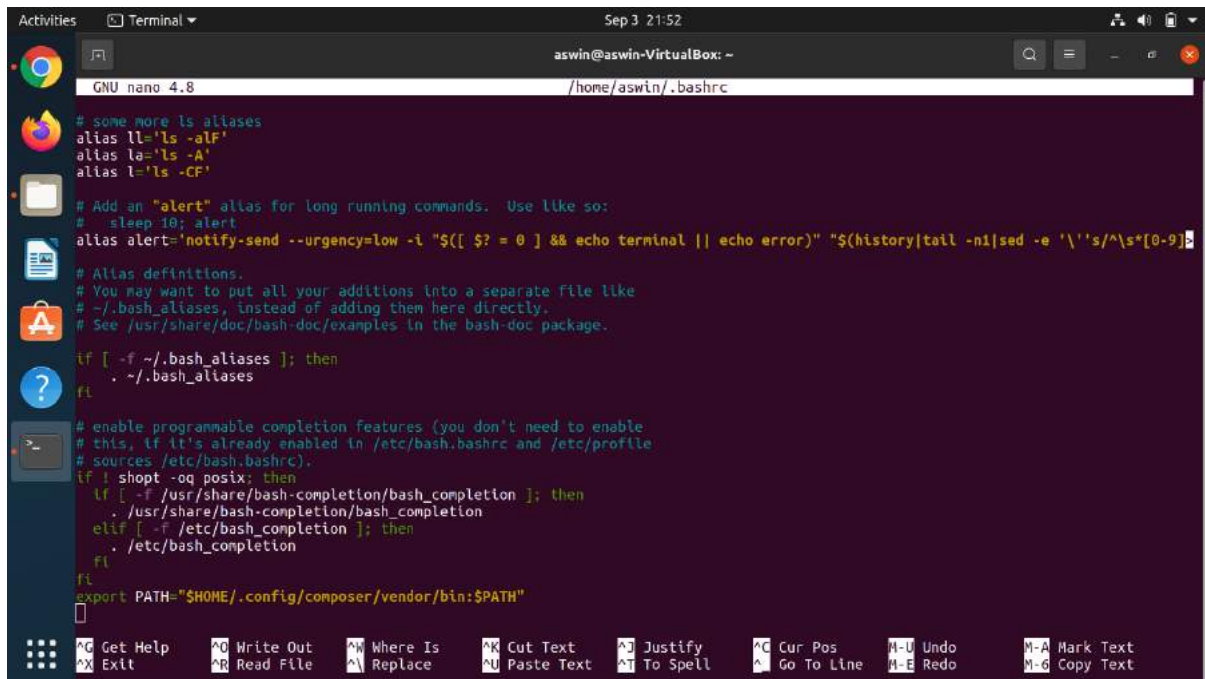
- As you had seen above image, all packages have been installed on the ‘~/.config/composer’ directory. Next, we need to add the ‘bin’ directory to the PATH environment through the ~/.bashrc configuration. So Now Edit the ~/.bashrc configuration using nano command.



A screenshot of a terminal window in a virtual machine, similar to the one above. The window title is 'aswin@aswin-VirtualBox: ~'. The prompt is 'aswin@aswin-VirtualBox:~\$' and the command 'nano ~/.bashrc' is being typed. The terminal has a dark purple background. On the left side of the screen, there is a vertical dock with icons for various applications including a web browser, a file manager, and a terminal. The top of the window shows system information like 'Sep 3 21:50' and window control buttons.

