



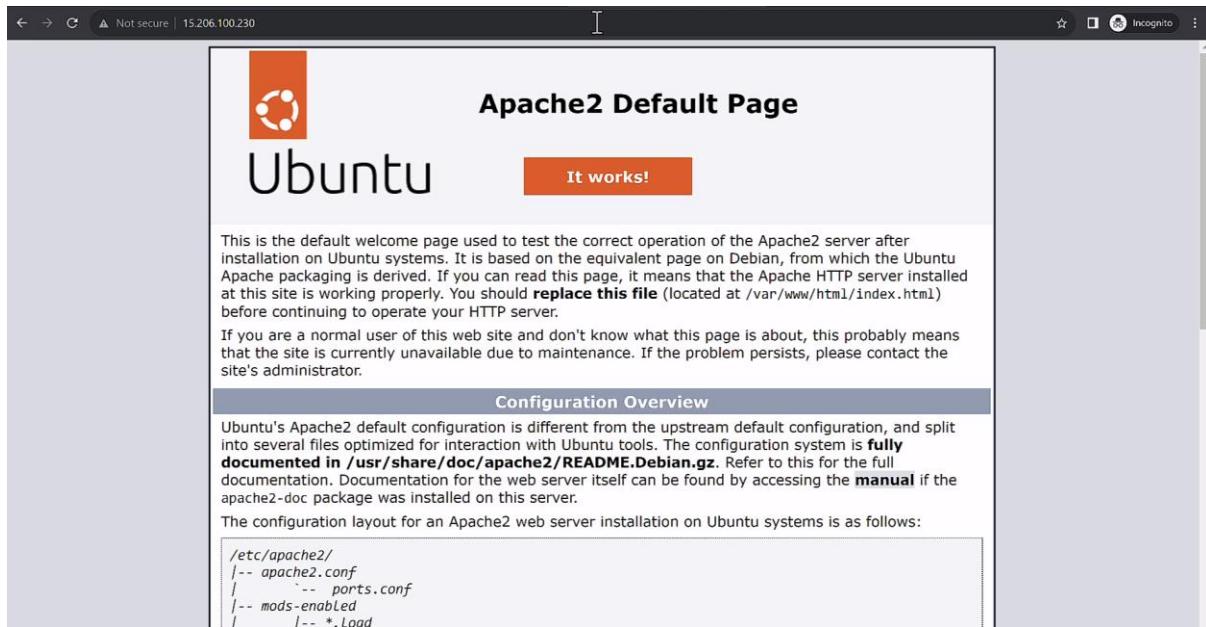
## Deploying Application to the Server

- Now, so far, we have a database server in place that's hosting a database in the MySQL database engine. We have also established connectivity from the machine itself.
- Now I want to have a simple PHP application running on an Apache web server that can connect onto the database engine, fetch and display information accordingly.
- Now, obviously you could use another type of program. You could develop a net application, develop let's say a Java application and deployed accordingly. Here just showing a very simple PHP based page that can go and fetch the information and display accordingly.
- In this page index dot PHP, I put in the host name that is the private IP address of DB Instance. What is the username it's going to use to connect on to the database server. What is the password and what is the database name.
- In the lib folder. I just have the bootstrap file which I'm making since I'm making use of bootstrap in this code. We have to copy this code onto our web server.
- But I said I also want to make use of the Apache web server. Currently on our machine, we have nginx running. So, what I'll do is I will first stop the Nginx web server because it listens on Port 80 and Apache is also going to listen on port 80 so, we can't have both of them listening on port 80. So firstly, I'm on the web server.
- That's web instance. I'm first going to stop Nginx. Then I will install the Apache web server. And I believe by default, it should be running. Let's just confirm. So, I'll take the public IP address of Web VM.

```
sudo systemctl stop nginx
```

```
sudo apt install apache2
```

```
ubuntu@ip-10-0-0-68:~$ sudo systemctl stop nginx
ubuntu@ip-10-0-0-68:~$ sudo apt install apache2
Reading package lists... 0%
```



- After that you need to install some libraries of apache onto the web server itself.

```
sudo apt install php libapache2-mod-php
```

```
ubuntu@ip-10-0-0-68:~$ sudo apt install php libapache2-mod-php
```

