

**SISTEM MANAJEMEN LAYANAN WEB BERBASIS PLATFORM AS A
SERVICE (PAAS) DENGAN API OPENSTACK**

KOMPETENSI JARINGAN

SKRIPSI



IDA BAGUS RATHU EKA SURYA WIBAWA

NIM. 1308605045

**JURUSAN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS UDAYANA
BUKIT JIMBARAN**

2018

LEMBAR PENGESAHAN TUGAS AKHIR

Judul : Sistem Manajemen Layanan Web Berbasis Platform as a
Service (PaaS) dengan API Openstack
Nama : Ida Bagus Rathu Eka Surya Wibawa
NIM : 1308605045
Tanggal Seminar : 6 Oktober 2018

Disetujui oleh:

Pembimbing I

Penguji I

I Dewa Made Bayu Atmaja Darmawan, S.Kom., M.Cs.
NIP. 198012062006041003

I Gede Santi Astawa, S.T., M.Cs.
NIP. 198012062006041003

Pembimbing II

Penguji II

I Komang Ari Mogi, S.Kom., M.Kom.
NIP. 198409242008011007

Gst. Ayu Vida Mastrika Giri, S.Kom., M.Cs.
NIP. 1990060620160322001

Penguji III

Made Agung Raharja, S.Si., M.Cs
NIK. 1985091920130122003

Mengetahui,
Ketua Jurusan Ilmu Komputer
FMIPA Universitas Udayana
Ketua,

Dr. I Ketut Gede Suhartana, S.Kom., M.Kom
NIP. 197201102008121001

SURAT PERNYATAAN KEASLIAN KARYA ILMIAH

Yang bertanda tangan di bawah ini menyatakan bahwa naskah Skripsi dengan judul:
**SISTEM MANAJEMEN LAYANAN WEB BERBASIS PLATFORM AS A
SERVICE (PAAS) DENGAN API OPENSTACK**

Nama : Ida Bagus Rathu Eka Surya Wibawa

NIM : 1308605045

Program Studi : Teknik Informatika

E-mail : eka.suryawibawa@mhs.cs.unud.ac.id

Nomor telp/HP : 083119171966

Alamat : Jalan Pratu Made Rambug Banjar Sasih No. 49 Sukawati, Gianyar,
Bali

Belum pernah dipublikasikan dalam dokumen skripsi, jurnal nasional maupun internasional atau dalam prosiding manapun, dan tidak sedang atau akan diajukan untuk publikasi di jurnal atau prosiding manapun. Apabila di kemudian hari terbukti terdapat pelanggaran kaidah – kaidah akademik pada karya ilmiah saya, maka saya bersedia menanggung sanksi-sanksi yang dijatuhkan karena kesalahan tersebut, sebagaimana diatur oleh Peraturan Menteri Pendidikan Nasional Nomor 17 Tahun 2010 tentang Pencegahan dan Penanggulangan Plagiat di Perguruan Tinggi.

Demikian Surat Pernyataan ini saya buat dengan sesungguhnya untuk dapat dipergunakan bilamana diperlukan.

Denpasar, 12 Juni 2018

Yang membuat pernyataan,

(Ida Bagus Rathu Eka Surya Wibawa)

NIM. 1308605045

Judul : Sistem Manajemen Layanan Web Berbasis Platform as a Service (PaaS) dengan API Openstack
Nama : Ida Bagus Rathu Eka Surya Wibawa
NIM : 1308605045
Pembimbing I : I Dewa Made Bayu Atmaja Darmawan, S.Kom., M.Cs.
Pembimbing II : I Komang Ari Mogi, S.Kom., M.Kom.

ABSTRAK

Cloud computing merupakan teknologi yang saat ini mulai berkembang dalam banyak aktivitas teknologi informasi. Cloud computing merupakan model komputasi yang semua sumber daya yang ada dalam layanan cloud dijalankan dengan media jaringan internet. Dengan adanya cloud computing memudahkan para pengguna dalam melakukan komputasi tanpa harus melakukan instalasi aplikasi pada komputer, pengguna hanya perlu mengaksesnya melalui internet. Cloud computing memiliki beberapa fasilitas yang dapat dipilih oleh pengguna sesuai kebutuhan pengguna seperti Infrastructure as a Service(IaaS), Platform as a Service(PaaS). Serta Software as a Service(SaaS).

Pengembangan Platform as a Service sebagai salah satu teknologi cloud computing yang dapat digunakan oleh pengembang aplikasi untuk mengembangkan aplikasi yang akan dibuat tanpa perlu menyediakan infrastruktur, database, framework aplikasi dan lain sebagainya serta bersifat dinamis. Dalam pengembangan layanan cloud yang dikelola oleh seorang sistem administrator atau sysadmin, tugas menginstalasi dan menkonfigurasi sistem pada server maupun software aplikasi dilakukan dengan otomatis dengan menggunakan platform otomatisasi sistem linux untuk efisiensi waktu dan manajemen penggunaanya dengan lebih mudah.

Kata kunci: *Cloud Computing, Platform as a Service, automation, ansible, API, Openstack*

Judul : Sistem Manajemen Layanan Web Berbasis Platform as a Service (PaaS) dengan API Openstack
Nama : Ida Bagus Rathu Eka Surya Wibawa
NIM : 1308605045
Pembimbing I : I Dewa Made Bayu Atmaja Darmawan, S.Kom., M.Cs.
Pembimbing II : I Komang Ari Mogi, S.Kom., M.Kom.

ABSTRACT

Cloud computing is a technology that is currently starting to develop in many information technology activities. Cloud computing is a computing model in which all resources in cloud services run on the internet network media. With cloud computing, it makes it easier for users to perform computations without having to install applications on computers, users only need to access them via the internet. Cloud computing has several facilities that can be selected by users according to user needs, such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS). As well as Software as a Service (SaaS).

Development of Platform as a Service as a cloud computing technology that can be used by application developers to develop applications that will be created without the need to provide infrastructure, databases, application frameworks and so on and are dynamic. In developing cloud services that are managed by a system administrator or sysadmin, the task of installing and configuring systems on servers and application software is carried out automatically using the Linux system automation platform for time efficiency and easier management of users.

Kata kunci: *Cloud Computing, Platform as a Service, automation, ansible, API, Openstack*

KATA PENGANTAR

Puji syukur peneliti panjatkan kehadapan Tuhan Yang Maha Esa, karena berkat rahmat dan karunia-Nya, Proposal Tugas Akhir dengan judul “Sistem Manajemen Layanan Web Berbasis Platform as a Service (PaaS) dengan API Openstack” dapat diselesaikan dengan baik dan tepat waktu. Proposal ini diharapkan dapat menjadi pedoman dan arahan dalam melaksanakan penelitian. Peneliti mengucapkan terima kasih kepada berbagai pihak yang telah membantu dalam pembuatan proposal ini, antara lain:

1. Bapak I Dewa Made Bayu Atmaja Darmawan, S.Kom., M.Cs. sebagai pembimbing 1 yang telah mengkritisi, membimbing, dan menyempurnakan proposal tugas akhir ini.
2. Bapak I Komang Ari Mogi, S.Kom., M.Kom. sebagai pembimbing 2 yang telah membimbing, dan menyempurnakan proposa tugas akhir ini.
3. Bapak Agus Muliantara, S.Kom., M.Kom. selaku Ketua Jurusan Ilmu Komputer yang telah banyak memberikan masukan dalam proses pembuatan proposal tugas akhir ini.
4. Bapak/Ibu dosen di Jurusan Ilmu Komputer, yang telah meluangkan waktu turut memberikan saran dan masukan dalam penyempurnaan proposal tugas akhir ini.
5. Semua pihak yang telah memberi dukungan sehingga proposal ini dapat diselesaikan sesuai dengan waktu yang ditentukan.

Disadari pula bahwa sudah tentu laporan ini masih mengandung kelemahan dan kekurangan. Maka dari pada itu masukan dan saran penyempurnaan sangat diharapkan.

Bukit Jimbaran, Oktober 2017

Penyusun

(Ida Bagus Rathu Eka Surya Wibawa)

DAFTAR ISI

LEMBAR PENGESAHAN TUGAS AKHIR	ii
SURAT PERNYATAAN KEASLIAN KARYA ILMIAH	iii
ABSTRAK	iv
ABSTRACT	v
KATA PENGANTAR	vi
DAFTAR ISI	vii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
DAFTAR LAMPIRAN	xii
BAB I	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian	2
1.4 Batasan Masalah	2
1.5 Manfaat Penelitian	2
1.6 Metode Penelitian	3
1.6.1 Analisis Kebutuhan	3
1.6.2 Kerangka Kerja Penelitian	4
1.6.3 Perancangan Sistem	5
1.6.4 Flowchart Sistem	8
1. Flowchart Konfigurasi Otomatis	8
2. Flowchart Front-end Web	9
1.6.5 Rancangan ERD	10
1.6.6 Evaluasi Perancangan Sistem	11
BAB II	12
2. 1 Tinjauan Studi	12

2. 2	Cloud Computing	15
2. 3	Jenis Cloud Computing.....	15
1.	IaaS (Infrastructure as a Service)	15
2.	Platform as a Service (PaaS).....	16
3.	Software as a Service (SaaS)	16
2. 4	Openstack	17
2. 5	Sistem Administrator	17
2. 6	Python.....	18
2. 7	RESTful API	18
2. 8	Ansible.....	19
BAB III		20
3.1	Analisis Kebutuhan Sistem	20
3.2	Perancangan Sistem.....	20
3.2.1	<i>Desain Arsitektur Kerja Sistem</i>	21
3.2.2	<i>Flowchart System</i>	22
3.2.2.1	<i>Flowchart Konfigurasi Otomatis</i>	23
3.2.2.1	<i>Flowchart Konfigurasi Otomatis</i>	24
3.2.3	<i>Entity Relational Diagram</i>	26
3.2.4	Perancangan Antar Muka	26
3.2.5	Skenario Pengujian Sistem.....	26
1.	BlackBox Testing	26
2.	Performance Testing.....	27
BAB IV		29
4.1	Gambaran Umum Sistem	29
4.2	Lingkungan Implementasi	29
4.3	Implementasi Basis Data.....	29
4.4	Implementasi Sistem.....	33
4.5	Antar Muka Sistem.....	38
4.6	Implementasi kebutuhan Ansible	50
4.7	Pengujian Sistem	60

4.7.1	Black Box Testing.....	61
4.7.2	Performance Testing.....	62
BAB V.....		68
5.1	Kesimpulan.....	68
5.2	Saran	68
DAFTAR PUSTAKA		70
LAMPIRAN.....		71

DAFTAR TABEL

Tabel 6.1 Tinjauan Studi	12
Tabel 6.2 Tabel Pengujian Black Box.....	27
Tabel 8.1 Rancangan Jadwal Pelaksanaan Penelitian .	Error! Bookmark not defined.

DAFTAR GAMBAR

Gambar 3.2.2.1 Flowchart Konfigurasi Otomatis	23
Gambar 3.2.2.1.1 <i>Flowchart Back-End</i>	24
Gambar 3.2.3.1 <i>Entity Relationship Diagram</i> Sistem	26

BAB I

PENDAHULUAN

1.1 Latar Belakang

Cloud computing merupakan teknologi yang saat ini mulai berkembang dalam banyak aktivitas teknologi informasi. Cloud computing merupakan model komputasi yang semua sumber daya yang ada dalam layanan cloud dijalankan dengan media jaringan internet. Dengan adanya cloud computing memudahkan para pengguna dalam melakukan komputasi tanpa harus melakukan instalasi aplikasi pada komputer, pengguna hanya perlu mengaksesnya melalui internet. Cloud computing memiliki beberapa fasilitas yang dapat dipilih oleh pengguna sesuai kebutuhan pengguna seperti Infrastructure as a Service(IaaS) sebagai penyedia infrastruktur pada pengguna, Platform as a Service (PaaS) yang dapat digunakan oleh pengembang aplikasi untuk mengembangkan aplikasi yang akan dibuat tanpa perlu menyediakan infrastruktur, database, framework aplikasi dan lain sebagainya. Serta Software as a Service(SaaS) yang memberikan software yang siap digunakan oleh pengguna.

