

PRAKTIKUM TEKNOLOGI CLOUD COMPUTING
LAPORAN PROYEK AKHIR
DRAFT DOKUMEN RUSUNAWA MENGGUNAKAN LAMPP DAN PROSES
PEMBUATAN DOCKERFILE NYA



DISUSUN OLEH:

NAMA ANGGOTA : RIFA OKTAVIAN 123170096
NICO 123170103
KELAS : B
ASISTEN PRAKTIKUM : JALUANDA PARAMA, S.KOM
MUHAMMAD IMAM AL FATAH

PROGRAM STUDI INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK INDUSTRI
UNIVERSITAS PEMBANGUNAN NASIONAL "VETERAN"
YOGYAKARTA
2020

HALAMAN PENGESAHAN

DRAFT DOKUMEN RUSUNAWA MENGGUNAKAN LAMPP DAN PROSES PEMBUATAN DOCKERFILE NYA

Disusun oleh :

Rifa Oktavian

123170093

Nico

123170096

Telah diperiksa dan disetujui oleh Asisten Praktikum Teknologi Cloud Computing
pada tanggal :

Menyetujui,

Asisten Praktikum

Asisten Praktikum

Jaluanda Parama, S.Kom.

Muhammad Imam Al Fatah

NIM. 123456789

Mengetahui,

Ka. Lab. Sistem Digital

Mangaras Yanu Florestiyanto, S.T., M.Eng.

NIK. 2 8201 13 0425 1

KATA PENGANTAR

Assalamualaikum Warahmatullah Wabarakatuh.

Puji syukur kami panjatkan kepada Tuhan Yang Maha Esa yang senantiasa mencurahkan rahmat dan hidayah-Nya sehingga kami dapat menyelesaikan Praktikum Teknologi Cloud Computing serta laporan proyek akhir praktikum yang berjudul Draft Dokumen Rusunawa menggunakan LAMPP dan Proses Pembuatan Docker nya. Adapun laporan ini berisi tentang proyek akhir yang kami pilih dari hasil pembelajaran selama praktikum berlangsung.

Tidak lupa kami ucapkan terimakasih kepada Asisten Praktikum yang selalu membimbing dan mengajari kami dalam melaksanakan praktikum dan dalam menyusun laporan ini. Laporan ini masih sangat jauh dari kesempurnaan, oleh karena itu kritik serta saran kami harapkan untuk menyempurnakan laporan akhir ini. Atas perhatian dari semua pihak yang membantu penulisan ini, kami ucapkan terimakasih. Semoga laporan ini dapat dipergunakan seperlunya. Wassalamualaikum Warahmatullah Wabarakatuh.

Yogyakarta, 17 Mei 2020

Penyusun

DAFTAR ISI

HALAMAN PENGESAHAN	ii
KATA PENGANTAR	iii
DAFTAR ISI	iv
BAB I PENDAHULUAN	1
1.1. Latar Belakang Masalah.....	1
1.2. Tujuan Proyek Akhir.....	2
1.3. Manfaat Proyek Akhir.....	2
1.4. Tahap Penyelesaian Proyek Akhir.....	3
BAB II ISI DAN PEMBAHASAN.....	4
2.1. Komponen yang Digunakan.....	4
2.2. Rancangan Arsitektur <i>Cloud Computing</i>	4
2.3. Parameter dan Konfigurasi.....	8
2.4. Tahap Implementasi.....	199
2.5. Hasil Implementasi	53
2.6. Pengujian Singkat	554
BAB III JADWAL Pengerjaan dan Pembagian Tugas	55
3.1. Agenda Pengerjaan	55
3.2. Keterangan Pembagian Tugas	55
BAB IV KESIMPULAN DAN SARAN.....	56
4.1. Kesimpulan.....	56
4.2. Saran	56
DAFTAR PUSTAKA	57
LAMPIRAN	58

BAB I

PENDAHULUAN

1.1. Latar Belakang Proyek Akhir

Pada awalnya Cloud Computing didasari ketika John McCarthy menyampaikan visi bahwa “suatu hari nanti komputasi akan menjadi infrastruktur publik seperti listrik dan telpon”. Kemudian pada tahun 1995 ketika Larry Ellison yang merupakan pendiri Oracle menawarkan ide bahwa sebetulnya user tidak memerlukan berbagai software, mulai dari Sistem Operasi dan berbagai software lain, dijejalkan ke dalam komputer mereka yang mengakibatkan Amazon pada tahun 2000 berperan penting dalam pengembangan Cloud Computing dengan memodernisasi pusat data lalu berkembang menjadi Amazon Web Service (AWS) pada tahun 2006. Cloud Computing sendiri merupakan kombinasi pemanfaatan teknologi komputer dengan pengembangan berbasis internet. Jenis layanan dari Cloud Computing sendiri antara lain Infrastructure as a Service (IaaS) yang merupakan Layanan Cloud Computing ini memberikan penyewaan infrastruktur yang memberi pilihan penggunaanya dalam menyesuaikan kebutuhan yang ingin digunakan. Pilihan yang diberikan diantaranya adalah CPU, RAM, Penyimpanan Data dan Keamanan Jaringan. Platform as a Service (PaaS) yang merupakan Layanan Cloud Computing ini memberikan pelayanan yang hanya mengizinkan penggunaanya mengolah Platform saja tanpa perlu melakukan konfigurasi dibagian lainnya. Contoh dari PaaS ini adalah Amazon Web Service, Microsoft Azure, Facebook, dll. Software as a Service (SaaS) yang merupakan Layanan Cloud Computing ini pengguna tidak perlu memikirkan bagaimana mengurus aplikasi dikarenakan hal tersebut merupakan tanggung jawab dari vendor. Contoh dari SaaS ini adalah Google Apps (Docs, Spreadsheet, dll), Office 365, dan Adobe Creative Cloud. Anything as a Service (XaaS) yang merupakan Layanan Cloud Computing ini didasari dari suatu bisnis yang dapat memangkas pengeluaran mereka dari membeli produk perangkat lunak yang diinstal di aplikasi mereka dengan cara berlangganan. Layanan ini memungkinkan penggunaanya meningkatkan pendapatan bisnis mereka karena mereka dapat menyesuaikan kebutuhan mereka. Beberapa poin yang diperhitungkan dalam penerapan produk ini agar dapat sesuai dengan kebutuhan Cloud Computing antara lain:

1. Menganalisis pelanggan yang menggunakan smartphone dan terkoneksi ke internet.
2. Menginstall OS Linux dengan VMWare Workstation 15.
3. Menginstall kebutuhan untuk web server dan akses basisdata hingga berhasil.

4. Menginstall projek dengan menggunakan WinSCP dan berhasil.
5. Menginstall docker tetapi belum dapat diterapkan dengan baik dengan sistem ticketing.
6. Menguji hasil dari yang telah diterapkan di langkah-langkah sebelumnya.

Dari beberapa penelitian yang telah dijabarkan, masih terdapat beberapa kekurangan, diantaranya adalah penggunaan algoritma yang memiliki keterbatasan pada proses deteksi, yaitu hanya mampu mendeteksi satu objek pada satu proses deteksi. Kekurangan lainnya adalah penggunaan teknik preprocessing yang masih sederhana, sehingga sistem belum mampu menghasilkan representasi objek dengan baik yang berpengaruh pada ketepatan proses deteksi. Sehingga dari permasalahan yang telah dijabarkan, solusi yang diusulkan pada penelitian ini adalah penggunaan multi-frame detection untuk proses deteksi kendaraan sehingga mampu mendeteksi kendaraan dengan jumlah yang lebih banyak serta dapat mengetahui kecepatan yang dihasilkan dari kendaraan yang melintasi area pemantauan, dan penggunaan cascaded mask untuk melakukan proses filtering citra sehingga mampu menghasilkan data citra yang baik untuk berbagai kondisi pencahayaan yang berbeda.

1.2 Tujuan Proyek Akhir

Berdasarkan latar belakang proyek akhir yang telah dijelaskan sebelumnya, mengenai tujuan dari pembuatan proyek akhir ini adalah sebagai berikut:

1. Mengimplementasikan arsitektur *cloud computing* untuk Draft Dokumen Rusunawa yang telah dibuat sebelumnya dengan menggunakan Ubuntu 18.04 dan LAMPP (Apache 8, PHP 7.2, MySQL 5.2). Tujuan dari diimplementasikannya program ini kedalam *Cloud Computing* adalah mempermudah akses dari penyewa Rusunawa dalam mencetak surat yang akan diajukan ke Pihak BPJS Ketenagakerjaan.

1.3 Manfaat Proyek Akhir

Manfaat yang dapat diperoleh dari pembuatan proyek akhir ini adalah sebagai berikut:

1. Sistem Monitoring yang telah dirancang dapat digunakan sewaktu-waktu tanpa khawatir *downtime* dikarenakan terdapatnya *primary* dan *backup server* yang menggunakan arsitektur *recovery* pada *cloud computing*.

2. Perusahaan tidak perlu memperlakukan *maintenance*, dikarenakan dengan menggunakan *cloud computing*, rutinitas *maintenance* akan dilakukan sepenuhnya oleh *vendor*.

3. Batasan memori penyimpanan multimedia menjadi tidak terbatas dikarenakan sistem telah sepenuhnya beralih menggunakan *cloud computing*.

4. Permintaan data secara *realtime* dapat dilakukan secara terpusat maupun secara terdistribusi dikarenakan setiap sistem terhubung satu sama lain melalui *private cloud network*.

1.4 Tahap Penyelesaian Proyek Akhir

Tahapan secara singkat untuk penyelesaian proyek akhir ini adalah sebagai berikut:

1. Menganalisis kebutuhan dari sistem Draft Dokumen Rusunawa untuk ditransformasikan ke dalam arsitektur *cloud computing* menggunakan basis SaaS.
2. Mengintegrasikan sistem ke dalam arsitektur *cloud computing* dengan menggunakan virtualisasi OS Linux Ubuntu menggunakan VMWare Workstation 15.
3. Menggunakan software WinSCP dalam pemindahan data dari Windows ke Ubuntu.
4. Melakukan konfigurasi Ubuntu Server dengan menginstall docker sehingga pendistribusian program lebih mudah.
5. Menguji keandalan arsitektur *cloud computing* yang dibangun dengan beberapa pengujian yaitu kecepatan waktu akses, batasan akses sesuai konfigurasi, dsb.

BAB II

ISI DAN PEMBAHASAN

2.1 Komponen yang Digunakan

Komponen yang digunakan pada tugas proyek akhir ini terbagi menjadi dua sesuai dengan judul yang telah dituliskan sebelumnya. Berikut akan dibahas keduanya masing-masing dalam sub bab terkait.

2.1.1 Komponen pada Proyek Pertama

Untuk membangun “Draft Dokumen Rusunawa menggunakan Ubuntu LAMPP” yang berbasiskan konsep *cloud computing*, maka diperlukan analisis berbagai komponen. Berikut akan dijelaskan terlebih dahulu dalam bentuk poin-poin singkat:

1. Sistem yang telah dibangun menggunakan bahasa pemrograman PHP dengan versi 7.2 dan bahasa HTML dengan versi minimal 4.0.
2. Selain itu juga diperlukan penyimpanan basis data dengan arsitektur penyimpanan MySQL versi 5.2 sehingga dapat digunakan untuk menyimpan berbagai data dokumen rusunawa yang dibutuhkan oleh sistem tersebut.
3. Untuk target pengguna dengan konsep *cloud computing*, maka penggunaannya ialah pegawai di bagian Direktorat Perencanaan Strategis dan Teknologi Informasi di BPJS TK Jakarta Selatan. Tidak ada yang dapat mengakses sistem tersebut kecuali harus terhubung melalui jaringan intranet BPJS TK. Sehingga diperlukan arsitektur *cloud computing* yang bersifat *private*, tidak dapat diakses secara bebas oleh semua orang kecuali orang yang berkepentingan.
4. Sistem tersebut juga harus dapat digunakan sewaktu-waktu dan ditargetkan memiliki nilai *uptime* SLA 99,9% sehingga diperlukan minimal dua buah Ubuntu Server, satu sebagai *primary server* dan salah satunya sebagai *backup server*. Mekanisme untuk peralihan antar *server* membutuhkan *proxy server* dan semuanya menggunakan sistem operasi Ubuntu.

Berdasarkan penjelasan poin-poin tersebut, sistem akan dibuat dua kali. Yang pertama digunakan sebagai percobaan. Yang kedua digunakan sebagai hasil dari percobaan yang pertama. Pertama-tama dilakukan percobaan terlebih dahulu dengan menggunakan Laptop Acer. Untuk komponen utama penyusun *cloud computing* yang digunakan untuk percobaan dapat dilihat dalam bentuk tabel sebagai berikut:

Tabel 2.1 Spesifikasi VM *cloud computing* untuk Laptop Percobaan

No.	Nama Parameter	Nilai	Keterangan
1.	VMware ESXi <i>version</i>	15.0.1	Keterangan versi yang digunakan.
2.	VMware ESXi <i>build</i>	10737736	Keterangan <i>build (patch) number</i> .
3.	IPv4 <i>hypervisor</i>	IP: 192.168.44.1 (Static)	IP <i>number</i> untuk akses manajemen <i>hypervisor</i> melalui <i>web based</i> . Pengaturan IP <i>address</i> menggunakan mode statik.
		SM: 255.255.255.0	Kelas IP/ <i>subnet mask</i> yang digunakan.
		DNS: 192.168.1.9	Alamat IP untuk fungsionalitas DNS.
		GW: 192.168.1.9	Alamat untuk <i>gateway</i> atau gerbang menuju akses jaringan luar.
4.	<i>Processor info</i>	AMD Ryzen 5 (2500U, 2 GHz, 4 MB, 4 Cores)	Jenis <i>processor</i> yang digunakan pada <i>hypervisor</i> .
5.	RAM <i>info</i>	16 GB Memory	Kapasitas RAM pada <i>hypervisor</i> .

Setelah dilakukan percobaan, Laptop yang digunakan sebagai penyusun *cloud computing* dengan menggunakan Laptop HP. Komponen utama penyusun *cloud computing* dapat dilihat dalam bentuk tabel sebagai berikut:

Tabel 2.2 Spesifikasi VM *cloud computing* untuk Proyek Pertama

No.	Nama Parameter	Nilai	Keterangan
1.	VMware ESXi <i>version</i>	15.5.1	Keterangan versi yang digunakan.
2.	VMware ESXi <i>build</i>	15018445	Keterangan <i>build (patch) number</i> .
3.	IPv4 <i>hypervisor</i>	IP: 192.168.1.109 (Static)	IP <i>number</i> untuk akses manajemen <i>hypervisor</i> melalui <i>web based</i> . Pengaturan IP <i>address</i> menggunakan mode statik.
		SM: 255.255.255.0	Kelas IP/ <i>subnet mask</i> yang digunakan.
		DNS: 192.168.1.109	Alamat IP untuk fungsionalitas DNS.
		GW: 192.168.1.1	Alamat untuk <i>gateway</i> atau gerbang menuju akses jaringan luar.
4.	<i>Processor info</i>	Intel® Core™ i7-9750H Processor (2.6 GHz base frequency, up to 4.5 GHz base with Intel® Turbo Boost Technology, 12 MB cache, 6 cores)	Jenis <i>processor</i> yang digunakan pada <i>hypervisor</i> .
5.	RAM <i>info</i>	8 GB Memory	Kapasitas RAM pada <i>hypervisor</i> .

Selain spesifikasi mengenai VM *cloud computing* tersebut, untuk spesifikasi yang digunakan dalam Ubuntu OS yang telah dibuat dalam VM tersebut adalah sebagai berikut:

Tabel 2.2 Spesifikasi Ubuntu OS untuk proyek pertama

No.	Nama Parameter	Nilai	Keterangan
1.	LAMPP	Apache 2.4.29	Preprosesor bahasa pemrograman HTML, termasuk CSS dan JS.
		PHP 7.2.24	Bahasa skrip yang digunakan untuk memproses web dinamis
		MySQL 5.7.29	Manajemen basis data
2.	Docker	-	-

2.1.2 Komponen pada Proyek Kedua

Pada proyek yang kedua ini, kita menggunakan docker untuk membangun “Draft Dokumen Rusunawa menggunakan Ubuntu LAMPP” yang berbasiskan konsep *cloud computing*, maka diperlukan analisis berbagai komponen. Berikut akan dijelaskan terlebih dahulu dalam bentuk poin-poin singkat:

1. Sistem yang telah dibangun menggunakan bahasa pemrograman PHP dengan versi 7.2 dan bahasa HTML dengan versi minimal 4.0.
2. Selain itu juga diperlukan penyimpanan basis data dengan arsitektur penyimpanan MySQL versi 5.2 sehingga dapat digunakan untuk menyimpan berbagai data dokumen rusunawa yang dibutuhkan oleh sistem tersebut.
3. Untuk target pengguna dengan konsep *cloud computing*, maka penggunanya ialah pegawai di bagian Direktorat Perencanaan Strategis dan Teknologi Informasi di BPJS TK Jakarta Selatan. Tidak ada yang dapat mengakses sistem tersebut kecuali harus terhubung melalui jaringan intranet BPJS TK. Sehingga diperlukan arsitektur *cloud computing* yang bersifat *private*, tidak dapat diakses secara bebas oleh semua orang kecuali orang yang berkepentingan.
4. Sistem tersebut juga harus dapat digunakan sewaktu-waktu dan ditargetkan memiliki nilai *uptime* SLA 99,9% sehingga diperlukan minimal dua buah Ubuntu Server, satu sebagai *primary server* dan salah satunya sebagai *backup server*. Mekanisme untuk peralihan antar *server* membutuhkan *proxy server* dan semuanya menggunakan sistem operasi Ubuntu.

Berdasarkan penjelasan poin-poin tersebut, sistem akan dibuat dua kali. Yang pertama digunakan sebagai percobaan. Yang kedua digunakan sebagai hasil dari percobaan

yang pertama. Pertama-tama dilakukan percobaan terlebih dahulu dengan menggunakan Laptop Acer. Untuk komponen utama penyusun *cloud computing* yang digunakan untuk percobaan dapat dilihat dalam bentuk tabel sebagai berikut:

Tabel 2.1 Spesifikasi VM *cloud computing* untuk Laptop Percobaan

No.	Nama Parameter	Nilai	Keterangan
1.	VMware ESXi <i>version</i>	15.0.1	Keterangan versi yang digunakan.
2.	VMware ESXi <i>build</i>	10737736	Keterangan <i>build (patch) number</i> .
3.	IPv4 <i>hypervisor</i>	IP: 192.168.44.1 (Static)	IP <i>number</i> untuk akses manajemen <i>hypervisor</i> melalui <i>web based</i> . Pengaturan IP <i>address</i> menggunakan mode statik.
		SM: 255.255.255.0	Kelas IP/ <i>subnet mask</i> yang digunakan.
		DNS: 192.168.1.9	Alamat IP untuk fungsionalitas DNS.
		GW: 192.168.1.9	Alamat untuk <i>gateway</i> atau gerbang menuju akses jaringan luar.
4.	<i>Processor info</i>	AMD Ryzen 5 (2500U, 2 GHz, 4 MB, 4 Cores)	Jenis <i>processor</i> yang digunakan pada <i>hypervisor</i> .
5.	RAM <i>info</i>	16 GB Memory	Kapasitas RAM pada <i>hypervisor</i> .

Setelah dilakukan percobaan, Laptop yang digunakan sebagai penyusun *cloud computing* dengan menggunakan Laptop HP. Komponen utama penyusun *cloud computing* dapat dilihat dalam bentuk tabel sebagai berikut:

Tabel 2.2 Spesifikasi VM *cloud computing* untuk Proyek Pertama

No.	Nama Parameter	Nilai	Keterangan
1.	VMware ESXi <i>version</i>	15.5.1	Keterangan versi yang digunakan.
2.	VMware ESXi <i>build</i>	15018445	Keterangan <i>build (patch) number</i> .
3.	IPv4 <i>hypervisor</i>	IP: 192.168.1.109 (Static)	IP <i>number</i> untuk akses manajemen <i>hypervisor</i> melalui <i>web based</i> . Pengaturan IP <i>address</i> menggunakan mode statik.
		SM: 255.255.255.0	Kelas IP/ <i>subnet mask</i> yang digunakan.
		DNS: 192.168.1.109	Alamat IP untuk fungsionalitas DNS.
		GW: 192.168.1.1	Alamat untuk <i>gateway</i> atau gerbang menuju akses jaringan luar.
4.	<i>Processor info</i>	Intel® Core™ i7-9750H Processor (2.6 GHz base frequency, up to 4.5 GHz base with Intel® Turbo Boost Technology, 12 MB cache, 6 cores)	Jenis <i>processor</i> yang digunakan pada <i>hypervisor</i> .
5.	RAM <i>info</i>	8 GB Memory	Kapasitas RAM pada <i>hypervisor</i> .

