

**PRAKTIKUM TEKNOLOGI CLOUD COMPUTING
LAPORAN PROYEK AKHIR**

**SISTEM INFORMASI INVENTARIS BARANG SEKOLAH TELADAN
MENGUNAKAN UBUNTU LAMPP DAN IMPLEMENTASINNYA
MENGUNAKAN DOCKER FILE**



DISUSUN OLEH:

**NAMA ANGGOTA : AZRA FIRDAUZ RIZKY R. 123170022
ADRIAN FATHUR S 123170051
KELAS : E
ASISTEN PRAKTIKUM : JALUANDA PARAMA, S.Kom.
WAHYU AJI NUGROHO, S.Kom.**

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

2020

HALAMAN PENGESAHAN

SISTEM INFORMASI INVENTARIS BARANG SEKOLAH TELADAN MENGUNAKAN UBUNTU LAMPP DAN IMPLEMENTASINNYA MENGUNAKAN DOCKER FILE

Disusun oleh :

Azra Firdauz Rizky R.

123170022

Adrian Fathur Setyawan

123170051

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

Menyetujui,

Asisten Praktikum

Asisten Praktikum

Jaluanda Parama, S.Kom.

Wahyu Aji Nugroho, S.Kom.

Mengetahui,

Ka. Lab. Sistem Digital

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

NIK. 2 8201 13 0425 1

KATA PENGANTAR

Puji dan syukur kami panjatkan kehadiran Allah swt atas segala limpahan rahmat-Nya sehingga kami dapat menyelesaikan penyusunan laporan akhir praktikum teknologi cloud computing. Adapun isi dari laporan akhir ini adalah project akhir kami yang berjudul Sistem Informasi Inventaris Barang Sekolah Teladan Menggunakan Ubuntu LAMPP dan Implementasinya Menggunakan Docker File yang membahas tentang informasi inventaris barang yang dimiliki oleh Sekolah Teladan agar terciptanya suasana transparansi informasi inventaris barang dan menjaga integritas data inventaris barang.

Tak lupa pula kami mengucapkan banyak terima kasih kepada dosen ataupun asisten laboratorium yang selalu membimbing dan mengajari kami dalam melaksanakan praktikum dan dalam penyusunan laporan akhir ini. Serta semua pihak yang telah membantu kami dalam penyusunan laporan akhir ini. Karena laporan akhir ini masih jauh dari sempurna oleh karena itu kritik dan saran yang membangun masih kami harapkan untuk penyempurnaan laporan akhir ini.

Atas perhatian dari semua pihak yang membantu penulisan laporan akhir ini kami ucapkan terima kasih. Semoga laporan ini dapat dipergunakan sesuai dengan keperluan.

Yogyakarta, 31 Maret 2020

Penyusun

DAFTAR ISI

HALAMAN PENGESAHAN	i
KATA PENGANTAR	ii
DAFTAR ISI	iii
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	5
2.1. Komponen yang Digunakan	5
2.2. Rancangan Arsitektur <i>Cloud Computing</i>	7
2.3. Parameter dan Konfigurasi	10
2.4. Tahap Implementasi	14
2.5. Hasil Implementasi	21
2.6. Pengujian Singkat	23
BAB III JADWAL Pengerjaan dan Pembagian Tugas	24
3.1. Agenda Pengerjaan	24
3.2. Keterangan Pembagian Tugas	24
BAB IV KESIMPULAN DAN SARAN	25
4.1. Kesimpulan	25
4.2. Saran	25
DAFTAR PUSTAKA	26
LAMPIRAN	27

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Cloud Computing pada saat ini sedang sering digunakan dan terdengar dalam perkembangan teknologi. Banyak teknologi dan system yang sudah menggunakan *cloud computing*, *cloud computing* juga sudah mulai diterapkan dalam lingkungan pekerjaan. Dari penjelasan tersebut apa sebenarnya makna dari *Cloud Computing*. *Cloud Computing* adalah pengembangan suatu teknologi virtualisasi yang dapat digunakan untuk membantu dalam mengelola suatu pekerjaan dan dapat membantu suatu computer tunggal dalam menduplikasi mesin secara maya (Sulistyowati et al., 2012).

Sebelum *cloud computing* mulai terkenal seperti sekarang ini, terjadi beberapa fenomena seperti virtualisasi, grid computing, ASP/*Application Service Provision* dan juga *Software as a Service/SaaS*. Pada dasarnya, *cloud computing* bias sangat cepat menyebar dan menjadi sangat disukai karena adanya layanan yang mudah untuk dikonsumsi, serta memiliki berbagai macam manfaat seperti data yang tersimpan di server secara terpusat, menjamin keamanan pengguna, fleksibilitas dan skalabilitas tinggi dan memberikan jaminan investasi jangka panjang.

Penggunaan *cloud computing* sekarang sudah banyak diterapkan. Salah satunya yaitu implementasi *cloud computing* pada instansi pemerintahan seperti Kementrian Kominfo. Tujuan dari penggunaan *cloud computing* tersebut pada instansi pemerintahan dikarenakan biaya penggunaan *cloud computing* yang murah karena tidak perlu untuk maintainance data center sendiri. Selain itu, penggunaan *cloud computing* ini juga dapat menjadi salah satu cara untuk mengintegrasikan data dan mempermudah mengambil data sesuai dengan hak aksesnya. Semisal Kementrian Kominfo ingin mengambil data yang ada di Kemenpora sesuai hak aksesnya maka teknologi *cloud computing* ini sangat bermanfaat karena pemngambilan data tersebut sangat flexibel sesuai hak aksesnya.(Wildana, 2018)

Sistem informasi inventaris barang Sekolah Teladan ini merupakan suatu sistem informasi yang berbasis web yang berfungsi untuk membantu administrator untuk mendata barang barang inventaris yang ada di Sekolah Teladan. Banyaknya inventaris barang menyebabkan penggunaan memori yang cukup banyak. Hal tersebut juga dapat menyebabkan lambatnya akses web. Selain itu, Sekolah Teladan juga perlu mengeluarkan *cost* yang lumayan untuk melakukan pembaruan, perbaikan dan perawatan memori. Oleh Sebab itu, untuk mengurangi biaya perawatan maupun pembaruan memori maka

diperlukan teknologi *cloud computing*. Selain itu juga, data data yang telah di *upload* ke server akan terjaga keamanannya apabila terjadi kerusakan pada memori local maka tidak akan berpengaruh pada hilangnya data yang telah terupload ke server. Penggunaan teknologi *cloud computing* ini juga dapat menciptakan suasana transparansi terhadap informasi tentang inventaris barang karena konsep *cloud computing* ini dapat diakses oleh siapa saja yang memiliki hak untuk mengakses cloud tersebut.

Didalam pengembangan sistem informasi inventaris barang Sekolah Teladan ini kami menggunakan *virtual hosting* dan hal tersebut memunculkan masalah yaitu meningkatnya beban web server sejalan dengan banyaknya aplikasi web yang harus ditangani. Data yang digunakan aplikasi web biasanya disimpan didalam satu server databse membuat kemungkinan adanya kerusakan data akibat *hacker*. Oleh karena itu untuk menangani dan meminimalisir kejadian seperti itu diperlukan implementasi docker. Teknologi docker merupakan teknologi suatu platform *open source* bagi pengembang perangkat lunak dan pengelolaan sistem jaringan untuk membangun, mengirimkan dan menjalankan aplikasi-aplikasi terdistribusi (Adiputra, 2015). Selain itu sistem ini juga dimplementasikan didalam virtual machine.

Dalam pengembangan website tersebut, tahap-tahap yang dilakukan untuk menyelesaikan projek tersebut yaitu dengan membuat website inventaris yang dipastikan sudah dapat berjalan dengan normal. Kemudian menyiapkan server Ubuntu LAMPP yang akan digunakan sebagai tempat *hosting* website tersebut. Apabila semua sudah dipastikan normal, maka barulah website tersebut di *hosting*. Untuk menyempurnakan dan mengurangi beban server diperlukannya implementasi docker file dengan penginstalan docker engine dan docker compose.

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 Sistem Inventaris Barang Sekolah Teladan yang telah dibuat sebelumnya dengan menggunakan Ubuntu 18.04.4 dan LAMPP (Apache 2.4, PHP 7.2.24, MySQL).
2. Mengimplementasikan hosting Sistem Invetaris Barang Sekolah Teladan secara local menggunakan LAMPP.
3. Mengimplementasikan pembuatan Sistem Inventaris Barang Sekolah Teladan menggunakan dockerfile.

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 inventaris barang menjadi tidak terbatas dikarenakan sistem telah sepenuhnya beralih menggunakan *cloud computing*.
4. Implementasi dari *dockerfile* membuat beban kerja server web semakin ringan, sehingga performa web yang telah dibangun semakin baik.

1.4 Tahap Penyelesaian Proyek Akhir

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

1. Menganalisis kebutuhan dari sistem ABC untuk ditransformasikan ke dalam arsitektur *cloud computing* menggunakan basis IaaS.
2. Implementasi Sistem Informasi Inventaris Barang Menggunakan LAMPP.

Tahap-tahap instalasi LAMPP :

- a. Melakukan pembuatan file VMware.
- b. Melakukan instalasi apache2 pada mesin VM.
- c. Melakukan instalasi php pada mesin VM.
- d. Melakukan instalasi MySQL pada mesin VM.
- e. Melakukan instalasi phpmyadmin pada mesin VM.
3. Hosting proyek pertama menggunakan tools WinSCP sebagai tools pemindah file local ke server.
4. Implementasi docker pada Sistem Informasi Inventaris Barang Menggunakan Docker.

Tahap-tahap penggunaan docker :

- a. Melakukan instalasi docker
- b. Melakukan instalasi docker-compose
- c. Membuat file docker
- d. Membuat file docker-compose

5. *Import* database kedalam service yang telah dituliskan didalam file docker-compose.yml
6. *Build up* docker-compose agar proyek kedua dapat berjalan dengan baik.

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 membuat dan membangun “Sistem Informasi Inventaris Barang Sekolah Teladan Menggunakan Ubuntu LAMPP” yang di dasari dengan konsep *cloud computing*, maka diperlukan beberapa komponen baik software maupun hardware yang perlu diperhatikan. Berikut akan dijelaskan terlebih dahulu dalam bentuk poin-poin singkat:

1. Sistem ini dibangun dengan bahasa pemrograman PHP dengan versi 7.2.24 dengan server Ubuntu versi 18.04.4.
2. Penggunaan CodeIgniter untuk mempercepat kinerja pengerjaan project
3. Untuk DBMS menggunakan MySQL versi 5.2 dengan sekali install complete pada LAMPP.