And add the following line at the end of the file.

```
export PATH="$HOME/.config/composer/vendor/bin:$PATH"
```



```
GNU nano 4.8 /home/aswin/.bashrc

# some more ls aliases
alias ll='ls -alF'
alias la='ls -A'
alias l='ls -CF'

# Add an "alert" alias for long running commands.  Use like so:
# sleep 10; alert
alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal || echo error)" "$(history|tail -n1|sed -e '\''s/^\s*[0-9]\s*'\`"

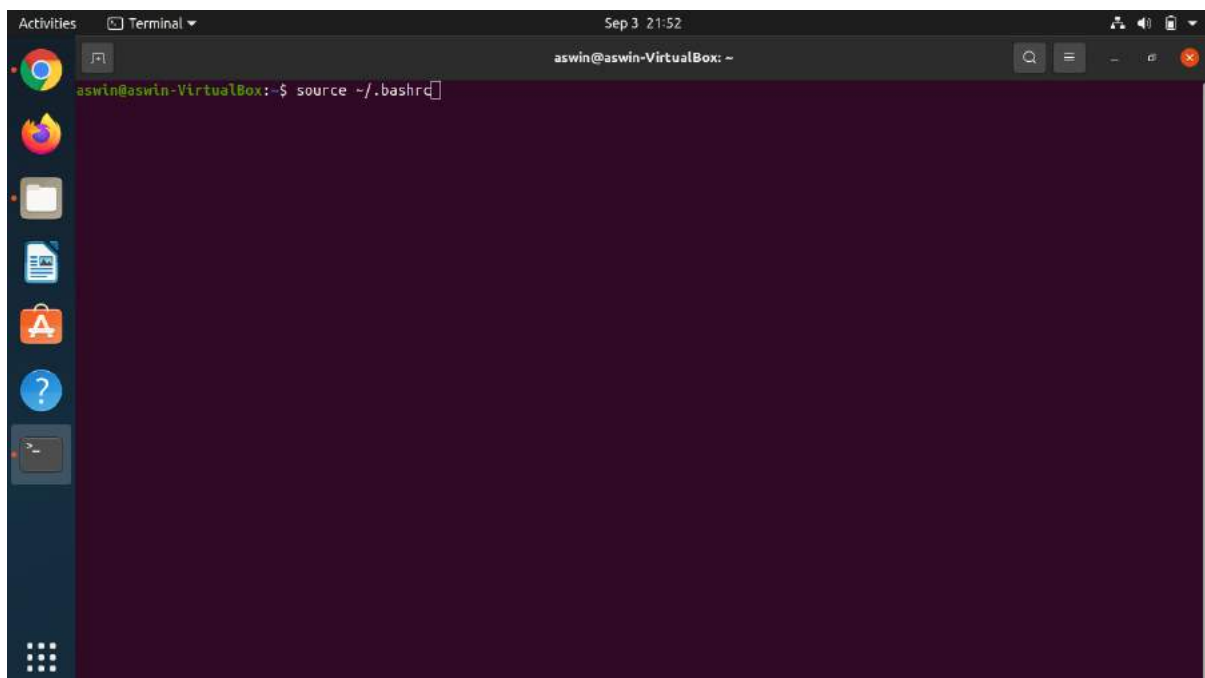
# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi

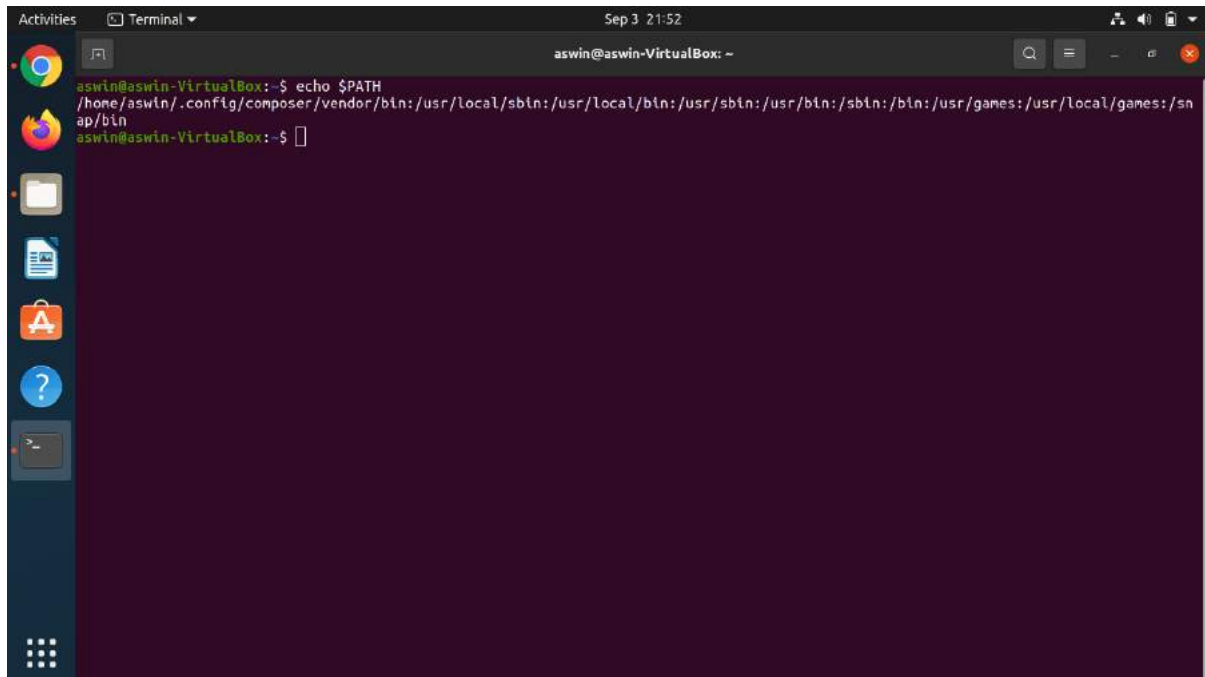
export PATH="$HOME/.config/composer/vendor/bin:$PATH"
```

- Now reload your bashrc configuration using the source command.