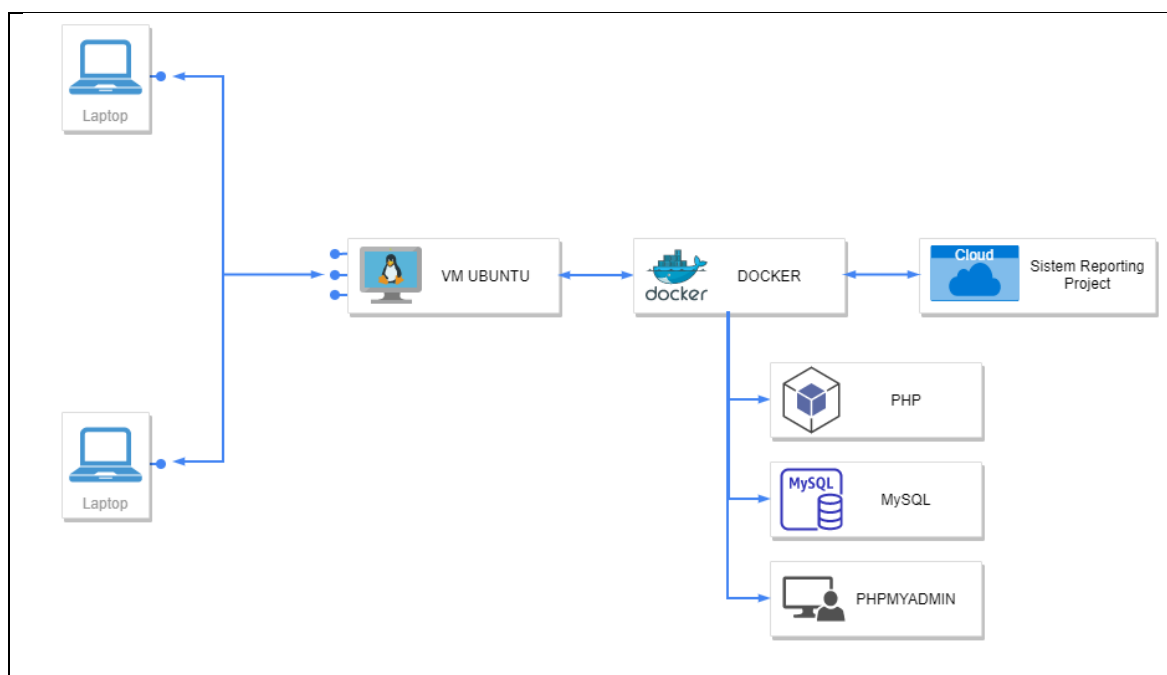
Selain spesifikasi mengenai VM *cloud computing* tersebut, untuk spesifikasi yang digunakan dalam Ubuntu OS yang telah dibuat dalam VM tersebut adalah sebagai berikut:

Tabel 2.2 Spesifikasi Ubuntu OS untuk proyek pertama

No.	Nama Parameter	Nilai	Keterangan
1.	LAMPP	Apache 2.4.29	Preprosesor bahasa pemrograman HTML, termasuk CSS dan JS.
		PHP 7.2.24	Bahasa skrip yang digunakan untuk memproses web dinamis
		MySQL 5.7.29	Manajemen basis data
2.	Docker	19.03.8	Versi Docker yang digunakan

2.2 Rancangan Arsitektur Cloud Computing

Pada proyek yang akhir ini digunakan rancangan arsitektur dari penerapan aplikasi “Draft Dokumen Rusunawa menggunakan Ubuntu LAMPP” menggunakan sistem operasi Linux Ubuntu dan virtualisasikan dengan VMWare Workstation 15. Ilustrasi dari rancangan arsitektur dari proyek ini dapat dilihat pada **Gambar 2.1** berikut ini :



Gambar 2.1 Rancangan Arsitektur

2.3 Parameter dan Konfigurasi

- Berikut adalah parameter dan konfigurasi yang digunakan dalam pengerjaan Proyek Pertama LAMPP :

- Parameter yang digunakan untuk mengecek IP address di *Windows CMD* (*Command Prompt*) pada penjelasan **Modul 2.1.1** berikut :

```
C:\Users\HP> ipconfig
```

Keterangan :

- ipconfig : digunakan untuk mengecek IP address yang digunakan di koneksi windows

Modul 2.1.1 Parameter mengecek IP Address di *Windows*

2. Parameter yang digunakan untuk menginstall ssh server di ubuntu terminal pada penjelasan **Modul 2.1.3** berikut :

```
nico@nico:~$ sudo apt-get install openssh-server openssh-client
```

Keterangan :

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- apt-get install : perintah untuk instalasi paket
- openssh-server : nama server dari server openssh
- openssh-client : client yang akan diinstall

3. Parameter yang digunakan mengecek IP Address di ubuntu terminal pada penjelasan **Modul 2.1.3** berikut :

```
nico@nico:~$ ifconfig
```

Keterangan :

- Ifconfig : perintah di ubuntu yang digunakan untuk mengecek IP Address yang dipakai

4. Parameter yang digunakan untuk instalasi Apache2 dapat dilihat pada penjelasan **Modul 2.1.4** berikut :

```
nico@nico:~$ sudo apt update
```

```
nico@nico:~$ sudo apt install apache2
```

Keterangan :

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- apt update : refresh repository index
- apt install : perintah untuk instal paket
- apache2 : nama file yang akan diinstall

5. Parameter yang digunakan untuk mengatur *firewall* Apache dapat dilihat pada penjelasan **Modul 2.1.5** berikut :

```
nico@nico:~$ sudo ufw allow in "Apache Full"
```

Keterangan :

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- ufw allow : perintah untuk mengizinkan aplikasi
- Apache Full : mengatur lalu lintas dari web server dengan protokol http (port 80) dan https (port 443)

6. Parameter yang digunakan untuk instalasi MySQL dapat dilihat pada penjelasan **Modul 2.1.6** berikut :

```
nico@nico:~$ sudo apt install mysql-server
```

keterangan :

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- apt install : perintah untuk instal paket
- mysql-server : nama paket yang akan diinstal, yaitu mysql-server

7. Parameter yang digunakan untuk pengaturan dasar pengamanan MySQL dapat dilihat pada penjelasan **Modul 2.1.7** berikut :

```
nico@nico:~$ sudo mysql_secure_installation
```

Keterangan :

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- mysql_secure_installation : perintah untuk mengatur password di mysql

8. Parameter yang digunakan untuk menginstall prosesor konten web server dengan PHP dapat dilihat pada penjelasan **Modul 2.1.8** berikut :

```
nico@nico:~$ sudo apt install php libapache2-mod-php php-mysql
```

Keterangan :

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- apt install : perintah untuk instal paket
- php libapache2-mod-php php-mysql : jenis paket yang akan diinstall, yaitu php

9. Parameter yang digunakan untuk membuat file nano dengan format php pada direktori *Web Server Apache* di `/var/www/html/` dapat dilihat pada penjelasan **Modul 2.1.9** berikut :

```
nico@nico:~$ sudo nano /var/www/html/info.php
```

Keterangan :

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- nano : membuka aplikasi nano (text editor ubuntu)
- /var/www/html/info.php : lokasi penyimpanan file beserta nama file

10. Parameter yang digunakan untuk membuat file php untuk mengecek versi php dapat dilihat pada penjelasan **Modul 2.1.10** berikut :

```
<?php
```

```
phpinfo();?>
```

Keterangan :

- phpinfo() : fungsi di php yang digunakan untuk melihat versi php

11. Parameter yang digunakan untuk instalasi phpmyadmin untuk layanan manajemen *database* MySQL berbasis Web GUI dapat dilihat pada penjelasan **Modul 2.1.11** berikut :

```
nico@nico:~$ sudo apt install phpmyadmin php-mbstring php-gettext
```

Keterangan :

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- apt install : perintah untuk instal paket
- phpmyadmin php-mbstring php-gettext : jenis paket yang akan diinstall, yaitu phpmyadmin

12. Parameter untuk masuk ke MySQL dengan mode CLI dapat dilihat pada penjelasan **Modul 2.1.12** berikut :

```
nico@nico:~$ sudo mysql -u root
```

ATAU

```
nico@nico:~$ sudo mysql -u root -p
```

Keterangan :

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- mysql -u root : perintah untuk masuk ke mysql dengan menggunakan username root tanpa memasukkan password

- `mysql -u root -p` : perintah untuk masuk ke mysql dengan menggunakan username root dengan memasukkan password

13. Parameter untuk mengatur parameter plugin dari user root dapat dilihat pada penjelasan **Modul 2.1.13** berikut :

```
nico@nico:~$ UPDATE mysql.user SET plugin = 'mysql_native_password',
authentication_string = PASSWORD('KATA_SANDI_ROOT_USER') WHERE User =
'root';
```

```
nico@nico:~$ FLUSH PRIVELEGES;
```

Keterangan :

- `UPDATE mysql.user` : update user mysql
- `SET plugin = 'mysql_native_password'` : menset plugin
- `authentication_string = PASSWORD('KATA_SANDI_ROOT_USER')` : menset kata autentikasi password dengan kata sandi root yang telah dibuat sebelumnya
- `WHERE User = 'root'` : dengan user yang bernama 'root'
- `FLUSH PRIVILEGES` : perintah untuk menyuruh server untuk membaca ulang tabel hak akses

14. Parameter untuk mengubah owner dari file dapat dilihat pada penjelasan **Modul 2.1.14** berikut :

```
nico@nico:~$ sudo chown nama_user /var/www/html
```

Keterangan :

- `sudo` : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- `chown nama_user` : change owner, mengubah owner dari file atau folder dengan menambahkan nama user yang akan menggantikan
- `/var/www/html` : direktori tempat pergantian owner

2. Berikut adalah parameter dan konfigurasi yang digunakan dalam pengerjaan Projek Kedua docker :

1. Parameter untuk install packages dan mengizinkan apt untuk menggunakan repository https dapat dilihat pada penjelasan **Modul 2.2.1** berikut :

```
$ sudo apt-get update
```

```
$ sudo apt-get install \apt-transport-https \ca-certificates \curl  
\gnupg-agent \ software-properties-common
```

Keterangan :

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- apt-get update : perintah untuk memperbaiki masalah dengan memperbaharui program komputer
- apt-get install : perintah untuk menginstall package untuk menggunakan repository HTTPS

2. Parameter untuk menambahkan key docker dapat dilihat pada penjelasan **Modul 2.2.2** berikut :

```
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-  
key add
```

Keterangan :

- Curl : singkatan dari Client URL, perintah untuk mengecek konektivitas ke URL docker
- Sudo : sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- Apt-key add : menambahkan key terpercaya ke apt untuk repositori

3. Parameter untuk memasukkan kode yang telah disediakan di dokumentasi docker dapat dilihat pada penjelasan **Modul 2.2.3** berikut :

```
sudo apt-key fingerprint 0EBFCD88
```

keterangan :

- Sudo : sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)

- Apt-key fingerprint : menambahkan key fingerprint terpercaya ke apt untuk repositori

4. Parameter untuk menambahkan repository stable dapat dilihat pada penjelasan **Modul 2.2.4** berikut :

```
$ sudo add-apt-repository \"deb [arch=amd64]
https://download.docker.com/linux/ubuntu \" $(lsb_release -cs) \"stable\"
```

keterangan :

- Sudo : sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- Add-apt repository : perintah untuk menambahkan repository

5. Parameter untuk menginstall docker engine dan containerd dapat dilihat pada penjelasan **Modul 2.2.5** berikut

```
$ sudo apt-get update
$ sudo apt-get install docker-ce docker-ce-cli containerd.io
```

Keterangan :

- Sudo : sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- Apt-get update : perintah untuk mendapatkan update
- Apt-get install : perintah untuk install docker

6. Parameter untuk mencoba docker apakah sudah berhasil atau belum dapat dilihat pada penjelasan **Modul 2.2.6** berikut :

```
$ sudo docker run hello-world
```

Keterangan :

- Sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- Docker run : perintah untuk menjalankan file docker

7. Parameter untuk cek status docker dapat dilihat pada penjelasan **Modul 2.2.7** berikut :

```
$ sudo systemctl status docker
```

Keterangan :

- Sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- Systemctl : perintah untuk mengaktifkan layanan docker

8. Parameter untuk menjalankan perintah docker tanpa sudo dapat dilihat pada penjelasan **Modul 2.2.8** berikut :

```
$ sudo usermod -aG docker $USER
```

Keterangan :

- Sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- Usermod : perintah untuk memasukan user ke dalam group docker

9. Parameter untuk menginstall docker-compose dapat dilihat pada penjelasan **Modul 2.2.9** berikut :

```
$ sudo curl -
```

```
L"https://github.com/docker/compose/releases/download/1.24.1/
docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-
compose
```

Keterangan :

- Sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- Curl : singkatan dari Client URL, perintah untuk mengecek konektivitas ke URL docker compose
- -o : perintah untuk membuka lokasi file

10. Parameter untuk mengecek versi dan mengatur hak akses docker-compose dapat dilihat pada penjelasan **Modul 2.2.10** berikut :

```
$ sudo chmod +x /usr/local/bin/docker-compose
```

```
$ docker-compose version
```

Keterangan :

- Sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- chmod : perintah untuk mengatur hak akses atau permission ke direktori docker compose diinstall

- `docker-compose version` : perintah untuk mengetahui versi dari `docker-compose`

11. Parameter untuk membuat direktori dan masuk ke direktori dapat dilihat pada penjelasan **Modul 2.2.11** berikut :

```
$ mkdir -p ~/docker/lamp/html
$ cd ~/docker/lamp
```

Keterangan :

- `Mkdir` : singkatan dari `make directory`, perintah untuk membuat direktori baru
- `Cd` : perintah untuk masuk ke folder

12. Parameter untuk isi file dari `php.Dockerfile` dapat dilihat pada penjelasan **Modul 2.2.12** berikut :

```
From php:7.4.3-apache
RUN docker-php-ext-install mysqli pdo pdo_mysql
```

Keterangan :

- `Run docker-php-ext-install mysqli pdo pdo_mysql` : perintah untuk menjalankan instalasi docker untuk php

13. Parameter untuk isi file dari `docker-compose.yml` dapat dilihat pada penjelasan **Modul 2.2.13** berikut :

```
version: "3.7"
services:
  web-server:
    build:
      dockerfile: php.Dockerfile
      context: .
    restart: always
    volumes:
      - "./html:/var/www/html/"
    ports:
      - "8080:80"
  mysql-server:
    image: mysql:8.0.19
    restart: always
```

```

environment:
  MYSQL_ROOT_PASSWORD: 123170096
volumes:
  - mysql-data:/var/lib/mysql

phpmyadmin:
  image: phpmyadmin/phpmyadmin:5.0.1
  restart: always
  environment:
    PMA_HOST: localhost:3306
    PMA_USER: root
    PMA_PASSWORD: 123170096
  ports:
    - "5000:80"
  volumes:
    mysql-data:

```

Keterangan :

- Services : jenis service apa saja yang dibutuhkan
- Web-server : jenis web-server yang akan digunakan
- Mysqlq-server : jenis mysql-server yang akan digunakan
- Phpmyadmin : jenis phpmyadmin yang akan digunakan

14. Parameter untuk menjalankan file docker-compose dapat dilihat pada penjelasan **Modul 2.2.14** berikut :

```

$ docker-compose up -d

```

Keterangan :

- Docker-compse up : perintah untuk menjalankan file docker-compose

15. Parameter untuk melihat service dari docker yang telah berjalan dapat dilihat pada penjelasan **Modul 2.2.15** berikut :

```

$ docker-compose ps

```

Keterangan :

- Docker-compose ps : perintah untuk mengecek service yang berjalan pada docker

16. Parameter untuk menyegarkan data docker dapat dilihat pada penjelasan **Modul 2.2.16** berikut :

```

$ docker-compose up -d

```

Keterangan :

- Docker-compse up : perintah untuk menjalankan file docker-compse

2.4 Tahap Implementasi

Di sini akan dijelaskan implementasi tentang penerapan layanan hosting dengan LAMPP (SAAS) beserta Dockernya

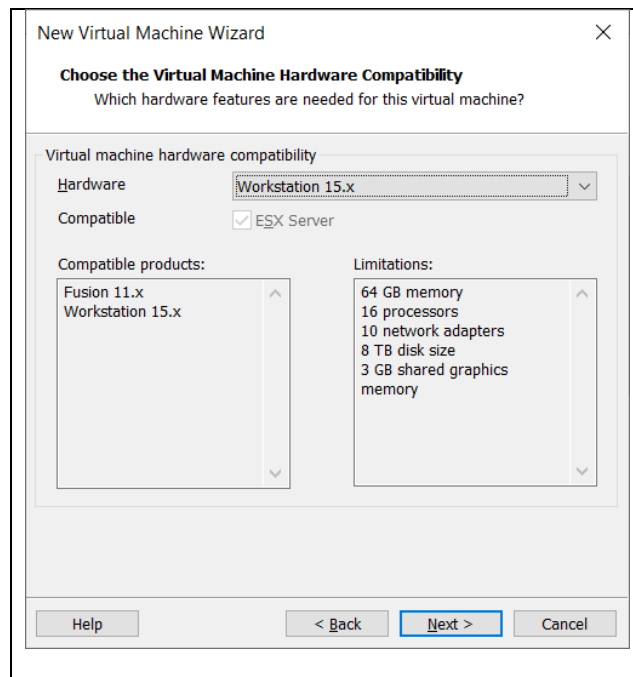
1. penerapan layanan hosting dengan LAMPP (SAAS)
 1. Pilih New Virtual Machine Wizard di VMWare Workstation. Lalu pilih yang Typical

Gambar 2.1 Konfigurasi awal pembuatan virtual machine wizard



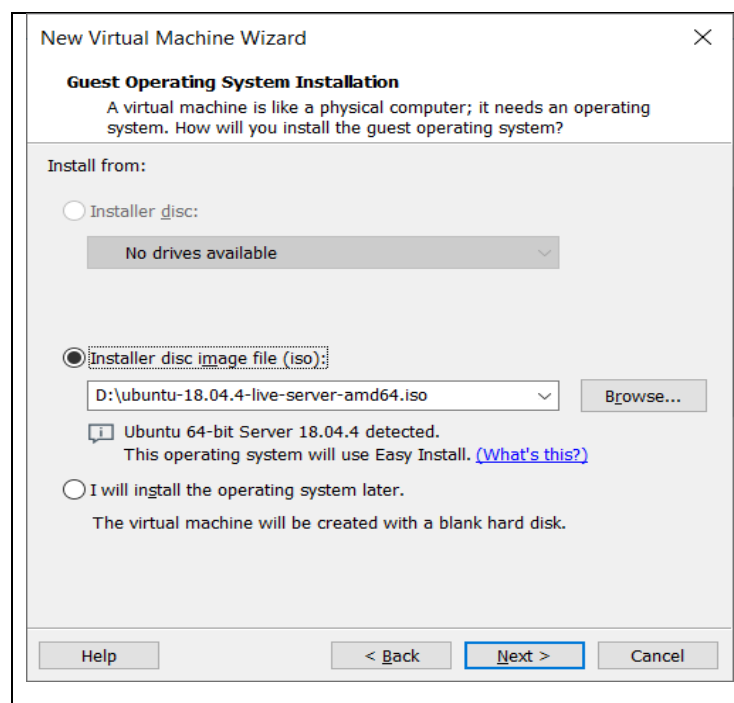
2. Pilih next

Gambar 2.2. Konfigurasi awal pembuatan virtual machine wizard



3. Pilih iso linux yang tadi sudah diunduh. Pilih next.

Gambar 2.3. Pemilihan iso



Isi data diri. Data ini tidak begitu penting. Silahkan diisi acak tidak masalah. Pilih next.

Gambar 2.4. Isi data diri

New Virtual Machine Wizard

Easy Install Information
This is used to install Ubuntu 64-bit.

Personalize Linux

Full name: nico

User name: nico

Password:

Confirm:

Help < Back Next > Cancel

1. Pilih direktori file sesuai yang diinginkan dan masukkan nama virtual machine nya. Pilih next

Gambar 2.5. Pemilihan nama dan pemilihan lokasi

New Virtual Machine Wizard

Name the Virtual Machine
What name would you like to use for this virtual machine?

Virtual machine name: Ubuntu LAMPP

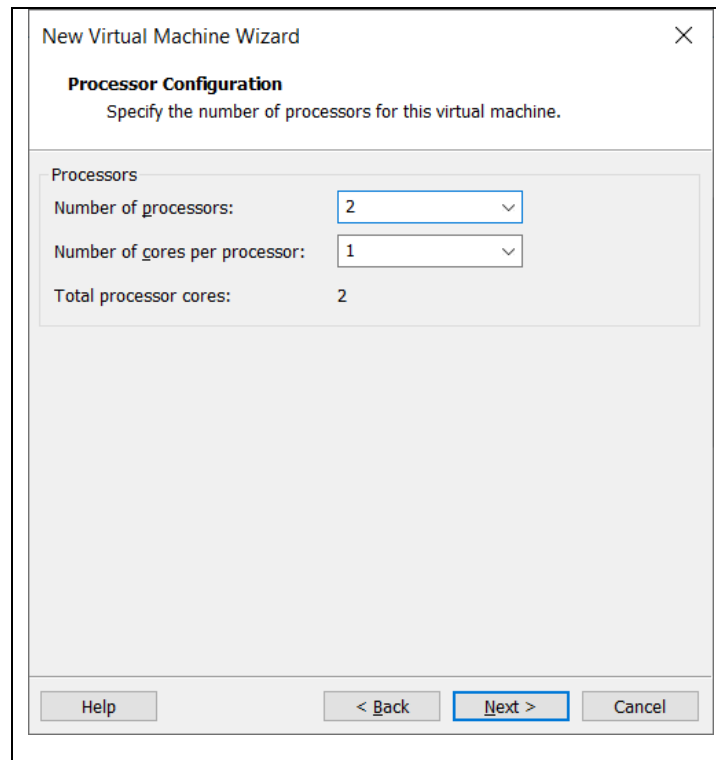
Location: D:\Ubuntu LAMPP 123170103 Browse...

The default location can be changed at Edit > Preferences.

< Back Next > Cancel

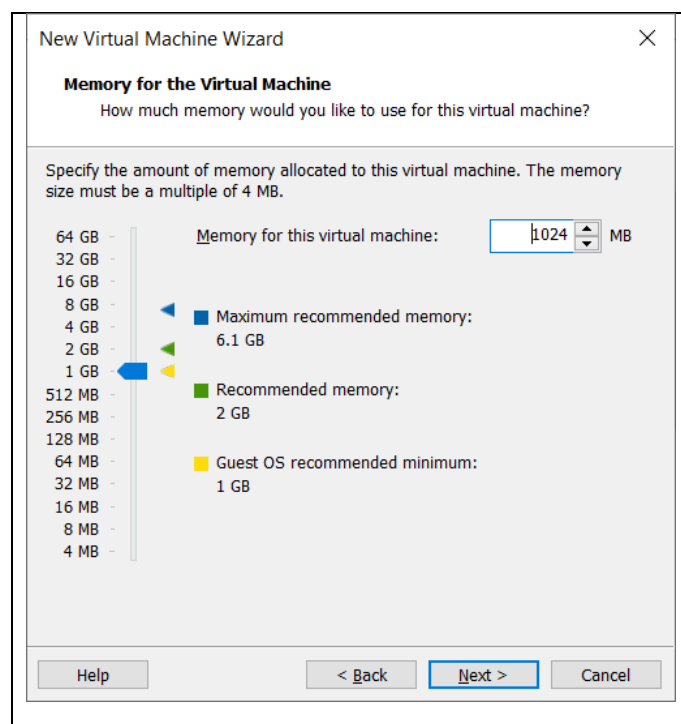
2. Pilih jumlah prosesor dan core sesuai kebutuhan. Di sini menggunakan 2 prosesor dan 1 core

Gambar 2.6. Pemilihan prosesor dan core



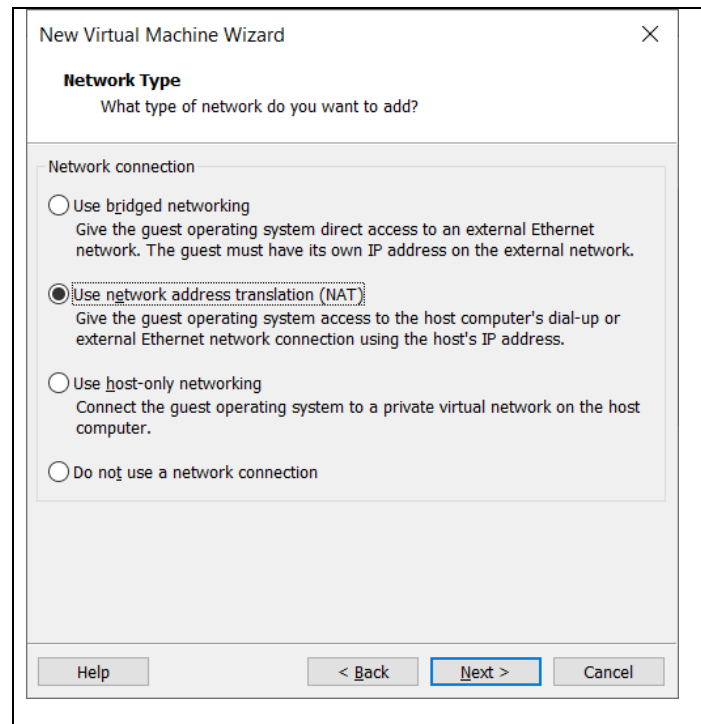
3. Pilih jumlah ram. Disini menggunakan 1 gb ram.