Dalam layanan cloud dikelola oleh seorang sistem administrator atau sysadmin yang bertugas menginstalasi dan menkonfigurasi *server*, *install* dan mengkonfigurasi *software* aplikasi, membuat dan mengelola *user*, *backup* dan *restore* file, konfigurasi keamanan server, serta memonitor keamanan jaringan agar layanan yang dibutuhkan dapat berjalan dengan baik. Permasalahan yang muncul adalah untuk melakukan hal tersebut *sysadmin* harus mengkonfirmasi kebutuhan pengguna, serta untuk melakukan instalasi dan konfigurasi pada virtual server membutuhkan waktu yang cukup lama.

Dari masalah tersebut penelitian ini diharapkan dapat mempermudah tugas – tugas dari sistem administrator dengan sistem manajemen berbasis cloud yang dibangun menggunakan API dari openstack dan flask python, sistem administrator dapat mendukung kinerja seorang sistem administrator mengelola setiap virtual server

yang dimiliki. Serta melakukan instalasi dan konfigurasi secara otomatis bila ada pengguna baru yang membutuhkan sebuah server virtual untuk website.

1.2 Rumusan Masalah

Berdasarkan latar belakang, dapat dirumuskan beberapa permasalahan dalam penelitian ini yaitu:

- a. Bagaimana sistem yang dibangun dapat mendukung pekerjaan sistem administrator dalam mengelola server cloud.
- b. Bagaimana membangun sistem yang memfasilitasi developer dalam mengembangkan sistem yang dimiliki berbasis cloud.

1.3 Tujuan Penelitian

Penelitian ini memiliki beberapa tujuan yang ingin dicapai, berdasarkan rumusan masalah yang ditemukan, yaitu:

- a. Membuat developer mengurangi beban resource dapat komputer dalam mengembangkan aplikasi yang dibuat.
- b. Mendukung sistem administrator dalam mengelola server agar kinerja sistem administrator menjadi optimal.
- c. Sistem yang dibangun dapat mempermudah pengguna dalam membangun sebuah server dan web server

1.4 Batasan Masalah

Penelitian ini memiliki beberapa batasan masalah, guna fokus untuk mencapai tujuan yang sudah dijabarkan sebelumnya, yaitu:

- a. Layanan cloud menggunakan openstack sebagai penyedia layanan cloud computing.
- b. Sistem melayani virtual server untuk membuat layanan aplikasi web.
- c. Sistem dibangun menggunakan python dengan framework flask.
- d. Sistem tidak menyediakan domain untuk layanan pengguna.

1.5 Manfaat Penelitian

Sistem yang dibuat dapat mendukung pekerjaan seorang administrator untuk mengelola server utama, serta memberikan wadah bagi para developer untuk bisa mengembangkan sistemnya secara online tanpa perlu menyiapkan server. Sehingga developer hanya fokus pada pengembangan aplikasinya.

1.6 Metode Penelitian

Bagian ini akan menjelaskan mengenai langkah – langkah yang akan dilakukan dalam merancang sistem manajemen layanan web berbasis Platform as a Service (PaaS) dengan API openstack.

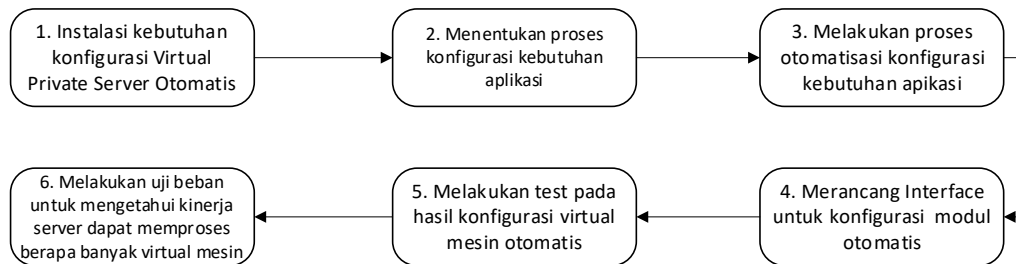
1.6.1 Analisis Kebutuhan

Pada tahap ini dilakukan analisis kebutuhan sistem, analisis kebutuhan sistem meliputi data yang digunakan, pembelajaran dari referensi yang sudah ada dan perangkat yang digunakan baik perangkat lunak maupun perangkat keras:

1. Tahapan pertama yang dilakukan dalam penelitian ini adalah mengidentifikasi permasalahan. Tahap ini merupakan tahap yang paling penting dalam penelitian karena jalannya penelitian didasarkan atas permasalahan yang terjadi. Setelah menentukan masalah yang terjadi, tahapan yang diperlukan selanjutnya adalah menentukan rumusan masalah dan tujuan yang ingin dicapai dalam penelitian. Pada penelitian ini identifikasi permasalahan dilakukan dengan menggunakan teknik observasi, dari teknik ini maka akan dapat diketahui mengenai keluhan – keluhan yang ada di lapangan.
2. Tahap kedua yang dilakukan dalam metodologi penelitian ini adalah studi literatur. Studi literatur dilakukan dengan mengambil literatur – literatur pendukung dari jurnal – jurnal ilmiah, baik jurnal dalam negeri ataupun jurnal luar negeri dan dari beberapa buku. Dalam studi literatur ini, penulis mencari sumber terkait permasalahan – permasalahan yang perlu menjadi perbaikan dalam penelitian selanjutnya.

1.6.2 Kerangka Kerja Penelitian

Bagian ini menjelaskan tentang bagaimana penelitian ini dilakukan. Berikut adalah beberapa proses penting yang dilakukan :



Gambar 1. 1 Kerangka Kerja Sistem

1. Instalasi kebutuhan konfigurasi VPS(Virtual Private Server)

Tahap pertama dalam penelitian ini dengan mempersiapkan sistem dan software yang dibutuhkan dalam mengkonfigurasi kebutuhan web server dalam menginstalasi modul-modul kebutuhan sistem seperti sistem operasi, database, framework, web server .

2. Menentukan proses konfigurasi kebutuhan aplikasi

Pada tahap ini mempersiapkan kebutuhan konfigurasi untuk membangun sebuah web server, dari web service sampai database yang diperlukan untuk menampung setiap user.

3. Melakukan proses otomatisasi konfigurasi modul aplikasi

Pada tahap ini melakukan otomatisasi dalam mengkonfigurasi sebuah web server. Dimana proses otomatisasi menggunakan API dari Ansible.

4. Merancang web front-end untuk user

Pada tahap ini bila semua konfigurasi sudah berjalan dengan baik, dibuat web yang dapat digunakan oleh pengguna mengatur kebutuhan web yang dibuat serta dapat melihat informasi dari setiap VPS yang dimiliki.

5. Melakukan test pada hasil konfigurasi

Pada tahap ini menguji hasil konfigurasi dari web interface. Untuk menguji konfigurasi berjalan dengan baik atau tidak

6. Melakukan uji beban server

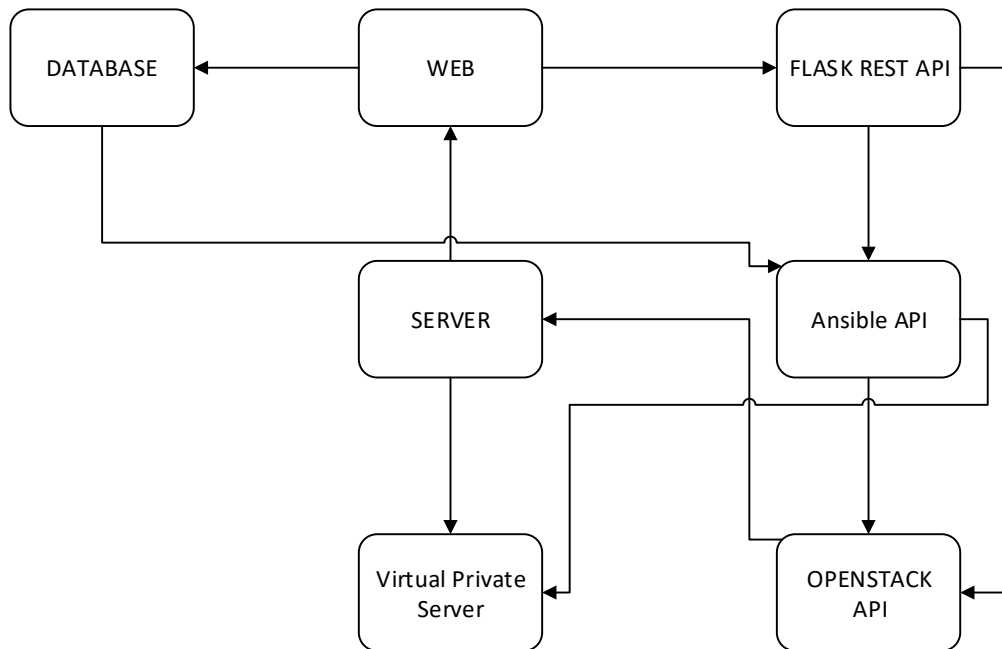
beban ini dilakukan untuk menguji tingkat kemampuan server dalam memberikan pelayanan pada pengembangan aplikasi. Bila server tidak sanggup untuk melayani pengembang baru akan menolak menyediakan layanan.

7. Melakukan Backup Server

Ketika server user sudah berjalan, sistem akan melakukan backup data pengguna untuk menjaga ketersediaan data pengguna bila terjadi sesuatu yang dapat merugikan pengguna, seperti file data hilang atau rusak, dapat dikembalikan dengan restore pada sistem dengan menggunakan waktu check poin.

1.6.3 Perancangan Sistem

Pada tahap Perancangan sistem, dilakukan berdasarkan hasil analisa kebutuhan sistem yang sudah dilakukan sebelumnya. Hal ini dilakukan agar, perancangan tidak keluar dari tujuan sistem yang dikembangkan. Berikut merupakan desain arsitektur kerja sistem:



Gambar 1. 2 Desain Kerja Sistem

Pada gambar 1.2 merupakan hubungan antara setiap perangkat lunak yang ada pada server utama. Berikut ini merupakan penjelasannya :

1. Web dan database

Database digunakan untuk menyimpan informasi pengguna serta informasi pada VPS yang dimiliki oleh setiap pengguna.

2. Web dan framework flask REST API

Flask REST API digunakan untuk menghubungkan web front-end dengan back-end. Dimana back-end dibuat dengan menggunakan framework flask dari python, selanjutnya dibuat API tersendiri agar web front-end dapat mengirim dan menerima informasi serta konfigurasi yang dilakukan yang nantinya akan diproses oleh back-end server.

3. Database dan ansible API

Hubungan database dengan ansible API akan mengambil informasi pengguna berupa username, email dan password yang nantinya akan dimasukkan pada

konfigurasi dalam VPS yang selanjutnya sebagai super amin pada VPS yang dibuat.

4. framework flask REST API dan asible API

Hubungan framework flask REST API dengan asible API. Ketika user melakukan konfigurasi pada web utama akan dikirim melalui REST API flask selanjutnya konfigurasi tersebut akan digunakan oleh API ansible untuk mengkonfigurasi VPS yang dibuat.

5. framework flask dan openstack API

Hubungan framework flask dan openstack API adalah pada flask dikonfigurasi untuk dapat terhubung dengan server devstack, dimana server openstack yang mengelola dalam membuat sebuah VPS.

6. openstack API dan server

API openstack digunakan untuk terhubung dengan server back-end yang dibuat dengan framework flask. Untuk dapat mengembangkan aplikasi openstack dari bahasa pemrograman yang berbeda.

7. ansible API dan Virtual Private Server

hubungan ansible API dengan VPS adalah ansible akan mengirimkan konfigurasi pada VPS melalui SSH yang yang dibuat. Selanjutnya pada vps akan melakukan konfigurasi yang diperlukan untuk membangun sebuah web aplikasi secara otomatis. Pengguna hanya perlu memilih keperluan yang ada menu web front-end.