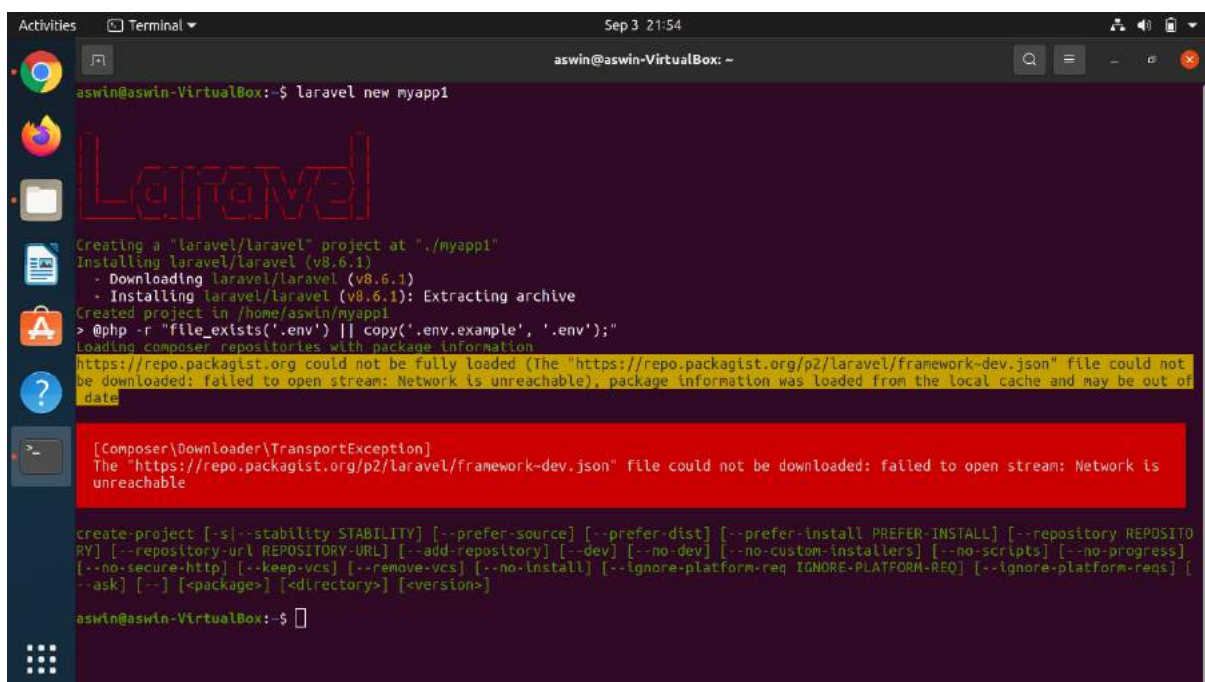
```
aswin@aswin-VirtualBox: ~$ source ~/.bashrc
```

- Now echo \$PATH. It will return your “Bin” directory path for the Composer package.



```
aswin@aswin-VirtualBox:~$ echo $PATH
/home/aswin/.config/composer/vendor/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
aswin@aswin-VirtualBox:~$
```

- The ‘bin’ directory for the composer packages has been added to the \$PATH environment variable. And as a result, you can use the command ‘laravel’ to start and create a new project. Now go ahead and type Laravel new then your project name to start a new Laravel project.



```
aswin@aswin-VirtualBox:~$ laravel new myapp1

Laravel

Creating a "laravel/laravel" project at "./myapp1"
Installing laravel/laravel (v8.6.1)
  - Downloading laravel/laravel (v8.6.1)
  - Installing laravel/laravel (v8.6.1): Extracting archive
Created project in /home/aswin/myapp1
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
https://repo.packagist.org could not be fully loaded (The "https://repo.packagist.org/p2/laravel/framework-dev.json" file could not be downloaded: failed to open stream: Network is unreachable), package information was loaded from the local cache and may be out of date

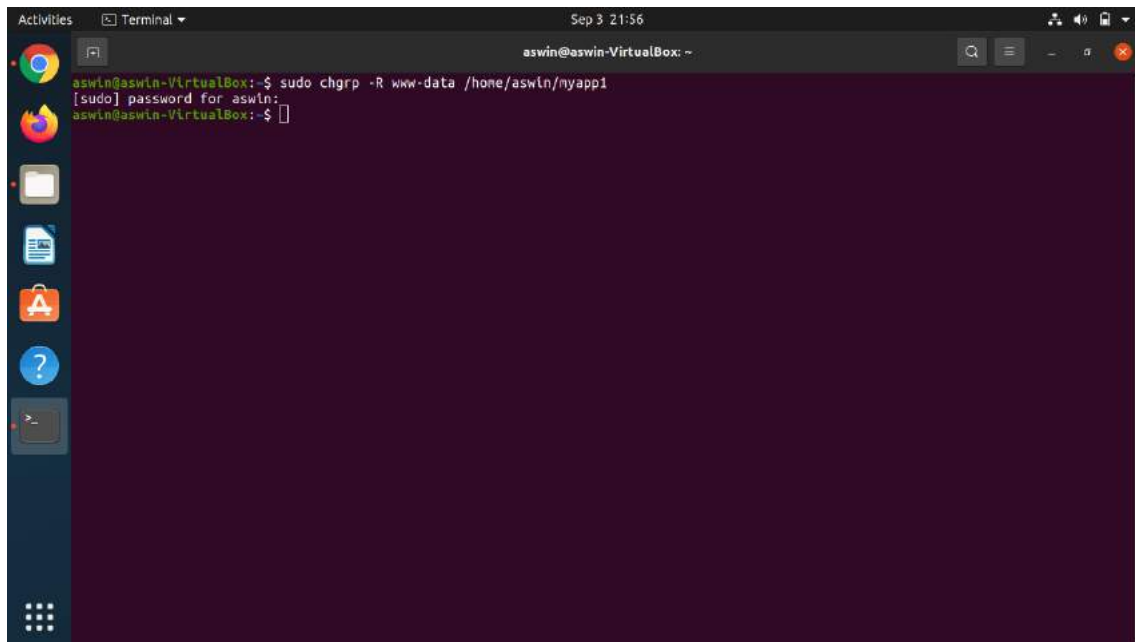
[Composer\Downloader\TransportException]
The "https://repo.packagist.org/p2/laravel/framework-dev.json" file could not be downloaded: failed to open stream: Network is unreachable

create-project [-s|--stability STABILITY] [--prefer-source] [--prefer-dist] [--prefer-install PREFER-INSTALL] [--repository REPOSITORY] [--repository-url REPOSITORY-URL] [--add-repository] [--dev] [--no-dev] [--no-custom-installers] [--no-scripts] [--no-progress] [--no-secure-http] [--keep-vcs] [--remove-vcs] [--no-install] [--ignore-platform-req IGNORE-PLATFORM-REQ] [--ignore-platform-reqs] [-a|--ask] [--] [<package>] [<directory>] [<version>]

aswin@aswin-VirtualBox:~$
```