Gambar 2.7. Pemilihan ram



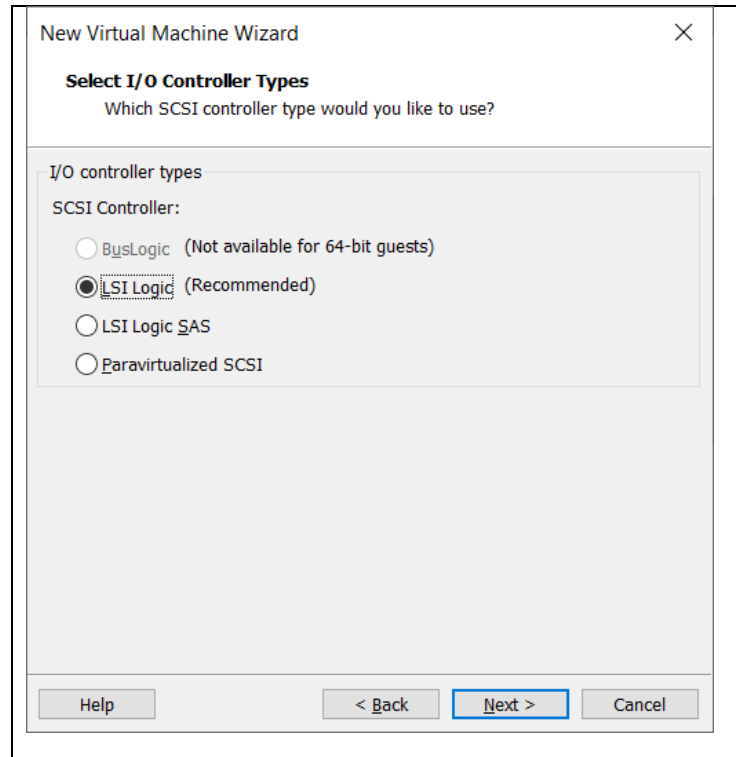
4. Pilih NAT. Pilih Next

Gambar 2.8. Pemilihan Network



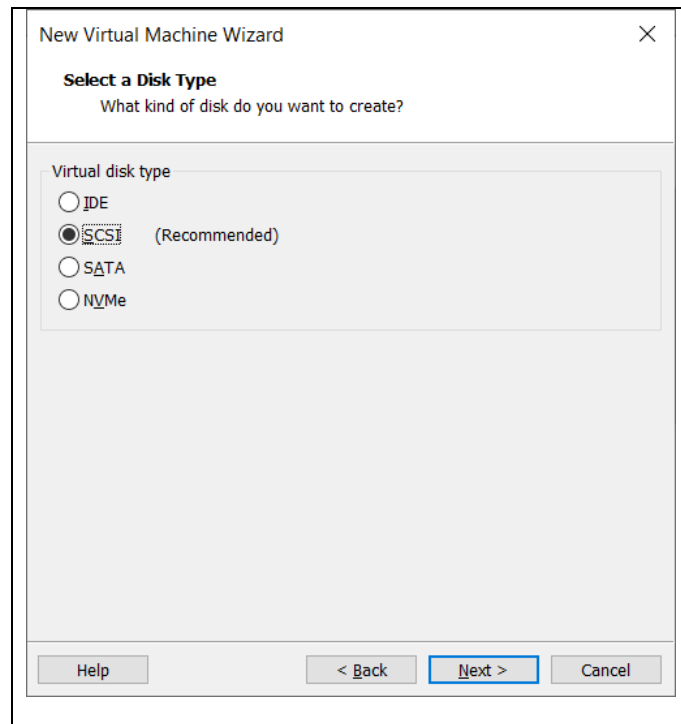
5. Pilih LSI logic

Gambar 2.9. Pemilihan IO Controller



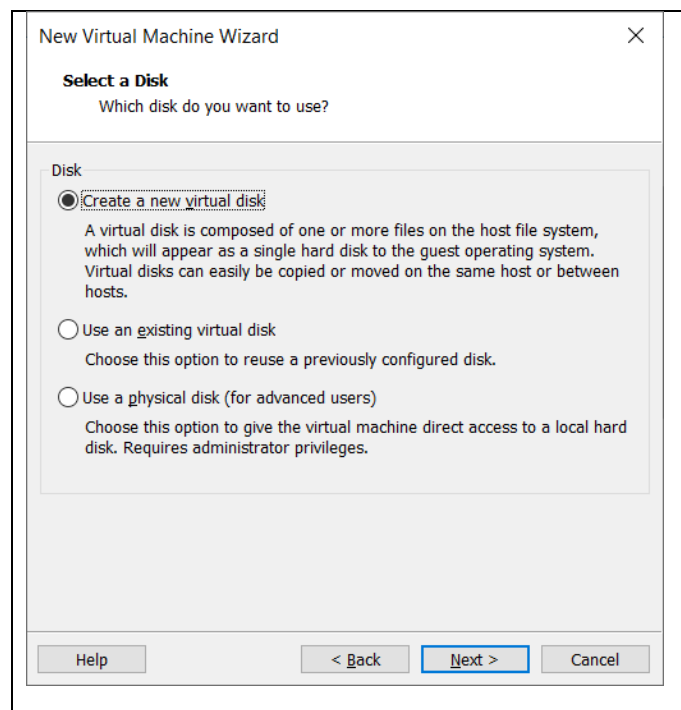
6. Pilih yang SCSI. Pilih next

Gambar 2.10. Pemilihan IO Controller



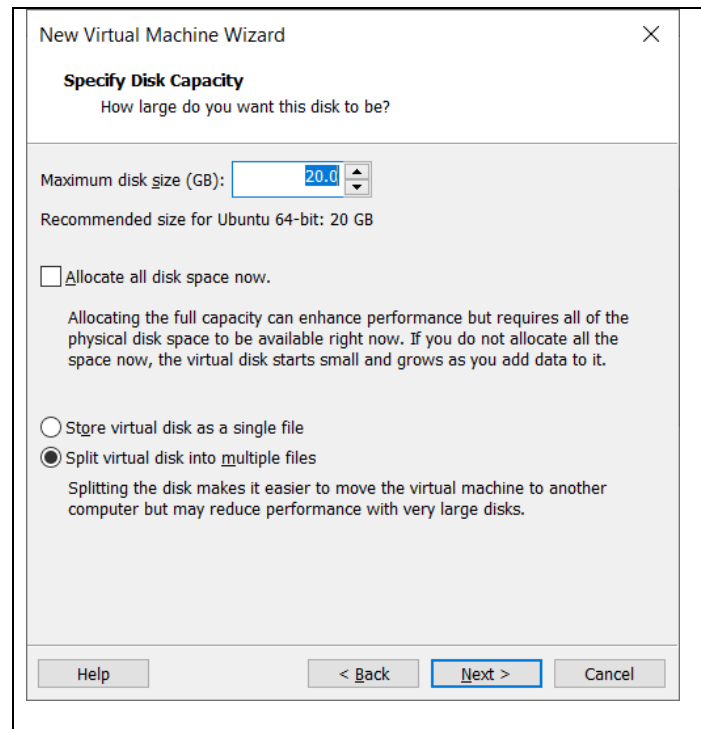
7. Pilih create a new virtual disk

Gambar 2.11. Pilih Create a New Virtual Disk



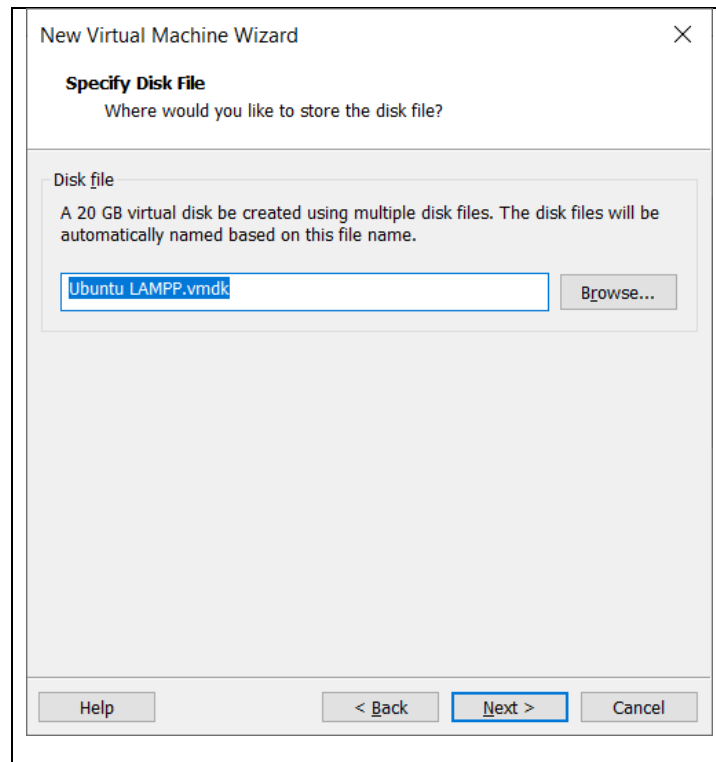
8. Pilih yang Split virtual disk into multiple device

Gambar 2.12. Pilih split virtual disk



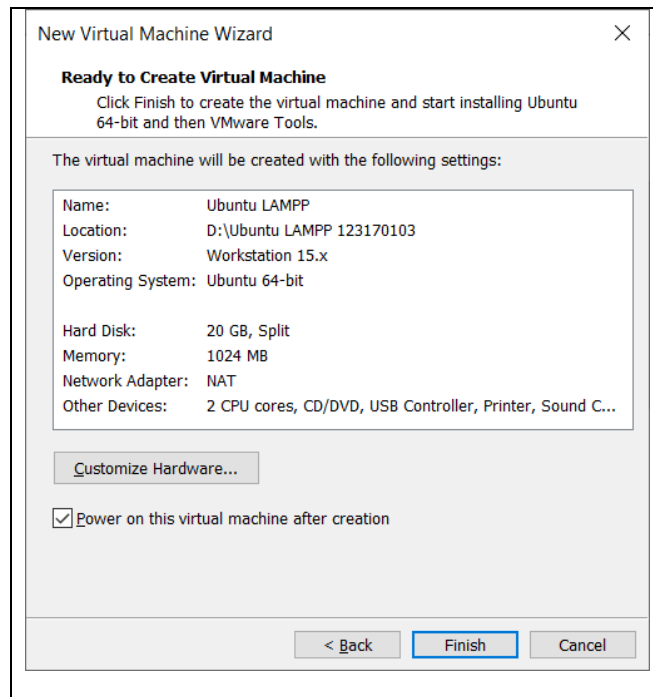
9. Pilih next

Gambar 2.13. Konfirmasi nama disk file



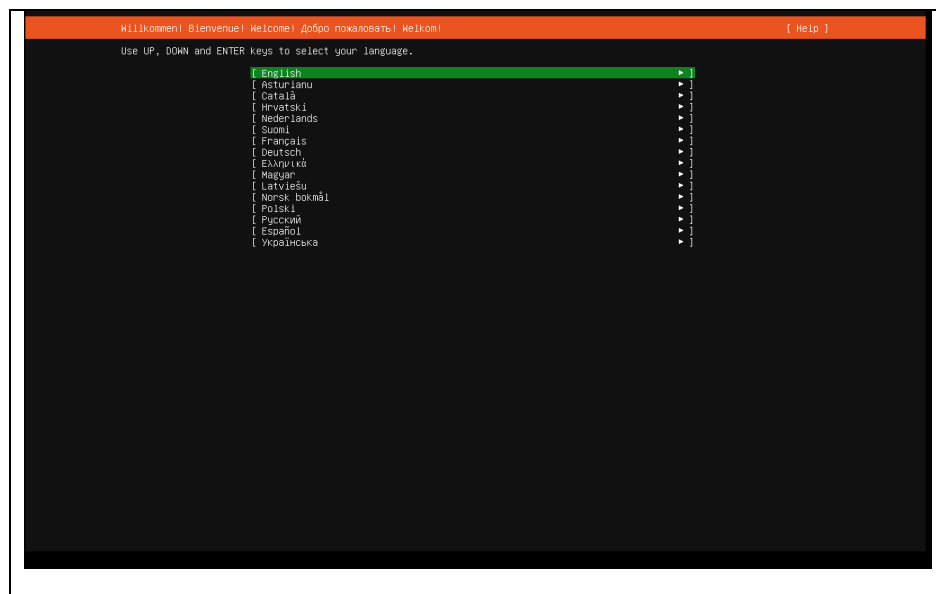
10. Ringkasan spesifikasi virtual machine. Pilih finish

Gambar 2.14. Ringkasan spesifikasi virtual machine



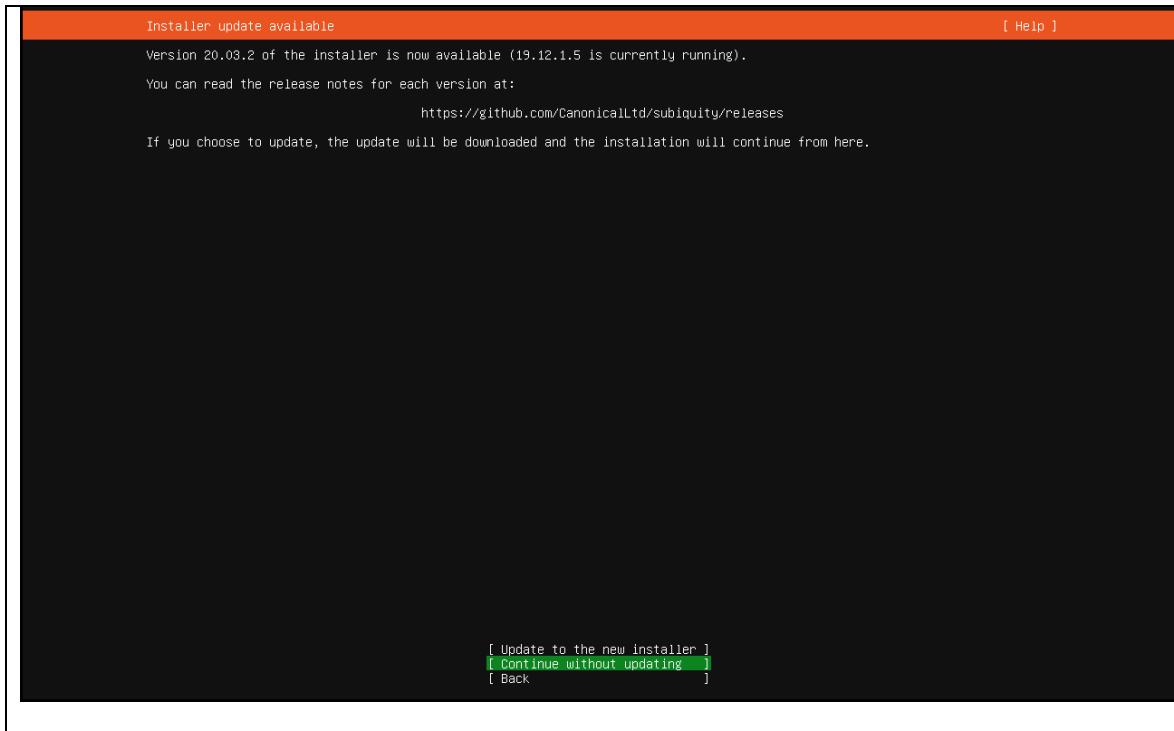
11. Akan muncul tampilan sbb. Pilih English

Gambar 2.15. Gambar instalasi bahasa linux



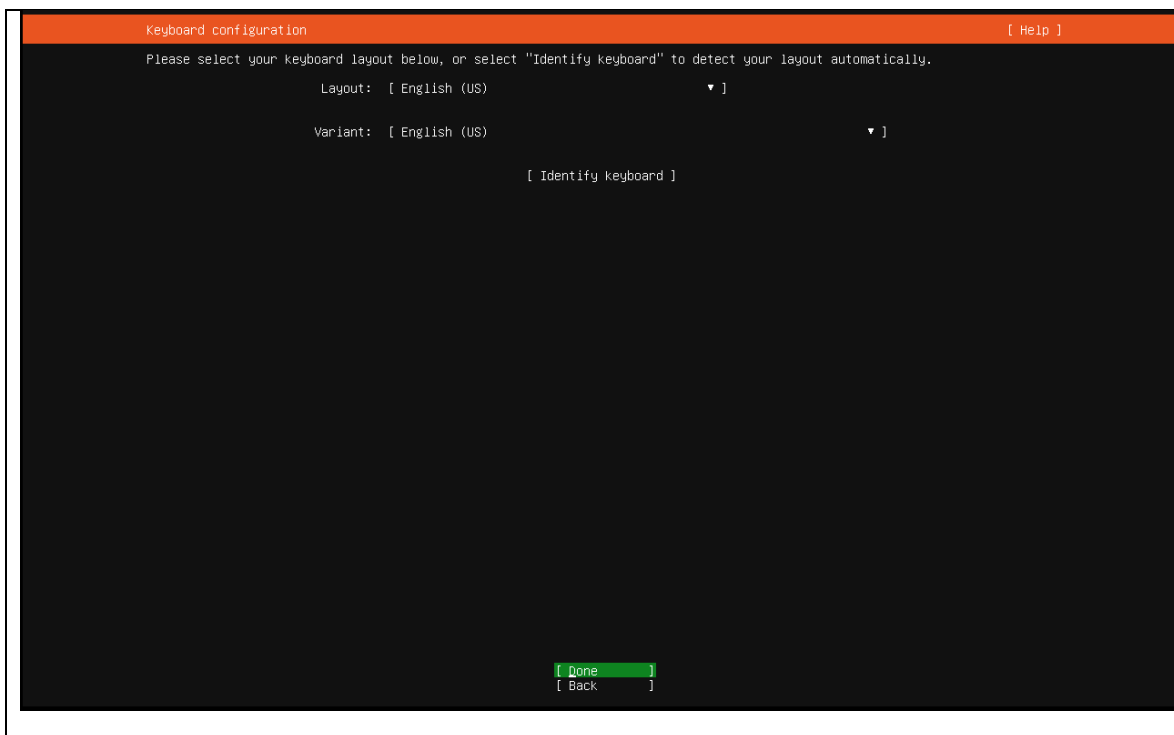
12. Akan ada pilihan update. Pilih yang continue without updating

Gambar 2.16. Gambar pilihan update



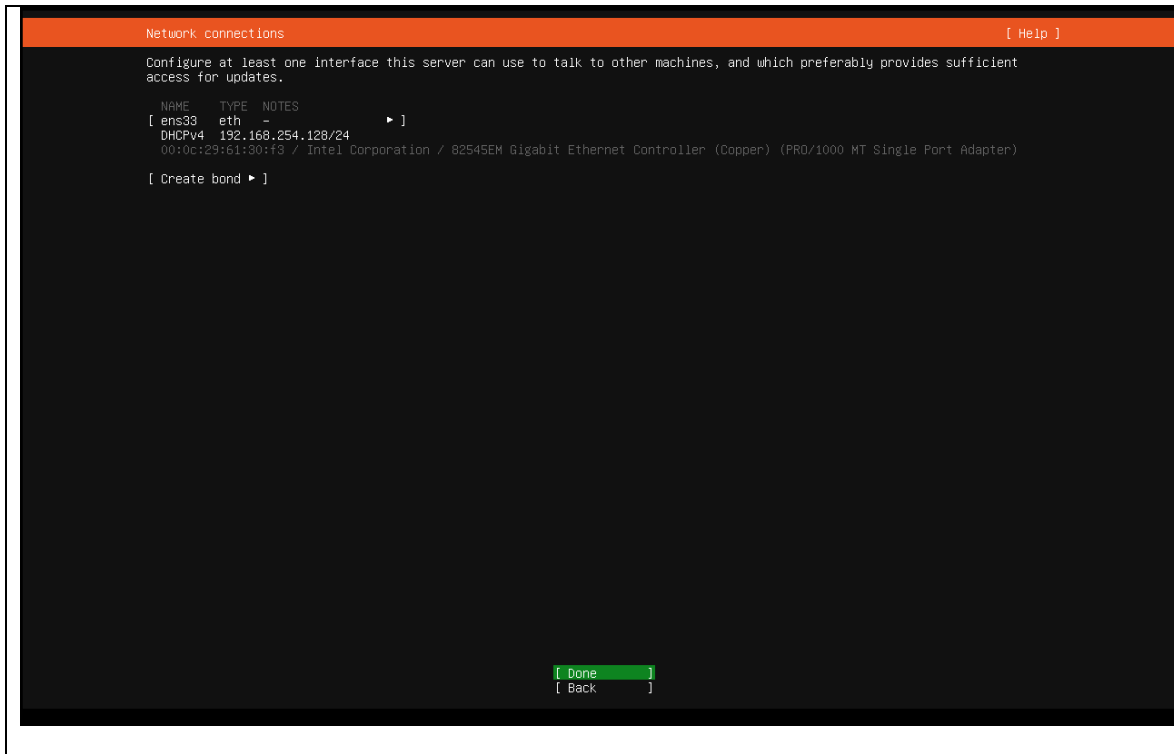
13. Akan ada pilihan konfigurasi keyboard. Pilih done.

Gambar 2.17. Konfigurasi keyboard



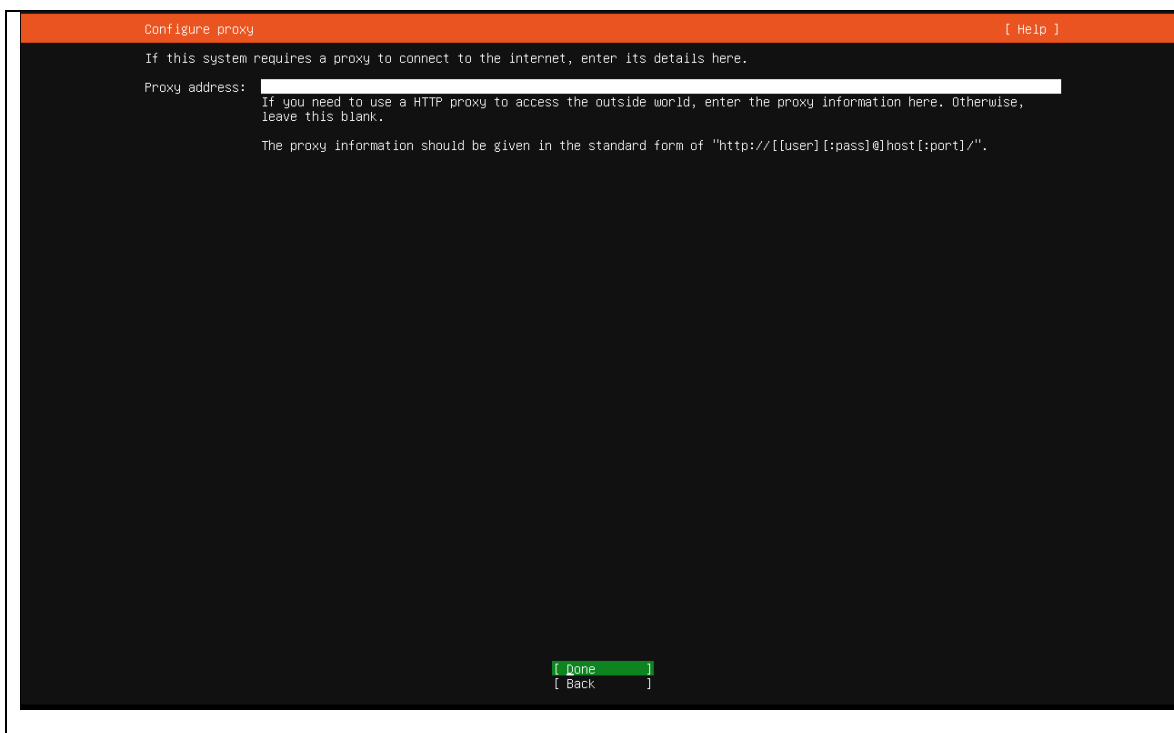
14. Konfigurasi network. Pilih done.