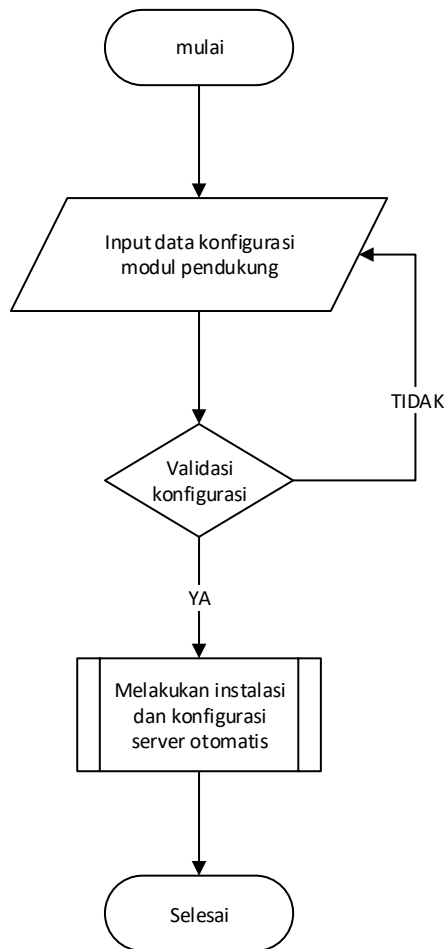
8. server dan Virtual Private Server

hubungan antara server dan VPS(Virtual Private Server) adalah server mengelola dan memberikan resource pada VPS serta mengatur konektivitas setiap VPS yang dimiliki.

1.6.4 Flowchart Sistem

Pada bagian ini akan menjelaskan proses yang dilakukan oleh perangkat lunak, bagaimana proses tersebut berjalan yang akan dijelaskan pada setiap flowchart berikut ini:

1. Flowchart Konfigurasi Otomatis

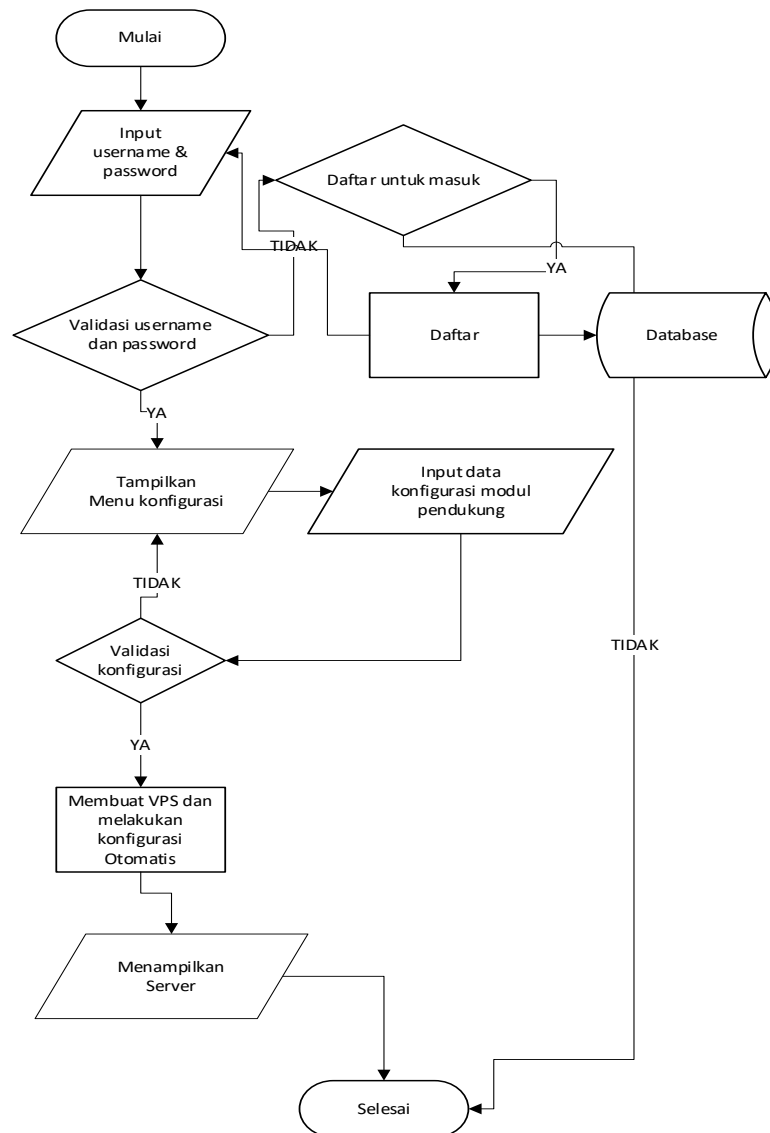


Gambar 1. 3 Flowchart Konfigurasi Otomatis

Pada desain flowchart diatas merupakan garis besar gambaran system yang akan dikerjakan, bagaimana alur platform yang dibuat dapat mengotomatitasi dalam instalasi serta konfigurasi sebuah layanan pada virtual machine. Mulai dari memasukkan aplikasi pendukung yang ingin digunakan seperti database, web service, username dan e-mail administrator layanan aplikasi. Kemudian dari hasil

inputan tersebut diolah oleh flask API yang dibuat untuk dimasukkan ke dalam konfigurasi yang terdapat ada Ansible API, selanjutnya dari ansible API dengan menggunakan SSH akan melakukan instalasi konfigurasi pada virtual server yang didapat setiap user. Bila konfigurasi telah selesai user akan menerima sebuah ip public untuk dapat mengakses layanan aplikasi.

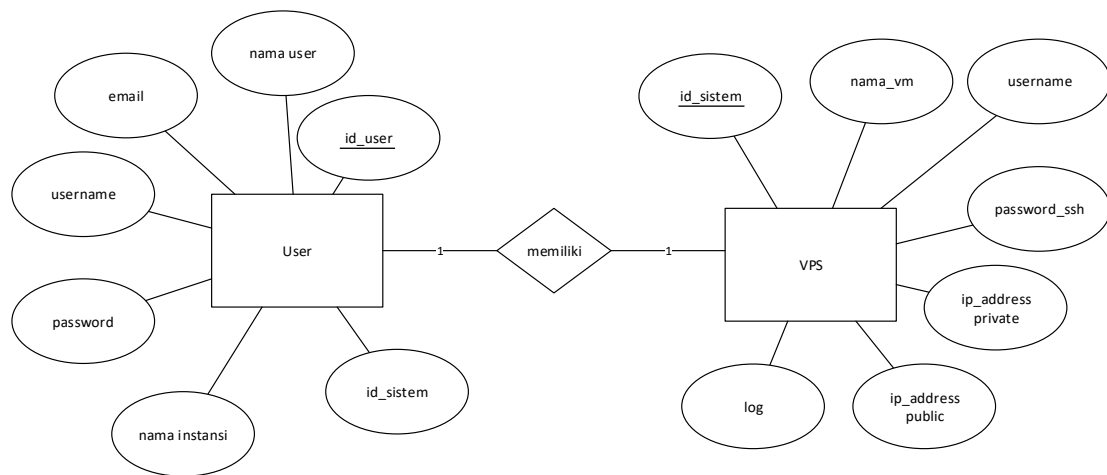
2. Flowchart Front-end Web



Gambar 1. 4 Flowchart Front-end Web

Pada desain flowchart Front-end Web merupakan alur kerja web dari sisi front-side atau pada sisi user. Setelah user melakukan registrasi pada website, selanjutnya user akan diarahkan ke menu instalasi dan konfigurasi dalam membangun layanan VPS. Menu akan dibuat secara user friendly sehingga user hanya perlu memilih menu konfigurasi system. Ketika user sudah selesai memilih menu konfigurasi, system akan memvalidasi informasi yang dimasukkan oleh user, bila sudah benar system akan mengirimkan hasil instalasi dan konfigurasi dalam sebuah data berbentuk JSON ke dalam system yang nantinya akan dikirim ke virtual sever untuk melakukan konfigurasi dan instalasi modul kebutuhan aplikasi selanjutnya sistem akan melakukan backup berkala untuk menjaga keamanan data pengguna bila terjadi satu kesalahan dapat dilakukan restore data.

1.6.5 Rancangan ERD



Gambar 1. 5 Entity Relationship Diagram Sistem

Pada desain ERD diatas menjelaskan hubungan user dengan VPS yang dimiliki user memiliki sebuah vps dengan satu ip public untuk dapat mengakses server e-learning. Serta pada vps akan menyimpan log keadaan vps mulai dari kinerja dan resource yang dimiliki.

1.6.6 Evaluasi Perancangan Sistem

Pada tahap ini dilakukan evaluasi terhadap desain perancangan sistem. Bila sistem telah sesuai dengan kebutuhan awal yang didefinisikan akan dilanjutkan ke tahap implementasi. Namun apabila desain sistem belum memenuhi kebutuhan awal yang didefinisikan, maka akan dilakukan perancangan ulang desain sistem.

BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Studi

Pada penelitian ini, peneliti menggunakan beberapa penelitian yang pernah dilakukan mengenai sistem layanan PaaS dan metode adaptive threshold. Beberapa penelitian tersebut yaitu:

Tabel 2. 1 Tinjauan Studi

No	Judul	Tahun	Penulis
1.	A Dynamic Self-adaptive Resource-Load Evaluation Method in Cloud Computing	2015	Liyun Zuo, Lei Shu, Shoubin Dong, Zhangbing Zhou, Lei Wang
2.	Adaptive Resource Management for Service Workflows in Cloud Environments	2013	Yi Wei dan M. Brian Blake dan Iman Saleh
3.	A Heuristic Adaptive Threshold Algorithm on IaaS Clouds	2015	Qingxin Xia dan Yuqing Lan, Limin Xiao
4.	The Design and Implementation of Resource Monitoring for Cloud Computing Service Platform	2013	Lei Xiaojiang, Shang Yanlei
5.	An Adaptive Threshold Method to Address Routing Issues in Delay-Tolerant Networks	2011	Nicole Ng, Hwa Chang, Zhongjian Zou, Sai Tang

Dalam jurnal A Dynamic Self-adaptive Resource-Load Evaluation Method in Cloud Computing memaparkan Sumber daya cloud membuat indikator evaluasi statis tradisional tidak dapat menggambarkan keadaan sumber daya secara akurat, sehingga pada penelitian ini menggunakan tiga indikator dinamis yaitu jumlah permintaan

sumber daya, kapasitas komputasi sumber daya dan kekuatan beban sumber daya untuk mengevaluasi keadaan beban sumber daya. Algoritma self-adaptive digunakan untuk mengevaluasi beban sumber daya. Dengan algoritma tersebut dapat membagi status load resource menjadi overload, normal dan idle. Pada sumber daya yang mengalami overload akan dimigrasi pada load balance, dan idle resource akan dilepas untuk menghemat energi bila waktu idle melebihi batas threshold. Sehingga pada penelitian ini mengusulkan dua threshold adaptif yang dipengaruhi oleh indikator evaluasi secara dinamis. Pada percobaan menampilkan SDWM dengan keuntungan yang sangat baik dari waktu respond dan pemanfaatan sumber daya ketika sumber daya secara otomatis masuk dan keluar (Liyun Zuo, 2015).

Dalam jurnal Adaptive Resource Management for Service Workflows in Cloud Environments menjelaskan Manajemen sumber daya merupakan bagian yang sangat penting pada alur kerja manajemen aktivitas di dalam cloud. Dengan algoritma adaptif Resources Management untuk layanan alokasi dan dealokasi secara dinamis pada alur kerja sumberdaya sebelum mereka mengeksekusi pada sistem cloud. Tujuan dari jurnal ini, algoritma dapat menjaga jumlah total sumber daya yang dimiliki oleh masing-masing layanan pada tingkat yang diinginkan sehingga sumber daya tersebut tidak terlalu banyak melakukan komunikasi dan tidak memanfaatkan banyak sumberdaya. Hasil prediksi adaptif digunakan untuk memandu algoritma dalam memilih kandidat sumber daya yang akan di lepas atau meminta sumber daya baru untuk dialokasikan. Dengan menjalankan simulasi pada data beban kerja secara sintetis. Algoritma tersebut melakukannya dengan baik daripada reaktif sumber daya Management yang ada dan dapat mencapai fluktuasi beban layanan yang lebih rendah dan kehilangan sumber daya (Saleh, 2013).