Step-5: Finally Configure Apache for Laravel and test it

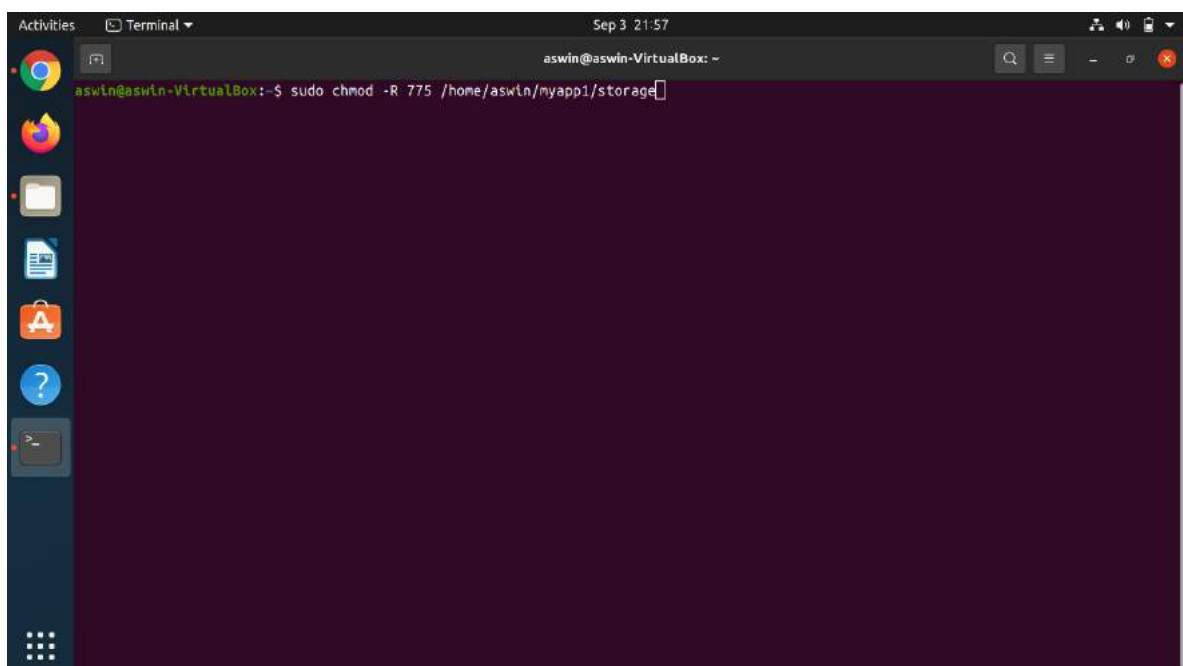
- Add your project directory to www-data group use the following command



```
aswin@aswin-VirtualBox: ~  
$ sudo chgrp -R www-data /home/aswin/nyapp1  
[sudo] password for aswin:  
aswin@aswin-VirtualBox: ~$
```