Berdasarkan penjelasan poin-poin tersebut, untuk komponen utama penyusun *cloud computing* maka dapat disimpulkan dalam bentuk tabel seperti dibawah ini :

Tabel 2.1 Spesifikasi VM *cloud computing* untuk proyek pertama

No.	Nama Parameter	Nilai	Keterangan
1.	Merek Server	Virtual Machine dengan VMWare Workstation	Tidak menggunakan <i>hardware</i> fisik secara langsung, melainkan menggunakan aplikasi <i>virtual machine</i> .
2.	Prosesor	2 core @2.4Ghz	Prosesor dari <i>hypervisor</i> yang dialokasikan ke <i>guest</i> .
3.	Konfigurasi Jaringan <i>Guest OS</i>	Mode NAT	Mode adapter jaringan VM <i>guest</i> yang digunakan.
		IP: 192.168.79.128/24	Alamat IP dan <i>network</i> yang digunakan oleh <i>guest OS</i> .
		DNS: 192.168.79.1	Alamat IP untuk DNS <i>guest OS</i> .
		GW: 192.168.79.255	Alamat untuk <i>gateway</i> atau gerbang menuju akses jaringan luar.
4.	Versi Ubuntu	Ubuntu 18.04.4 LTS	ISO Ubuntu yang digunakan untuk <i>guest OS</i> .
5.	<i>Harddisk</i>	20 GB	Alokasi <i>harddisk</i>
6.	<i>Memory RAM</i>	2 GB	Alokasi <i>memory</i> untuk <i>guest OS</i>
7.	<i>Network Adapter</i>	NAT	Digunakan untuk menghubungkan Koneksi internet terhadap OS host sehingga mesin virtualisasinya juga dapat memanfaatkan koneksi internet

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	Preprosesor bahasa pemrograman HTML, termasuk CSS dan JS.
		PHP 7.2.24	Bahasa Pemrograman yang digunakan dalam membangun sistem informasi kost
		MySQL	Basis Data yang digunakan dalam menyimpan data sistem informasi kost
		phpmyadmin	Perangkat lunak yang dapat dimanfaatkan untuk mengelola basis data dalam MySQL

2.1.1 Komponen pada Proyek Kedua

Pada proyek kedua adalah membangun sebuah website dengan implementasi menggunakan Dockerfile. Pada proyek dua ini juga menggunakan *source file* VMware Workstation Pro yang sama dengan proyek pertama yang dijelaskan pada **Tabel 2.1**. Berikut ini penjelasan secara singkat dari proyek kedua dalam bentuk point-point :

1. Sistem Inventaris Barang Sekolah Teladan akan diimplementasikan menggunakan teknologi docker dengan Ubuntu Server yang sama dengan proyek pertama atau *source file* VMware yang sama dengan proyek yang pertama .
2. Adanya penginstallan docker engine dan docker compose pada Ubuntu OS sebelum melakukan implentasi.
3. Pada service web atau container templateyysi-tcc terdapat beberapa *images* yaitu image apache dan image php, sefdangkan pada service db atau container templatteyysi-db diperlukan image MySQL.

Spesifikasi yang digunakan dalam Ubuntu OS yang telah dibuat dalam VM tersebut adalah sebagai berikut.

Tabel 2.3 Spesifikasi Ubuntu OS untuk proyek kedua

No.	Nama Parameter	Nilai/images	Keterangan
1	Docker-compose	Version 1.25.5	Untuk mengeksekusi beberapa service(terdiri dari beberapa container)/container secara bersamaan
2.	Docker-engine	Version 19.03.8	Teknologi untuk membuat container-container
		PHP 7.2.24	Bahasa Pemrograman yang digunakan dalam membangun sistem informasi kost
		MySQL	Basis Data yang digunakan dalam menyimpan data sistem informasi kost

		Apache 7.3	Preproesor bahasa pemrograman HTML, termasuk CSS dan JS.
--	--	------------	--

2.2 Rancangan Arsitektur *Cloud Computing*

Rancangan Arsitektur *Cloud Computing* tugas proyek akhir ini terbagi menjadi dua sesuai dengan judul yang telah dituliskan sebelumnya. Berikut akan dibahas keduanya masing-masing dalam sub bab terkait.

2.2.1 Rancangan Arsitektur *Cloud Computing* Proyek Pertama

Rancangan arsitektur *cloud computing* pada proyek pertama ini merupakan hosting lokal Sistem Inventaris Barang Sekolah Teladan menggunakan LAMPP pada VM. Berikut ini penjelasan point-point secara singkat terhadap rancangan arsitektur *cloud computing* proyek pertama :

1. Pada proyek ini kami menggunakan hardware Laptop LENOVO Ideapad 310 sebagai hardware utamanya.
2. Untuk layer Hypervisornya, proyek ini menggunakan hypervisor VMware Workstation Pro.
3. Untuk layer VM, didalam layer VM ini dilakukan isntallasi Ubuntu server fungsinya untuk mendapatkan IP Address virtualisasi yaitu 192.168.79.128.
4. Untuk layer OS, daidalam layer OS dilakukan installasi OS Ubuntu 18.04.4 sebagai OS dari VM nya.
5. Untuk layer terakhir rancangan arsitektur *cloud computing* proyek pertama yaitu layer app. Layer app ini fungsinya untuk tempat dimana aplikasi-aplikasi pendukung pembuatan proyek pertama ini diinstall, seperti: Apache, MySQL, PHPmyAdmin (LAMPP) , serta Sistem Inventaris Barang Sekolah Teladan.

Berdasarkan point-point diatas, rancangan arsitektur *cloud computing* proyek pertama dapat dilihat pada **Gambar 2.1** dibawah ini:

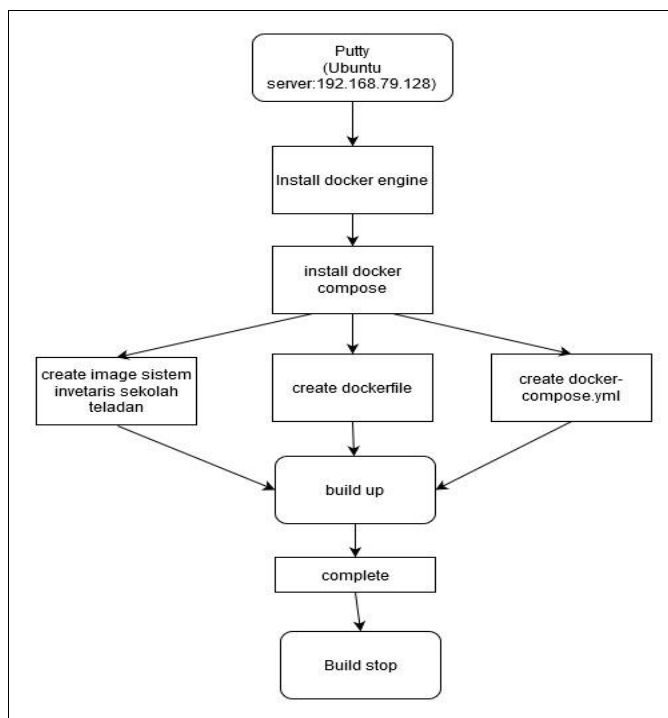
2.2.2 Rancangan Arsitektur *Cloud Computing* Proyek Kedua

Pada rancangan arsitektur *cloud computing* proyek kedua yaitu membangun Sistem Inventaris Barang Sekolah Teladan dengan implementasi teknologi docker. Berikut ini penjelasan secara singkat dalam bentuk point-point terhadap rancangan arsitektur *cloud computing* proyek kedua:

1. Untuk layer hypervisor masih menggunakan VMware Workstation Pro.
2. Untuk layer VM, masih menggunakan Ubuntu server yang sama dengan proyek pertama dengan *IP Address* 192.168.79.128.
3. Untuk layer OS, *operating system* yang dipakai yaitu Ubuntu versi 18.04.4 LTS.
4. Untuk layer docker engine, teknologi ini ada karena kami install docker engine dimana fungsi dari docker engine ini digunakan untuk *support* pembuatan dockerfile.
5. Untuk layer docker compose, layer ini fungsinya yaitu untuk menjalankan beberapa layer service yang ada diatas layer docker compose secara bersamaan.
6. Didalam proyek kedua ini terdapat dua layer service, layer service pertama yaitu service web dan layer service kedua yaitu service db. Kedua service tersebut memiliki fungsi yang berbeda. Untuk service web mengandung terdapat container yang bernama templateyysi-tcc yang didalamnya terdapat dua images yaitu images templateyysi yang mana berupa isi kodingan dari sistem inventaris barang sekolah teladan dan images apache dimana memiliki fungsi agar untuk menerjemahkan bahasa html,css, dan javascript, Sedangkan untuk service db mengandung container yang bernama templateyysi-db yang didalamnya terdapat images MySQL yang fungsinya untuk pembuatan database sistem inventaris barang sekolah teladan.

Berdasarkan point-point diatas, rancangan arsitektur *cloud computing* proyek pertama dapat dilihat pada **Gambar 2.3** dibawah ini:

Untuk proses pengerjaan dari proyek kedua ini dapat dilihat pada **Gambar 2.4**, seperti dibawah ini:



2.3 Parameter dan Konfigurasi

Pada tahap parameter dan konfigurasi akan menjelaskan tentang isian parameter dan konfigurasi terhadap komponen alat dan bahan yang digunakan dalam pengerjaan proyek pertama dan proyek kedua. Proyek pertama dan kedua memiliki kesamaan terhadap

file VMware yang digunakan. Kedua proyek tersebut dibangun dalam area file VMware yang sama sehingga untuk konfigurasi terhadap pembuatan VMware hanya dilakukan sekali saja, untuk pembuatannya dapat dilihat pada sub bab **2.4 Tahap Implementasi**, Sedangkan karena adanya perbedaan terhadap parameter dan konfigurasi antara proyek pertama dan kedua maka proyek parameter dan konfigurasi pada tugas proyek akhir ini dibagi menjadi dua sesuai dengan parameter dan konfigurasi masing masing proyek. Berikut akan dibahas keduanya masing-masing dalam sub bab terkait.

2.3.1 Parameter dan Konfigurasi Proyek Pertama

Pada bagian ini akan dijelaskan parameter dan konfigurasi proyek pertama yaitu tentang instalasi apache2, instalasi MySQL, instalasi PHP, dan instalasi phpmyadmin. Masing-masing instalasi tersebut akan dijelaskan sedetail mungkin. Parameter yang digunakan untuk instalasi apache2 dapat dilihat pada penjelasan parameter yang digunakan untuk instalasi apache2 dapat dilihat pada penjelasan pada **Modul 2.1** berikut ini:

```
$ sudo apt install apache2
```

Keterangan:

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- apt : merupakan package manager pada Ubuntu
- install : parameter tambahan pada apt untuk mengeksekusi perintah instalasi paket aplikasi
- apache2 : nama paket aplikasi untuk Apache

```
$ sudo ufw allow in "Apache Full"
```

Keterangan: Untuk mengatur Firewall agar port 80 untuk protocol HTTP dan port 443 untuk protocol HTTPS diizinkan oleh sistem Ubuntu.