Gambar 2.18. Konfigurasi network



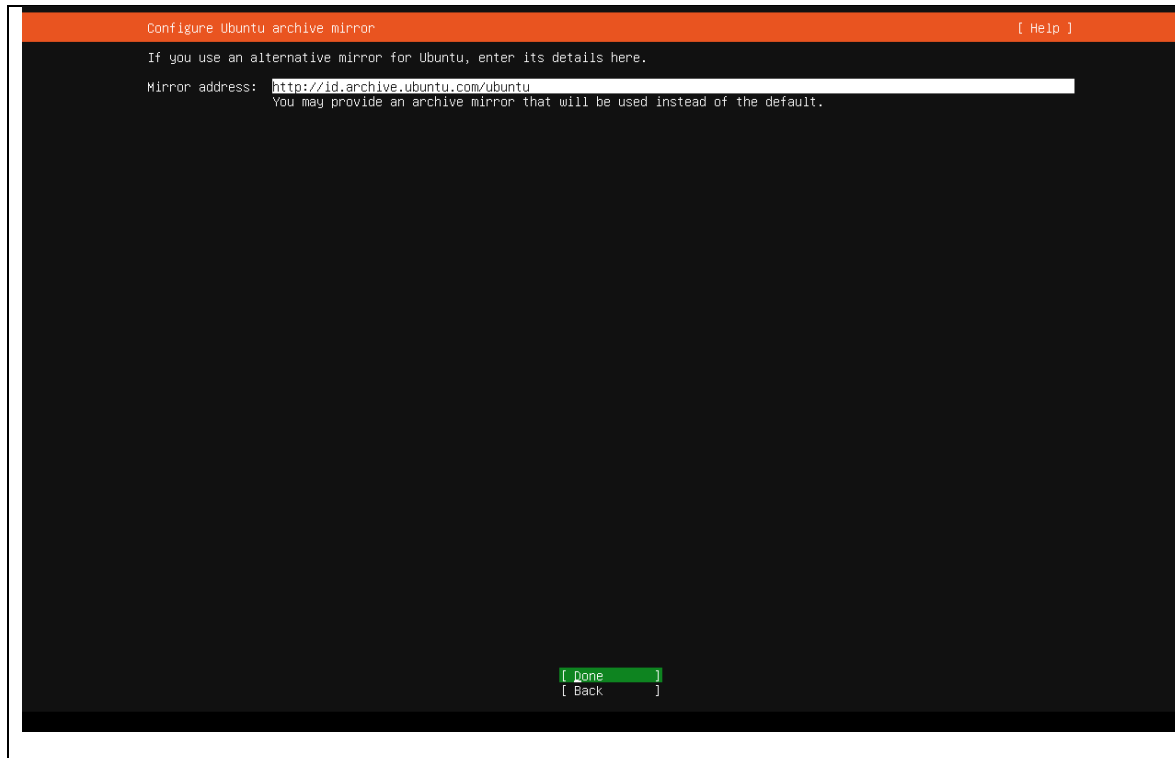
15. Konfigurasi proxy. Pilih done

Gambar 2.19. Konfigurasi proxy



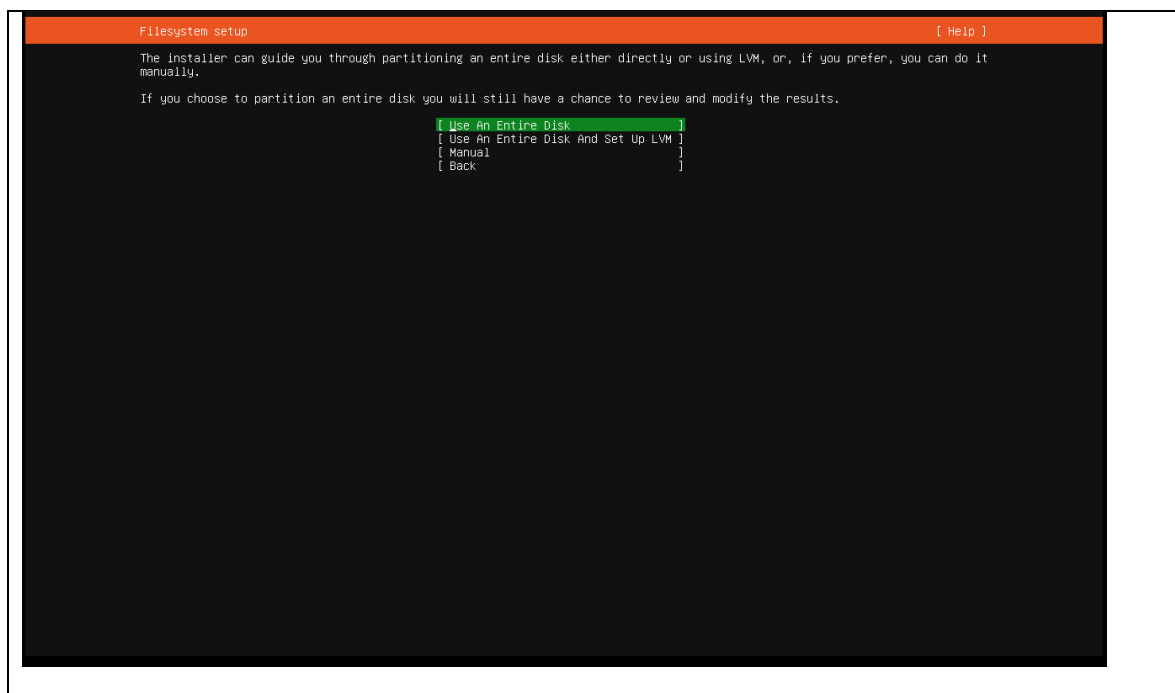
16. Konfigurasi ubuntu archive mirror. Pilih done

Gambar 2.20. Konfigurasi ubuntu archive mirror



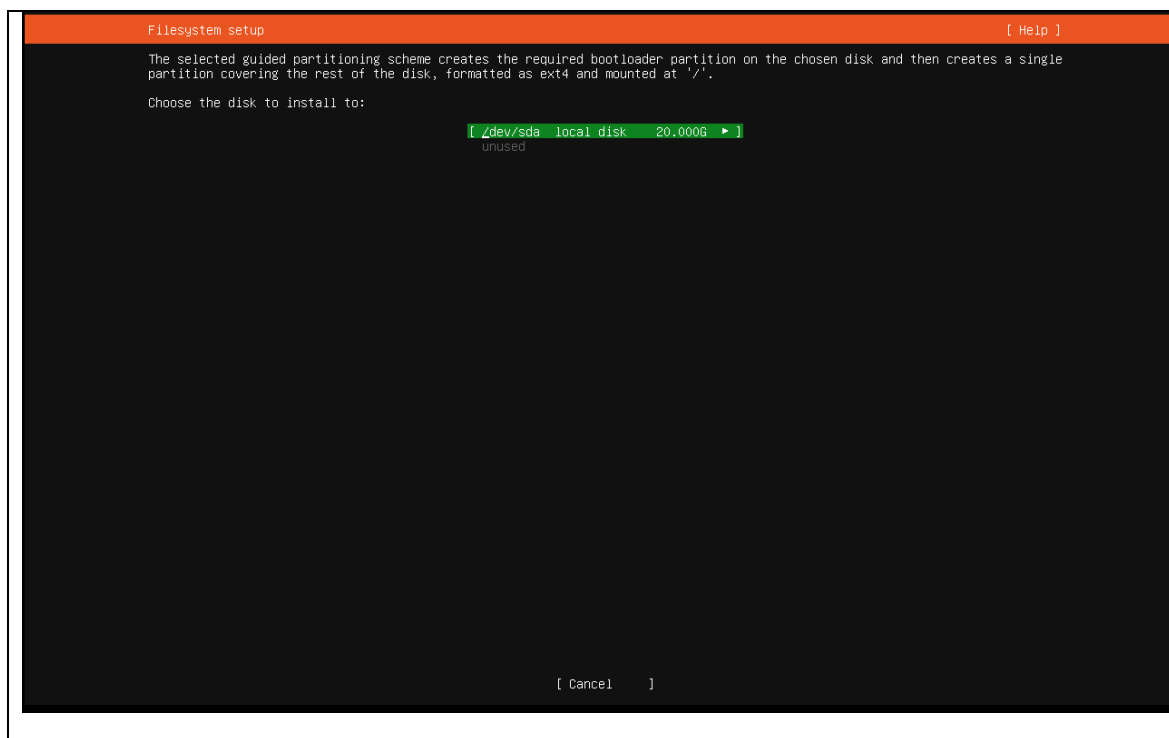
17. Konfigurasi filesystem. Pilih yang use an entire disk

Gambar 2.21. Konfigurasi filesystem



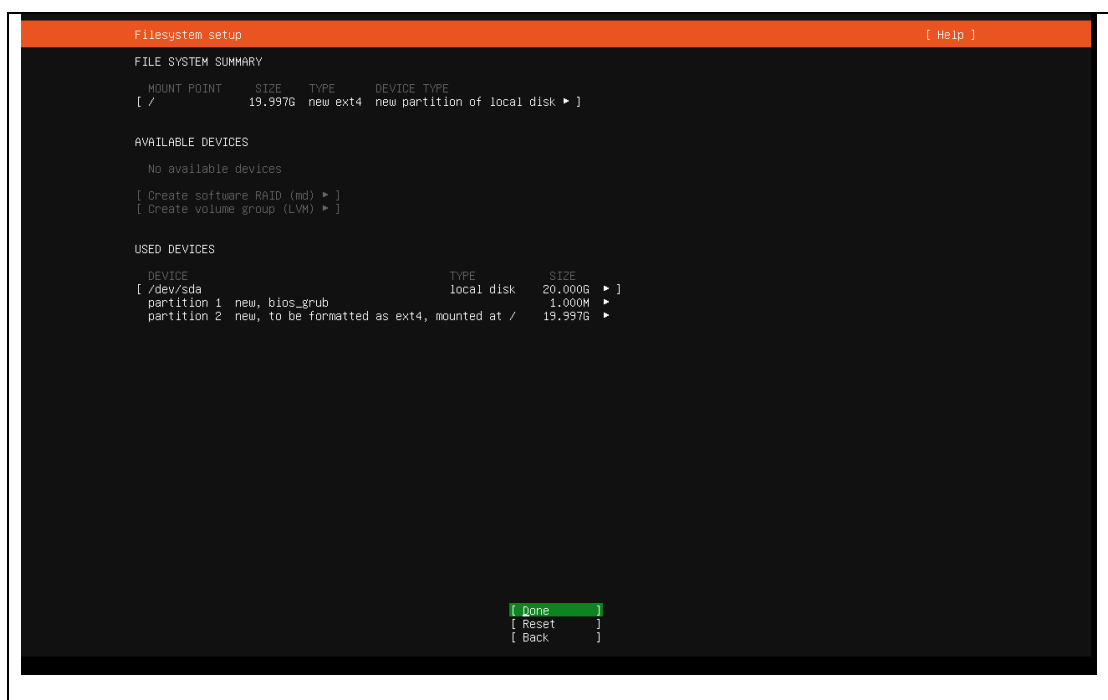
18. Pilih disk instalasi.

Gambar 2.22. Konfigurasi filesystem



19. Tampilan ringkasan filesystem. Pilih done

Gambar 2.23. Ringkasan filesystem



20. Pilih nama server dan password. Pilih done

Gambar 2.24. Gambar nama server dan password

Profile setup [Help]

Enter the username and password you will use to log in to the system. You can configure SSH access on the next screen but a password is still needed for sudo.

Your name: _____

Your server's name: _____
The name it uses when it talks to other computers.

Pick a username: _____

Choose a password: _____

Confirm your password: _____

[Done] [Back]

21. Tampilan featured server. Pilih done. Ditunggu sampai instalasi selesai

Gambar 2.25. Tampilan featured server

Featured Server Snaps [Help]

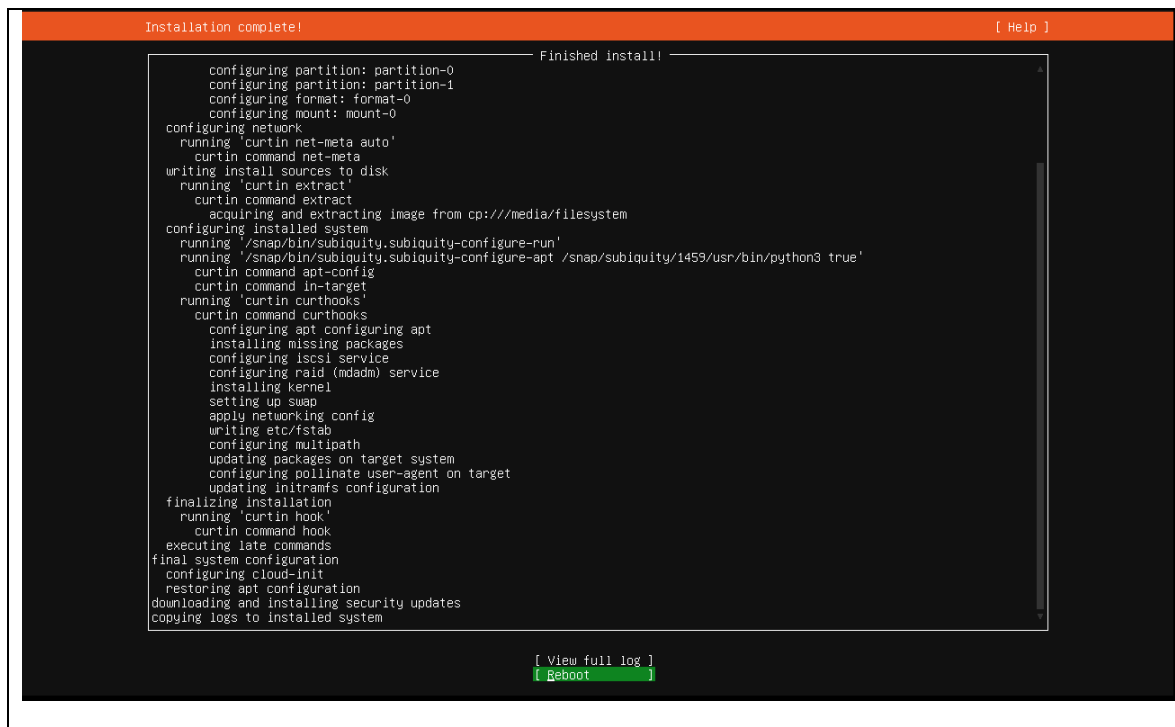
These are popular snaps in server environments. Select or deselect with SPACE, press ENTER to see more details of the package, publisher and versions available.

<input checked="" type="checkbox"/> microk8s	canonical	Kubernetes for workstations and appliances
<input type="checkbox"/> nextcloud	nextcloud	Nextcloud Server - A safe home for all your data
<input type="checkbox"/> wekan	xet7	Open-Source kanban
<input type="checkbox"/> kata-containers	katacontainers	Lightweight virtual machines that seamlessly plug into the containers ecosyste
<input type="checkbox"/> docker	canonical	Docker container runtime
<input type="checkbox"/> canonical-livepatch	canonical	Canonical Livepatch Client
<input type="checkbox"/> rocketchat-server	rocketchat	Group chat server for 100s, installed in seconds.
<input type="checkbox"/> mosquito	mosquitto	Eclipse Mosquitto MQTT broker
<input type="checkbox"/> etcd	canonical	Resilient key-value store by CoreOS
<input type="checkbox"/> powershell	microsoft-powershell	PowerShell for every system!
<input type="checkbox"/> stress-ng	cking-kernel-tools	A tool to load, stress test and benchmark a computer system
<input type="checkbox"/> sabnzbd	safihre	SABnzbd
<input type="checkbox"/> wormhole	snappy	get things from one computer to another, safely
<input type="checkbox"/> aws-cli	aws	Universal Command Line Interface for Amazon Web Services
<input type="checkbox"/> google-cloud-sdk	google-cloud-sdk	Command-line interface for Google Cloud Platform products and services
<input type="checkbox"/> slcli	softlayer	Python based SoftLayer API Tool.
<input type="checkbox"/> doctl	digitalocean	DigitalOcean command line tool
<input type="checkbox"/> conjure-up	canonical	Package runtime for conjure-up spells
<input type="checkbox"/> minidlna-escoand	escoand	server software with the aim of being fully compliant with DLNA/UPnP clients.
<input type="checkbox"/> postgresql10	cmd	PostgreSQL is a powerful, open source object-relational database system.
<input type="checkbox"/> heroku	heroku	CLI client for Heroku
<input type="checkbox"/> keepalived-project	keepalived-project	High availability VRRP/BFD and load-balancing for Linux
<input type="checkbox"/> prometheus	canonical-is-snaps	The Prometheus monitoring system and time series database
<input type="checkbox"/> juju	canonical	Simple, secure and stable devops. Juju keeps complexity low and productivity h

[Done] [Back]

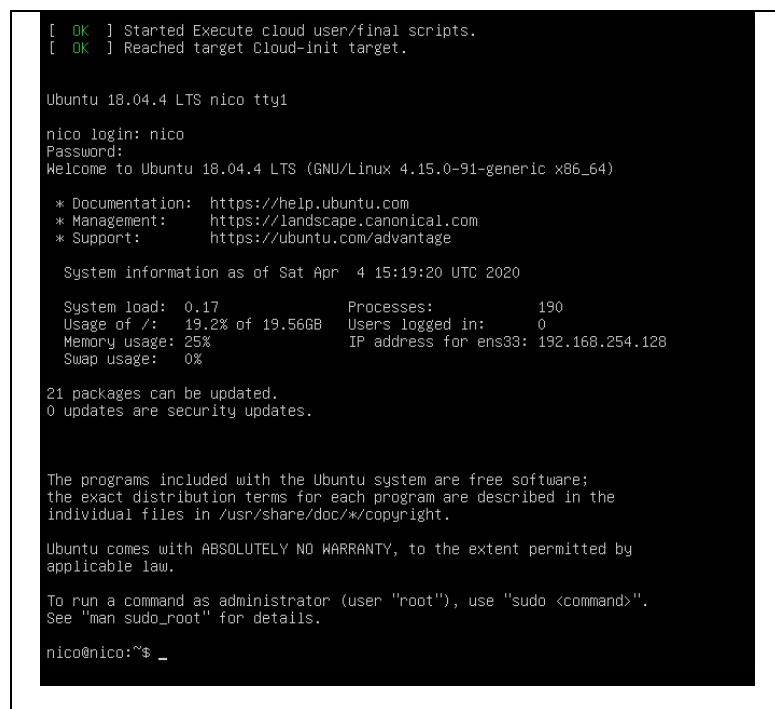
22. Hasil akhir akan seperti berikut. Pilih reboot.

Gambar 2.26. Tampilan selesai instalasi



23. Setelah selesai, login server. Setelah itu akan muncul tampilan seperti di bawah ini.

Gambar 2.27. Tampilan hasil login



24. Setelah itu ketikkan di cmd (command prompt) **Modul 2.1.1.**

Ipconfig digunakan untuk mengambil ip address yang sedang dipakai. Jika menggunakan ip address yang berada di linux maka akan muncul “Connection refused”. Alasannya karena tidak menginstall ssh terlebih dahulu pada saat instalasi OS linux.

Kalau ingin menggunakan ip address yang berada di os linux, maka ketik di terminal Ubuntu **Modul 2.1.2.** Fungsi dari **Modul 2.1.1** adalah untuk menginstall ssh yang apabila terlewat di langkah instalasi linux.

Jika ingin mengetahui ip address dari linux, maka lakukan perintah sesuai dengan **Modul 2.1.3.** Maka tampilannya akan seperti berikut :

Gambar 2.28. Mendapat hasil ip dari linux

```
Preparing to unpack .../ssh-import-id_5.7-0ubuntu1.1_all.deb ...
Unpacking ssh-import-id (5.7-0ubuntu1.1) ...
Setting up ncurses-term (6.1-1ubuntu1.18.04) ...
Setting up openssh-sftp-server (1:7.6p1-4ubuntu0.3) ...
Setting up ssh-import-id (5.7-0ubuntu1.1) ...
Setting up libwrap0:amd64 (7.6.q-27) ...
Setting up openssh-server (1:7.6p1-4ubuntu0.3) ...

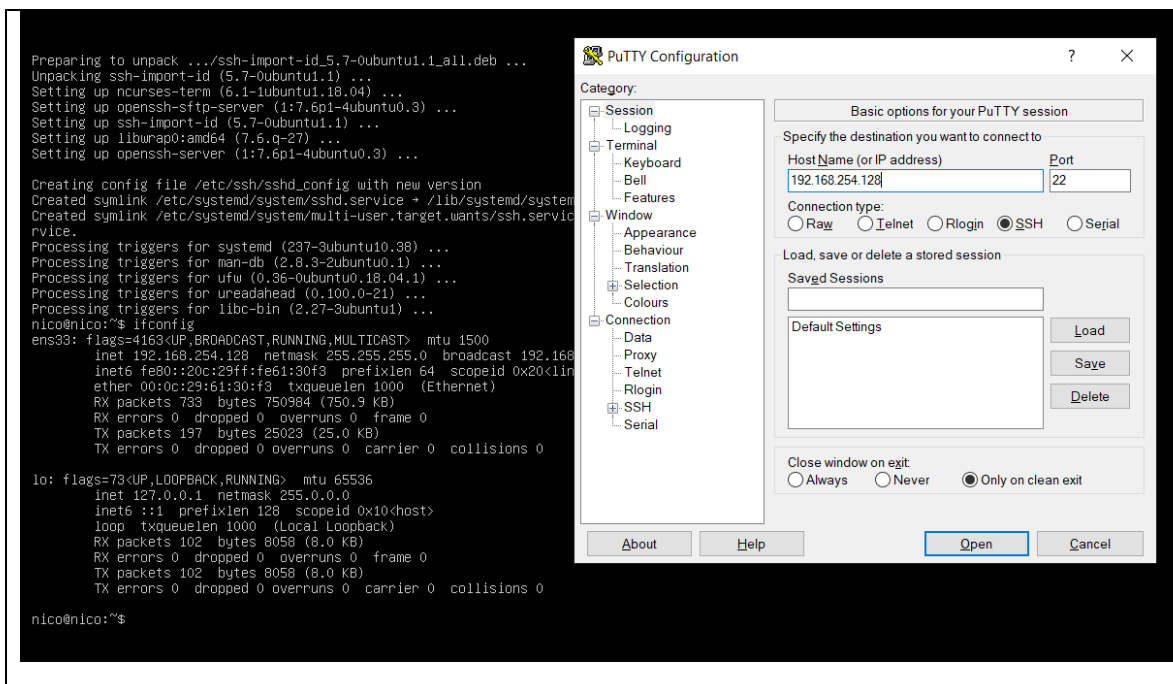
Creating config file /etc/ssh/sshd_config with new version
Created symlink /etc/systemd/system/ssh.service → /lib/systemd/system/ssh.service.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /lib/systemd/system/ssh.service.
Processing triggers for systemd (237-3ubuntu10.38) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ufw (0.36-0ubuntu0.18.04.1) ...
Processing triggers for ureadahead (0.100.0-21) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
nico@nico:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.254.128 netmask 255.255.255.0 broadcast 192.168.254.255
    inet6 fe80::20c:29ff:fe61:30f3 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:61:30:f3 txqueuelen 1000 (Ethernet)
    RX packets 733 bytes 750984 (750.9 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 197 bytes 25023 (25.0 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 102 bytes 8058 (8.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 102 bytes 8058 (8.0 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

nico@nico:~$ _
```

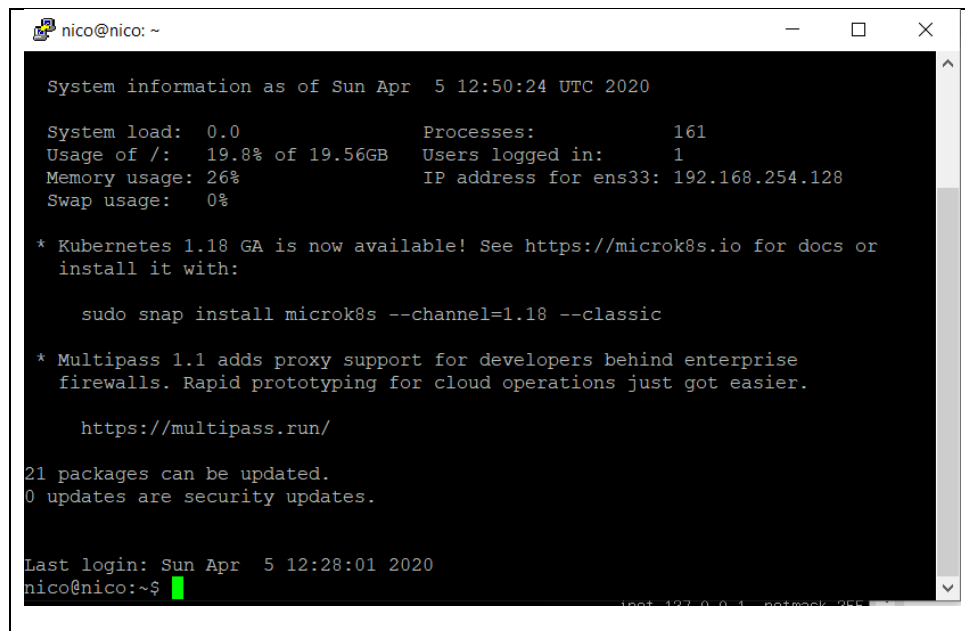
25. Buka PuTTY. Setelah itu masukkan ip address yang berada di linux ke PuTTY.