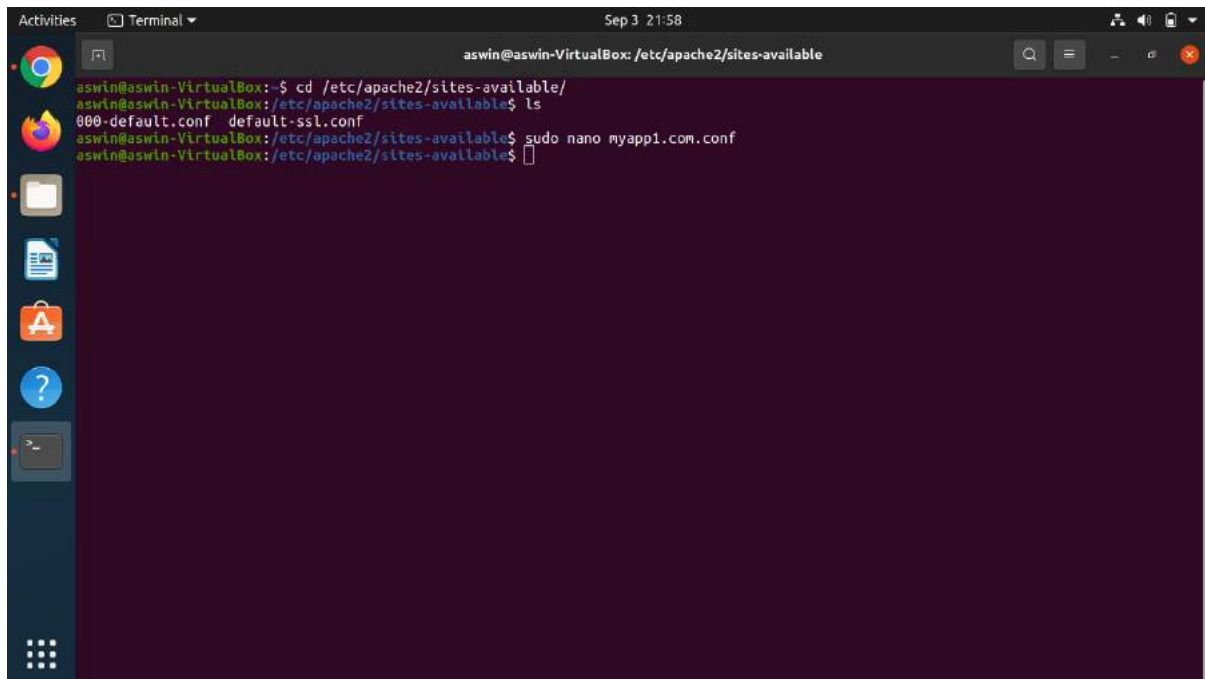
-R flag is recursive, Recursive means all subdirectory and files under your project directory become changed to the “www-data” group.

- Also, you need to change access permission 775 of the storage directory under your project. So, go ahead and use the following command.



```
aswin@aswin-VirtualBox: ~  
$ sudo chmod -R 775 /home/aswin/nyapp1/storage
```

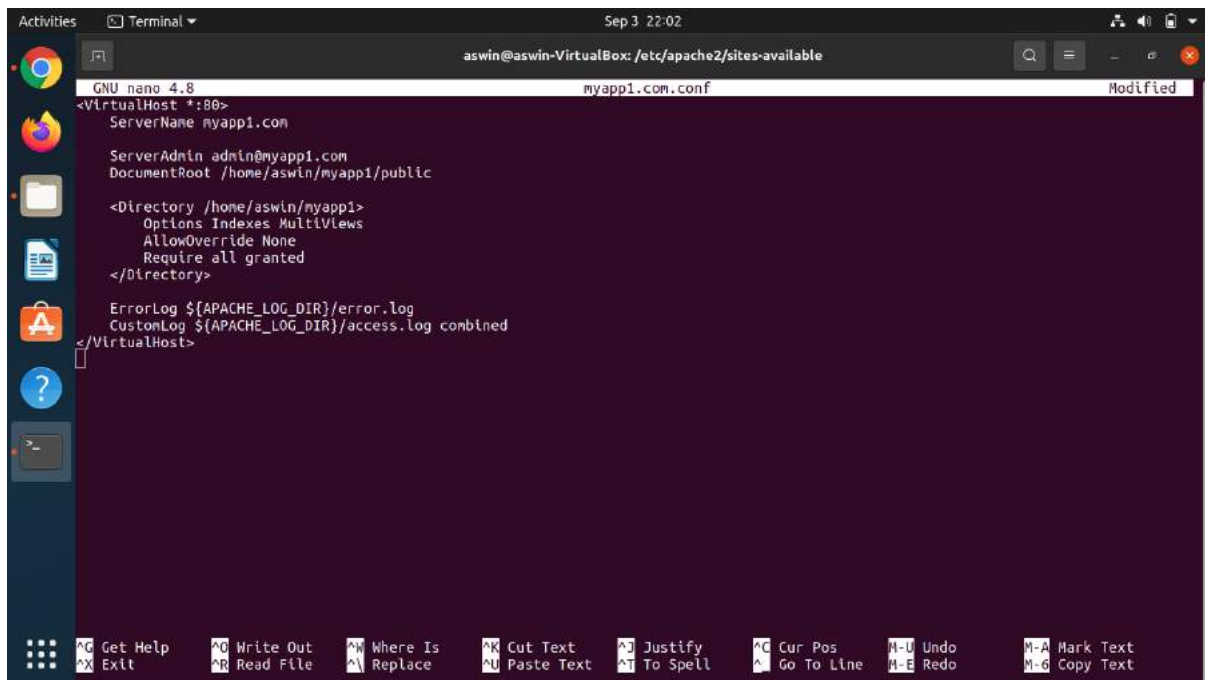
- Now create an apache vhost configuration go to the following directory and create a vhost config file using nano file editor.