Modul 2.1 Parameter instalasi apache2

Parameter yang digunakan untuk instalasi MySQL dapat dilihat pada penjelasan **Modul 2.2** berikut ini:

```
$ sudo apt install mysql-server
```

Keterangan:

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- apt : merupakan package manager pada Ubuntu
- install : parameter pada apt untuk mengeksekusi perintah instalasi paket
- mysql-server : nama paket untuk MySQL

*Untuk kesimpulan bahwa perintah diatas digunakan untuk menginstal paket MySQL di Ubuntu

```
$ sudo mysql_secure_installation
```

Keterangan: Untuk mengatur keamanan pada MySQL contohnya penamaan username dan password

Modul 2.2 Parameter instalasi MySQL

Parameter yang digunakan untuk instalasi PHP dapat dilihat pada penjelasan **Modul 2.3** berikut ini:

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

Keterangan:

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- apt : merupakan package manager pada Ubuntu
- install : parameter tambahan pada apt untuk mengeksekusi perintah instalasi paket
- php libapache2-mod-php php-mysql : nama paket untuk PHP
- * untuk perintah diatas fungsinya untuk instalasi PHP dan melakukan tahap default sehingga PHP siap digunakan

```
$ sudo nano /var/www/html/info.php
```

Keterangan: Untuk membuat file baru bernama info.php yang berada di direktori var/www/html

Modul 2.3 Parameter instalasi PHP

Parameter yang digunakan untuk instalasi phpmyadmin dapat di lihat pada penjelasan **Modul 2.4** berikut ini:

```
$ sudo apt install phpMyAdmin php-mbstring php-gettext
```

Keterangan:

- sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root)
- apt : merupakan package manager pada Ubuntu
- install : parameter tambahan pada apt untuk mengeksekusi perintah instalasi paket
- phpMyAdmin php-mbstring php-gettext: nama paket untuk phpmyadmin

```
$ sudo mysql -u root
```

Keterangan: Untuk masuk ke MySQL sebagai user root.

```
mysql> UPDATE mysql.user SET plugin = 'mysql_native_password',  
authentication_string = PASSWORD('123170051') WHERE User = 'root';
```

Keterangan: Untuk mengubah password dari user 'root' menjadi '123170051'

```
mysql> FLUSH PRIVILEGES;
```

Keterangan: Untuk mererefresh akun yang terkoneksi dengan phpmyadmin

```
$ sudo chown fathur /var/www/html/
```

Keterangan: Untuk memberikan akses ke User fathur agar dapat mengakses direktori var/www/html

Modul 2.4 Parameter instalasi phpmyadmin

2.3.2 Parameter dan Konfigurasi Proyek Kedua

Pada bagian ini dijelaskan parameter dan konfigurasi proyek kedua yaitu mengenai instalasi docker engine, instalasi docker compose, *import* database pada images MySQL ketiga instalasi tersebut dilakukan diatas layer VM dengan OS Ubuntu. Masing – masing instalasi tersebut akan dijelaskan pada sub bab ini sedetail mungkin. Berikut ini penjelasan parameter yang digunakan untuk instalasi dapat dilihat pada penjelasan **Modul 2.5** berikut ini:

<pre>\$ sudo apt-get update</pre> <p>Keterangan :</p> <ul style="list-style-type: none"> -sudo : perintah untuk eksekusi suatu command dengan hak akses tertinggi (root) - apt-get update akan memberikan informasi tentang jumlah komponen yang perlu diperbaharui.
<pre>\$ sudo apt-get install docker-ce docker-ce-cli containerd.io</pre> <p>Keterangan :</p> <ul style="list-style-type: none"> -perintah command line tersebut digunakan untuk update apt package index dan melakukan instalasi terhadap docker engine dan container.
<pre>\$ apt-cache madison docker-ce</pre> <p>Keterangan :</p> <ul style="list-style-type: none"> -perintah command line tersebut digunakan untuk memunculkan spesifik versi docker engine, lalu nanti akan memunculkan list version docker engine sesuai proyek.
<pre>\$ sudo apt-get install docker-ce=<VERSION_STRING> docker-ce- cli=<VERSION_STRING> containerd.io</pre> <p>Keterangan:</p> <ul style="list-style-type: none"> -perintah command line tersebut digunakan untuk mengeksekusi pilihan docker engine sesuai dengan proyek. -ganti <version_string> dengan versi ubuntu pada proyek
<pre>\$ sudo docker run hello-world</pre> <p>Keterangan :</p> <ul style="list-style-type: none"> -perintah command line tersebut digunakan untuk mengecek bahwa docker engine telah terpasang dan berjalan dengan baik.

Modul 2.5 Parameter instalasi docker engine

Parameter yang digunakan untuk instalasi docker compose dapat dilihat pada penjelasan **Modul 2.6**, untuk penerapan docker compose dapat dilihat pada sub bab **2.4 Tahap Implementasi** berikut ini:

<pre>\$ sudo curl -L "https://github.com/docker/compose/releases/download/1.25.5/docker- compose-\$(uname -s)-\$(uname -m)" -o /usr/local/bin/docker-compose</pre> <p>Keterangan : Perintah command line diatas digunakan untuk mendownload docker compose paling baru dari alamat github yang tertulis.</p>
<pre>\$ sudo chmod +x /usr/local/bin/docker-compose</pre> <p>Keterangan : Perintah command line diatas digunakan untuk mengizinkan eksekusi docker compose dalam bentuk binary.</p>
<pre>\$ docker-compose --version</pre> <p>Keterangan : perintah command line diatas digunakan untuk mengecek apakah docker compose siap digunakan beserta versi docker compose nya.</p>

Modul 2.6 Parameter instalasi docker compose

Parameter yang digunakan untuk proses eksekusi docker compose dapat dilihat pada penjelasan **Modul 2.7**, proses eksekusi dilakukan setelah dilakukan pembuatan file Dockerfile dan docker-compose.yml. berikut ini:

<code>\$ sudo docker-compose up</code>
Keterangan : Perintah command line diatas digunakan untuk membangun dan mengeksekusi file docker-compose.yml yang berisi container container yang didalamnya ada images yang digunakan dalam membangun aplikasi proyek.
<code>\$ sudo docker-compose down</code>
Keterangan : Perintah command line diatas digunakan untuk meremove service yang dianggap tidak sesuai dengan ekspektasi setelah dilakukan build service.
<code>\$ sudo docker-compose start</code>
Keterangan : Perintah command line diatas digunakan untuk melakukan power on pada service yang telah di build
<code>\$ sudo docker-compose stop</code>
Keterangan : Perintah command line diatas digunakan untuk melakukan power off pada service yang sedang berjalan.

Modul 2.7 Parameter proses eksekusi docker compose

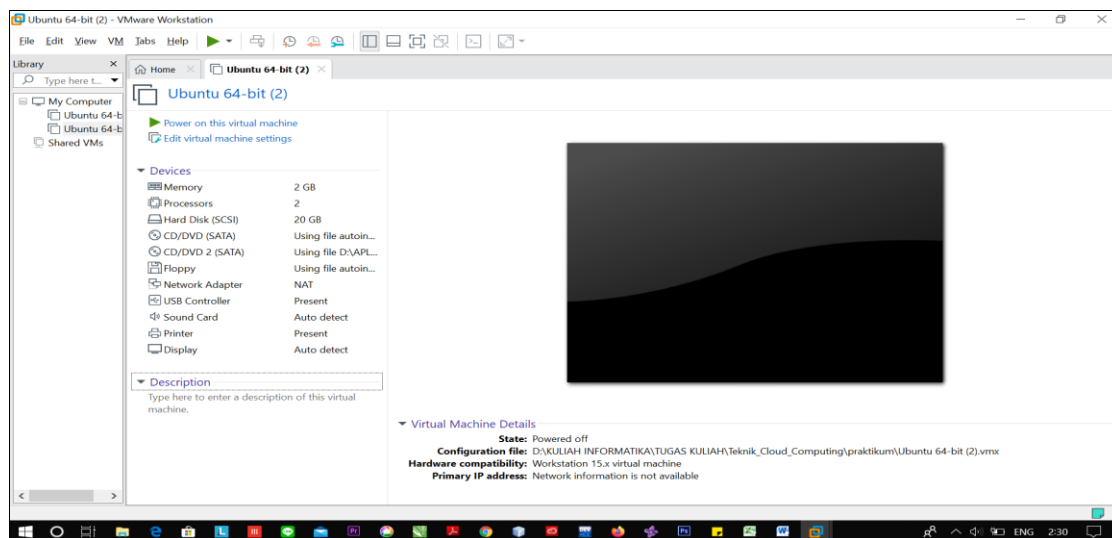
Parameter yang digunakan untuk *import* file database pada image MySQL dapat dilihat pada penjelasan **Modul 2.8**, pada proyek kedua ini menggunakan nama database yaitu teladan.sql, tahap ini dilakukan setelah pembuatan dockerfile, docker-compose.yml dan sudah dilakukan *build up* berikut ini:

<code>\$ sudo docker exec -i templateyysi-db mysql -ufathur -p123170051 -e "create database teladan"</code>
Keterangan : Perintah diatas digunakan untuk membuat database teladan pada service mysql dan container templateyysi-db yang sedang berjalan dengan username fathur dan password 123170051
<code>\$ sudo docker exec -i templateyysi-db mysql -ufathur -p123170051 teladan < ./teladan.sql</code>
Keterangan : Perintah diatas digunakan untuk mengimport file database teladan.sql kedalam database yang telah dibuat yaitu teladan pada container templateyysi-db.

Modul 2.8 Parameter proses import file database pada image MySQL

2.4 Tahap Implementasi

Pada sub bab tahap implementasi ini akan menunjukan dan menjelaskan tahapan implementasi dari parameter dan konfigurasi baik proyek pertama maupun proyek kedua yang dituliskan pada sub bab sebelumnya. Untuk konfigurasi ini untuk proyek pertama dan kedua menggunakan file VMware yang sama, sehingga pembuatan konfigurasi ini dilakukan sekali saja. Tahap implementasi ditunjukan pada **Tabel 2.1** mengenai konfigurasi untuk VMware Workstation dapat dilihat pada **Gambar 2.5** berikut ini:

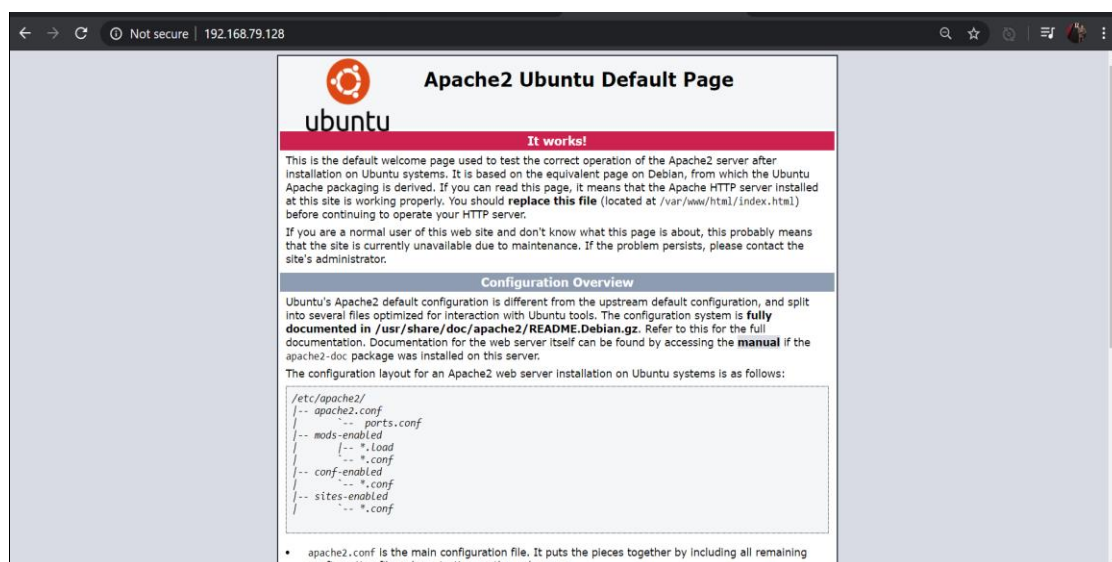


Gambar 2.5 Pembuatan virtual machine menggunakan VMware dengan OS Ubuntu

Pada tahap implementasi proyek akhir ini dibagi menjadi dua tahap implementasinya dikarenakan adanya proses implementasi antara proyek pertama dan proyek kedua yang berbeda, sehingga akan dijelaskan secara masing-masing pada sub sub bab di bawah ini:

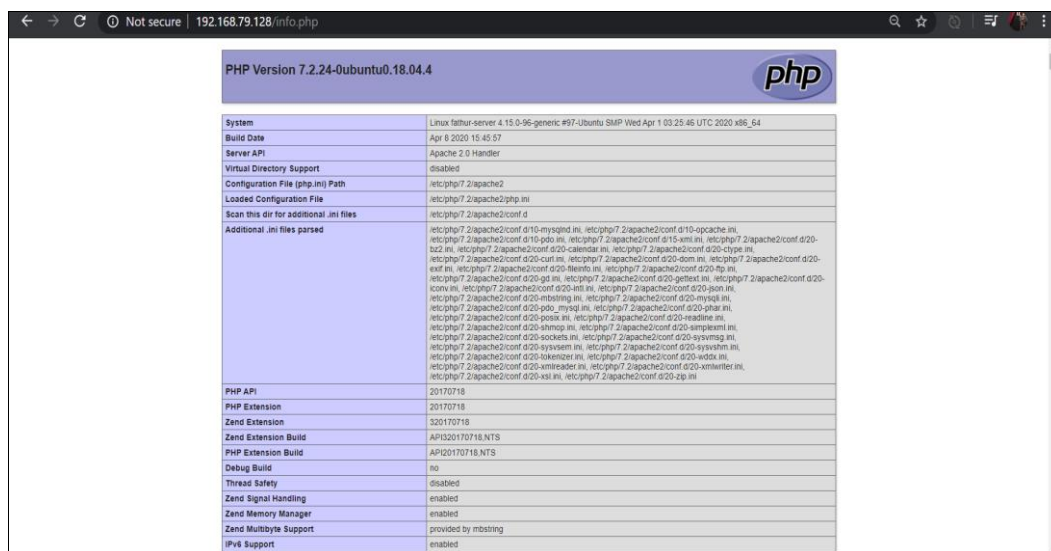
2.4.1 Tahap Implementasi Proyek Pertama

Tahap implementasi **Modul 2.1** mengenai instalasi apache2 pada OS Ubuntu VMware Workstation. Hasil dari perintah command line pada **Modul 2.1**, apabila tahap implementasinya benar maka akan ditunjukkan pada **Gambar 2.6** berikut ini:



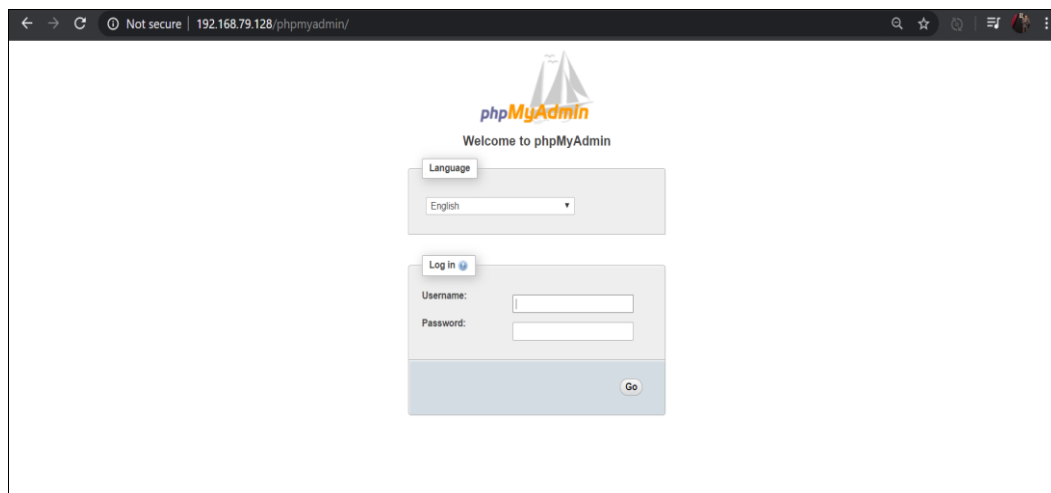
Gambar 2.6 Hasil instalasi apache dan pengaturan firewall

Tahap implementasi **Modul 2.3** mengenai instalasi PHP pada OS Ubuntu VMware Workstation apabila berhasil dalam proses menjalankan perintah yang ada, dapat dilihat pada **Gambar 2.7** berikut ini:



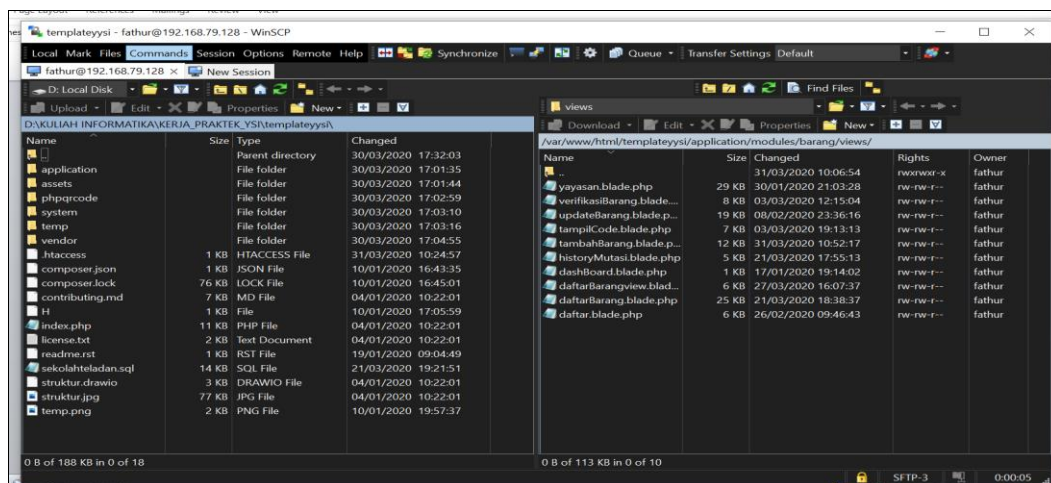
Gambar 2.7 Hasil instalasi PHP

Tahap implementasi **Modul 2.4** mengenai instalasi phpmyadmin pada OS Ubuntu VMware Workstation apabila berhasil dalam proses menjalankan perintah yang ada, dapat dilihat pada **Gambar 2.7** berikut ini:



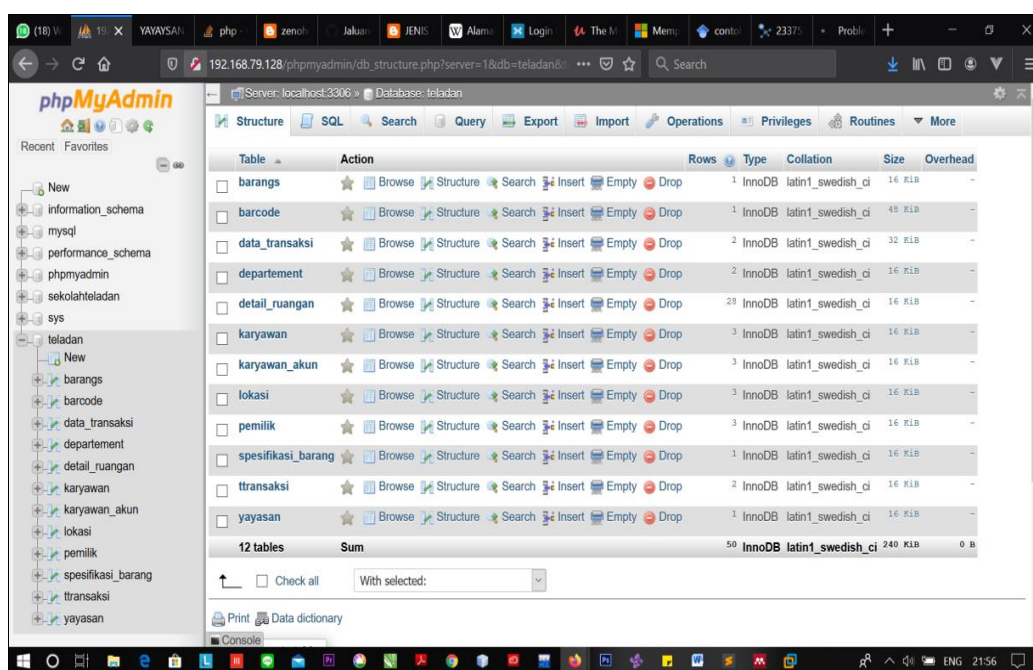
Gambar 2.7 Hasil instalasi PHPMysqladmin

Setelah dilakukan penginstalan apache, mysql, php, serta PHPMysqladmin, maka tahap berikutnya yaitu memindahkan file source code proyek ke web server dengan bantuan aplikasi WinSCP. Untuk konfigurasi WinSCP membutuhkan IP address virtualisasi Ubuntu, username dan password yang telah dibuat. Lalu source code file proyek yang akan dihosting di pindah ke folder var/www/html. Seperti **Gambar 2.8** dibawah ini:



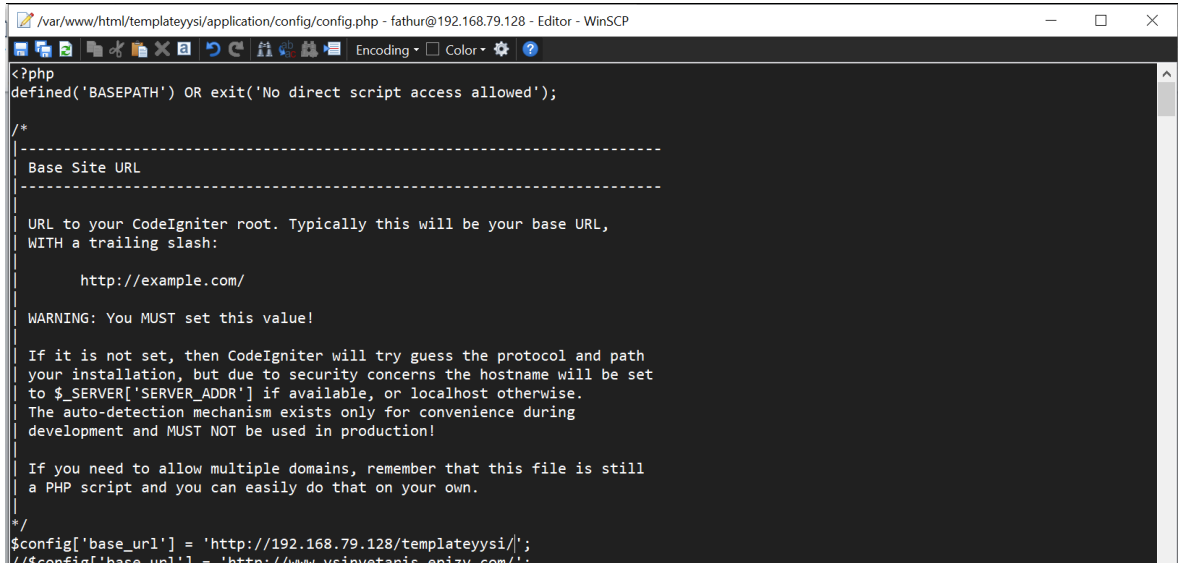
Gambar 2.8 Pemindahan File local ke remote menggunakan WinSCP

Setelah dilakukan pemindahan file proyek dari file local ke remote maka tahap selanjutnya dilakukan import database pada DBMS MySQL pada phpmyadmin LAMPP, dengan cara menuliskan *IP Address virtualisas/phpmyadmin* nya pada web browser chrome maupun mozilla . tahap tersebut dapat dilihat pada **Gambar 2.9** dibawah ini,



Gambar 2.9 Import database

Setelah import dilakukan apabila anda menggunakan *CodeIgniter* untuk membangun proyek anda jangan lupa untuk mengatur baser url nya dengan tahap buka *application->config->config.php* dan carilah base url. Untuk penulisan base url dituliskan sesuai dengan ip address virtualisasi. Tahap tersebut dapat dilihat pada **Gambar 2.10** dibawah ini :



```

<?php
defined('BASEPATH') OR exit('No direct script access allowed');

/*
-----
Base Site URL
-----

URL to your CodeIgniter root. Typically this will be your base URL,
WITH a trailing slash:

    http://example.com/

WARNING: You MUST set this value!

If it is not set, then CodeIgniter will try guess the protocol and path
your installation, but due to security concerns the hostname will be set
to $_SERVER['SERVER_ADDR'] if available, or localhost otherwise.
The auto-detection mechanism exists only for convenience during
development and MUST NOT be used in production!

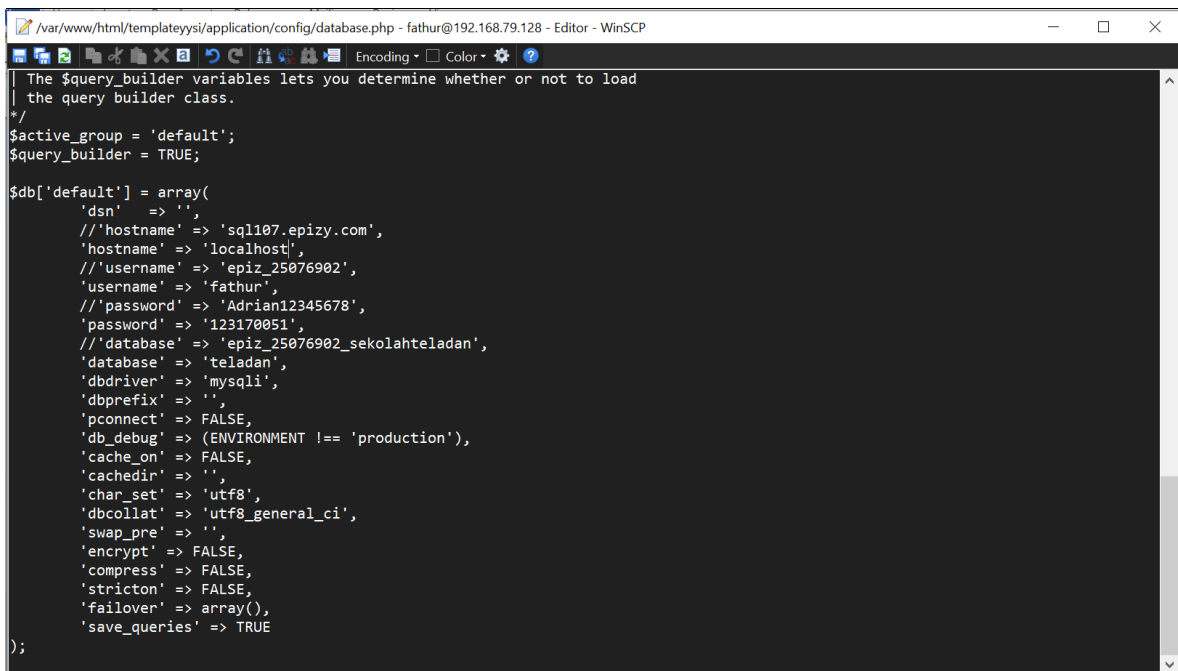
If you need to allow multiple domains, remember that this file is still
a PHP script and you can easily do that on your own.
*/

$config['base_url'] = 'http://192.168.79.128/templateyysi/';
// $config['base_url'] = 'http://www.usinvetanis.epizy.com/';

```

Gambar 2.10 Setting base url pada proyek pertama pada framework CI

Setelah langkah diatas pastikan konfigurasi database pada framework CI di sesuaikan dengan nama database yang telah dimport sebelumnya. Proses tersebut bisa dilihat pada **Gambar 2.11** dibawah ini:



```

The $query_builder variable lets you determine whether or not to load
the query builder class.
*/
$active_group = 'default';
$query_builder = TRUE;

$db['default'] = array(
    'dsn' => '',
    // 'hostname' => 'sql107.epizy.com',
    'hostname' => 'localhost',
    // 'username' => 'epiz_25076902',
    'username' => 'fathur',
    // 'password' => 'Adrian12345678',
    'password' => '123170051',
    // 'database' => 'epiz_25076902_sekolahteladan',
    'database' => 'teladan',
    'dbdriver' => 'mysqli',
    'dbprefix' => '',
    'pconnect' => FALSE,
    'db_debug' => (ENVIRONMENT !== 'production'),
    'cache_on' => FALSE,
    'cachedir' => '',
    'char_set' => 'utf8',
    'dbcollat' => 'utf8_general_ci',
    'swap_pre' => '',
    'encrypt' => FALSE,
    'compress' => FALSE,
    'stricton' => FALSE,
    'failover' => array(),
    'save_queries' => TRUE
);

```

Gambar 2.11 Setting database pada file database.php pada framework CI

Untuk mengecek apakah proyek telah terhosting pada LAMPP dilakukan tahap yaitu menuliskan *IP Address virtualisasi/namafolderproyek* pada web browser, contoh penulisan *192.168.79.128/templateyysi/*.

2.4.2 Tahap Implementasi Proyek Kedua

Tahap implementasi **Modul 2.5** tentang penginstalan docker, apabila perintah pada modul tersebut dilakukan secara runtut dan benar maka , bisa dilakukan pengecekan version dockernya ditunjukkan pada **Gambar 2.12** dibawah ini :

```
0 updates are security updates.

fathur@fathur-server:~$ sudo docker --version
[sudo] password for fathur:
Docker version 19.03.8, build afacb8b7f0
fathur@fathur-server:~$ _
```

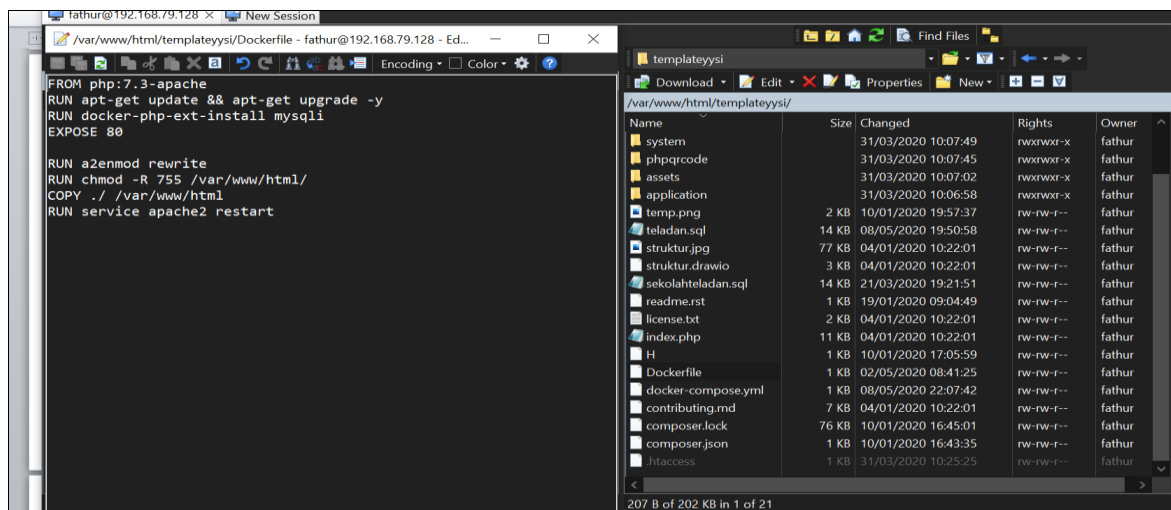
Gambar 2.12 Pengecekan versi docker yang siap digunakan

Setelah dilakukan penginstalan docker, tahap berikutnya yaitu penginstalan docker compose untuk perintah penginstalannya dapat dilihat pada **Modul 2.6**, apabila perintah pada modul tersebut telah diikuti dengan benar maka akan muncul, seperti **Gambar 2.13** di bawah ini :

```
fathur@fathur-server:~$ sudo docker-compose --version
docker-compose version 1.25.5, build 8a1c60f6
fathur@fathur-server:~$ _
```