Dalam jurnal A Heuristic Adaptive Threshold Algorithm on IaaS Clouds memaparkan teknologi untuk menghemat energi IaaS telah banyak menarik perhatian, namun bagi penyedia layanan cloud IaaS menjamin penghematan energi dan kinerja di bawah kondisi Service Level Agreement (SLA). Pada jurnal ini menggunakan metode berbasis adaptif threshold untuk mengurangi tradeoff antara penghematan energi dan

SLA. Dar dapat menemukan ambang batas secara optimal, penulis mengusulkan sebuah framework yang basisnya pada Ben kerja, yang mengintegrasikan secara mulus metode prediksi berbasis pilihan dan model untuk mengukur hubungan antara biaya migrasi mesin virtual (VM) dan listrik saat mesin fisik mati. Algoritma threshold dirancang untuk menangkap ambang batas secara efektif. Dengan metode ini dapat memperbaiki konsumsi energi sebesar 10-20 persen. Algoritma threshold adaptif lebih efisien dan mampu beradaptasi dengan baik terhadap beban kerja dengan variabel tinggi (Xia, Lan, & Xiao, 2015).

Dalam jurnal “The Design and Implementation of Resource Monitoring for Cloud Computing Service Platform”. Membahas dengan cloud computing pengguna dapat menemukan solusi baru yang efektif untuk membangun sistem dan aplikasi berdasarkan kebutuhan mereka. karena cloud computing menghasilkan ketersediaan dan skalabilitas yang tinggi. pada bagian background tim tersebut mengembangkan subsistem yang digunakan untuk memonitor sumber daya, dengan bantuan dari platform yang dibuat, pada layer atas berjalan dengan lancar (Lei Xiaojiang, 2013).

Pada jurnal An Adaptive Threshold Method to Address Routing Issues in Delay-Tolerant Networks memaparkan algoritma routing berbasis epidermi memberikan solusi dimana menggunakan beberapa salinan paket untuk meningkatkan probabilitas pengiriman paket sukses, namun banyak salinan paket tidak hanya meningkatkan beban jaringan tetapi juga menurunkan throughput, dari hal tersebut pada jurnal ini menggunakan metode threshold, metode threshold menggunakan pendekatan nilai statis. pada jurnal metode threshold adaptif mempengaruhi nilai threshold secara terus menerus sehingga dapat menyesuaikan dengan topologi secara dinamis. pada hasil simulasi tersebut memverifikasi bahwa nilai threshold adaptif melebihi metode threshold dengan mengurangi tingkat pengiriman paket. Pada penelitian ini menguji kinerja metode threshold adaptif melalui berbagai topologi sederhana, dengan metode threshold secara konsisten menunjukkan throughput yang tinggi dengan mengurangi tingkat pengiriman sekitar 3,9 detik. oleh karena itu metode adaptif threshold

merupakan pendekatan secara praktis untuk menangani masalah routing pada Delay-Tolerant Networks(DTN) (Nicole Ng, 2011).

2. 2 Cloud Computing

Cloud computing mengacu pada aplikasi dan service yang berjalan dalam jaringan data terdistribusi dengan menggunakan sumberdaya virtual dan internet akses protokol pada umumnya. Hal ini dibedakan pada gagasan sumber daya virtual dan detail dari mesin fisik sistem dalam software yang berjalan secara abstraksi dari user. (Sosinsky, 2011). Cloud Computing secara sederhana adalah “layanan teknologi informasi yang bisa dimanfaatkan atau diakses oleh pelanggannya melalui jaringan internet”. Komputasi awan adalah suatu konsep umum yang mencakup SaaS, Web 2.0, dan tren teknologi terbaru lain yang dikenal luas, dengan tema umum berupa ketergantungan terhadap Internet untuk memberikan kebutuhan komputasi pengguna. Sebagai contoh, Google Apps menyediakan aplikasi bisnis umum secara sharing yang diakses melalui suatu penjelajah web dengan perangkat lunak dan data yang tersimpan di server.

2. 3 Jenis Cloud Computing

Jenis-jenis dari Cloud Computing dapat dijabarkan sebagai berikut:

1. IaaS (Infrastructure as a Service)

Terletak satu level lebih rendah dibanding PaaS. Ini adalah sebuah layanan yang “menyewakan” sumberdaya teknologi informasi dasar, yang meliputi media penyimpanan, processing power, memory, sistem operasi, kapasitas jaringan dan lainlain, yang dapat digunakan oleh penyewa untuk menjalankan aplikasi yang dimilikinya. Model bisnisnya mirip dengan penyedia data center yang menyewakan ruangan untuk co-location, tapi ini lebih ke level mikronya. Penyewa tidak perlu tahu, dengan mesin apa dan bagaimana caranya penyedia layanan menyediakan layanan IaaS. Yang penting, permintaan mereka atas sumberdaya dasar teknologi informasi itu dapat dipenuhi.

2. Platform as a Service (PaaS)

Konsepnya hampir serupa dengan IaaS. Namun Platform disini adalah penggunaan operating system dan infrastruktur pendukungnya. Yang cukup terkenal adalah layanan dari situs Force.Com serta layanan dari para vendor server. Seperti namanya, PaaS adalah layanan yang menyediakan modul-modul siap pakai yang dapat digunakan untuk mengembangkan sebuah aplikasi, yang tentu saja hanya bisa berjalan diatas platform tersebut. Seperti juga layanan SaaS, pengguna PaaS tidak memiliki kendali terhadap sumber daya komputasi dasar seperti memory, media penyimpanan, processing power dan lain-lain, yang semuanya diatur oleh provider layanan ini. Pionir di area ini adalah Google AppEngine, yang menyediakan berbagai tools untuk mengembangkan aplikasi di atas platform Google, dengan menggunakan bahasa pemrograman Python dan Django. Kemudian Salesforce juga menyediakan layanan PaaS melalui Force.com, menyediakan modul-modul untuk mengembangkan aplikasi diatas platform Salesforce yang menggunakan bahasa Apex. Dan mungkin yang jarang sekali kita ketahui, bahwa Facebook juga bisa dianggap menyediakan layanan PaaS, yang memungkinkan kita untuk membuat aplikasi diatasnya. Salah satu yang berhasil menanggung untung besar dari layanan PaaS

3. Software as a Service (SaaS)

Berada satu tingkat diatas PaaS dan IaaS, dimana disini yang ditawarkan adalah software atau suatu aplikasi bisnis tertentu. Contoh yang paling mutakhir adalah Salesforce.Com, Service-Now.Com, Google Apps, dsb. SaaS ini merupakan layanan Cloud Computing yang paling dahulu populer. Software as a Service ini merupakan evolusi lebih lanjut dari konsep ASP (Application Service Provider). Sesuai namanya, SaaS memberikan kemudahan bagi pengguna untuk bisa memanfaatkan sumberdaya perangkat lunak dengan cara berlangganan. Sehingga tidak perlu mengeluarkan investasi baik untuk in house development ataupun pembelian lisensi. Dengan cara berlangganan via web, pengguna dapat langsung menggunakan berbagai fitur yang disediakan oleh penyedia layanan. Hanya saja dengan konsep SaaS ini, pelanggan tidak

memiliki kendali penuh atas aplikasi yang mereka sewa. Hanya fitur-fitur aplikasi yang telah disediakan oleh penyedia saja yang dapat disewa oleh pelanggan.

2.4 Openstack

OpenStack adalah sistem aplikasi cloud yang mengelola sumberdaya seperti komputasi, penyimpanan dan jaringan, yg tersedia pada infrastruktur fisik seperti dalam sebuah fasilitas pusat-data (data center). Admin atau pengguna dapat mengendalikan dan melakukan provisioning atas sumber-daya ini melalui dashboard / antar-muka web. Developer dapat mengakses sumber-daya tersebut melalui sejumlah API standar (Mulyana, 2017).

Openstack merupakan platform cloud-computing open source yang memungkinkan pengguna untuk membangun sebuah "IAAS" Infrastruktur sebagai service cloud yang bergerak secara massal pada komoditas hardware dan skala. Openstack mengontrol kolam besar komponen komputasi awan di seluruh datacenter, semua dikelola melalui dashboard yang menyediakan administrator kontrol penuh sambil memberikan pengguna kemampuan untuk sumber penyediaan melalui antarmuka web.

2.5 Sistem Administrator

System administrator atau sysadmin merupakan sebuah pekerjaan yang bertugas mengatur dan memelihara dan mengoperasikan sistem komputer dan jaringan komputer. Dalam perusahaan maupun sebuah organisasi sysadmin sangat dibutuhkan dalam mengelola dan mengamankan data pada server yang dimiliki perusahaan (Collings & Kurt, 2015). Adapun tugas sysadmin sebagai berikut:

1. Menginstalasi dan menkonfigurasi server.
2. Menginstall dan mengkonfigurasi software aplikasi.
3. Membuat dan mengelola user.
4. Backup dan restore file.
5. Konfigurasi keamanan server.
6. Memonitor keamanan jaringan

2.6 Python

Python merupakan bahasa pemrograman yang sangat powerful yang memiliki beberapa kesamaan dengan Fortran, salah satu bahasa pemrograman paling awal, namun lebih baik dari Fortran. Python memungkinkan anda untuk menggunakan variabel tanpa perlu mendeklarasikannya. Sehingga kita tidak dipaksa untuk mendefinisikan class dengan python.

Python dikembangkan oleh Guido van Rossum pada tahun 1990 di CWI, Amsterdam sebagai kelanjutan dari bahasa pemrograman ABC. Versi terakhir yang dikeluarkan CWI adalah 1.2. (Doty, 2008).

Tahun 1995, Guido pindah ke CNRI sambil terus melanjutkan pengembangan Python. Versi terakhir yang dikeluarkan adalah 1.6. Tahun 2000, Guido dan para pengembang inti Python pindah ke BeOpen.com yang merupakan sebuah perusahaan komersial dan membentuk BeOpen PythonLabs. Python 2.0 dikeluarkan oleh BeOpen. Setelah mengeluarkan Python 2.0, Guido dan beberapa anggota tim PythonLabs pindah ke DigitalCreations.

2.7 RESTful API

REST (REpresentational State Transfer) merupakan standar arsitektur komunikasi berbasis web yang sering diterapkan dalam pengembangan layanan berbasis web. Umumnya menggunakan HTTP (Hypertext Transfer Protocol) sebagai protocol untuk komunikasi data. REST pertama kali diperkenalkan oleh Roy Fielding pada tahun 2000.

REST API merupakan web service yang bertujuan untuk mendukung kebutuhan server web pada suatu kebutuhan situs atau aplikasi lainnya. program client menggunakan Application Programming Interface (API) untuk berkomunikasi dengan layanan web. Secara umum, API mengekspos seperangkat data dan fungsi untuk memfasilitasi interaksi antara program komputer dan memungkinkan mereka saling bertukar informasi (Masse, 2012).

Dalam pengaplikasiannya, REST lebih banyak digunakan untuk web service yang berorientasi pada resource. Maksud orientasi pada resource adalah orientasi yang menyediakan resource-resource sebagai layanannya dan bukan kumpulan-kumpulan dari aktifitas yang mengolah resource tersebut.

2.8 Ansible

Ansible merupakan sebuah software yang bisa membantu seorang sistem administrator untuk melakukan otomasi pada server. ansible merupakan teknologi yang digunakan untuk melakukan otomasi, memudahkan dalam melakukan konfigurasi server, tujuan dibuat ansible membuat hal tersebut menjadi sederhana dan mudah. namun tetap fokus pada keamanan dan keandalan dalam melakukan otomasi. ansible menggunakan OpenSSH untuk transportasi (dengan mode socket yang cepat).

Dengan ansible sysadmin dapat melakukan instalasi, deployment hingga melakukan update server. Sistem kerja yang dimiliki oleh ansible membutuhkan koneksi khusus berupa SSH. Ansible bekerja di koneksi SSH remote client yang ingin di deploy atau dilakukan otomasi. Pada ansible memerlukan inventory atau data server tujuan untuk dapat dilakukan otomasi. Pada penerapannya, ansible menggunakan playbook dan roles, dimana konfigurasi tersebut dalam format markup YAML dan environment variabel dapat ditulis dalam bentuk JSON. .

ansible dirancang untuk memudahkan para sysadmin dan para pakar IT mengelola lingkungan server dengan mudah. ansible mengelola mesin dengan cara yang tidak biasa, tidak pernah bertanya cara melakukan upgrade daemon jarak jauh atau masalah karena tidak dapat mengelola sistem karena daemon sistem terhapus.