A terminal window titled "aswin@aswin-VirtualBox: /etc/apache2/sites-available" with a timestamp of "Sep 3 21:58". The terminal shows the following commands and output:

```
aswin@aswin-VirtualBox:~$ cd /etc/apache2/sites-available/  
aswin@aswin-VirtualBox:/etc/apache2/sites-available$ ls  
000-default.conf default-ssl.conf  
aswin@aswin-VirtualBox:/etc/apache2/sites-available$ sudo nano myapp1.com.conf  
aswin@aswin-VirtualBox:/etc/apache2/sites-available$
```

- And type the following line inside the file.

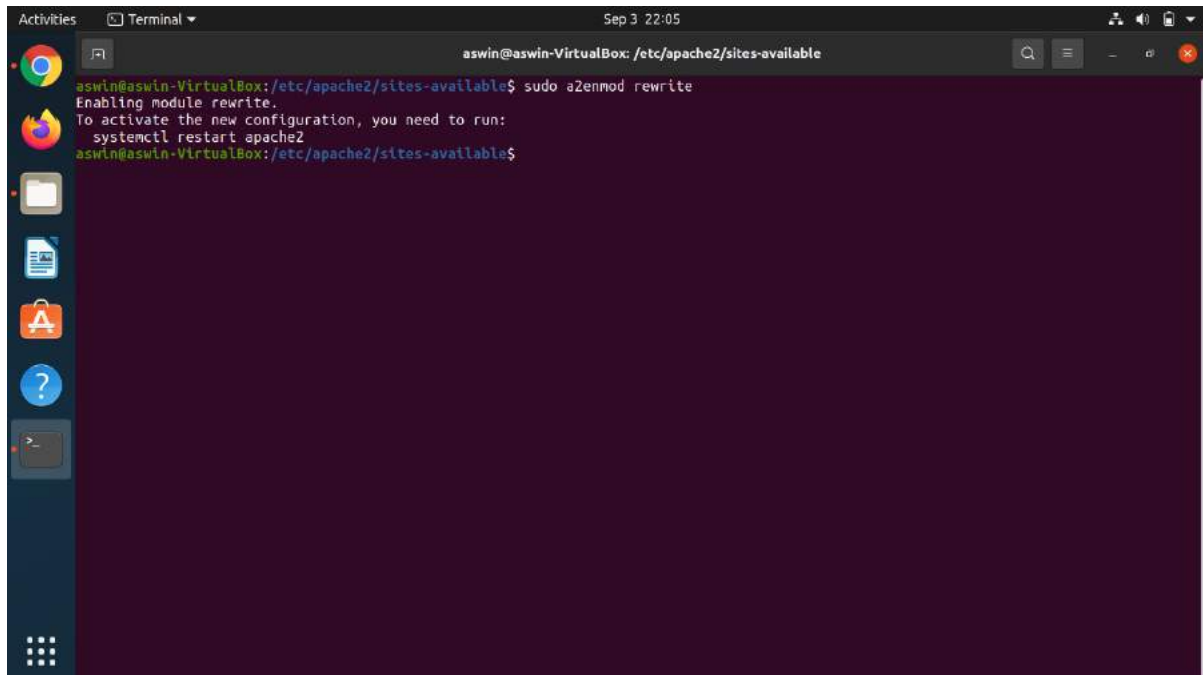


A terminal window titled "aswin@aswin-VirtualBox: /etc/apache2/sites-available" with a timestamp of "Sep 3 22:02". The terminal shows the contents of the file "myapp1.com.conf" being edited with nano 4.8. The file content is as follows:

```
GNU nano 4.8 myapp1.com.conf Modified  
<VirtualHost *:80>  
    ServerName myapp1.com  
  
    ServerAdmin admin@myapp1.com  
    DocumentRoot /home/aswin/myapp1/public  
  
    <Directory /home/aswin/myapp1>  
        Options Indexes MultiViews  
        AllowOverride None  
        Require all granted  
    </Directory>  
  
    ErrorLog ${APACHE_LOG_DIR}/error.log  
    CustomLog ${APACHE_LOG_DIR}/access.log combined  
</VirtualHost>
```

At the bottom of the terminal, there is a status bar with various keyboard shortcuts: ^G Get Help, ^O Write Out, ^W Where Is, ^K Cut Text, ^J Justify, ^C Cur Pos, ^U Undo, ^M Mark Text, ^X Exit, ^R Read File, ^H Replace, ^V Paste Text, ^T To Spell, ^_ Go To Line, ^E Redo, ^C Copy Text.