Gambar 2.29. Memasukkan ip address di PuTTY



26. Login menggunakan username server linux. Maka akan tampil seperti berikut.

Gambar 2.30. Hasil dari login PuTTY.



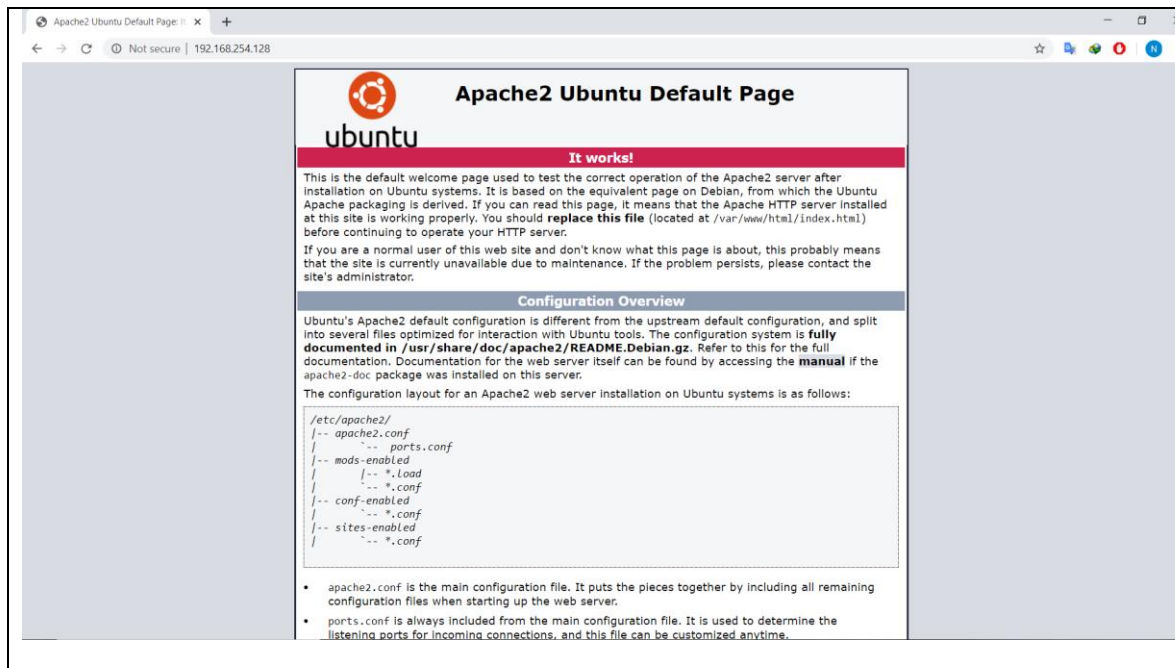
```
nico@nico: ~  
System information as of Sun Apr  5 12:50:24 UTC 2020  
  
System load:  0.0          Processes:           161  
Usage of /:   19.8% of 19.56GB Users logged in:       1  
Memory usage: 26%         IP address for ens33: 192.168.254.128  
Swap usage:   0%  
  
* Kubernetes 1.18 GA is now available! See https://microk8s.io for docs or  
install it with:  
  
    sudo snap install microk8s --channel=1.18 --classic  
  
* Multipass 1.1 adds proxy support for developers behind enterprise  
firewalls. Rapid prototyping for cloud operations just got easier.  
  
    https://multipass.run/  
  
21 packages can be updated.  
0 updates are security updates.  
  
Last login: Sun Apr  5 12:28:01 2020  
nico@nico:~$
```

27. Install apache dan pengaturan full. Install apache dengan menggunakan **Modul 2.1.4.**

28. Setelah itu lakukan sesuai **Modul 2.1.5.**

Perintah ini digunakan untuk mengatur Firewall sehingga lalu lintas data dari Apache Web Server dengan tipe protokol HTTP (port 80) dan HTTPS (port 443) diijinkan. Apabila berhasil maka akan menghasilkan gambar sbb :

Gambar 2.31. Tampilan hasil instalasi apache2



29. Lanjutkan instalasi mysql sebagai manajemen basis data. Ketikkan sesuai dengan **Modul 2.1.6.**

Perintah ini digunakan untuk menginstall package mysql server untuk linux. Jika sudah selesai maka akan tampil sbb :

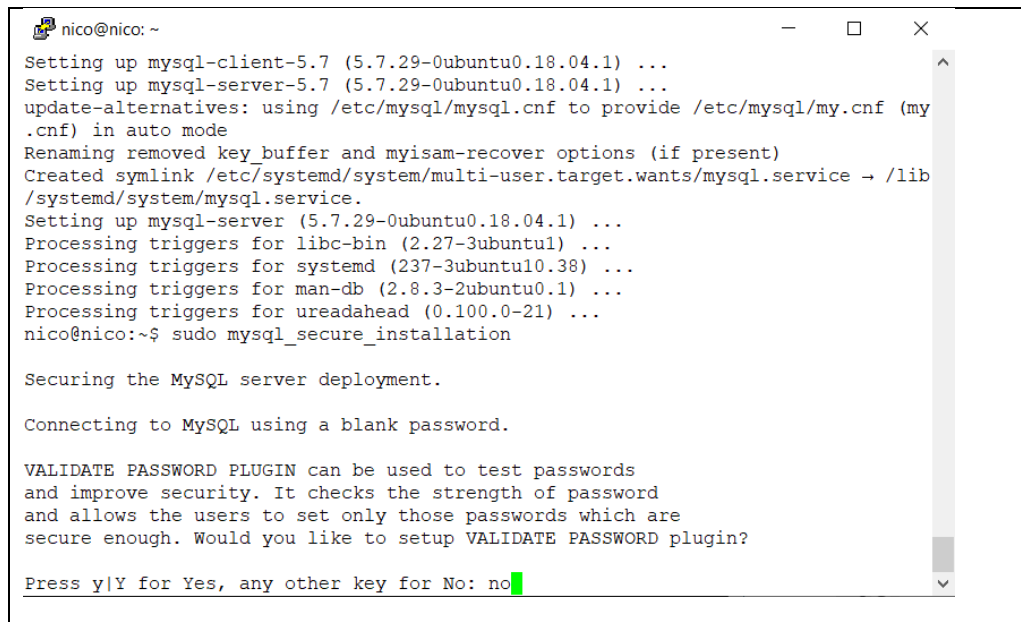
Gambar 2.32. Tampilan hasil instalasi mysql

```
nico@nico: ~
Setting up libhtml-parser-perl (3.72-3build1) ...
Setting up libcgi-pm-perl (4.38-1) ...
Setting up mysql-client-core-5.7 (5.7.29-0ubuntu0.18.04.1) ...
Setting up libfcgi-perl (0.78-2build1) ...
Setting up libhttp-date-perl (6.02-1) ...
Setting up libhtml-template-perl (2.97-1) ...
Setting up mysql-server-core-5.7 (5.7.29-0ubuntu0.18.04.1) ...
Setting up libcgi-fast-perl (1:2.13-1) ...
Setting up libhttp-message-perl (6.14-1) ...
Setting up mysql-client-5.7 (5.7.29-0ubuntu0.18.04.1) ...
Setting up mysql-server-5.7 (5.7.29-0ubuntu0.18.04.1) ...
update-alternatives: using /etc/mysql/mysql.cnf to provide /etc/mysql/my.cnf (my.cnf) in auto mode
Renaming removed key_buffer and myisam-recover options (if present)
Created symlink /etc/systemd/system/multi-user.target.wants/mysql.service → /lib/systemd/system/mysql.service.
Setting up mysql-server (5.7.29-0ubuntu0.18.04.1) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
Processing triggers for systemd (237-3ubuntu10.38) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ureadahead (0.100.0-21) ...
nico@nico:~$
```

30. Setelah itu lakukan perintah sesuai dengan **Modul 2.1.7.** Perintah ini digunakan untuk proses pengaturan dasar pengamanan mysql yang berupa users, kata sandi, hak akses, dan sebagainya. Akan ada pertanyaan tentang validasi password.

Untuk ini lebih baik tidak menggunakan passowrd untuk mempermudah pengerjaan.

Gambar 2.33. Tampilan pilihan validate password



```
nico@nico: ~
Setting up mysql-client-5.7 (5.7.29-0ubuntu0.18.04.1) ...
Setting up mysql-server-5.7 (5.7.29-0ubuntu0.18.04.1) ...
update-alternatives: using /etc/mysql/mysql.cnf to provide /etc/mysql/my.cnf (my
.cnf) in auto mode
Renaming removed key_buffer and myisam-recover options (if present)
Created symlink /etc/systemd/system/multi-user.target.wants/mysql.service → /lib
/systemd/system/mysql.service.
Setting up mysql-server (5.7.29-0ubuntu0.18.04.1) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
Processing triggers for systemd (237-3ubuntu10.38) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for ureadahead (0.100.0-21) ...
nico@nico:~$ sudo mysql_secure_installation

Securing the MySQL server deployment.

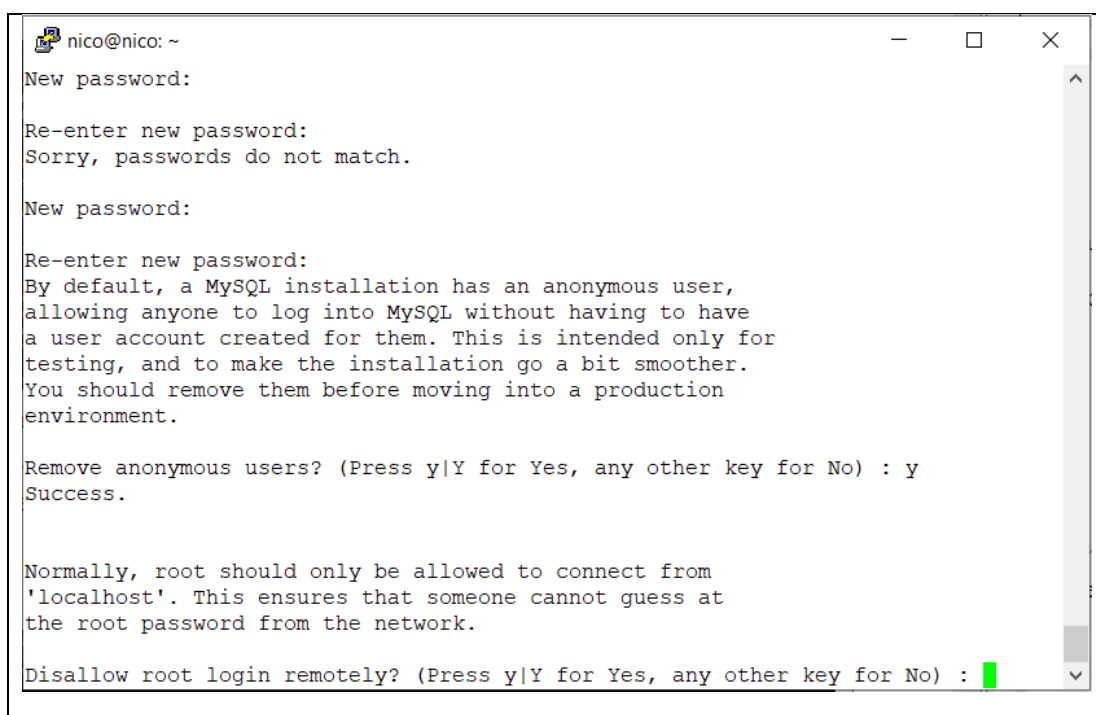
Connecting to MySQL using a blank password.

VALIDATE PASSWORD PLUGIN can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD plugin?

Press y|Y for Yes, any other key for No: no
```

31. Setelah itu tentukan password untuk mysql. Setelah itu akan ada pilihan anonymous users. Pilih yes untuk menghapus anonymous user sebagai standar keamanan mysql cloud. Maka akan tampil seperti berikut :

Gambar 2.34. Tampilan pilihan setelah anonymous users



```
nico@nico: ~
New password:
Re-enter new password:
Sorry, passwords do not match.

New password:
Re-enter new password:
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

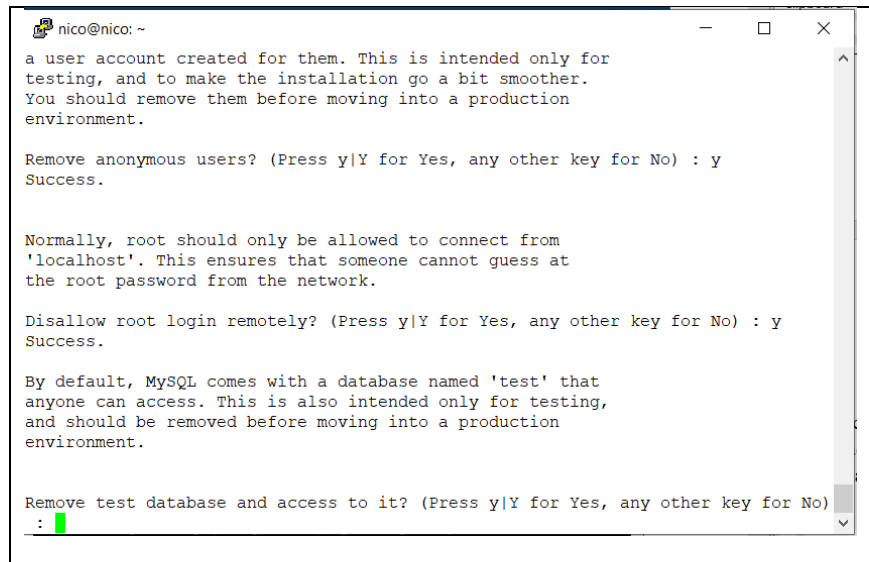
Remove anonymous users? (Press y|Y for Yes, any other key for No) : y
Success.

Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) :
```

32. Setelah itu ada pilihan tentang remote login ke basis data mysql dari jaringan luar localhost (Ubuntu Server). Pilih yes sehingga kemampuan remote login dengan akun root pada server cloud tidak diaktifkan, dikarenakan mempengaruhi keamanan server cloud. Maka tampilan yang dihasilkan akan sbb :

Gambar 2.35. Tampilan pilihan setelah pilihan remote users



```
nico@nico: ~
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : y
Success.

Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

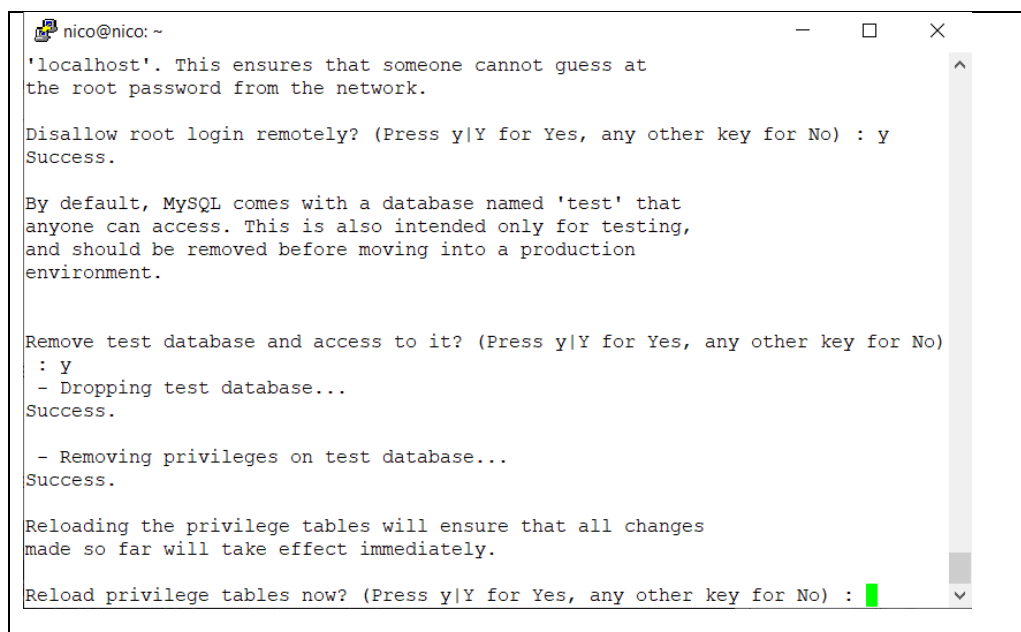
Disallow root login remotely? (Press y|Y for Yes, any other key for No) : y
Success.

By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No)
: y
```

33. Setelah itu terdapat peringatan tentang database dengan nama test yang secara default yang terpasang di terpasang di mysql. Pilih yes / y untuk menghapus database tersebut untuk alasan keamanan.

Gambar 2.36. Tampilan pilihan setelah remove test database.



```
nico@nico: ~
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : y
Success.

By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No)
: y
- Dropping test database...
Success.

- Removing privileges on test database...
Success.

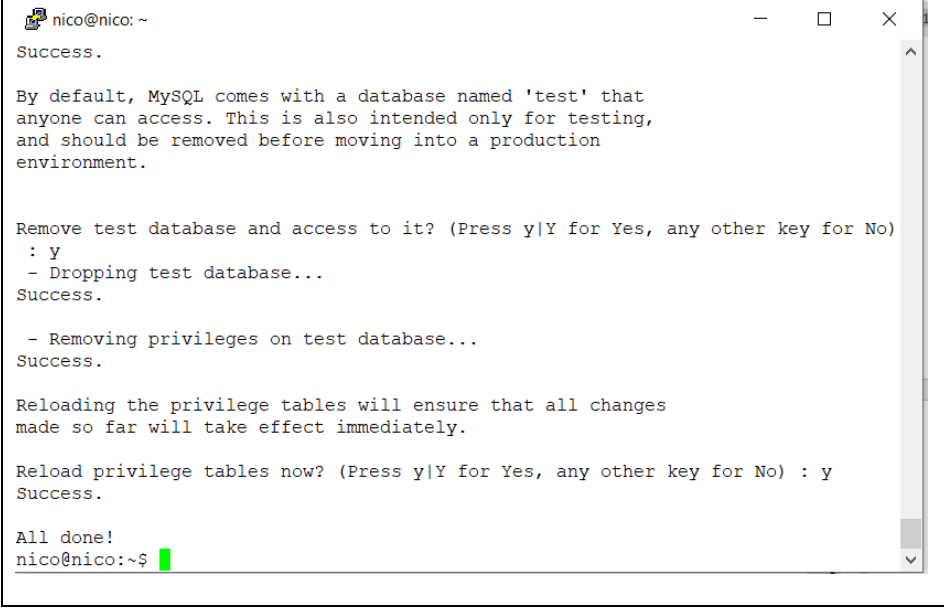
Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
```

34. Yang terakhir yaitu proses dari refresh/reload tabel privilege atau hak akses dari mysql. Masukkan input y pada menu ini sehingga tabel privilege memiliki

definisi nilai yang terbaru dan dapat diakses dengan username dan kata sandi yang telah diatur pada tahap sebelumnya. Hasilnya akan seperti berikut :

Gambar 2.37. Tampilan selesai instalasi mysql



```
nico@nico: ~
Success.

By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No)
: y
- Dropping test database...
Success.

- Removing privileges on test database...
Success.

Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

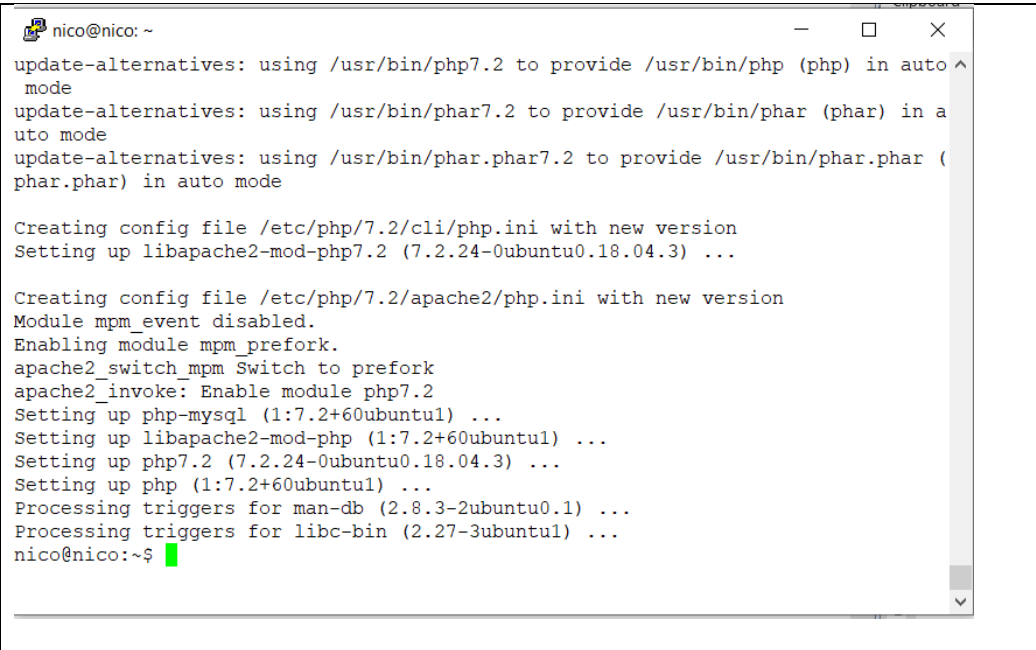
Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
Success.

All done!
nico@nico:~$
```

35. Instal php sebagai prosesor konten web server dengan jenis PHP. Hal ini dikarenakan Apache hanya dapat memproses konten web server HTML dan sejenisnya. Lakukan sesuai dengan **Modul 2.1.8**.

