**LAMP SERVER PROJECT**

**Introduction:**

In this project I build the LAMP server. LAMP server uses Linux as the operating system, Apache as the web server, MySQL for the relational database, and the PHP for scripting language.

LAMP stack is used for building and delivering web-based applications. Its flexibility and efficiency allow smaller developers to compete with commercial software development solutions.

**Technologies:**

Operating System - Ubuntu

PHP – Popular web scripting language.

Apache Web Server – serve webpages the Internet via the HTTP protocol.

MySQL DB Server – open-source database management system.

**Installation:**

Step 1 – Installation of Linux (Ubuntu 20.04) in virtual box.

For the installation of Linux, I first downloaded the latest version of VirtualBox in my windows laptop, and then download the Ubuntu 20.04.iso file directly from the Ubuntu website. Now the next thing is installation of ubuntu on VirtualBox. To install the Ubuntu hit the new tab which shows in below screenshot and then follow the instructions.

Graphical user interface, text, application, email

Description automatically generated

After installing the Ubuntu you’ll be asked for Try Ubuntu or Install Ubuntu, please select install ubuntu, and after installing the Ubuntu your Ubuntu home screen should look like below screenshot.

A screenshot of a video game

Description automatically generated

Step 2 – Installing the Apache

The Apache web server is among the most popular web servers in the world. I installed apache2 using terminal and the cmd - **sudo apt-get install apache2**

For start the apache2 service cmd – **systemctl start apache2**

For check the apache2 service status cmd - **systemctl status apache2**

For enable the apache2 service cmd – **systemctl enable apache2**

A screenshot of a computer

Description automatically generated

The above SS shows how I initiate the installation of the apache service.

We can do a spot check right away to verify that everything went as planned by visiting our server’s public IP address in our web browser. To find what is our public IP address we can type the cmd **- http://your\_server\_ip** or we can type localhost on our web browser to check. Below SS shows Apache2 Ubuntu Default page.

Graphical user interface, text

Description automatically generated

Step 3 – Installing MySQL.

Now that we have a web server up and running, we need to install the database system to be able to store and manage data for our site. MySQL is a popular database management system used within PHP environments. I installed the MySQL database and set it up with the password.

Installed MySQL with following Commad:

**sudo apt install mysql-server**

When prompted, confirm installation by typing Y, and then enter. When the installation is finished I initiate the cmd **sudo mysql\_secure\_installation** for set the password and remove some insecure default settings. Below SS shows how I start the installing MYSQL.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Step 4 – Installing PHP.

PHP is the component of our setup that will process code to display dynamic content to the final user. we’ll need php-mysql, a PHP module that allows PHP to communicate with MySQL-based databases.

For Installing the PHP, I used Following cmd:

**sudo apt install php libapache2-mod-php php-mysql**

Below the SS shows how I undertake the installing of PHP.

Text

Description automatically generated

After that I created new file named info.php inside my custom web root folder.

**cd /var/www/html/**

**vi info.php**

After that I added the following text, which is valid PHP code, inside the file.

**<?php**

**phpinfo();**

When I finished I saved file and close it, then went to the web browser again and type it

**http://localhost/info.php** to test the script. Below SS provides information about your server from the perspective of PHP. It is useful for debugging and to ensure that your settings are being applied correctly.

If we can see this page in our browser, then our PHP installation is working as expected.

Graphical user interface

Description automatically generated

**Problems:**

First problem I faced was my ubuntu instance had blackout screen while starting. I solved this problem by reinstalling the Ubuntu iso file.

Second Problem was after creating the info.php file, when I went for checking the script my php test script didn’t show up and had some errors after searching on the internet I resolved that problem with typing the command **systemctl restart apache2.**

The other problem I faced was connecting my front end to my back end, this issue was solved by installing php-mysql package.

**Reference:**

I learned so many things after finishing this project but finish this project I need some help which I include the website where I went for when I need help for completion of the project.

<https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu-20-04>

https://www.youtube.com/watch?v=BxONpdx2wJM&t=8s