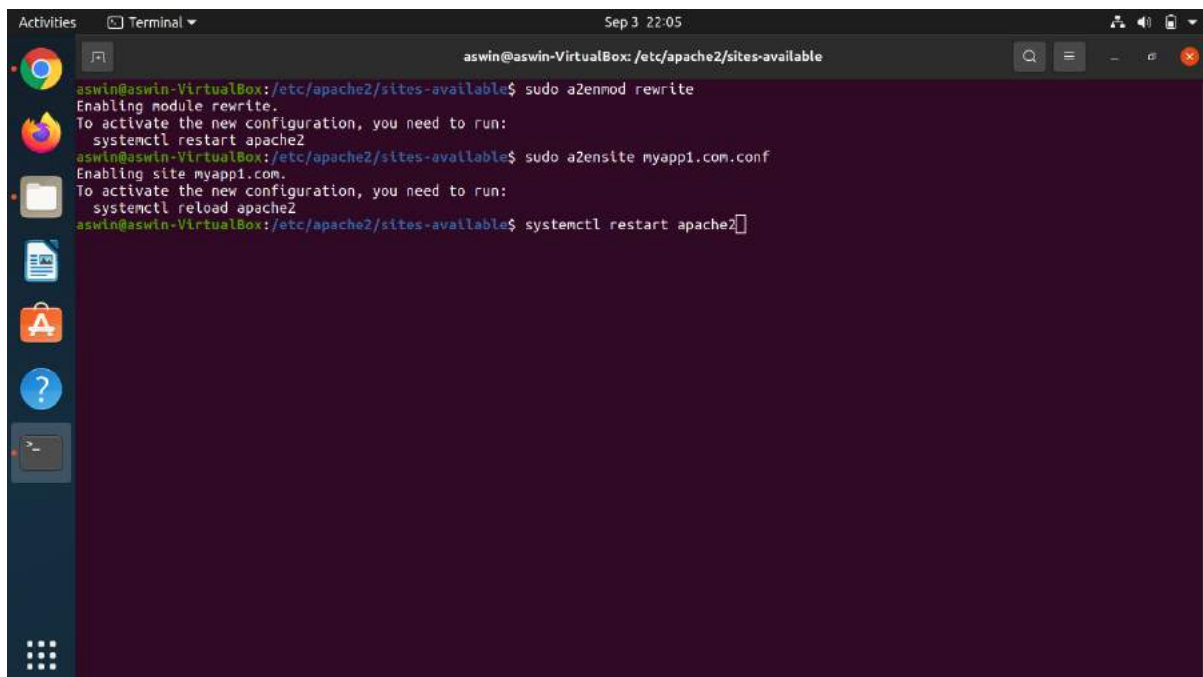
- Now enable mod rewrite for apache2 just type



A terminal window titled "aswin@aswin-VirtualBox: /etc/apache2/sites-available" showing the following commands and output:

```
aswin@aswin-VirtualBox:/etc/apache2/sites-available$ sudo a2enmod rewrite
Enabling module rewrite.
To activate the new configuration, you need to run:
  systemctl restart apache2
aswin@aswin-VirtualBox:/etc/apache2/sites-available$
```

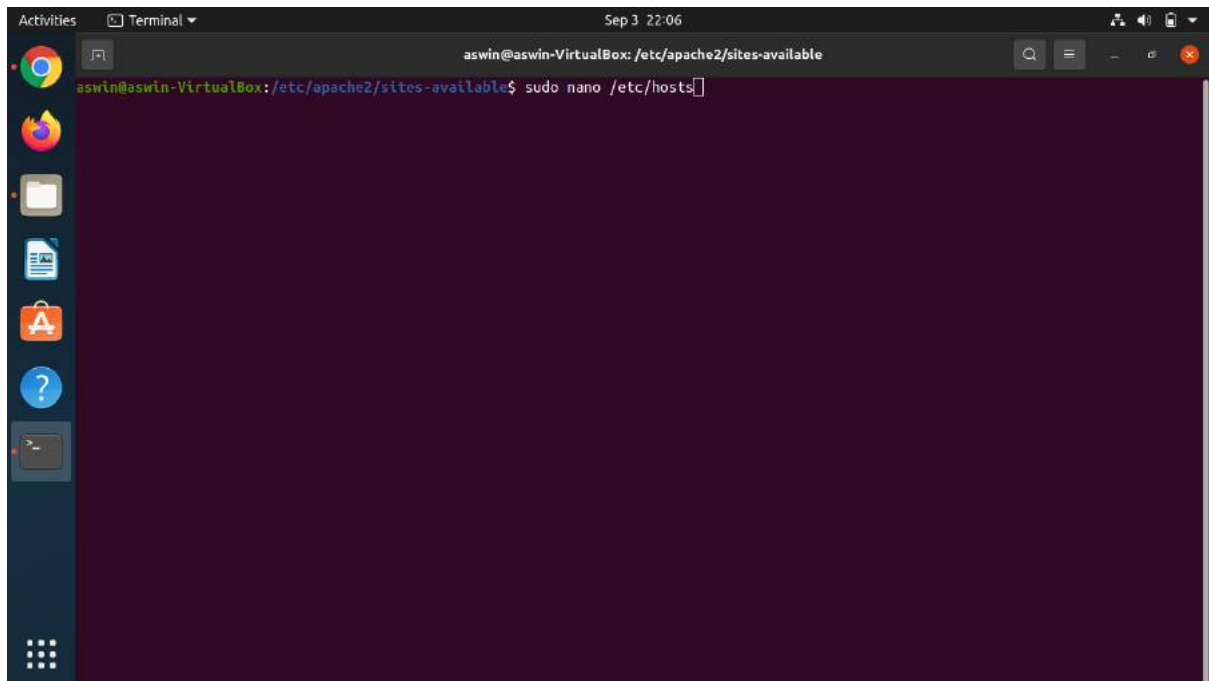
- Now enable your site, just type
Finally, Restart the apache service, type