Gambar 2.13 Pengecekan versi docker compose yang siap digunakan

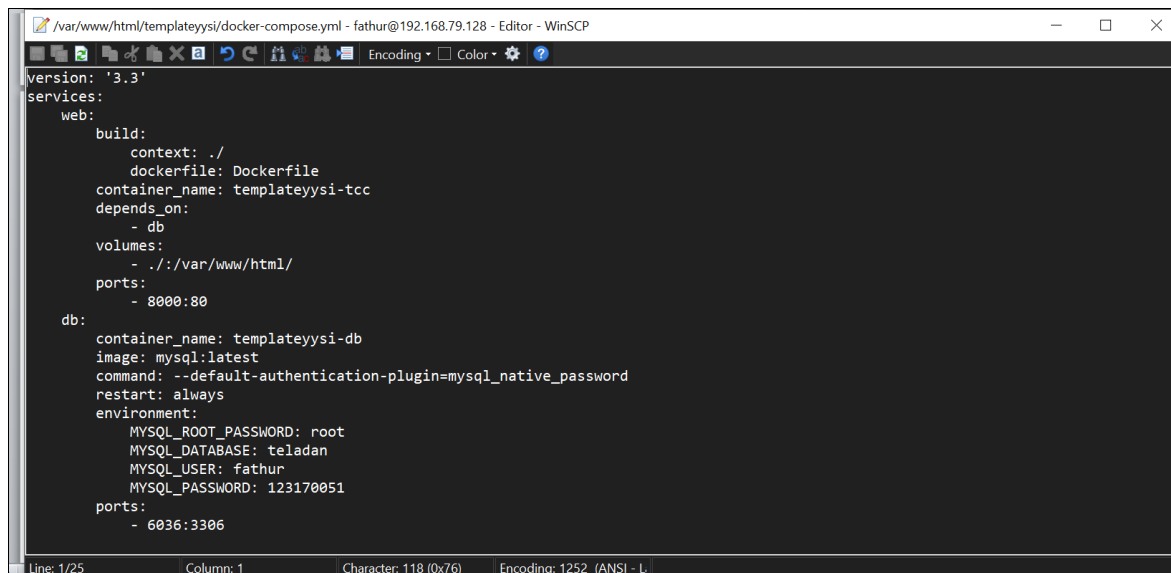
Apabila tahap penginstalan docker dan docker compose sudah selesai, maka tahap berikutnya yaitu membuat docker file dimana letak docker filenya berada di dalam satu project coding Sistem inventaris Barang Sekolah Teladan.dan untuk isi docker filenya berisi source yang dibutuhkan dalam proyek dapat di lihat dan konfigurasi ini dilakukan di WinSCP seperti **Gambar 2.14** di bawah ini:



Gambar 2.14 Isi docker file dan source yang dibutuhkan

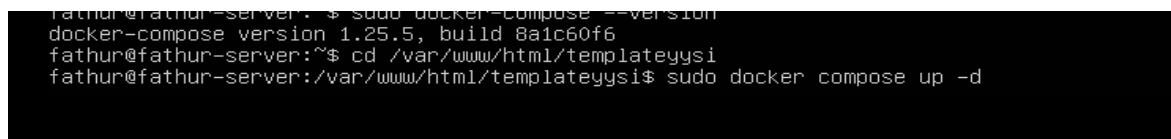
Setelah docker file dibuat, langkah berikutnya yaitu membuat file docker-compose.yml di WinSCP yang isinya adalah service apa saja yang akan dijalankan oleh

docker nantinya secara bersamaan, pada proyek ini *service* akan dijalankan secara bersamaan yaitu *service* web dan *service* db. Kedua *service* memiliki container-cotainer berisi image yang dibutuhkan untuk menjalankan proyek kedua ini, proses tersebut dapat dilihat pada **Gambar 2.15** di bawah ini :



Gambar 2.15 Isi docker-compose dan *service* yang dibutuhkan

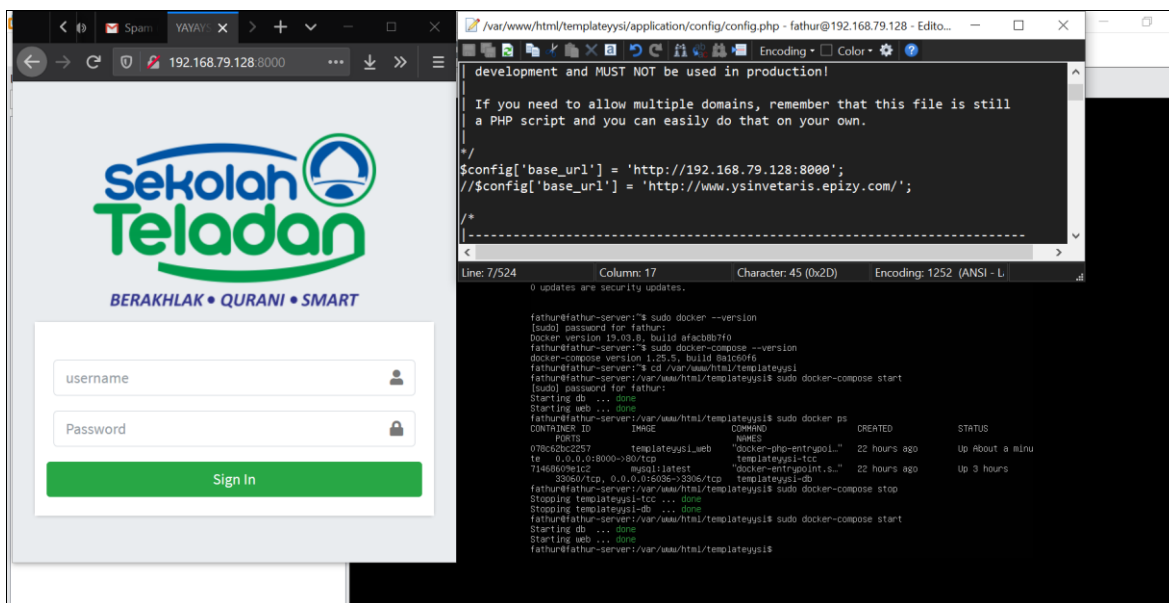
Setelah dilakukan pembuatan docker-compose.yml langkah selanjutnya yaitu tahap *build up* docker composenya untuk proses *build up* command line harus diarahkan ke folder proyek kedua yang terdapat file dockerfile dan docker-compose, berhubung saya meletakkan kedua file tersebut di /var/www/html/templateyysi maka command prompt VMware harus diarahkan ke folder tersebut dengan cara `cd /var/www/html/templateyysi`. Setelah itu lakukan perintah `sudo docker-compose up -d` untuk membangun service service yang ada di file docker-compose.yml agar berjalan secara bersamaan. Proses tersebut dapat dilihat pada **Gambar 2.16** seperti bawah ini:



Gambar 2.16 Proses *build up* docker-compose.yml

Setelah *build up* docker compose telah usai maka langkah berikutnya yaitu *migrate* database atau disebut juga *import* database kedalam service dengan container templateyysi-db. Untuk penulisan perintah command line nya dapat dilihat pada **Modul 2.8**. Apabila *migrate* database berhasil maka tahap berikutnya, yaitu mengecek apakah proyek kedua dapat diakses atau tidak, dengan menuliskan url di web browser. Dan jangan lupa untuk

mengganti Base URL pada konfigurasi pada *Code Igniternya*. Contoh penulisannya dapat dilihat pada **Gambar 2.17** seperti dibawah ini:

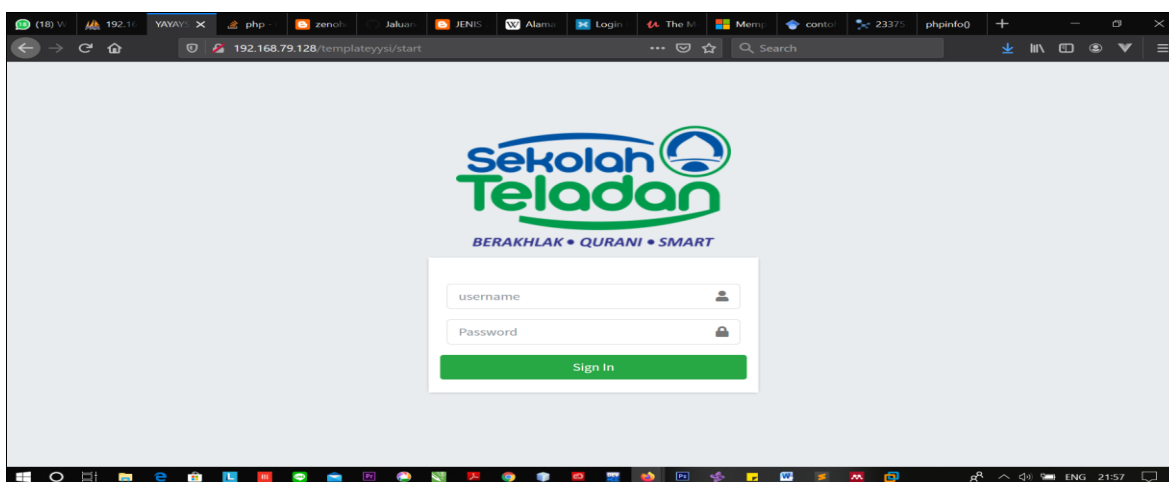


Gambar 2.17 Akses proyek kedua dan penulisan base url yang disesuaikan.

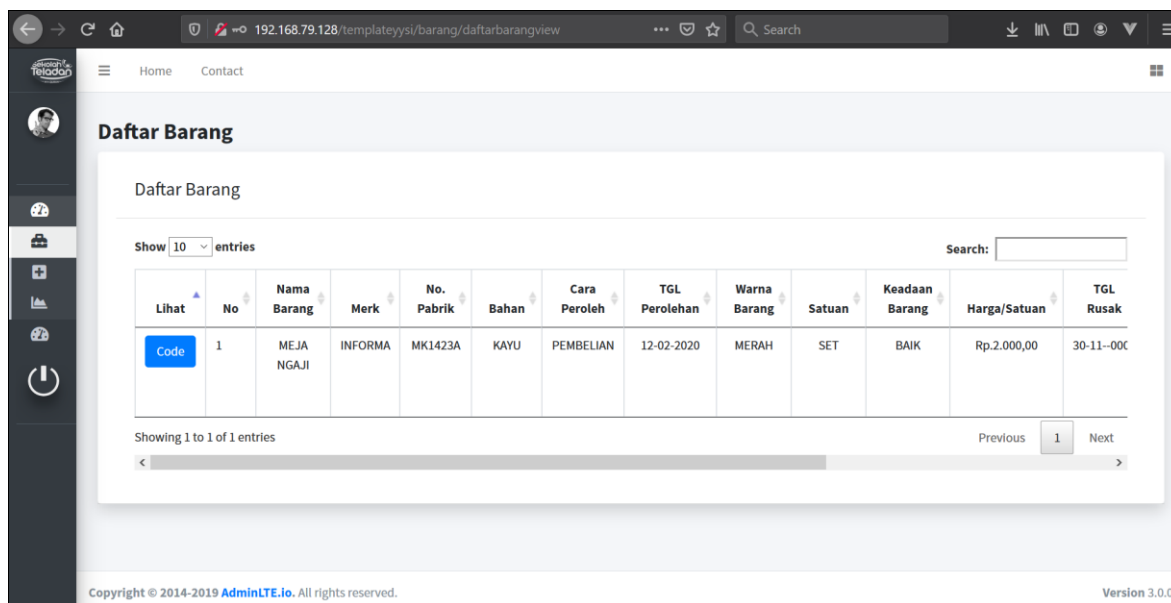
Apabila tahap web telah berjalan dengan semestinya maka proses implementasi docker pada Sistem Inventaris Barang Sekolah Teladan telah berhasil.