- Now you are going to install the package of MySQL for PHP.

```
sudo apt install php-mysql
```

```
ubuntu@ip-10-0-0-68:~$ sudo apt install php-mysql
```

- Here you have to restart the apache.

```
sudo systemctl restart apache2
```

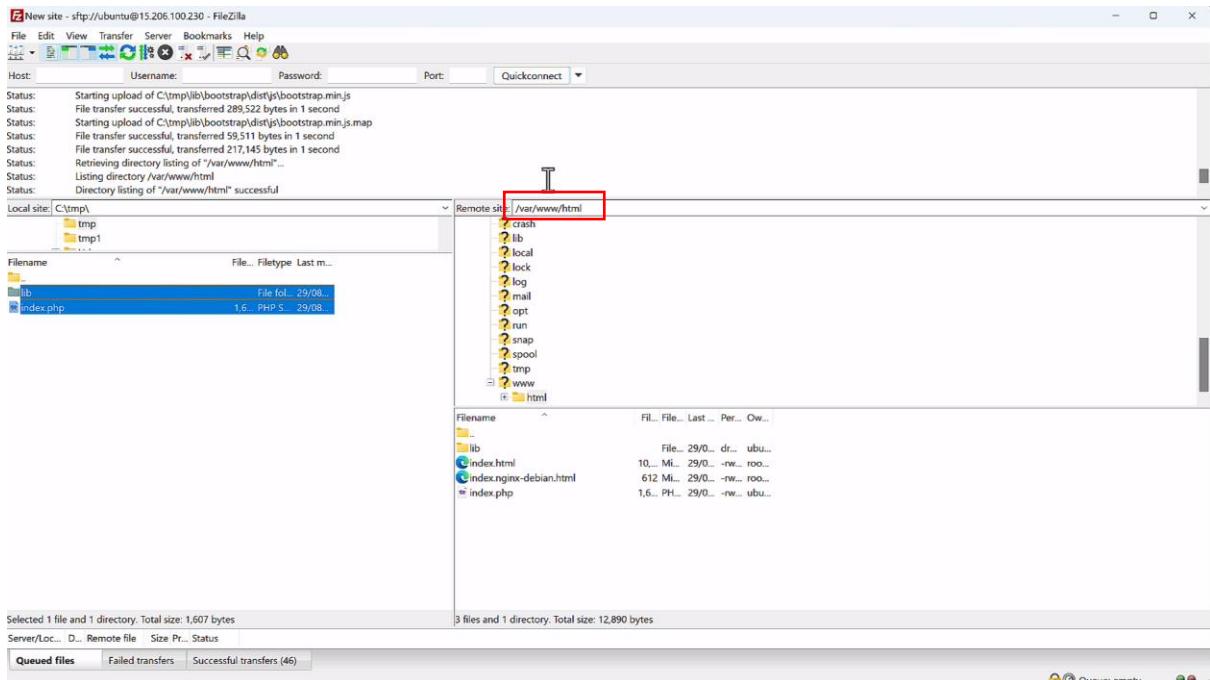
```
ubuntu@ip-10-0-0-68:~$ sudo systemctl restart apache2  
ubuntu@ip-10-0-0-68:~$ █
```

- Now what you have to do, you need to copy the application to this folder /var/www/html. But first you need to give permission to it. The command used here a very open command, you can go ahead and use more command for permissions.

```
sudo chmod 777 /var/www/html
```

```
ubuntu@ip-10-0-0-68:~$ sudo chmod 777 /var/www/html █
```

- After the permission change open file zilla and copy the lib folder and PHP application to your instance. But first remember to go to the folder /var/www/html



- If you now do a **slash index dot php**. You will see a table and this table is fetching or displaying information. Basically, this information is coming in from your table within your MySQL database. So, this entire exercise was just to kind of showcase how you can deploy again, your workloads onto your own VPC.

**B.tech Certifications**

This is a list of some of the B.tech Certifications

Course ID	Course Name	Rating
1	Computer Science - B.tech	4.5
2	Information Technology - B.tech	4.6
3	Mechanical Engineering - B.tech	4.7