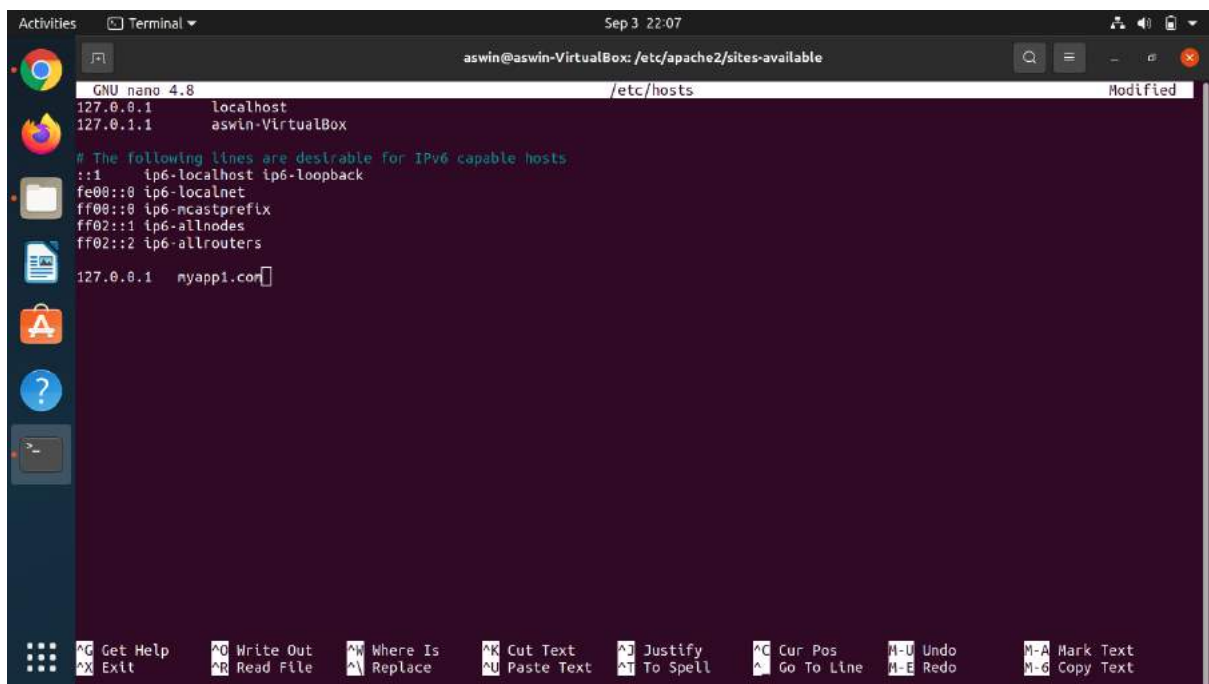
A terminal window titled "aswin@aswin-VirtualBox: /etc/apache2/sites-available" showing the following commands and output:

```
aswin@aswin-VirtualBox:/etc/apache2/sites-available$ sudo a2enmod rewrite
Enabling module rewrite.
To activate the new configuration, you need to run:
  systemctl restart apache2
aswin@aswin-VirtualBox:/etc/apache2/sites-available$ sudo a2ensite myapp1.com.conf
Enabling site myapp1.com.
To activate the new configuration, you need to run:
  systemctl reload apache2
aswin@aswin-VirtualBox:/etc/apache2/sites-available$ systemctl restart apache2
```

- As you are in a local environment you need a local dns resolver for your site. Go ahead and edit /etc/hosts file, add a dns record for your site then save the file.



A terminal window titled "Terminal" with a timestamp of "Sep 3 22:06". The window shows the user "aswin" at the prompt "aswin@aswin-VirtualBox: /etc/apache2/sites-available". The user has entered the command "sudo nano /etc/hosts" and the cursor is at the end of the command.



A terminal window titled "Terminal" with a timestamp of "Sep 3 22:07". The window shows the user "aswin" at the prompt "aswin@aswin-VirtualBox: /etc/apache2/sites-available". The user is editing the file "/etc/hosts" using "GNU nano 4.8". The file contains the following text:

```
127.0.0.1 localhost
127.0.1.1 aswin-VirtualBox

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

127.0.0.1 myapp1.com
```

The bottom of the terminal window shows a status bar with various keyboard shortcuts for nano, such as "Get Help", "Write Out", "Where Is", "Cut Text", "Justify", "Cur Pos", "Undo", "Mark Text", "Exit", "Read File", "Replace", "Paste Text", "To Spell", "Go To Line", "Redo", and "Copy Text".

- Now get back to the web browser and open a tab then type your project hostname.