2.5 Hasil Implementasi

Pada proyek akhir dengan judul Sistem Inventaris Barang Sekolah Teladan menggunakan LAMPP dan Implementasinya menggunakan docker. Untuk Sistem Informasi Inventaris Barang Sekolah Teladan Menggunakan LAMPP sudah dapat di implementasikan. Hasil proyek tersebut dapat dilihat pada **Gambar 2.18** dan **Gambar 2.19** seperti dibawah ini:

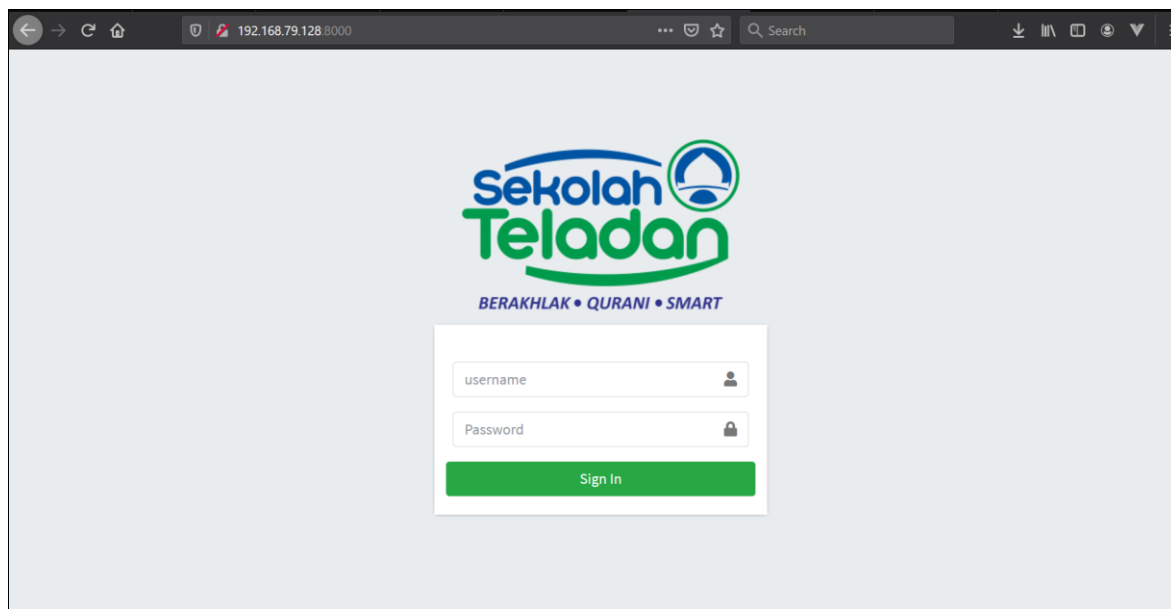


Gambar 2.18 Hasil dari hosting file local ke LAMPP menggunakan virtualisasi OS Ubuntu

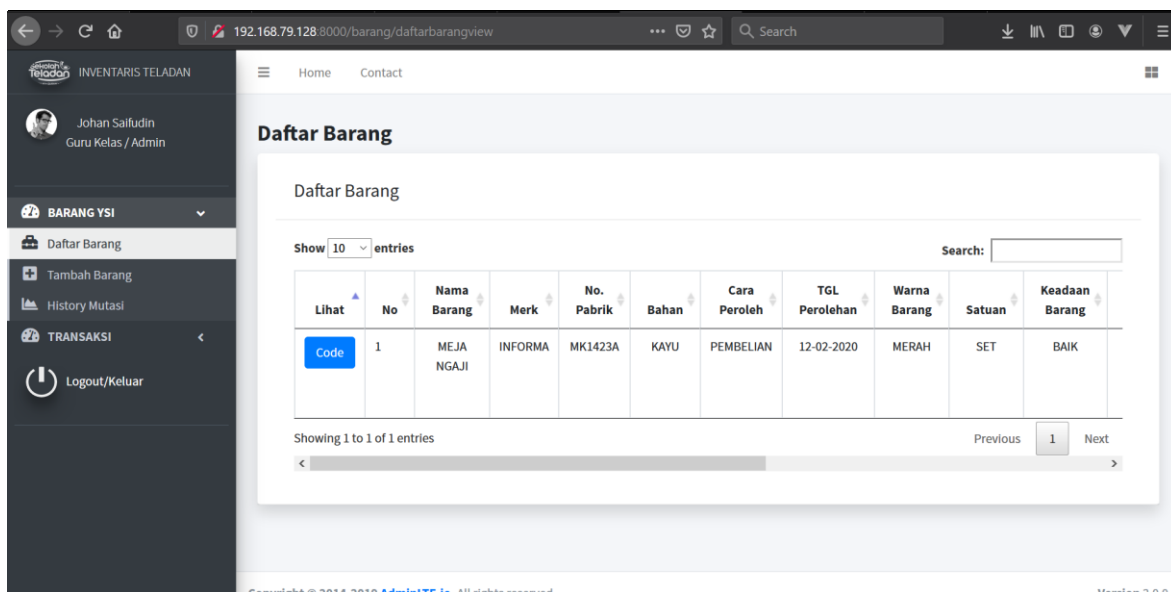


Gambar 2.19 Hasil dari hosting file local ke LAMPP menggunakan virtualisasi OS Ubuntu dan merupakan menu utama proyek pertama.

Untuk membedakan keberhasilan antara proyek pertama dan kedua dapat dilihat dari bar url pada web browser. Untuk Implementasi docker pada Sistem Informasi Inventaris Sekolah Teladan dapat dilihat pada **Gambar 2.20** dan **Gambar 2.21** dibawah ini :



Gambar 2.19 Hasil implementasi docker menggunakan virtualisasi OS Ubuntu dan merupakan menu login proyek kedua



Gambar 2.19 Hasil implementasi docker menggunakan virtualisasi OS Ubuntu dan merupakan menu utama proyek kedua

2.6 Pengujian Singkat

Pada proyek akhir terdapat masalah yang sudah dapat terpecahkan karena pengembangan aplikasi web sistem informasi inventaris barang sekolah teladan ini diharapkan dapat dikembangkan maka untuk mempermudah pengembangan tersebut diperlukan adanya membangun web inventaris tersebut menggunakan teknologi docker. Selain itu pula manfaat membangun sistem menggunakan docker yaitu dapat mengurangi beban kerja dari web server, sehingga nantinya dalam pencarian barang ddalam sistem ini dapat berjalan secara optimal. Selain itu, untuk masalah agar sistem informasi inventaris ini dapat diakses oleh orang yang mendapat hak akses dimudahkan dengan menggunakan implementasi hosting menggunakan LAMPP, sehingga transparansi dari data inventaris dapat terwujud. Kedua masalah tersebut telah diselesaikan dalam pengerjaan proyek akhir ini.

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	
		1	2	3	4	1	2	3	4	1	2
1.	Analisa Persoalan										
2.	Pembagian Tugas										
3.	Pengerjaan Proyek Pertama										
4.	Pengerjaan Laporan Proyek Pertama										
5.	Submisi Laporan dan Proyek Pertama										
6.	Revisi Laporan Proyek Pertama										
7.	Pengerjaan Proyek Kedua										
8.	Submisi Laporan dan Proyek Kedua										
9.	Revisi Laporan Proyek Kedua dan Melanjutkan Pengerjaan Proyek Akhir										
9.	Submisi 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.	Pembahasan Pokok Permasalahan	Fathur
2.	Perancangan Arsitektur Rancangan Proyek	Fathur
3.	Pengerjaan Proyek Pertama	Fathur
4.	Pengerjaan Laporan Proyek Pertama	Fathur
4.	Pengerjaan Laporan Proyek Pertama (bagian manfaat)	Azra
5.	Pengerjaan Proyek Kedua	Fathur
6.	Pengerjaan Laporan Proyek Kedua	Fathur
7.	Pengerjaan Proyek Akhir	Fathur
8.	Pengerjaan Laporan Proyek Akhir	Fathur

BAB IV

KESIMPULAN DAN SARAN

4.1 Kesimpulan

Berdasarkan pengerjaan proyek akhir ini dengan judul Sistem Informasi Inventaris Barang Sekolah Teladan dapat dikatakan berhasil. Masalah dalam pengerjaan proyek akhir ini dapat diselesaikan dengan berhasil hal tersebut dapat terlihat pada sub bab **2.5 Hasil Implementasi**. Keberhasilan dari proyek akhir ini dapat mendapat manfaat bahwa Sistem Informasi Inventaris Barang ini dapat diimplementasikan kedalam docker dimana hal tersebut dapat memudahkan pembangun proyek aplikasi dalam membangun sistem.

Pembagian kelompok dalam proyek ini menurut kami kurang efektif dikarenakan untuk teknologi docker ataupun virtualisasi machine ini perlu dipahami oleh setiap anggota kelompok. Apabila pembagian tugas tidak merata karena kondisi *work from home* maka dapat membebankan pekerjaan salah satu partner kelompok.

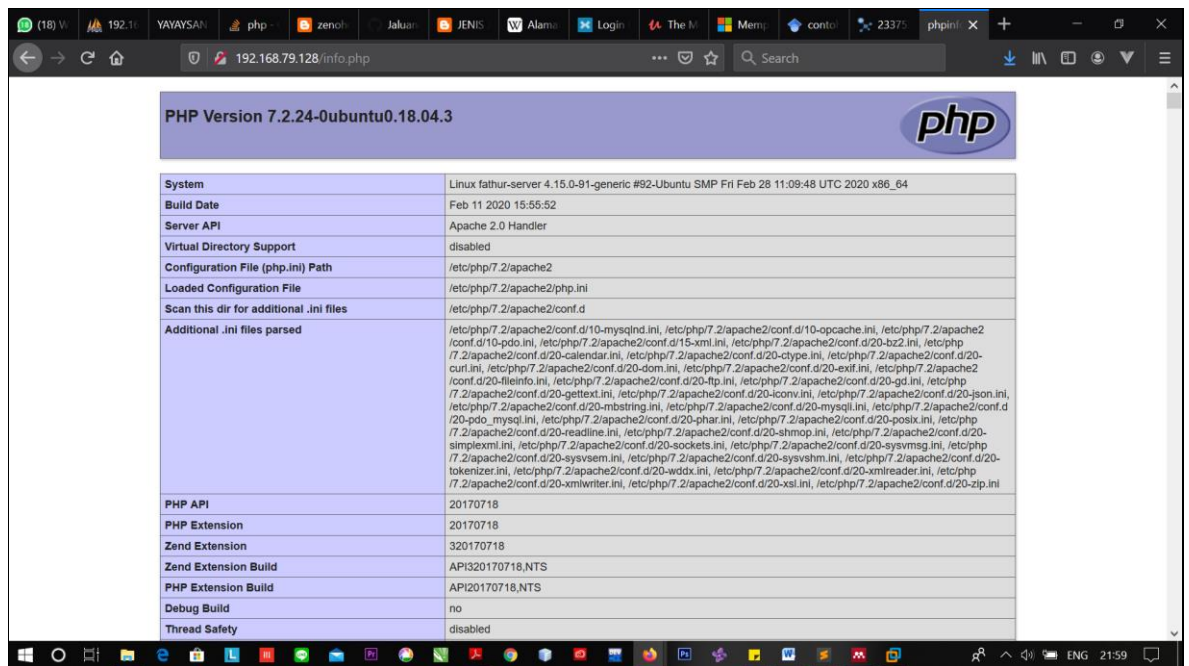
4.2 Saran

Terdapat saran dalam pengerjaan proyek ini yaitu tentang pembagian kelompok, seharusnya untuk proyek seperti ini apalagi teknologi docker yang sedang populer ini harusnya setiap anggota kelompok harus menguasai karena teknologi docker ini sangat bagus digunakan. Apabila pengerjaan ini dibuat kelompok maka cenderung ada salah satu partner yang kurang mengerti terhadap teknologi docker tersebut karena beban pengerjaan berada pada salah satu partner kelompok.