Perintah ini digunakan untuk menginstall php dan mengatur dengan pengaturan default sehingga siap digunakan. Jika sudah selesai, maka akan tampil sebagai berikut :

Gambar 2.38. Tampilan selesai instalasi php



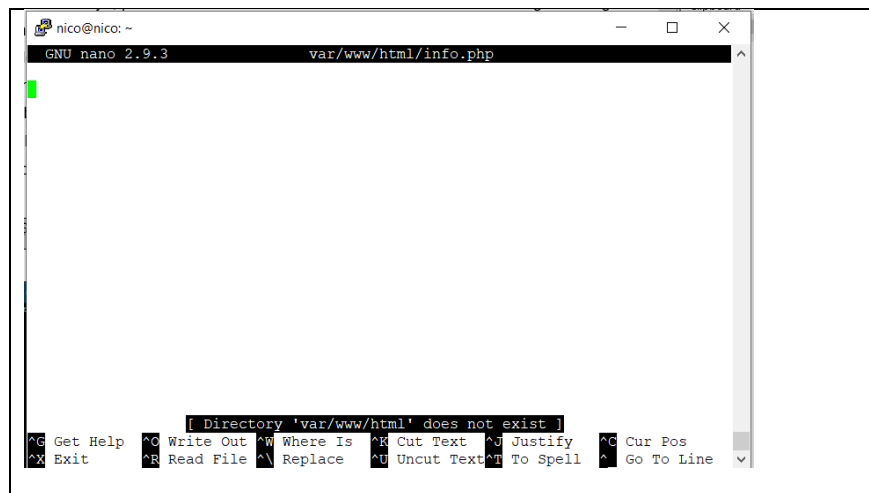
```
nico@nico: ~
update-alternatives: using /usr/bin/php7.2 to provide /usr/bin/php (php) in auto
mode
update-alternatives: using /usr/bin/phar7.2 to provide /usr/bin/phar (phar) in a
uto mode
update-alternatives: using /usr/bin/phar.phar7.2 to provide /usr/bin/phar.phar (
phar.phar) in auto mode

Creating config file /etc/php/7.2/cli/php.ini with new version
Setting up libapache2-mod-php7.2 (7.2.24-0ubuntu0.18.04.3) ...

Creating config file /etc/php/7.2/apache2/php.ini with new version
Module mpm_event disabled.
Enabling module mpm_prefork.
apache2_switch_mpm Switch to prefork
apache2_invoke: Enable module php7.2
Setting up php-mysql (1:7.2+60ubuntu1) ...
Setting up libapache2-mod-php (1:7.2+60ubuntu1) ...
Setting up php7.2 (7.2.24-0ubuntu0.18.04.3) ...
Setting up php (1:7.2+60ubuntu1) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
nico@nico:~$
```

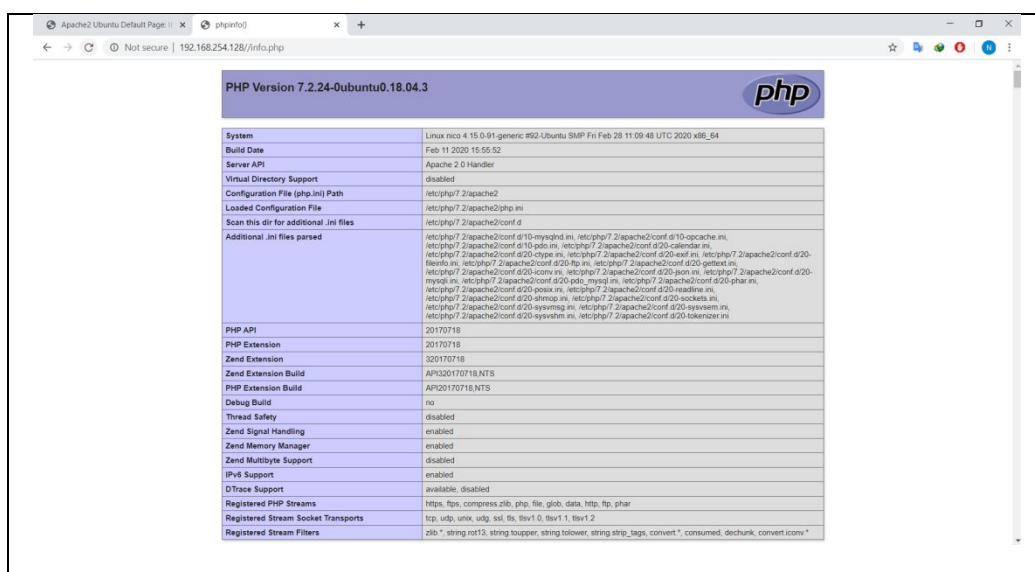
Untuk mengecek apakah instalasi php bisa dijalankan maka buat sebuah nano (notepad pada linux) dengan format php pada lokasi direktori Web Server Apache yaitu di /var/www/html/. Masukkan perintah sesuai dengan **Modul 2.1.9.** Maka akan tampil sbb :

Gambar 2.39. Tampilan nano



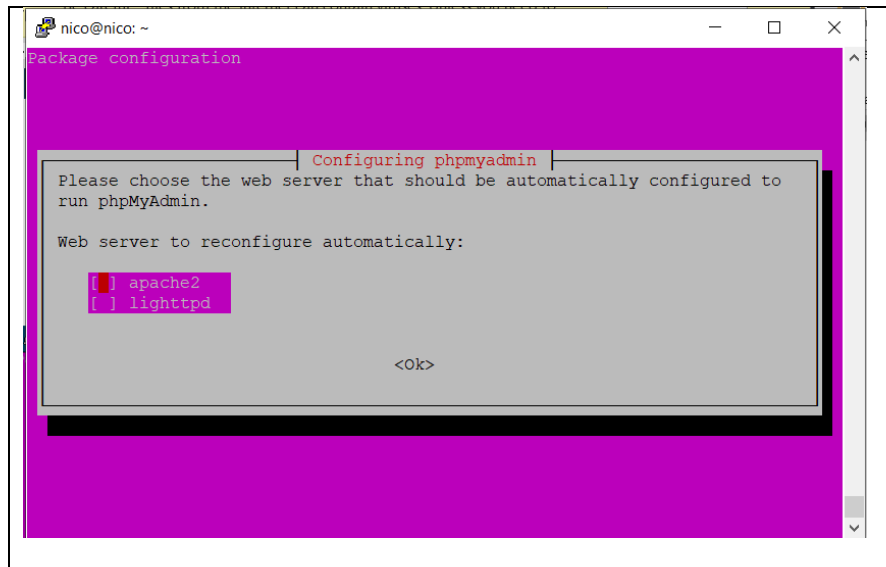
36. Kemudian masukkan ke nano sesuai dengan **Modul 2.1.10.** Fungsi dari kodingan tersebut adalah untuk mengetahui versi php yang telah diinstall. Setelah selesai simpan file tersebut dan keluar dari aplikasi nano. Setelah itu buka browser dengan sintaks http://IP_SERVER_CLOUD/info.php untuk mengetes apakah file php bisa dijalankan. Jika berhasil maka akan muncul gambar sbb :

Gambar 2.40. Tampilan info.php di browser



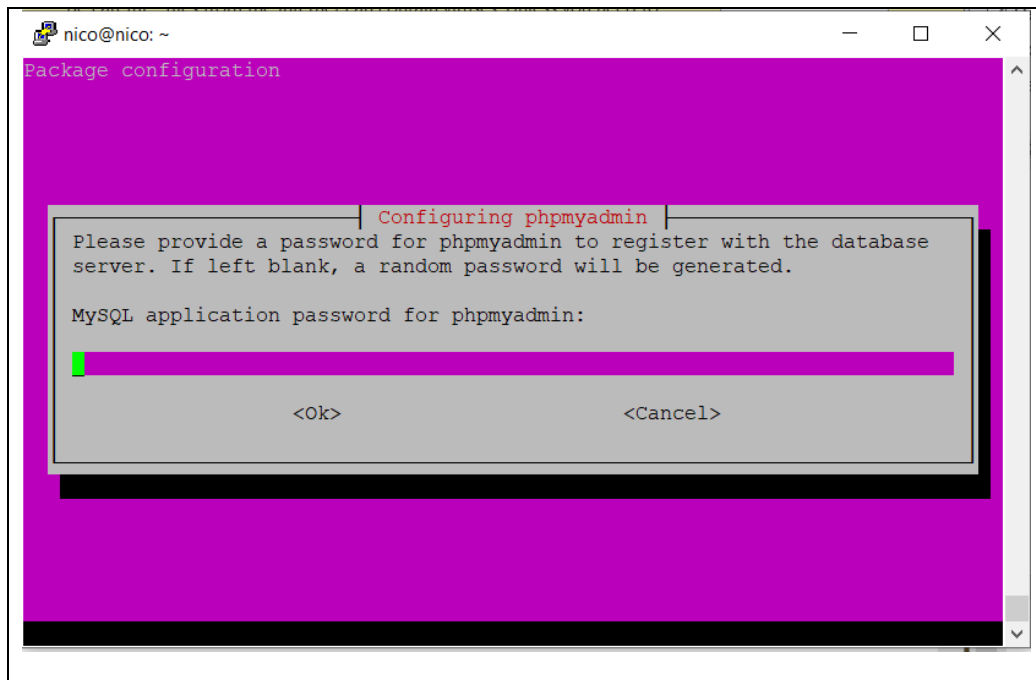
37. Install phpmyadmin. Fungsi dari phpmyadmin yaitu untuk layanan manajemen database mysql berbasis Web GUI. Lakukan perintah sesuai dengan **Modul 2.1.11**. Jika ada pilihan y/n pilih y. setelah selesai maka akan muncul gambar sbb :

Gambar 2.41. Peringatan instalasi phpmyadmin



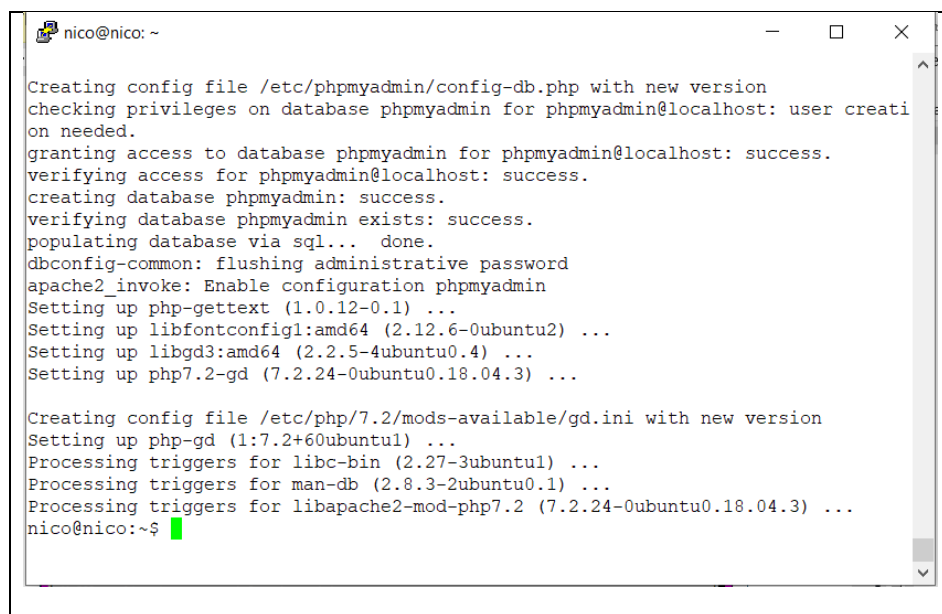
38. Pilih pada bagian apache2 lalu tekan spasi hingga muncul tanda * (asterisk). Setelah itu tekan enter. Akan ada peringatan tentang dpconfig-common, tekan Enter pada pilihan Yes. Setelah itu akan tampil gambar sbb :

Gambar 2.42. Masukkan Password untuk user mysql



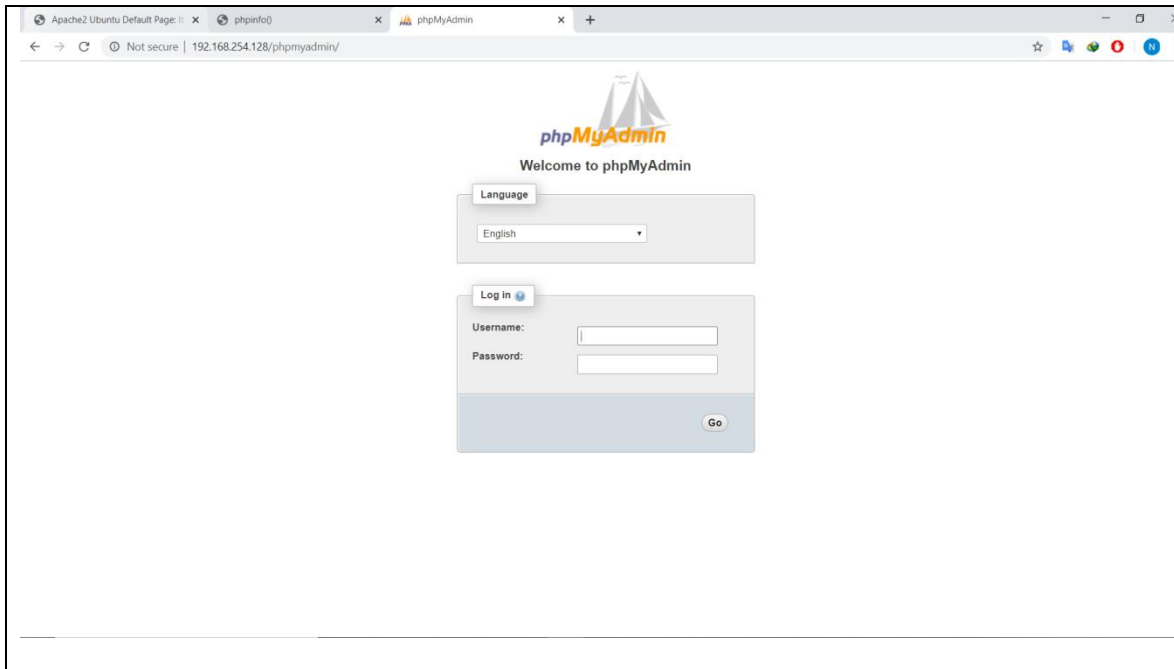
39. Di bagian ini, masukkan password untuk database phpmyadmin. Jika sudah maka akan tampil sbb :

Gambar 2.42. Tampilan selesai menginstall



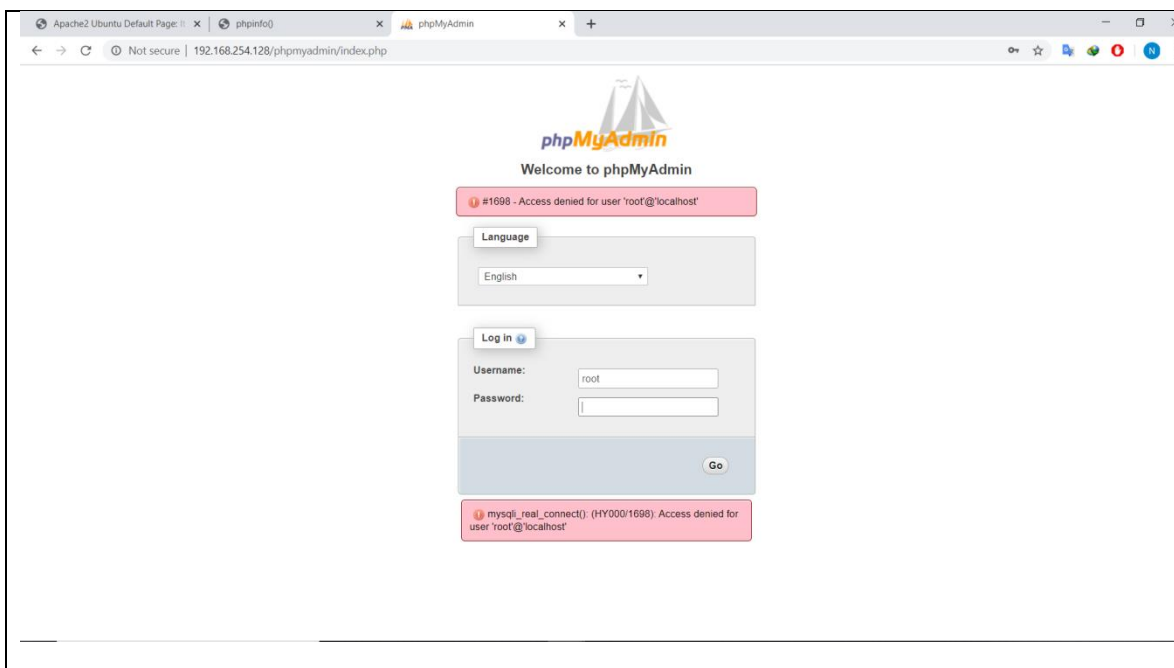
40. Setelah itu cek apakah phpmyadmin telah berhasil, buka alamat dengan ketentuan sbb : http://IP_SERVER_CLOUD/phpmyadmin. Maka akan tampil sbb :

Gambar 2.43. Tampilan selesai menginstall phpmyadmin



41. jika sudah, login dengan user root dan password yang sudah di set di setting phpmyadmin tadi. Cek apakah sudah bisa. Tampilannya maka akan sbb :

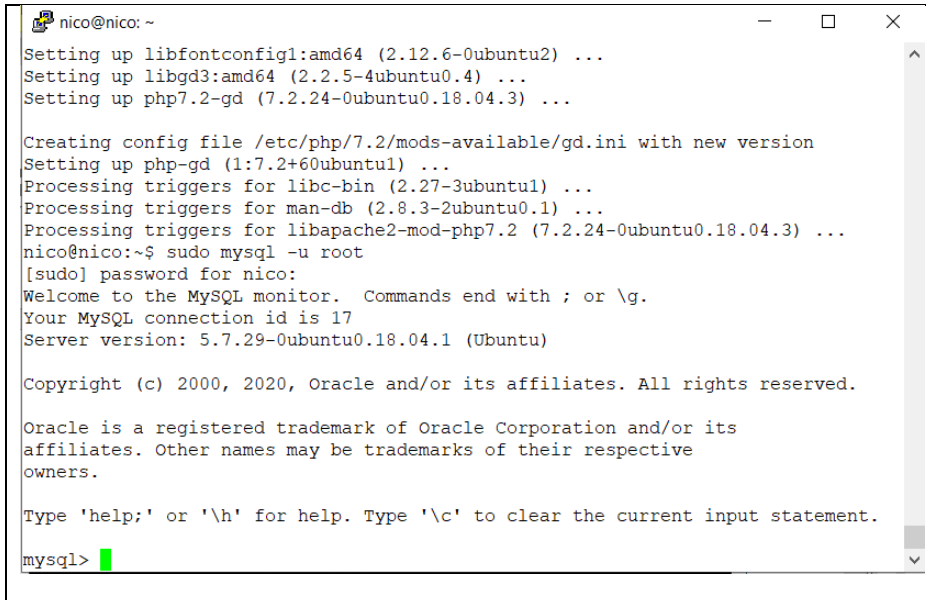
Gambar 2.44. Tampilan error ketika login



Error tersebut terjadi karena konfigurasi user root yang akan digunakan untuk login diatur dengan model plugin Unix Socket based authentication, sehingga perlu diatur plugin menggunakan Native MySQL authentication.

Untuk mempengaruhi. Error tersebut, maka ketikkan perintah sesuai **Modul 2.1.12**. Fungsi dari **Modul 2.1.12** adalah untuk masuk ke menu mysql melewati CLI. Jika sudah maka akan muncul tampilan sbb :

Gambar 2.45. Tampilan CLI mysql



```
nico@nico: ~
Setting up libfontconfig1:amd64 (2.12.6-0ubuntu2) ...
Setting up libgd3:amd64 (2.2.5-4ubuntu0.4) ...
Setting up php7.2-gd (7.2.24-0ubuntu0.18.04.3) ...

Creating config file /etc/php/7.2/mods-available/gd.ini with new version
Setting up php-gd (1:7.2+60ubuntu1) ...
Processing triggers for libc-bin (2.27-3ubuntu1) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
Processing triggers for libapache2-mod-php7.2 (7.2.24-0ubuntu0.18.04.3) ...
nico@nico:~$ sudo mysql -u root
[sudo] password for nico:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17
Server version: 5.7.29-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

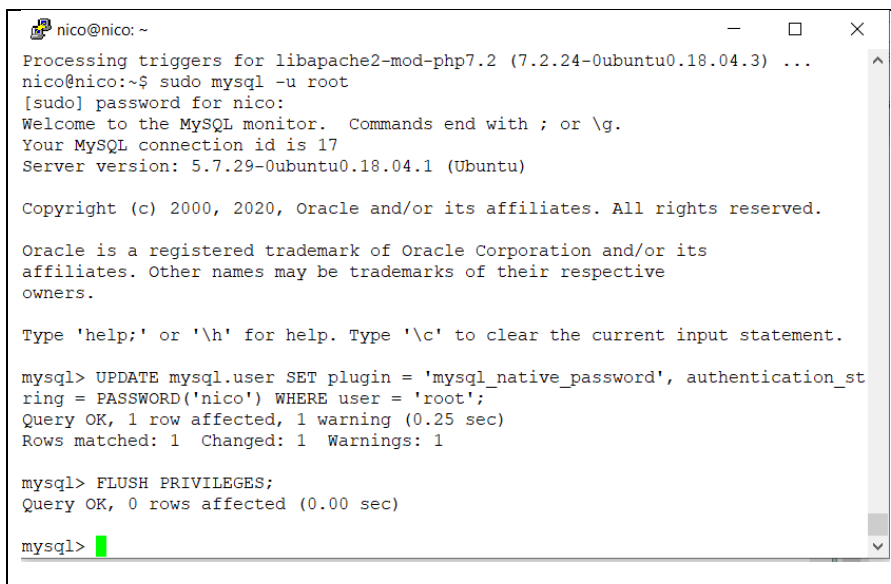
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

42. Setelah itu ketik perintah sesuai **Modul 2.1.13**.. jika berhasil, maka akan muncul hasil sbb :

Gambar 2.46. Tampilan selesai setting user root



```
nico@nico: ~
Processing triggers for libapache2-mod-php7.2 (7.2.24-0ubuntu0.18.04.3) ...
nico@nico:~$ sudo mysql -u root
[sudo] password for nico:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17
Server version: 5.7.29-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

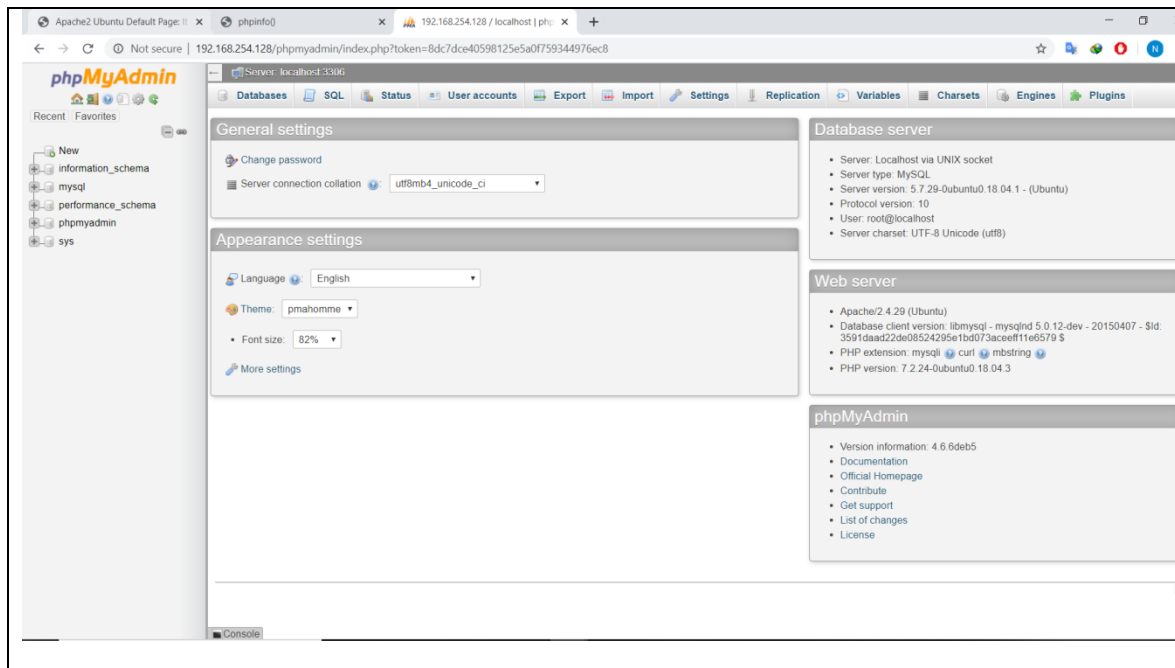
mysql> UPDATE mysql.user SET plugin = 'mysql_native_password', authentication_st
ring = PASSWORD('nico') WHERE user = 'root';
Query OK, 1 row affected, 1 warning (0.25 sec)
Rows matched: 1  Changed: 1  Warnings: 1