Ansible merupakan salah satu jenis Configuration Management Tools ayan dapat digunakan merubah proses infrastruktur manajemen dari program manual menjadi otomatis. Dalam zaman cloud kehadiran ansible membantu para sysadmin atau para devops dalam instalasi dan konfigurasi server dengan otomatis, oleh karena itu ansible menjadi satu platform yang digunakan untuk mengelola server – server.

BAB III

ANALISIS DAN PERANCANGAN

3.1 Analisis Kebutuhan Sistem

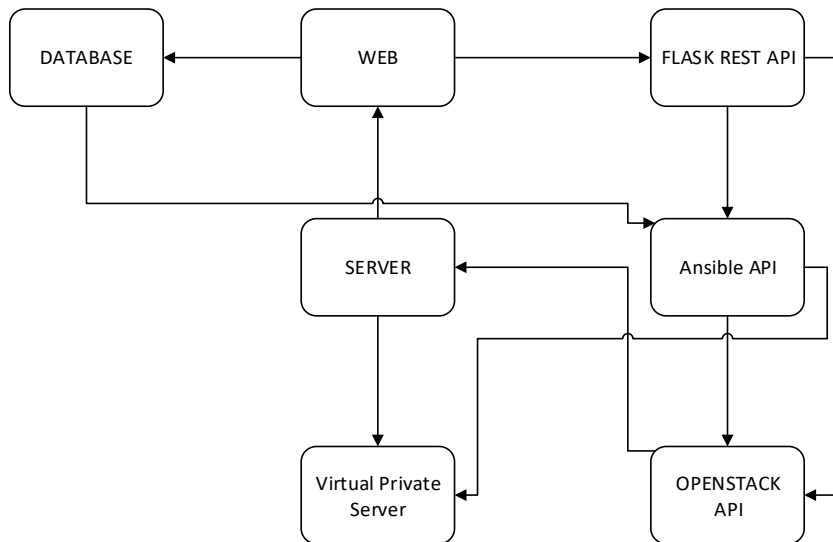
Pada tahap ini dilakukan analisis kebutuhan sistem, analisis kebutuhan sistem meliputi data yang digunakan, pembelajaran dari referensi yang sudah ada dan perangkat yang digunakan baik perangkat lunak maupun perangkat keras:

1. Tahapan pertama yang dilakukan dalam penelitian ini adalah mengidentifikasi permasalahan. Tahap ini merupakan tahap yang paling penting dalam penelitian karena jalannya penelitian didasarkan atas permasalahan yang terjadi. Setelah menentukan masalah yang terjadi, tahapan yang diperlukan selanjutnya adalah menentukan rumusan masalah dan tujuan yang ingin dicapai dalam penelitian. Pada penelitian ini identifikasi permasalahan dilakukan dengan menggunakan teknik observasi, dari teknik ini maka akan dapat diketahui mengenai keluhan – keluhan yang ada di lapangan.
2. Tahap kedua yang dilakukan dalam metodologi penelitian ini adalah studi literatur. Studi literatur dilakukan dengan mengambil literatur – literatur pendukung dari jurnal – jurnal ilmiah, baik jurnal dalam negeri ataupun jurnal luar negeri dan dari beberapa buku. Dalam studi literatur ini, penulis mencari sumber terkait permasalahan – permasalahan yang perlu menjadi perbaikan dalam penelitian selanjutnya.

3.2 Perancangan Sistem

Pada tahap Perancangan sistem, dilakukan berdasarkan hasil analisa kebutuhan sistem yang sudah dilakukan sebelumnya. Hal ini dilakukan agar, perancangan tidak keluar dari tujuan sistem yang dikembangkan

3.2.1 Desain Arsitektur Kerja Sistem



Pada gambar 7.3.1.1 merupakan hubungan antara setiap perangkat lunak yang ada pada server utama. Berikut ini merupakan penjelasannya :

9. *Web dan database*

Database digunakan untuk menyimpan informasi pengguna serta informasi pada *VPS* yang dimiliki oleh setiap pengguna.

10. *Web dan framework flask REST API*

Flask REST API digunakan untuk menghubungkan web *front-end* dengan *back-end*. Dimana *back-end* dibuat dengan menggunakan *framework flask* dari *python*, selanjutnya dibuat *API* tersendiri agar web *front-end* dapat mengirim dan menerima informasi serta konfigurasi yang dilakukan yang nantinya akan diproses oleh *back-end server*.

11. *Database dan ansible API*

Hubungan *database* dengan *ansible API* akan mengambil informasi pengguna berupa *username*, *email* dan *password* yang nantinya akan dimasukkan pada konfigurasi dalam *virtual machine* yang selanjutnya sebagai *super admin* pada *VPS* yang dibuat.

12. *Framework flask REST API dan asible API*

Hubungan *framework flask REST API* dengan *ansible API*. Ketika *user* melakukan konfigurasi pada *web* utama akan dikirim melalui *REST API flask* selanjutnya konfigurasi tersebut akan digunakan oleh *API ansible* untuk mengkonfigurasi *virtual machine* yang dibuat.

13. *Framework flask dan openstack API*

Hubungan *framework flask* dan *openstack API* adalah pada *flask* dikonfigurasi untuk dapat terhubung dengan *server devstack*, dimana *server openstack* yang mengelola dalam membuat sebuah *virtual machine*.

14. *Openstack API dan server*

API openstack digunakan untuk terhubung dengan server *back-end* yang dibuat dengan *framework flask*. Untuk dapat mengembangkan aplikasi *openstack* dari bahasa pemrograman yang berbeda.

15. *Ansible API dan virtual server*

Hubungan *ansible API* dengan *virtual* mesin adalah *ansible* akan mengirimkan konfigurasi pada *VPS* melalui *SSH* yang yang dibuat. Selanjutnya pada *virtual* mesin akan melakukan konfigurasi yang diperlukan untuk membangun sebuah *web* aplikasi secara otomatis. Pengguna hanya perlu memilih keperluan yang ada *menu web front-end*.

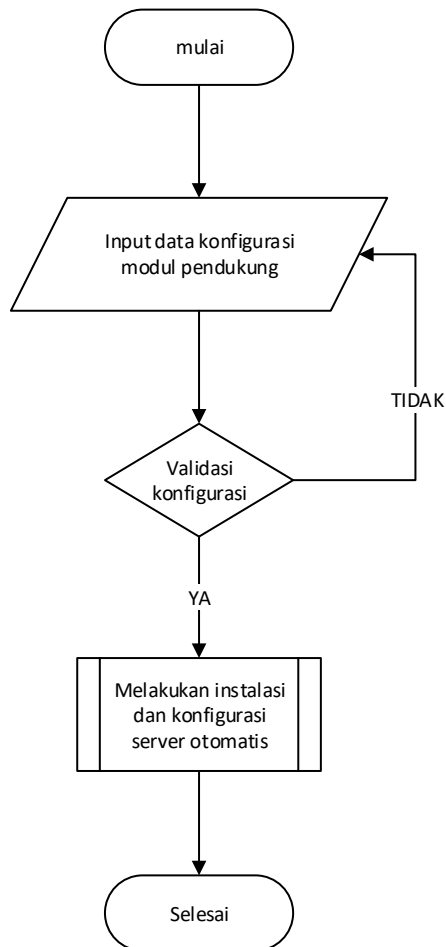
16. *Server dan Virtual Mesin*

Hubungan antara *server* dan *virtual* mesin adalah *server* mengelola dan memberikan *resource* pada *virtual* mesin serta mengatur konektivitas setiap *virtual* mesin yang dimiliki.

3.2.2 Flowchart System

Pada bagian ini akan menjelaskan proses yang dilakukan oleh perangkat lunak, bagaimana proses tersebut berjalan yang akan dijelaskan pada setiap *flowchart* berikut ini:

3.2.2.1 Flowchart Konfigurasi Otomatis

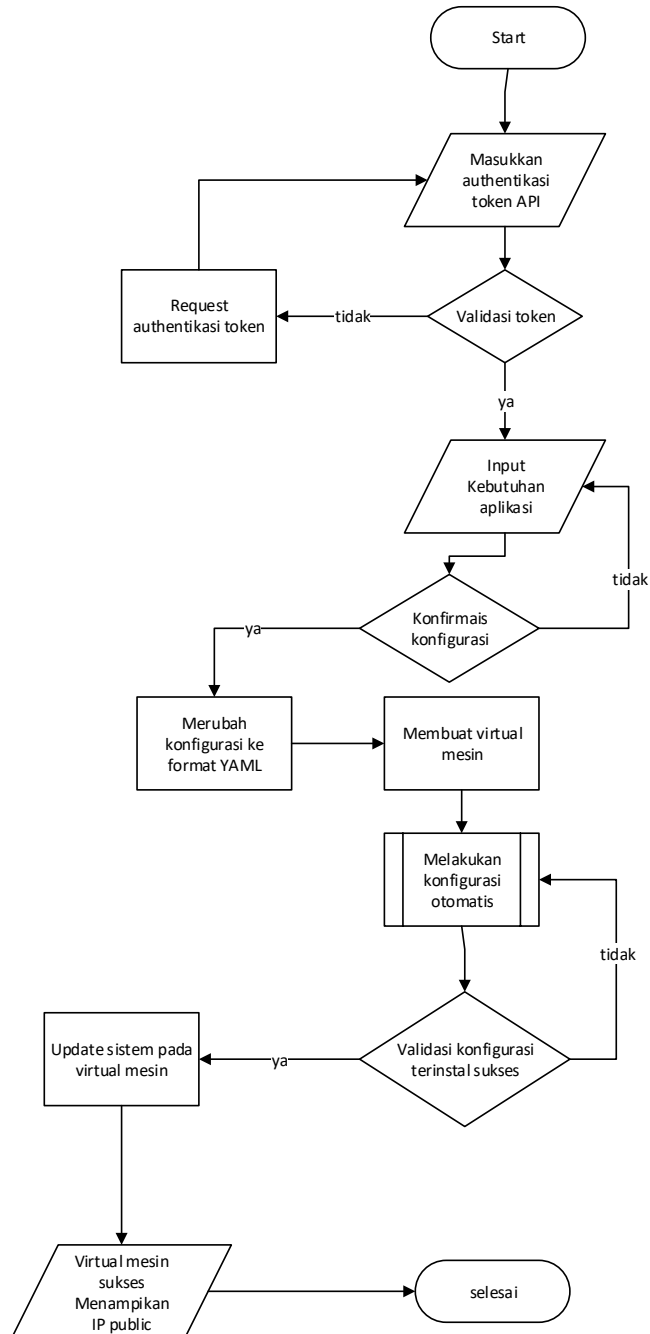


Gambar 3.2.2.1 Flowchart Konfigurasi Otomatis

Pada desain *flowchart* 3.2.2.1 merupakan garis besar gambaran *system* yang akan dikerjakan, bagaimana alur *platform* yang dibuat dapat mengotomatitasi dalam instalasi serta konfigurasi sebuah layanan pada *virtual machine*. Mulai dari memasukkan aplikasi pendukung yang ingin digunakan seperti *database*, *web service*, *username* dan *e-mail administrator* layanan aplikasi. Kemudian dari hasil *input* tersebut diolah oleh *flask API* yang dibuat untuk dimasukkan ke dalam konfigurasi yang terdapat ada *Ansible API*, selanjutnya dari *ansible API* dengan menggunakan *SSH* akan melakukan instalasi konfigurasi pada *virtual server* yang didapat setiap *user*. Bila

konfigurasi telah selesai *user* akan menerima sebuah *ip public* untuk dapat mengakses layanan aplikasi.