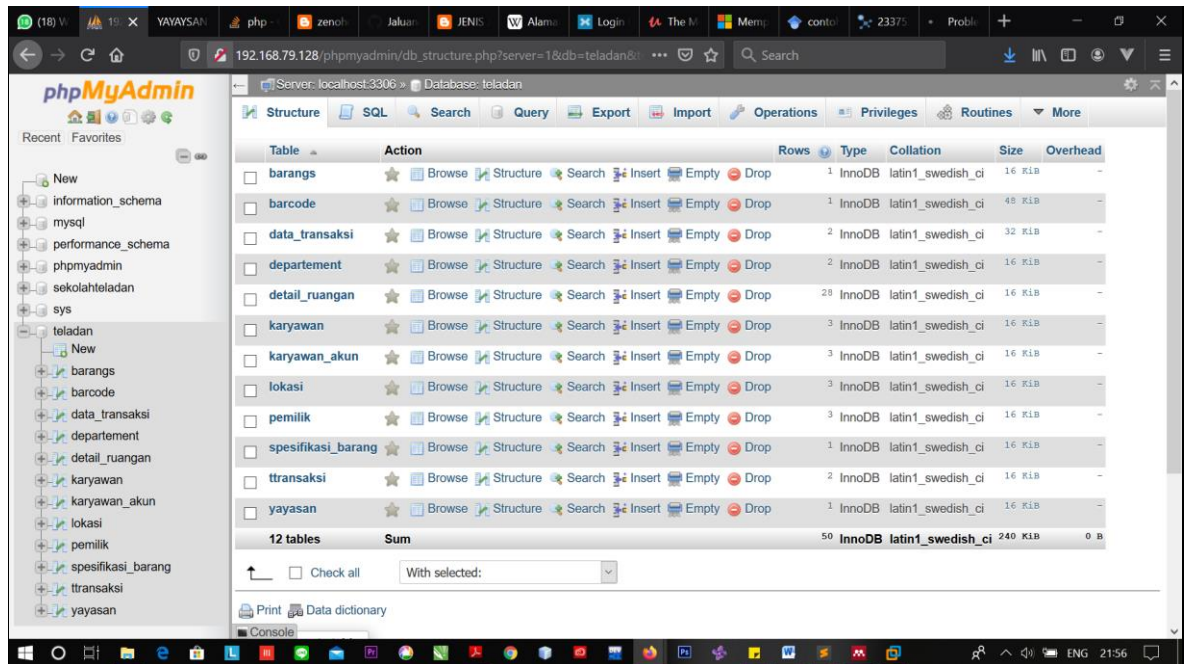
DAFTAR PUSTAKA

- [1] Adiputra, F. (2015). Container dan Docker: Teknik Virtualisasi dalam Pengelolaan Banyak Aplikasi Web. *Jurnal SimanteC*, 4(3).
- [2] Sulistyowati, L., Sulisty, W., & Bayu, T. I. (2012). Implementasi Cloud Computing sebagai Infrastructure as a Service untuk Penyediaan Web Server. *Teknologi Informasi-Aiti*, 9(2), 185–201.
[http://repository.uksw.edu/jspui/bitstream/123456789/2840/2/ART_Luchi S, Wiwin S, Teguh IB_Implementasi Cloud Computing_Full text.pdf](http://repository.uksw.edu/jspui/bitstream/123456789/2840/2/ART_Luchi%20S,%20Wiwin%20S,%20Teguh%20IB_Implementasi%20Cloud%20Computing_Full%20text.pdf)
- [3] Wildana, F. (2018). Implementasi Cloud Computing Di Beberapa Instansi Pemerintahan. *Masyarakat Telematika Dan Informasi : Jurnal Penelitian Teknologi Informasi Dan Komunikasi*, 8(2), 97. <https://doi.org/10.17933/mti.v8i2.105>

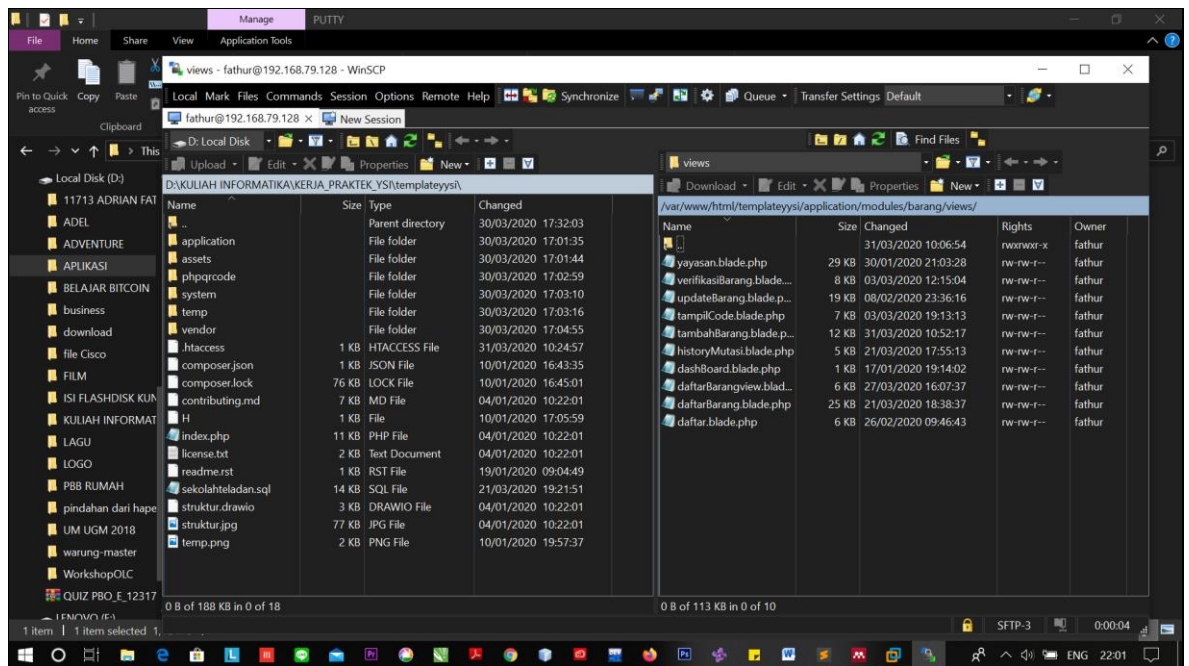
LAMPIRAN



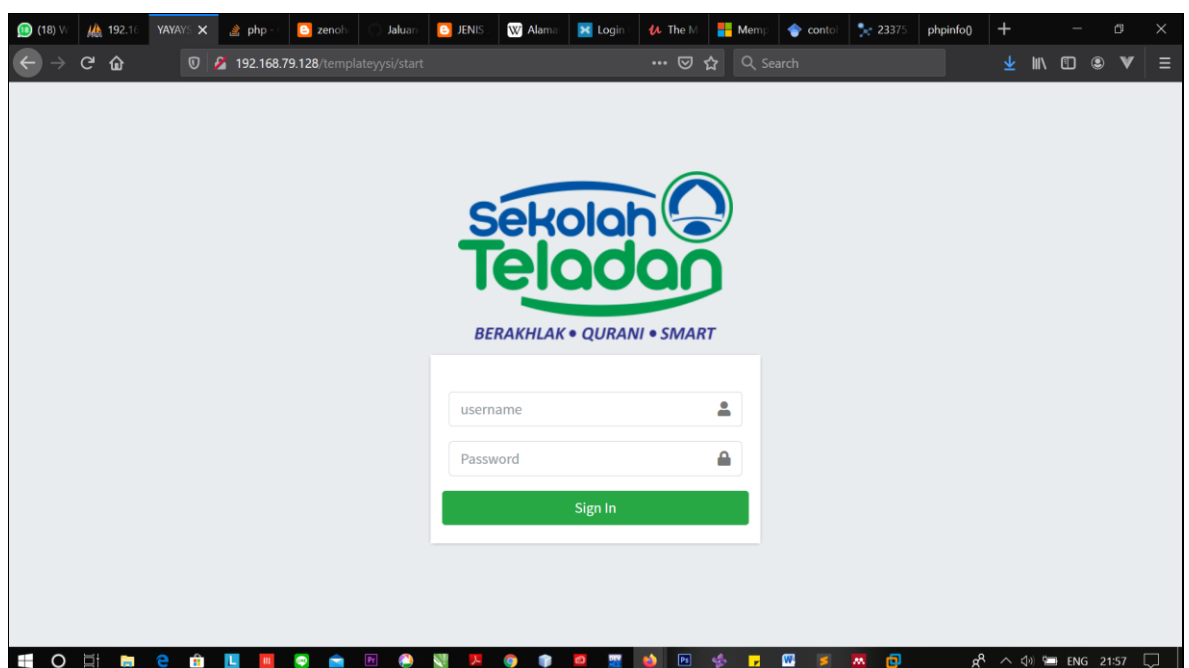
Gambar 1.1 Instalasi Apache2



Gambar 1.2 Instalasi phpmyadmin dan import database



Gambar 1.3 Aplikasi transfer file WinSCP



Gambar 1.4 Hasil dari hosting ke LAMPP

The screenshot displays the 'Daftar Barang' (Inventory List) page of the 'INVENTARIS TELADAN' system. The page features a sidebar on the left with navigation links: 'Daftar Barang', 'Tambah Barang', 'History Mutasi', 'TRANSAKSI', and 'Logout/Keluar'. The main content area shows a table with one entry. The table has columns for 'Lihat', 'No', 'Nama Barang', 'Merk', 'No. Pabrik', 'Bahan', 'Cara Peroleh', 'TGL Perolehan', 'Warna Barang', 'Satuan', and 'Keadaan Barang'. The entry shows a wooden table (MEJA KAYU) with ID 1, purchased on 12-02-2020, in good condition (BAIK). The footer includes copyright information for AdminLTE.io and version 3.0.0.

Lihat	No	Nama Barang	Merk	No. Pabrik	Bahan	Cara Peroleh	TGL Perolehan	Warna Barang	Satuan	Keadaan Barang
Code	1	MEJA KAYU	INFORMA	MK1423A	KAYU	PEMBELIAN	12-02-2020	MERAH	SET	BAIK

Showing 1 to 1 of 1 entries

Previous 1 Next

Copyright © 2014-2019 AdminLTE.io. All rights reserved. Version 3.0.0

Gambar 1.5 Tampilan utama sistem inventaris sekolah teladan