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)

mysql>
```

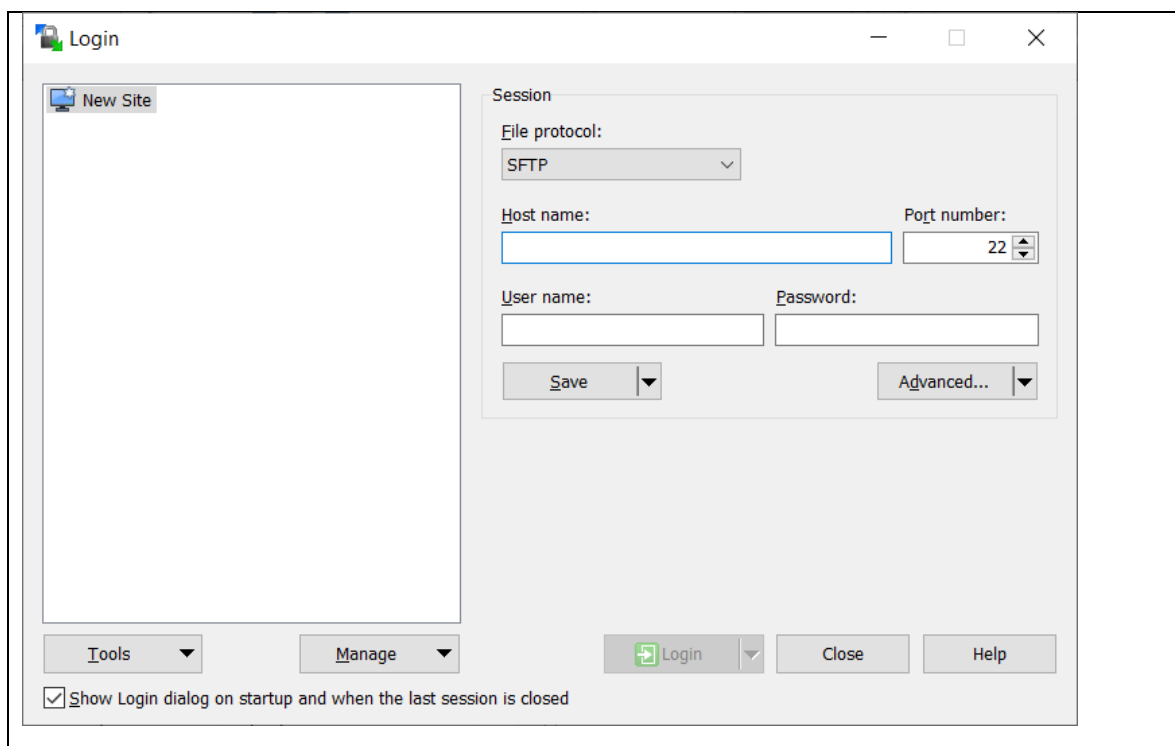
43. Setelah itu login lagi di browser. Cek apakah berhasil atau belum. Jika berhasil maka akan terlihat sbb :

Gambar 2.47. Tampilan phpmyadmin



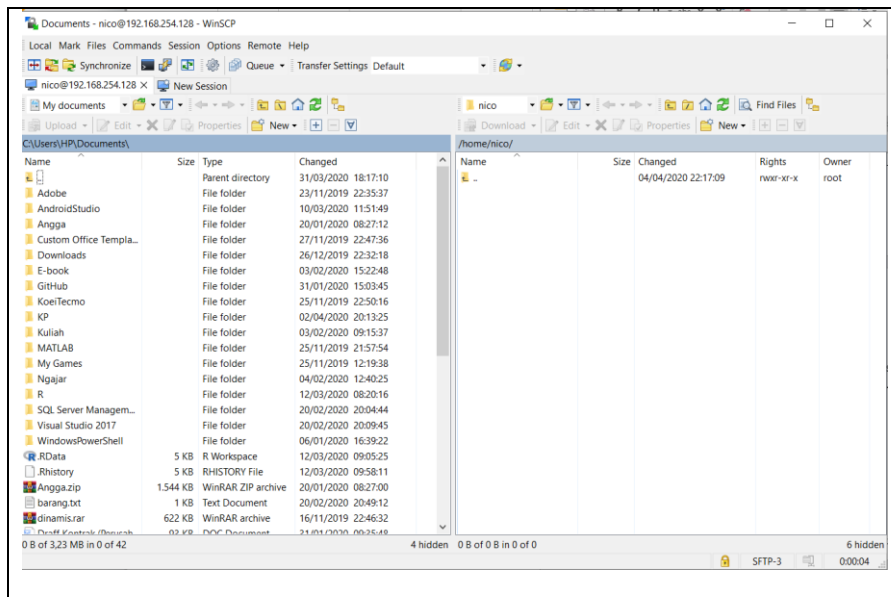
44. Buka aplikasi WinSCP. Maka tampilannya akan sbb :

Gambar 2.48. Tampilan phpmyadmin



45. Masukkan IP dari server yang digunakan, username dan password berupa username dan password yang digunakan di linux. Tekan login. Jika sudah maka akan tampil sbb :

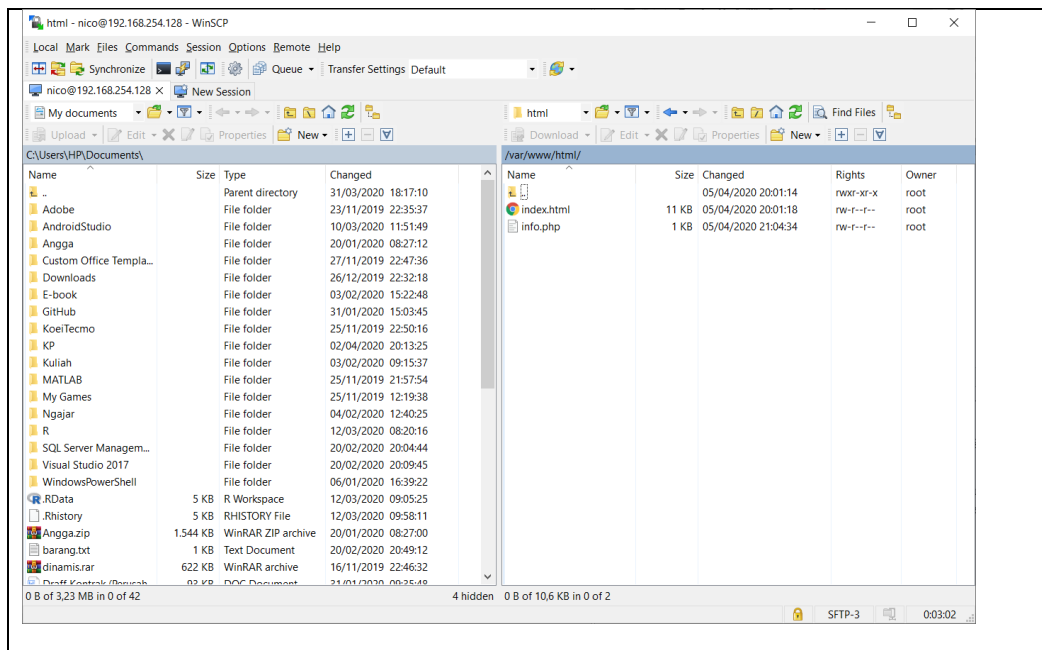
Gambar 2.49. Tampilan WinSCP setelah login



Kolom kanan merupakan tampilan berkas yang berada pada server. Sedangkan kolom kiri merupakan berkas yang ada pada workstation.

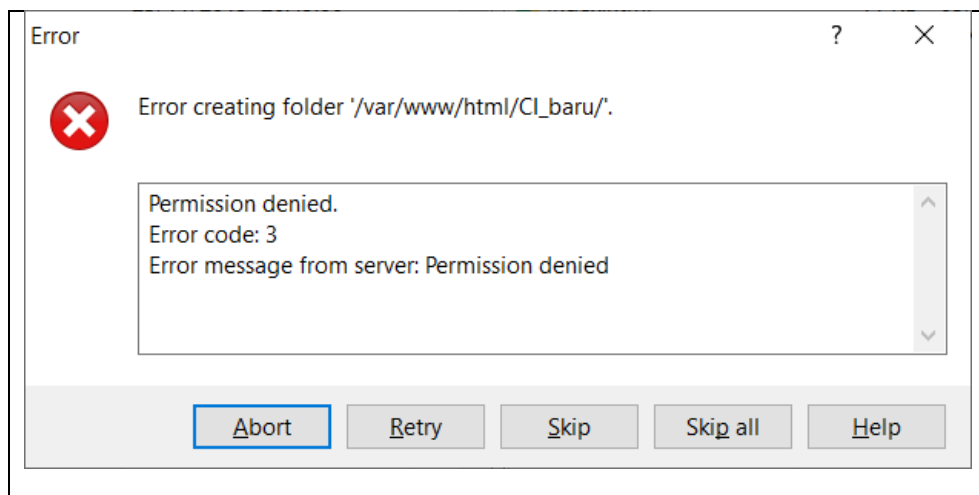
Di kolom kiri, arahkan ke direktori /var/www/html/ untuk meletakkan berkas yang berada di workstation ke server. Maka tampilannya akan seperti berikut :

Gambar 2.50. Tampilan WinSCP setelah mengubah direktori



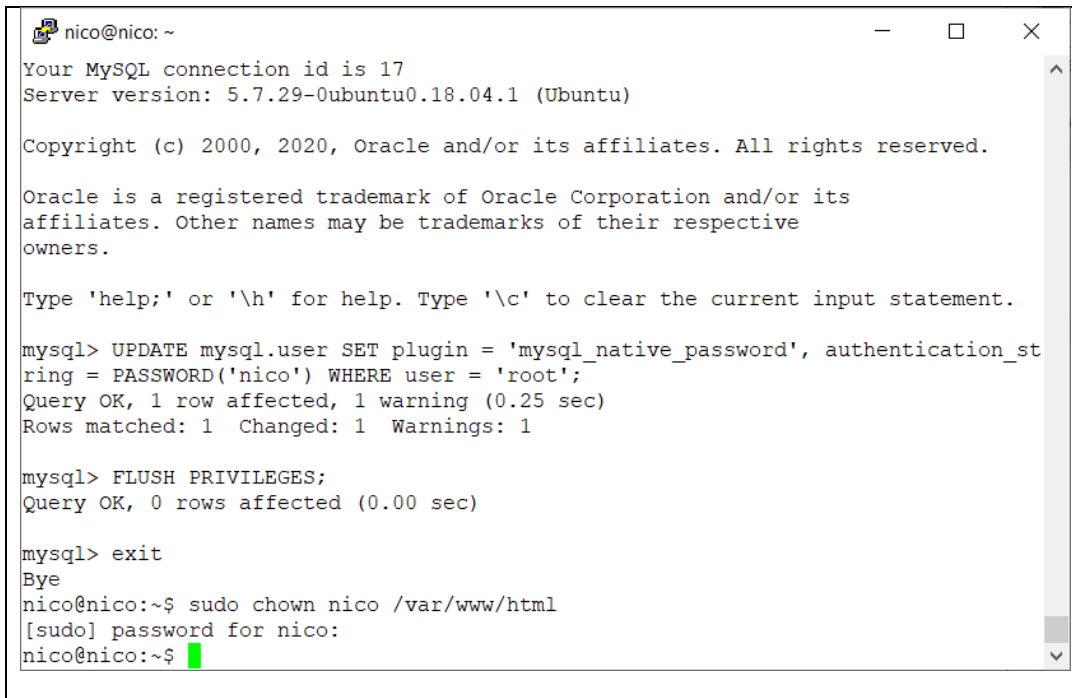
46. Untuk mengupload berkas kodingan, lakukan dengan cara drag and drop. Arahkan direktori kolom kiri ke berkas yang ingin diupload ke server. Maka akan ada error seperti berikut :

Gambar 2.51. Tampilan error pada pemindahan file.



Cara mengatasi error ini dengan mengganti owner dari user. Caranya yaitu mengetikkan perintah sesuai **Modul 2.1.14**. Setelah dijalankan, maka akan muncul gambar seperti berikut :

Gambar 2.51. Tampilan hasil pengubahan owner .



```
nico@nico: ~
Your MySQL connection id is 17
Server version: 5.7.29-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

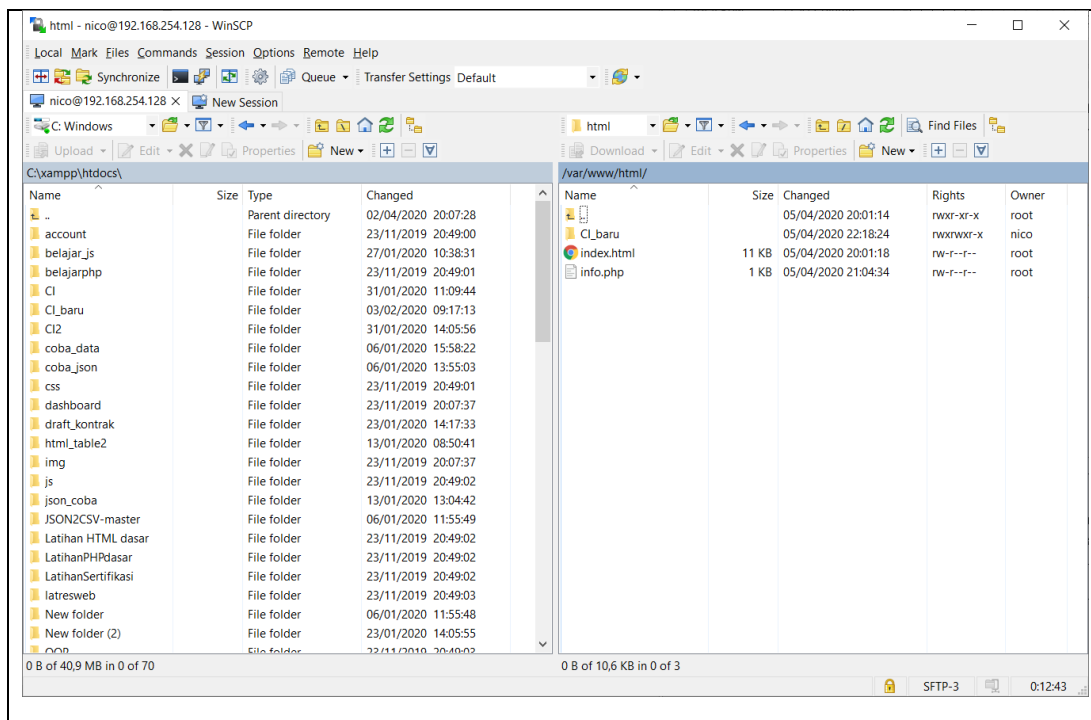
mysql> UPDATE mysql.user SET plugin = 'mysql_native_password', authentication_string = PASSWORD('nico') WHERE user = 'root';
Query OK, 1 row affected, 1 warning (0.25 sec)
Rows matched: 1  Changed: 1  Warnings: 1

mysql> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)

mysql> exit
Bye
nico@nico:~$ sudo chown nico /var/www/html
[sudo] password for nico:
nico@nico:~$
```

47. Jika sudah, maka ulangi langkah memindahkan file dari workstation ke server. Caranya dengan drag and drop. Disini file yang dipindahkan yaitu file “CI_Baru”. Jika sudah maka tampilan akan terlihat sbb :

Gambar 2.52. Tampilan berhasil memindahkan file

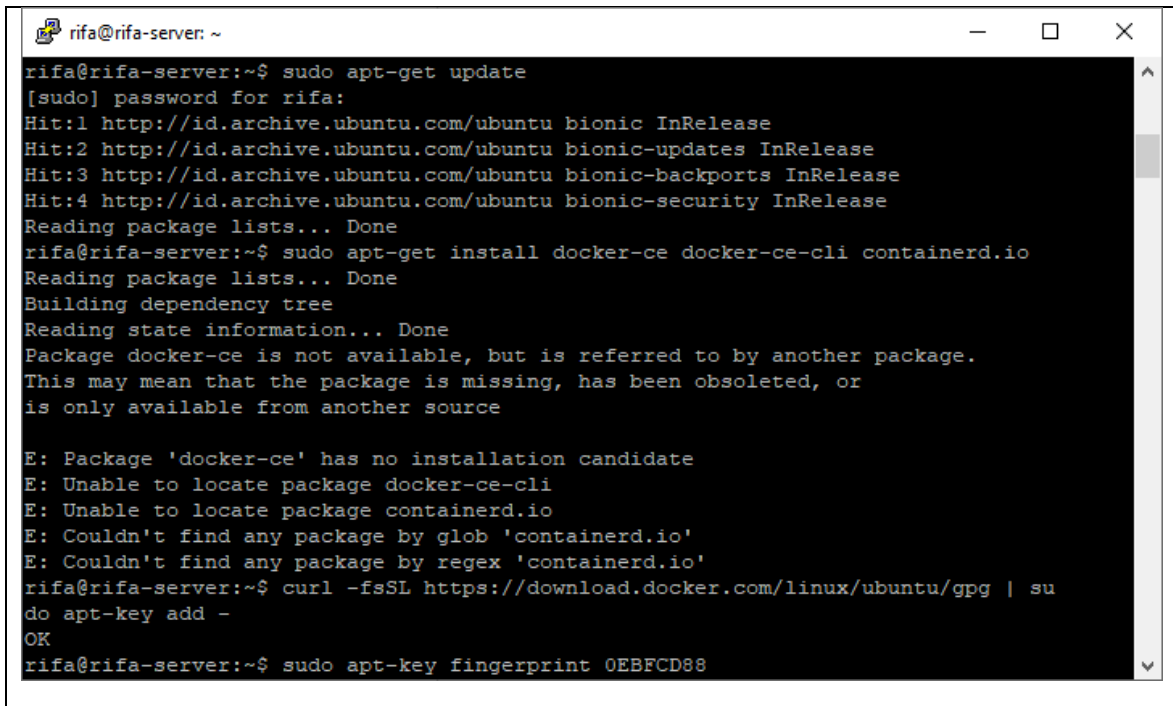


48. Cara mengecek apakah sudah bisa atau belum dengan mengetikan url di browser : http://IP_SERVER_CLOUD/nama_file . Disini menggunakan url : 192.168.254.128/CI_baru.

2. Penerapan layanan hosting dengan Docker

1. Update apt package index dan install package agar mengijinkan apt menggunakan repositori sesuai dengan **Modul 2.2.1**.
2. Tambahkan key GPG pada docker dengan menggunakan **Modul 2.2.2**.
3. Setelah menjalankan perintah **Modul 2.2.2**, maka akan mendapat kode fingerprint. Kode yang didapat yaitu 9DC8 5822 9FC7 DD38 854A E2D8 8D81 803C 0EBF CD88. Verifikasi dengan menambahkan 8 angka terakhir dari kode fingerprint. Lalu jalankan **Modul 2.2.3**.

Gambar 2.54. Tampilan memasukkan kode fingerprint

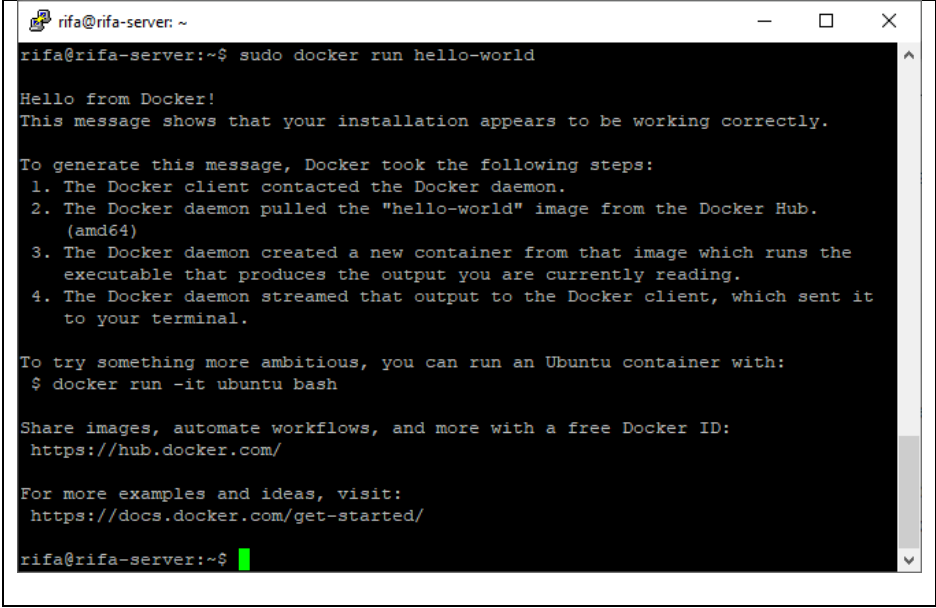


```
rifa@rifa-server: ~
rifa@rifa-server:~$ sudo apt-get update
[sudo] password for rifa:
Hit:1 http://id.archive.ubuntu.com/ubuntu bionic InRelease
Hit:2 http://id.archive.ubuntu.com/ubuntu bionic-updates InRelease
Hit:3 http://id.archive.ubuntu.com/ubuntu bionic-backports InRelease
Hit:4 http://id.archive.ubuntu.com/ubuntu bionic-security InRelease
Reading package lists... Done
rifa@rifa-server:~$ sudo apt-get install docker-ce docker-ce-cli containerd.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
Package docker-ce is not available, but is referred to by another package.
This may mean that the package is missing, has been obsoleted, or
is only available from another source

E: Package 'docker-ce' has no installation candidate
E: Unable to locate package docker-ce-cli
E: Unable to locate package containerd.io
E: Couldn't find any package by glob 'containerd.io'
E: Couldn't find any package by regex 'containerd.io'
rifa@rifa-server:~$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | su
do apt-key add -
OK
rifa@rifa-server:~$ sudo apt-key fingerprint 0EBFCD88
```

4. Gunakan **Modul 2.2.4** untuk mengatur repository stable
5. Update apt package index kemudian install Docker Community Edition dan container dengan menggunakan **Modul 2.2.5**.
6. Pastikan bahwa Docker engine terinstall dengan baik menggunakan command **Modul 2.2.6**, jika muncul pesan di gambar berikut, maka docker berhasil terinstall.

Gambar 2.55. Tampilan menjalankan Hello World docker



```
rifa@rifa-server: ~
rifa@rifa-server:~$ sudo docker run hello-world

Hello from Docker!
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
1. The Docker client contacted the Docker daemon.
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
   (amd64)
3. The Docker daemon created a new container from that image which runs the
   executable that produces the output you are currently reading.
4. The Docker daemon streamed that output to the Docker client, which sent it
   to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
$ docker run -it ubuntu bash