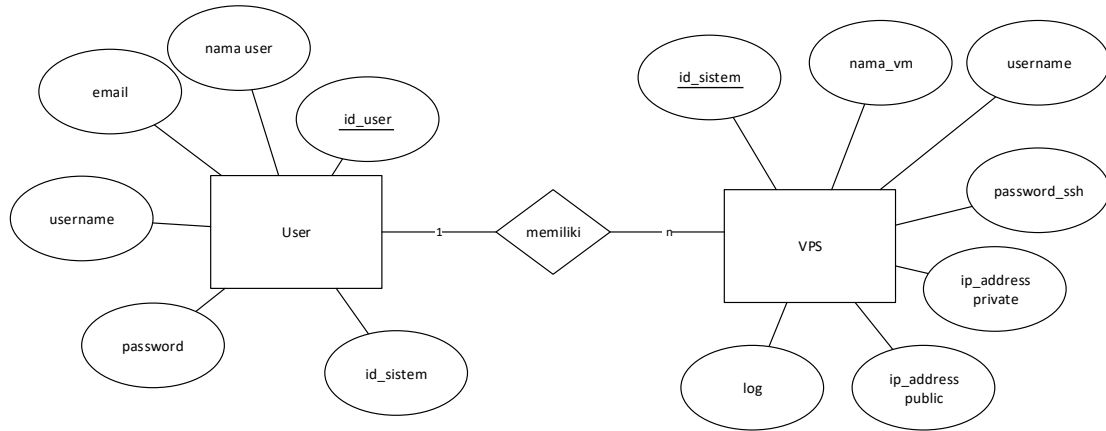
3.2.2.1 Flowchart Konfigurasi Otomatis



Gambar 3.2.2.1.1 Flowchart Back-End

Pada gambar flowchart 7.4..3.1 menjelaskan alur kerja sistem *back-end* sistem. Tahap pertama yang harus dilakukan adalah melakukan komunikasi dengan *openstack* agar dapat menggunakan *API* dari *openstack*. Melakukan komunikasi dengan melakukan *request token autentikasi*, hal ini dimaksudkan agar setiap komunikasi yang dibuat dapat melakukan *update data* pada *API* dan yang memiliki *token* dari *autentikasi* saja yang dapat berkomunikasi. Selanjutnya setelah melakukan *request*, sistem tetap akan melakukan validasi *token id* dari *openstack*, karena setiap melakukan *request token* memiliki batas waktu yang ditentukan. Setelah bisa melakukan komunikasi dengan *openstack* selanjutnya menerima masukan konfigurasi dari *front-end* dan sudah divalidasi oleh administrator, selanjutnya merubah setiap masukan konfigurasi dalam ke *YAML*, *YAML* merupakan format untuk memberi perintah pada *ansible* agar dapat melakukan instalasi maupun konfigurasi secara otomatis. Setelah *file* siap, sistem akan membuat *virtual* mesin dan mengirimkan konfigurasi tersebut ke *virtual* mesin dan langsung melakukan konfigurasi secara otomatis. Selanjutnya sistem akan kembali melakukan mengecek apakah sudah semua *terinstall* dan *terkonfigurasi* dengan baik, sehingga bila ada gagal dalam dilakukan *instalasi* kembali. Setelah semua dicek dan sukses maka selanjutnya sistem akan menampilkan *virtual* mesin siap dan sebuah *ip public* untuk diakses oleh *developer* dan *ssh key* bila *developer* memiliki kebutuhan aplikasi khusus yang tidak dimiliki oleh sistem.

3.2.3 Entity Relational Diagram



Gambar 3.2.3.1 Entity Relationship Diagram Sistem

Pada desain *ERD* diatas menjelaskan hubungan user dengan *VPS* yang dimiliki user memiliki sebuah *vps* dengan satu *ip public* untuk dapat mengakses *server*. Serta pada *vps* akan menyimpan *log* keadaan *vps* mulai dari kinerja dan *resource* yang dimiliki.

3.2.4 Perancangan Antar Muka

3.2.5 Skenario Pengujian Sistem

Bagian ini menjelaskan mengenai skenario pengujian sistem yang akan dilakukan pada Sistem Manajemen Layanan Web Berbasis Platform as a Service (PaaS) dengan API Openstack. Pengujian sistem ini dilakukan dengan menguji black box dan *Performance Testing*.

1. BlackBox Testing

Black Box Testing atau dikenal sebagai “Behaviour Testing” merupakan suatu metode pengujian yang digunakan untuk menguji executable code dari suatu perangkat lunak terhadap perilakunya. Pendekatan Black Box Testing dapat dilakukan jika kita sudah memiliki executable code. Orang-orang yang terlibat dalam Black Box Testing adalah tester, end-user, dan developer.

Fokus dari pengujian ini ialah pada kebutuhan fungsional perangkat lunak, sehingga memungkinkan tester mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program. Kesalahan yang ditemukan dalam pengujian, nantinya dapat disimpulkan apakah kesalahan tersebut murni dikarenakan kesalahan dari aplikasi atau kesalahan implementasi dari tester.

Tabel 3. 1 Tabel Pengujian Black Box

Berkas 1. Tabel Pengujian Black Box		
Identifikasi		
Nama Kasus Uji		
Deskripsi		
Kondisi Awal		
Tanggal Pengujian		
Penguji		
Skenario		
1.		
2.		
(Dst...)		
Hasil Yang Diharapkan	Hasil Yang Didapatkan	Kesimpulan

Untuk pengujian antarmuka pengguna atau rancangan skenario pengujian balck box dari sistem ini, dilakukan dua jenis pengujian yaitu pengujian secara happy path yaitu pengujian yang dilakukan dengan cara yang benar, serta pengujian secara alternative path yaitu mencoba segala kemungkinan yang mungkin terjadi pada sistem.

2. Performance Testing

Teknik pengujian memvalidasi perilaku perangkat lunak terhadap teknik pengujian software dari sisi kecepatan. Kecepatan ini dalam konteks pengujian dalam mengukur waktu respon perangkat lunak ketika berada jumlah kerja yang berlebih yang

biasa dikenal dengan beban kerja. Untuk memperlihatkan kecepatan sebenarnya sebuah perangkat lunak harus dilakukan pengujian performance testing.

Tujuan dari performance testing untuk memvalidasi kecepatan sebuah perangkat lunak terhadap kebutuhan sistem yang cepat. Secara umum harus mendefinisikan kombinasi waktu respons dan beban kerja

BAB IV

HASIL DAN PEMBAHASAN

4.1 Gambaran Umum Sistem

Sistem yang dibuat bertujuan membantu para pengembang aplikasi berbasis web, dengan meneriakkan platform web server berbasis cloud computing, dengan menyediakan virtual machine berbasis cloud/ instance. Pengembang web dapat dengan mudah menjalankan sebuah web yang dibuat dengan mengakses alamat IP yang di berikan. Karena berbasis cloud, pengembang web dapat mengakses virtual machine yang di buat dimana saja secara online. Namun pada penelitian ini pengembang hanya dapat menggunakan jaringan Local untuk mengkases server cloud / private cloud.

Dalam implementasinya, server virtual yang diberikan pada pengembang web, secara otomatis sudah terinstall LAMP package. Namun, bila pengembang web ingin menggunakan package web server yang berbeda dapat melakukan instalasi manual dengan mengakses virtual machine menggunakan SSH.

4.2 Lingkungan Implementasi

Sistem dirancang dan diimplementasikan pada sistem operasi Ubuntu dalam virtualisasi menggunakan Vmware, server dan sistem berjalan pada Laptop dengan spesifikasi i7, RAM 16 GB, dengan sistem operasi windows 10 64bit dan dengan support virtualisasi.

4.3 Implementasi Basis Data

```
app.config['SECRET_KEY'] = 'Idabagusrathuekasuryawibawa!'
app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = True
app.config['SQLALCHEMY_DATABASE_URI']=
'sqlite:///home/gustu/myweb/database/database.db'
Bootstrap(app)
db = SQLAlchemy(app)
login_manager = LoginManager()
login_manager.init_app(app)
login_manager.login_view = 'login'
```



```

level = 0

class User (UserMixin, db.Model):
    id = db.Column(db.Integer,primary_key=True)
    username = db.Column(db.String(20), unique=True)
    email = db.Column(db.String(50), unique=True)
    password = db.Column(db.String(80))
    level = db.Column(db.Integer)

class Project(UserMixin, db.Model):
    id = db.Column(db.Integer,primary_key=True)
    project = db.Column(db.String(20), unique=True)
    description = db.Column(db.String(255))
    idos = db.Column(db.String(128))
    osdesc = db.Column(db.String(50))
    idcpu = db.Column(db.String(128))
    cpudesc = db.Column(db.String(50))
    user = db.Column(db.String(20))
    addr = db.Column(db.String(255))

@login_manager.user_loader
def load_user(user_id):
    return User.query.get(int(user_id))

class LoginForm(FlaskForm):
    username = StringField('username', validators=[InputRequired(),
Length(min=4, max=20)])
    password = PasswordField('password',
validators=[InputRequired(), Length(min=8, max=80)])
    remember = BooleanField('remember me')

class RegisterForm(FlaskForm):

```

```

    email = StringField('email', validators=[InputRequired(),
Email(message='email salah'), Length(max=50)])

    username = StringField('username', validators=[InputRequired(),
Length(min=4, max=20)])

    password = PasswordField('password',
validators=[InputRequired(), Length(min=8, max=80)])

class ProjectForm(FlaskForm):
    namaproject = StringField('namaproject',
validators=[InputRequired(), Length(max=20)])

    deskripsi = StringField('deskripsi')
    sistem_operasi = StringField('sistem_operasi')
    cpu = StringField('cpu')

```

Pada baris kode tabel merupakan baris kode untuk menginisiasi basis data user atau pengguna dan project, pada field user berisi informasi id yang digenerate secara otomatis, username, email, password dalam bentuk hash dan level pengguna. Selanjutnya pada field project berisi ID project, nama project, ID sistem operasi ID CPU, ID penyimpanan, User, dan IP Address server.

```

def signup():
    form=RegisterForm()

    if form.validate_on_submit():
        hashed_password = generate_password_hash(form.password.data, method='sha256')
        new_user = User(username=form.username.data,
email=form.email.data, password=hashed_password, level=1)
        db.session.add(new_user)
        db.session.commit()
        return render_template('login_new.html', form=LoginForm())

    return render_template('signup.html', form=form)

```

Pada baris kode merupakan baris untuk melakukan pendaftaran ke website, hashed_password digunakan untuk mengamankan password pengguna baru lalu diberi level akses sebagai pengguna bukan pada level administrator.

```
def login():

    global level

    form=LoginForm()

    if form.validate_on_submit():
        user = User.query.filter_by(username=form.username.data).first()
        if user:
            if check_password_hash(user.password,
form.password.data) and user.level >= 0:
                login_user(user, remember=form.remember.data)
                level = user.level
                return redirect(url_for('dashboard'))
            return redirect(url_for('login'))

    return render_template('login.html', form=form)
```

Pada baris kode ini merupakan kode fungsi untuk melakukan validasi login ke dalam website, fungsi ini akan melakukan pengecekan apakah username merupakan administrator atau tidak, if check_password_hash(user.password, form.password.data) and user.level >= 0: bila level login sama dengan 1 (satu) maka username akan login sebagai pengguna.

```
def profil_save():
    data = request.json['data']

    hashed_password = generate_password_hash(data["password"],
method='sha256')
```

```

user = User.query.filter_by(username=data["username"]).first()
user.email = data["email"]
user.password = hashed_password

db.session.merge(user)
db.session.commit()

json_data = {"status": True}

response = app.response_class(
    response=json.dumps(json_data),
    mimetype='application/json'
)

return response

```

Pada baris kode ini merupakan fungsi untuk merubah password pengguna, password digenerate kembali dahulu pada fungsi `hashed_password = generate_password_hash(data["password"], method='sha256')`, selanjutnya pada user, melakukan filter sesuai user yang login lalu di update kembali ke basis data `db.session.merge(user)`, dan tutup kembali database dengan `commit db.session.commit()`.

4.4 Implementasi Sistem

Sistem membutuhkan server Openstack sebagai infrastruktur layanan cloud computing. Openstack merupakan dasar sistem yang digunakan untuk membangun sebuah sistem cloud. Openstack memberikan kemampuan untuk mengabstraksi sumber daya seperti komputasi (`compute`), penyimpanan (`storage`), dan sumber jaringan (`networking resource`), yang disatukan menjadi kumpulan sumber daya yang nantinya akan dibagi ke pengguna dengan aman. Openstack merupakan software server yang bersifat open source, dan pengembangan – pengembangan aplikasi dengan openstack

sebagai inti sistem sangat terbuka lebar dengan disediakannya API untuk pengembangan sistem yang berjalan di atas sistem openstack.

Pada penelitian ini. Openstack yang dijalankan untuk mendukung kebutuhan pengembangan sistem menggunakan framework tools dari openstack yaitu Kolla-Ansible, framework ini memudahkan dalam pengembangan openstack yang dapat disesuaikan dengan layanan yang dibutuhkan dan komponen infrastruktur dalam docker containers.

```

root@openstack:/home/gustu
root@openstack:/home/gustu# docker image ls
REPOSITORY          TAG                IMAGE ID           CREATED            SIZE
kolla/ubuntu-source-neutron-server    train             f3d9ee12e61c      3 months ago      1.03GB
kolla/ubuntu-source-neutron-l3-agent  train             d5aa69285e46      3 months ago      1.02GB
kolla/ubuntu-source-neutron-metadata-agent train            491c94a1acaf      3 months ago      1.01GB
kolla/ubuntu-source-neutron-openvswitch-agent train           563e9295b8ea      3 months ago      1.01GB
kolla/ubuntu-source-neutron-dhcp-agent train            d286c32a64e8      3 months ago      1.01GB
kolla/ubuntu-source-keystone-ssh      train            c8ca218fad6       3 months ago      919MB
kolla/ubuntu-source-keystone-fernet   train            cb2ab0946e45      3 months ago      918MB
kolla/ubuntu-source-keystone          train            7b6f9b94358a      3 months ago      909MB
kolla/ubuntu-source-nova-compute       train            0c9baa6fe97d      3 months ago      1.31GB
kolla/ubuntu-source-glance-api         train            f897ed1db579      3 months ago      929MB
kolla/ubuntu-source-placement-api      train            f6299bcb768       3 months ago      909MB
kolla/ubuntu-source-nova-novncproxy    train            cc899fca0606      3 months ago      1.19GB
kolla/ubuntu-source-nova-api           train            89edf13ecbbd      3 months ago      1.1GB
kolla/ubuntu-source-nova-scheduler     train            06fc844e48ec      3 months ago      1.09GB
kolla/ubuntu-source-nova-ssh           train            fbc6c6107862      3 months ago      1.09GB
kolla/ubuntu-source-nova-conductor     train            b0be03c12f8a      3 months ago      1.09GB
kolla/ubuntu-source-heat-api           train            4bc24f78a549      3 months ago      932MB
kolla/ubuntu-source-heat-engine        train            348740912fe3      3 months ago      932MB
kolla/ubuntu-source-heat-api-cfn       train            1666647b2eb6      3 months ago      932MB
kolla/ubuntu-source-horizon            train            687e861df058      3 months ago      1.05GB
kolla/ubuntu-source-rabbitmq           train            1f2de4f1e3d4      3 months ago      205MB
kolla/ubuntu-source-fluentd            train            d7a1df289137      3 months ago      567MB
kolla/ubuntu-source-openvswitch-vswitchd train           0985de85bab       3 months ago      254MB
kolla/ubuntu-source-openvswitch-db-server train           173341a3a1c2      3 months ago      254MB
kolla/ubuntu-source-haproxy            train            4546c6cbb6f0      3 months ago      244MB
kolla/ubuntu-source-keepalived         train            ca9c7ac346c       3 months ago      246MB
kolla/ubuntu-source-kolla-toolbox      train            1522621ca7c3      3 months ago      923MB
kolla/ubuntu-source-memcached           train            777464badcf       3 months ago      239MB
kolla/ubuntu-source-nova-libvirt       train            a6296a630e8       3 months ago      798MB
kolla/ubuntu-source-mariadb            train            f7f68292ab15      3 months ago      565MB
kolla/ubuntu-source-chrony             train            823aa594dfdf       3 months ago      243MB
kolla/ubuntu-source-cron               train            aa60c550bbc9      3 months ago      238MB
root@openstack:/home/gustu#

```

```

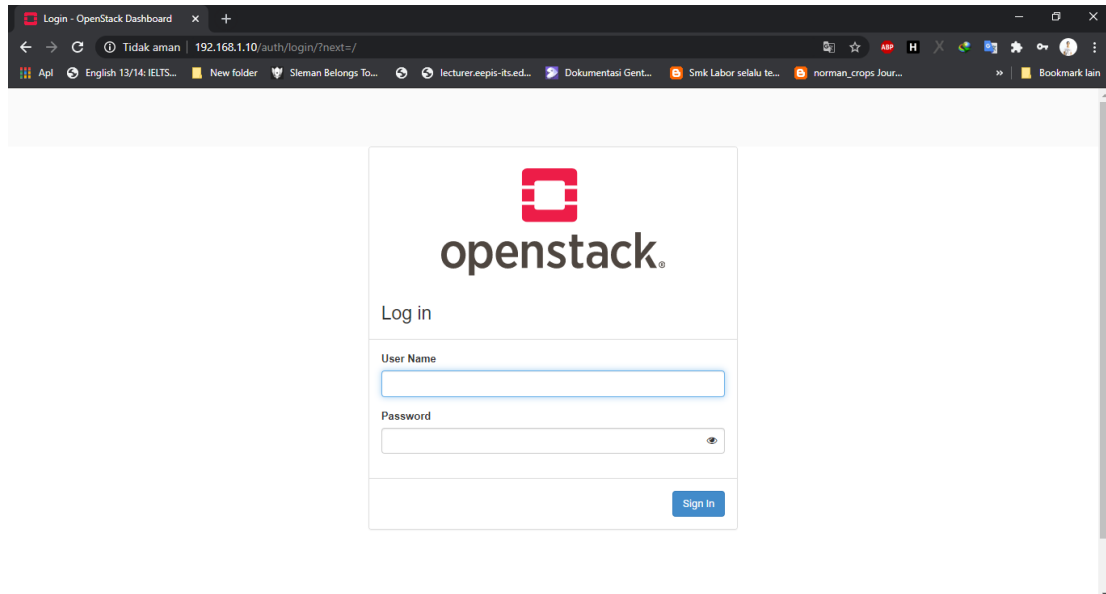
root@openstack:/home/gustu
root@openstack:/home/gustu# docker container ls
CONTAINER ID        NAME               COMMAND              CREATED            STATUS            PORTS
9f4882a3b01d       kolla/ubuntu-source-heat-engine:train "dumb-init --single-..." 3 months ago      Up 3 weeks              heat_engine
9f413b960c4c       kolla/ubuntu-source-heat-api-cfn:train "dumb-init --single-..." 3 months ago      Up 3 weeks              heat_api_cfn
f4e635589932       kolla/ubuntu-source-heat-api:train    "dumb-init --single-..." 3 months ago      Up 3 weeks              heat_api
913ac24be0f3       kolla/ubuntu-source-neutron-metadata-agent:train "dumb-init --single-..." 3 months ago      Up 3 weeks              neutron_meta
8c29f310f44       kolla/ubuntu-source-neutron-l3-agent:train "dumb-init --single-..." 3 months ago      Up 3 weeks              neutron_l3_a
gent
84f6f5d4278e       kolla/ubuntu-source-neutron-dhcp-agent:train "dumb-init --single-..." 3 months ago      Up 3 weeks              neutron_dhcp
gent
5f6fc0ec9a7        kolla/ubuntu-source-neutron-server:train "dumb-init --single-..." 3 months ago      Up 3 weeks              neutron_serv
gent
eef9d455dc4d       kolla/ubuntu-source-openvswitch-vswitchd:train "dumb-init --single-..." 3 months ago      Up 3 weeks              openvswitch
gent
027060377562       kolla/ubuntu-source-openvswitch-db-server:train "dumb-init --single-..." 3 months ago      Up 3 weeks              openvswitch
db
f4f693d3289f       kolla/ubuntu-source-nova-libvirt:train "dumb-init --single-..." 3 months ago      Up 3 weeks              nova_libvirt
d7f59c16e575       kolla/ubuntu-source-nova-ssh:train     "dumb-init --single-..." 3 months ago      Up 3 weeks              nova_ssh
4c2be0b3775f       kolla/ubuntu-source-nova-novncproxy:train "dumb-init --single-..." 3 months ago      Up 3 weeks              nova_novncpr
oxy
ab8526251a13       kolla/ubuntu-source-nova-conductor:train "dumb-init --single-..." 3 months ago      Up 3 weeks              nova_conduct
gent
eef71806a59f       kolla/ubuntu-source-glance-api:train   "dumb-init --single-..." 3 months ago      Up 3 weeks              glance_api
8b9246433694       kolla/ubuntu-source-keystone-fernet:train "dumb-init --single-..." 3 months ago      Up 3 weeks              keystone_fer
net
8d2b88951928       kolla/ubuntu-source-keystone-ssh:train "dumb-init --single-..." 3 months ago      Up 3 weeks              keystone_ssh
gent
5ba05f567e32       kolla/ubuntu-source-memcached:train    "dumb-init --single-..." 3 months ago      Up 3 weeks              memcached
gent
4232a6523f4       kolla/ubuntu-source-keepalived:train   "dumb-init --single-..." 3 months ago      Up 3 weeks              keepalived
gent
088e5de0e0e0       kolla/ubuntu-source-haproxy:train      "dumb-init --single-..." 3 months ago      Up 3 weeks              haproxy
gent
6a313b55088c       kolla/ubuntu-source-chrony:train       "dumb-init --single-..." 3 months ago      Up 3 weeks              chrony
gent
f618a2a59397       kolla/ubuntu-source-cron:train         "dumb-init --single-..." 3 months ago      Up 3 weeks              cron
gent
621a07049e87       kolla/ubuntu-source-kolla-toolbox:train "dumb-init --single-..." 3 months ago      Up 3 weeks              kolla_toolbo
x
8ddc3dffbeeb       kolla/ubuntu-source-fluentd:train      "dumb-init --single-..." 3 months ago      Up 3 weeks              fluentd
root@openstack:/home/gustu#

```

Tampilan awal openstack

Dalam server yang digunakan, server openstack dapat diakses melalui web interface. Openstack menyediakan pilihan pengembangan sistemnya dapat melalui CLI (Command-Line Interface) maupun melalui web interface. Namun untuk

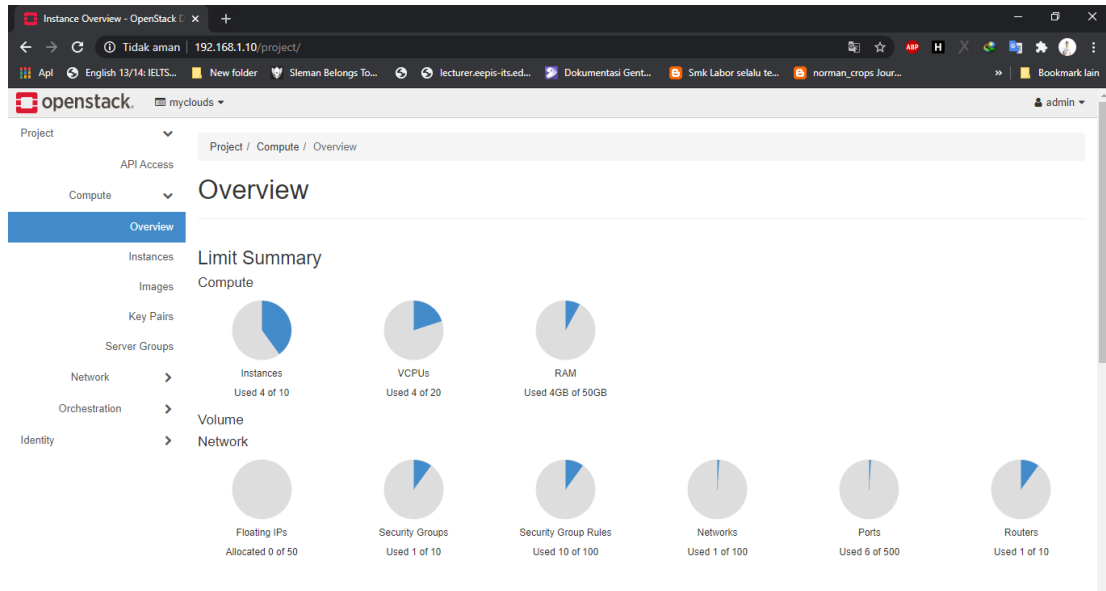
pengembangan yang lebih dalam disarankan menggunakan CLI, pada penelitian ini bisa dikembangkan dengan openstack horizon / openstack melalui web interfaces.