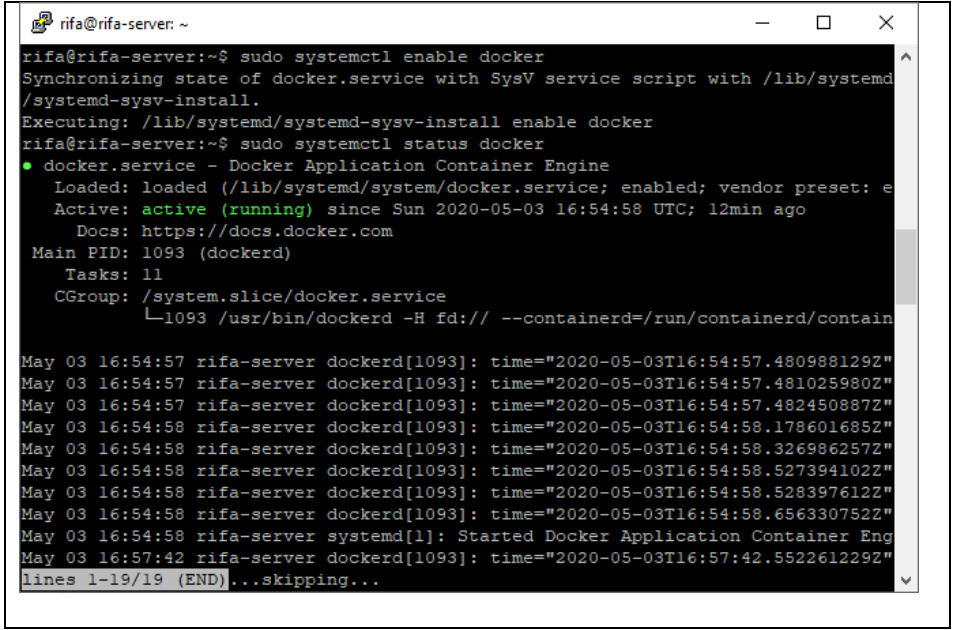
Share images, automate workflows, and more with a free Docker ID:
https://hub.docker.com/

For more examples and ideas, visit:
https://docs.docker.com/get-started/

rifa@rifa-server:~$
```

7. Setelah berhasil menginstall docker engine, proses selanjutnya adalah cek status docker dengan menggunakan **Modul 2.2.7**.

Gambar 2.56. Tampilan melihat status docker



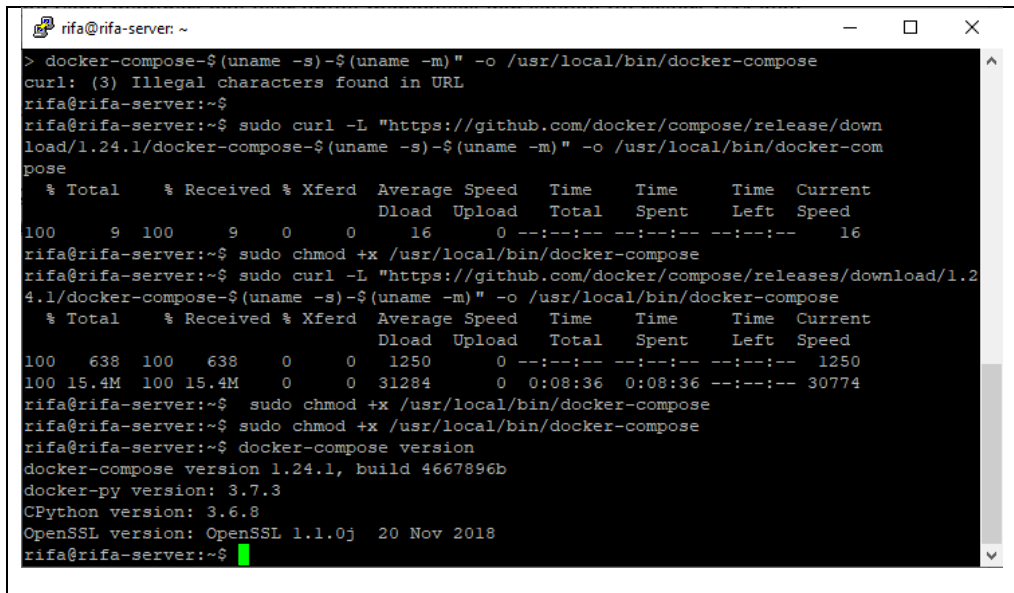
```
rifa@rifa-server: ~
rifa@rifa-server:~$ sudo systemctl enable docker
Synchronizing state of docker.service with SysV service script with /lib/systemd
/systemd-sysv-install.
Executing: /lib/systemd/systemd-sysv-install enable docker
rifa@rifa-server:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: e
   Active: active (running) since Sun 2020-05-03 16:54:58 UTC; 12min ago
     Docs: https://docs.docker.com
    Main PID: 1093 (dockerd)
      Tasks: 11
   CGroup: /system.slice/docker.service
           └─1093 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/contain

May 03 16:54:57 rifa-server dockerd[1093]: time="2020-05-03T16:54:57.480988129Z"
May 03 16:54:57 rifa-server dockerd[1093]: time="2020-05-03T16:54:57.481025980Z"
May 03 16:54:57 rifa-server dockerd[1093]: time="2020-05-03T16:54:57.482450887Z"
May 03 16:54:58 rifa-server dockerd[1093]: time="2020-05-03T16:54:58.178601685Z"
May 03 16:54:58 rifa-server dockerd[1093]: time="2020-05-03T16:54:58.326986257Z"
May 03 16:54:58 rifa-server dockerd[1093]: time="2020-05-03T16:54:58.527394102Z"
May 03 16:54:58 rifa-server dockerd[1093]: time="2020-05-03T16:54:58.528397612Z"
May 03 16:54:58 rifa-server dockerd[1093]: time="2020-05-03T16:54:58.656330752Z"
May 03 16:54:58 rifa-server systemd[1]: Started Docker Application Container Eng
May 03 16:57:42 rifa-server dockerd[1093]: time="2020-05-03T16:57:42.552261229Z"
lines 1-19/19 (END)...skipping...
```

8. Untuk menjalankan perintah docker tanpa sudo, maka kemudian ditambahkan command yang memberi hak user untuk mengakses dan setelah itu restart VM atau OS yang digunakan untuk merefresh dengan menggunakan **Modul 2.2.8**.

9. Untuk dapat menerapkan aplikasi dalam docker, hal yang dilakukan setelah menginstall docker engine adalah install docker-compose. Docker-compose adalah fungsi dari docker yang digunakan untuk mengakses servis apa saja yang akan digunakan di docker. Untuk menginstall docker-compose dapat melihat pada **Modul 2.2.9**.

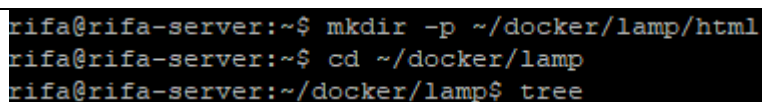
Gambar 2.57. Tampilan dari menginstall docker-compose



```
rifa@rifa-server: ~
> docker-compose-$(uname -s)-$(uname -m) -o /usr/local/bin/docker-compose
curl: (3) Illegal characters found in URL
rifa@rifa-server:~$
rifa@rifa-server:~$ sudo curl -L "https://github.com/docker/compose/releases/download/1.24.1/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
pose
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100    9  100    9    0    0    16      0  --:--:-- --:--:-- --:--:--    16
rifa@rifa-server:~$ sudo chmod +x /usr/local/bin/docker-compose
rifa@rifa-server:~$ sudo curl -L "https://github.com/docker/compose/releases/download/1.24.1/docker-compose-$(uname -s)-$(uname -m)" -o /usr/local/bin/docker-compose
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100   638  100   638    0    0   1250      0  --:--:-- --:--:-- --:--:--   1250
100 15.4M  100 15.4M    0    0 31284      0  0:08:36  0:08:36 --:--:-- 30774
rifa@rifa-server:~$ sudo chmod +x /usr/local/bin/docker-compose
rifa@rifa-server:~$ sudo chmod +x /usr/local/bin/docker-compose
rifa@rifa-server:~$ docker-compose version
docker-compose version 1.24.1, build 4667896b
docker-py version: 3.7.3
CPython version: 3.6.8
OpenSSL version: OpenSSL 1.1.0j  20 Nov 2018
rifa@rifa-server:~$
```

10. Setelah menginstall docker-compose dan mengecek versi docker nya, maka pindahkan ke direktori baru sesuai dengan **Modul 2.2.10**.
11. Siapkan direktori yang akan digunakan untuk menyimpan projek dan arahkan ke direktori yang sesuai dengan **Modul 2.2.11**.



Gambar 2.58. Tampilan dari menginstall docker-compose



```
rifa@rifa-server:~$ mkdir -p ~/docker/lamp/html
rifa@rifa-server:~$ cd ~/docker/lamp
rifa@rifa-server:~/docker/lamp$ tree
```

12. Buat dokumen yang bernama php.Dockerfile lalu isi command sesuai dengan **Modul 2.2.12** dan simpan di direktori docker/lamp
13. Buat dokumen yang bernama docker-compose.yml di direktori docker/lampp yang berisi sesuai dengan **Modul 2.2.13**, yaitu php,mysql, dan docker.
14. Untuk aplikasi yang ingin dijalankan disimpan di direktori docker/lamp/html

Gambar 2.59. Tampilan isi dari direktori docker/lamp/html

/home/rifa/docker/lamp/html/					
Name	Size	Changed	Rights	Owner	
		04/05/2020 01:37:43	rw-rwxr-x	rifa	
 index.php	1 KB	04/05/2020 01:47:25	rw-r--r--	rifa	

15. Tampilan dari peletakan file untuk menjalankan docker-compose akan menjadi seperti gambar berikut :

Gambar 2.60. Tampilan isi dari docker-compose

```
rifa@rifa-server:~/docker/lamp$ tree
.
├── docker-compose.yaml
├── html
│   └── index.php
└── php.Dockerfile

1 directory, 3 files
```

16. Setelah semua file yang dibutuhkan siap, maka LAMP server akan dijalankan dengan menggunakan **Modul 2.2.14**.

Jika berhasil, maka tampilannya seperti gambar berikut :

Gambar 2.61. Tampilan isi dari docker-compose

```
Creating lamp_mysql-server_1 ... done
Creating lamp_phpmyadmin_1   ... done
Creating lamp_web-server_1   ... done
```

17. Setelah selesai menjalankan lamp server, service yang telah berjalan dapat dilihat dengan menggunakan **Modul 2.2.15**.

Gambar 2.62. Tampilan isi dari service yang sedang berjalan

rifa@rifa-server:~/docker/lamp\$ docker-compose ps				
Name	Command	State	Ports	
lamp_mysql-server_1	docker-entrypoint.sh mysqld	Up	3306/tcp, 33060/tcp	
lamp_phpmyadmin_1	/docker-entrypoint.sh apac ...	Up	0.0.0.0:5000->80/tcp	
lamp_web-server_1	docker-php-entrypoint apac ...	Up	0.0.0.0:8080->80/tcp	

Gambar 2.63. Tampilan isi dari image yang sedang berjalan


```
rifa@rifa-server:~/docker/lamp$ docker image ls
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
lamp_web-server	latest	d3569bcd40be	40 minutes ago	414MB
mysql	8.0.19	0c27e8e5fcfa	10 days ago	546MB
composer	latest	f9118ee57c51	2 weeks ago	179MB
php	7.4.3-apache	d753d5b380a1	2 months ago	414MB
phpmyadmin/phpmyadmin	5.0.1	9ff80e124320	3 months ago	469MB
hello-world	latest	bf756fb1ae65	4 months ago	13.3kB

18. Untuk menampilkan proyek yang ingin dimasukkan ke dalam docker, maka letakkan data ke dalam direktori ~/docker/lamp/html

Gambar 2.63. Tampilan isi dari image yang sedang berjalan

/home/rifa/docker/lamp/html/				
Name	Size	Changed	Rights	Owner
build		04/05/2020 21:36:47	rw-rw-r--	rifa
dist		06/05/2020 04:55:28	rw-r--r--	rifa
doc		06/05/2020 04:55:29	rw-r--r--	rifa
docs		06/05/2020 04:55:29	rw-r--r--	rifa
font		06/05/2020 04:55:30	rw-r--r--	rifa
makefont		06/05/2020 04:55:30	rw-r--r--	rifa
plugins		06/05/2020 04:55:35	rw-r--r--	rifa
tutorial		06/05/2020 04:55:35	rw-r--r--	rifa
aktivitas_261.json	5 KB	10/01/2020 10:05:27	rw-r--r--	rifa
changelog.htm	9 KB	07/12/2019 11:23:46	rw-r--r--	rifa
composer.json	1 KB	20/11/2019 15:41:28	rw-r--r--	rifa
ex.pdf	6 KB	09/11/2002 16:34:04	rw-r--r--	rifa
ex.php	1 KB	04/04/2009 12:37:48	rw-r--r--	rifa
FAQ.htm	11 KB	07/12/2019 11:55:14	rw-r--r--	rifa
fpdf.css	2 KB	19/07/2008 14:04:50	rw-r--r--	rifa
fpdf.php	47 KB	07/12/2019 10:04:16	rw-r--r--	rifa
index.php	9 KB	21/01/2020 08:58:03	rw-r--r--	rifa
info.htm	2 KB	04/04/2009 12:44:26	rw-r--r--	rifa
install.txt	1 KB	18/06/2011 13:47:48	rw-r--r--	rifa
LICENSE	2 KB	20/11/2019 15:41:28	rw-r--r--	rifa
license.txt	1 KB	03/08/2008 09:52:26	rw-r--r--	rifa
mc_table.php	2 KB	08/02/2004 20:20:06	rw-r--r--	rifa

19. Setelah meletakkan data proyek, lakukan perintah sesuai

Modul 2.2.16 untuk merefresh data docker

20. Ketika ingin membuka aplikasi yang telah dimasukkan ke dalam docker, maka ketik ip kemudian

port dari web server. Contohnya seperti gambar berikut :

```
http://192.168.239.129:8080
```

2.5 Hasil Implementasi

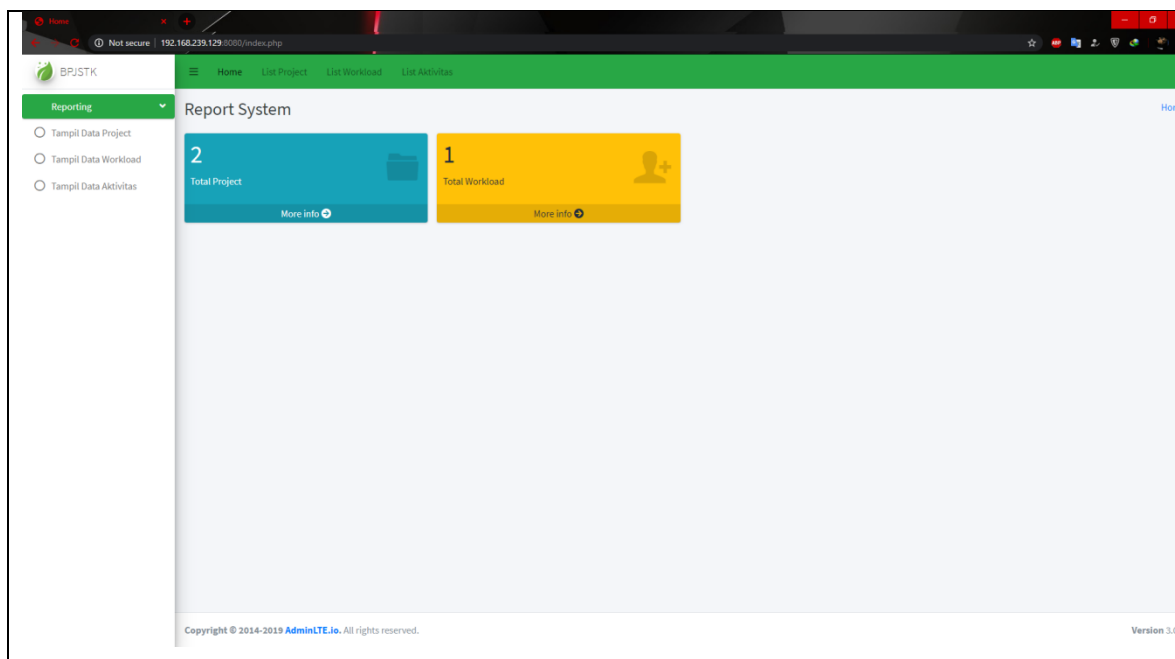
Pada tahap ini, hasil dari implementasi LAMPP dan docker telah berjalan sesuai rancangan dimana dapat menampilkan seluruh halaman dari aplikasi.

Gambar 2.5.1. Tampilan hasil dari LAMPP

Judul	Nomor Surat	Tanggal	Nama	Jabatan	Nama Perusahaan	Nama	Jabatan	Nama Perusahaan	Alamat	Nama Bangunan	Jumlah Kamar	Nomor Kamar	Tanggal Masuk	Tanggal Keluar	Opsi
BPIS Ketenagakerjaan Kabil – Batam	/PSRS Bijak-Cab/BTM/Kabil//	2020-01-08	Dev Ivan Saut Marhua	Kepala Cabang	PT. Binajasa Abadikarya (PT BIAK)	Nico	Asisten	Universitas Pembangunan "Veteran" Yogyakarta	Jl. SWK No.104, Ngropoh, Condongcatur, Kec. Depok, Kabupaten Sleman, Daerah Istimewa Yogyakarta	Twin Tower	3	502.503.504	2020-01-01	2020-04-01	Hapus Edit Hapus
BPIS Ketenagakerjaan Kabil – Batam	/PSRS Bijak-Cab/BTM/Kabil//	2020-01-02	Dev Ivan Saut Marhua	Kepala Cabang	PT. Binajasa Abadikarya (PT BIAK)	Nico	Asisten	Universitas Pembangunan "Veteran" Yogyakarta	Jl. SWK No.104, Ngropoh, Condongcatur, Kec. Depok, Kabupaten Sleman, Daerah Istimewa Yogyakarta	4	3	500.501.502	2020-01-01	2020-06-07	Hapus Edit Hapus
BPIS Ketenagakerjaan Kabil – Batam	/PSRS Bijak-Cab/BTM/Kabil//	2020-01-02	Dev Ivan Saut Marhua	Kepala Cabang	PT. Binajasa Abadikarya (PT BIAK)	Nico	Asisten	Universitas Pembangunan "Veteran" Yogyakarta	Jl. SWK No.104, Ngropoh, Condongcatur, Kec. Depok, Kabupaten Sleman, Daerah Istimewa Yogyakarta	4	3	500.501.502	2020-01-01	2020-06-07	Hapus Edit Hapus
BPIS Ketenagakerjaan Kabil – Batam	/PSRS Bijak-Cab/BTM/Kabil//	2020-01-02	Dev Ivan Saut Marhua	Kepala Cabang	PT. Binajasa Abadikarya (PT BIAK)	Nico	Asisten	Universitas Pembangunan "Veteran" Yogyakarta	Jl. SWK No.104, Ngropoh, Condongcatur, Kec. Depok, Kabupaten Sleman, Daerah Istimewa Yogyakarta	6	1	34	2020-01-04	2020-12-19	Hapus Edit Hapus
BPIS Ketenagakerjaan Kabil – Batam	/PSRS Bijak-Cab/BTM/Kabil//	2020-01-11	Dev Ivan Saut Marhua	Kepala Cabang	PT. Binajasa Abadikarya (PT BIAK)	Nico	Asisten	Universitas Pembangunan "Veteran" Yogyakarta	Jl. SWK No.104, Ngropoh, Condongcatur, Kec. Depok, Kabupaten Sleman, Daerah Istimewa Yogyakarta	3	2	333	2020-01-02	2020-07-25	Hapus Edit Hapus
BPIS Ketenagakerjaan Kabil – Batam		0000-00-00								0			0000-00-00	0000-00-00	Hapus Edit Hapus
BPIS Ketenagakerjaan Kabil – Batam		0000-00-00								0			0000-00-00	0000-00-00	Hapus Edit Hapus

[Tambah Data](#)

Gambar 2.5.1. Tampilan hasil dari docker



2.6 Pengujian Singkat

Pada tahap ini, di docker disini hanya sebatas dari PHP native. Alasan nya yaitu di bagian docker-compose.yml tidak melakukan instalasi Code Igniter. Di docker-compose apa yang akan digunakan harus dimasukkan ke dalam docker. di docker-compose ini penulis tidak menginstall Code Igniter di docker-compose.

BAB III

JADWAL Pengerjaan dan Pembagian Tugas

3.1 Agenda Pengerjaan

Berikut pada **Tabel 3.1** merupakan pembagian jenis tugas proyek akhir terhadap alokasi waktu pengerjaan pada bulan Maret, April, dan Mei tahun 2020:

Tabel 3.1 Agenda Pengerjaan Proyek

No.	Jenis Tugas	Waktu Pengerjaan								
		Maret		April				Mei		
		3	4	1	2	3	4	1	2	3
1.	Analisa Persoalan									
2.	Pembagian Tugas									
3.	Percobaan LAMPP									
4.	Pengerjaan Laporan dan Pengerjaan LAMPP									
5.	Percobaan Docker									
6.	Pengerjaan Laporan Hasil Revisi Submisi 1									
7.	Pengerjaan Docker									
8.	Pengerjaan Laporan Hasil Revisi Submisi 2									
9.	Presentasi Proyek Akhir									

3.2 Keterangan Pembagian Tugas

Berikut pada **Tabel 3.2** merupakan pembagian tugas-tugas pada proyek akhir terhadap anggota pada tim pembuatan proyek akhir:

Tabel 3.2 Pembagian Tugas Proyek

No.	Keterangan Tugas	Penanggung Jawab
1.	Perancangan Arsitektur Cloud Computing	Rifa, Nico
2.	Pengujian Singkat	Rifa, Nico
3.	Pembuatan laporan	Rifa, Nico
4.	Percobaan Lampp	Rifa
5.	Pengujian Lampp	Nico
6.	Percobaan Docker	Rifa, Nico
7.	Pengujian Docker	Rifa, Nico

BAB IV

KESIMPULAN DAN SARAN

4.1 Kesimpulan

Hasil yang bisa didapat dari laporan yang telah dibuat antara lain :

- a. Masalah untuk penggunaan docker untuk program yang menggunakan framework belum bisa diuji di laporan ini.
- b. Pengujian lampp tanpa menginstall Code Igniter bisa dilakukan
- c. Pengujian docker tidak akan menginstall framework terlebih dahulu akan muncul error.

4.2 Saran

Saran untuk laporan ini yaitu agar bisa mencoba versi lain dari docker, seperti menggunakan program dengan bahasa selain php dan menggunakan database selain mysql.

DAFTAR PUSTAKA

_____. Install Docker Engine on Ubuntu. Diakses pada pekan ke-2 Mei 2020, dari <https://docs.docker.com/engine/install/ubuntu/>

Shovon, Shahriar. Set up a LAMP server with Docker. Diakses pada pekan ke-1 dan ke-2 Mei 2020, dari https://linuxhint.com/lamp_server_docker/

_____. Cara Install dan Konfigurasi Docker di Ubuntu 18.04. Diakses pada pekan ke-1 dan pekan ke-2 Mei 2020, dari <https://www.linuxid.net/31061/cara-install-dan-konfigurasi-docker-di-ubuntu-18-04/>

Andreal, Andri, Docker-Nginx-Php-MySql at Linux Ubuntu Desktop 18.04 . Diakses pada tanggal 21 April 2020, dari <https://www.youtube.com/watch?v=ra1tI2J2WAA&t=281s>

Neykov, Nevyan, Apache, PHP & MySql on Docker on Ubuntu 19.04. Diakses pada tanggal 21 April 2020, dari https://www.youtube.com/watch?v=_mwWxgfZ7Zc

Anderson, Melissa. , How To Install Docker Compose on Ubuntu 16.04. Diakses pada pekan ke-3 April 2020, dari <https://www.digitalocean.com/community/tutorials/how-to-install-docker-compose-on-ubuntu-16-04>

LAMPIRAN

Lampiran pada bagian ini dapat berupa screenshoot, listing program yang terlalu panjang, dan sebagainya, atau dapat juga tugas bilamana diminta oleh asisten praktikum.