Menu awal untuk masuk ke dalam sistem openstack diharuskan login terlebih dahulu. Untuk dapat login openstack secara otomatis menggenerate kata sandi untuk kepentingan keamanan. Namun kita dapat mengubahnya bila dirasa sangat sulit diingat. Untuk dapat mengetahui kata sandi dari openstack dapat diakses melalui command-line

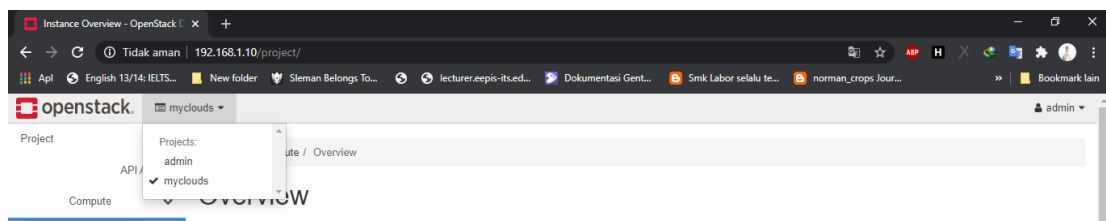
Dashboard Openstack

Bila username dan kata sandi benar, maka akan diarahkan ke menu dashboard. Pada menu dashboard kita dapat mengetahui berapa maksimum user yang dapat dibuat oleh server openstack, berapa core CPU yang masih tersedia dan dapat digunakan dalam pembuatan instance / virtual machine.



Membuat Project untuk mengembangkan sistem

pada menu Identity, peneliti membuat project baru pada sistem yang berjalan di atas server openstack. Sehingga bila terjadi sesuatu pada sistem yang dibuat tidak mempengaruhi server openstack secara keseluruhan, dan dikemudian hari ada penelitian lebih lanjut dalam pengembangan cloud computing, bisa dilakukan pada nama project yang berbeda.



Jadi server openstack secara keseluruhan tidak terganggu atau mengalami masalah bila salah satu project mengalami masalah dalam pengembangannya. Administrator hanya perlu menghapus projectnya tanpa perlu menginstall ulang sistem ataupun server openstack.

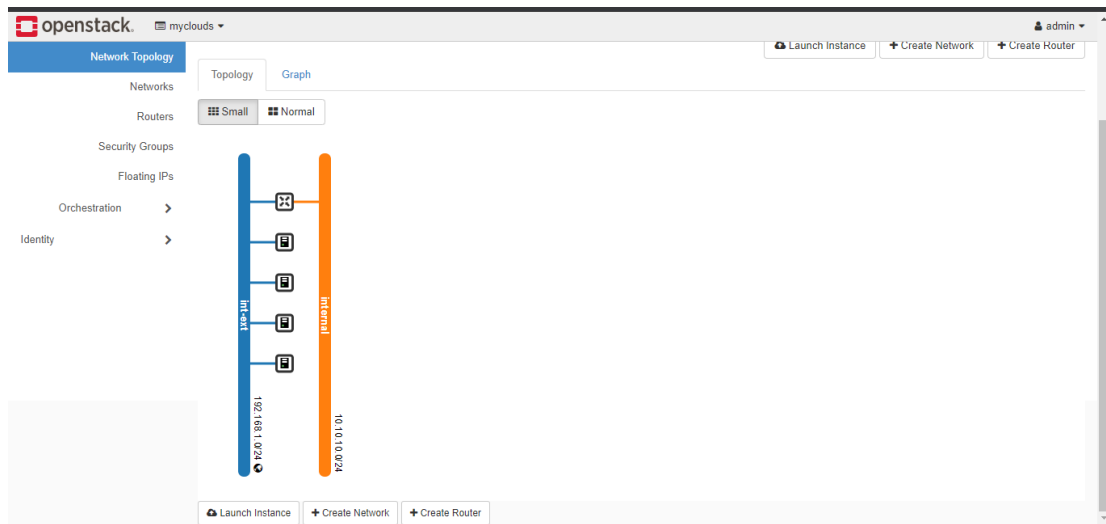
Pengaturan security groups

Untuk menu Security Group, berguna untuk memberi akses port yang bisa digunakan oleh instance. Port yang digunakan oleh instance tergantung kebutuhan yang perlukan, karena setiap instance memungkinkan menggunakan port yang berbeda – beda. Mungkin saja ada instance yang dibuat untuk penyimpanan aplikasi, ataupun mungkin digunakan untuk jadikan aplikasinya dapat diakses secara publik.

Direction	Ether Type	IP Protocol	Port Range	Remote IP Prefix	Remote Security Group	Description	Actions
Egress	IPv4	Any	Any	0.0.0.0/0	-	-	Delete Rule
Egress	IPv6	Any	Any	::/0	-	-	Delete Rule
Ingress	IPv4	Any	Any	-	default	-	Delete Rule
Ingress	IPv4	ICMP	Any	0.0.0.0/0	-	-	Delete Rule
Ingress	IPv4	TCP	1 - 65535	0.0.0.0/0	-	-	Delete Rule
Ingress	IPv4	TCP	22 (SSH)	0.0.0.0/0	-	-	Delete Rule
Ingress	IPv4	TCP	80 (HTTP)	0.0.0.0/0	-	-	Delete Rule
Ingress	IPv4	TCP	443 (HTTPS)	0.0.0.0/0	-	-	Delete Rule
Ingress	IPv4	UDP	1 - 65535	0.0.0.0/0	-	-	Delete Rule
Ingress	IPv6	Any	Any	-	default	-	Delete Rule

Topologi jaringan openstack dengan instance

Pada menu Network Topology, berisi informasi network IP yang yang instance peroleh, dan di jaringan mana instance terhubung. Pada server openstack secara otomatis setiap membuat project baru akan dialihkan ke jaringan private network, sehingga dalam mengembangkan sebuah project setiap user mendapat alokasi ip private masing – masing dan secara otomatis dilakukan NAT (Network Address Translation) oleh server openstack sehingga meski mendapatkan ip private dari server openstack dapat tetap mendapatkan akses ke jaringan internet, namun tidak dapat digunakan meremote instance melalui jaringan public.



Untuk dapat menjadikan instance yang dibuat dapat diremote melalui jaringan public. Pada konfigurasi jaringan penelitian ini dipilih yang public atau dalam hal ini bernama int-ext.

API Openstack

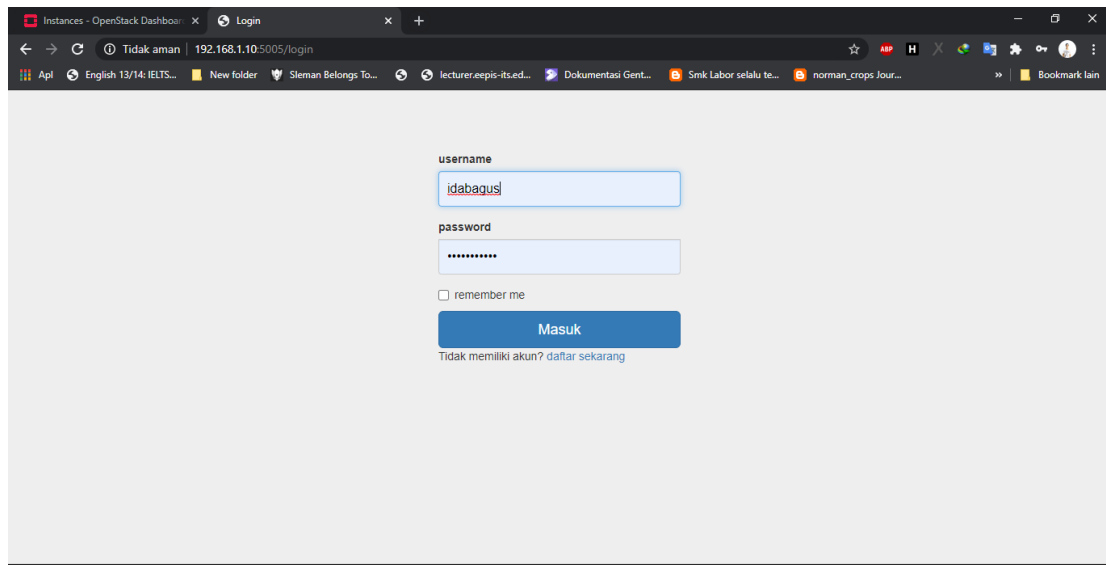
Service	Service Endpoint
Cloudformation	http://192.168.1.10:8000/v1
Compute	http://192.168.1.10:8774/v2.1
Compute_Legacy	http://192.168.1.10:8774/v2/084aa4c02ba3453389695d81f931049
Identity	http://192.168.1.10:5000
Image	http://192.168.1.10:9292
Network	http://192.168.1.10:9696
Orchestration	http://192.168.1.10:8004/v1/084aa4c02ba3453389695d81f931049
Placement	http://192.168.1.10:8780

4.5 Antar Muka Sistem

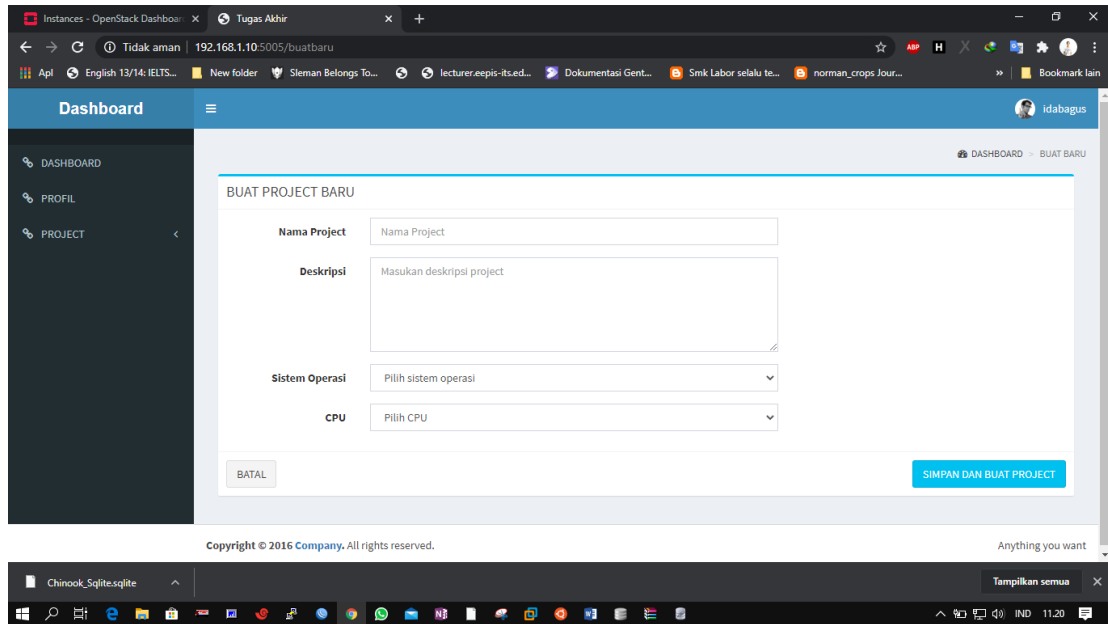
Implementasi antarmuka sistem pada penelitian ini bangun dengan menggunakan bahasa pemrograman python dengan framework flask. Pada sub-bab ini akan menampilkan implementasi antarmuka website, diantaranya landing page, login,

dashboard, menu membuat instance baru, dan daftar instance yang telah dibuat, menu profil. Berikut ini adalah pemaparannya :

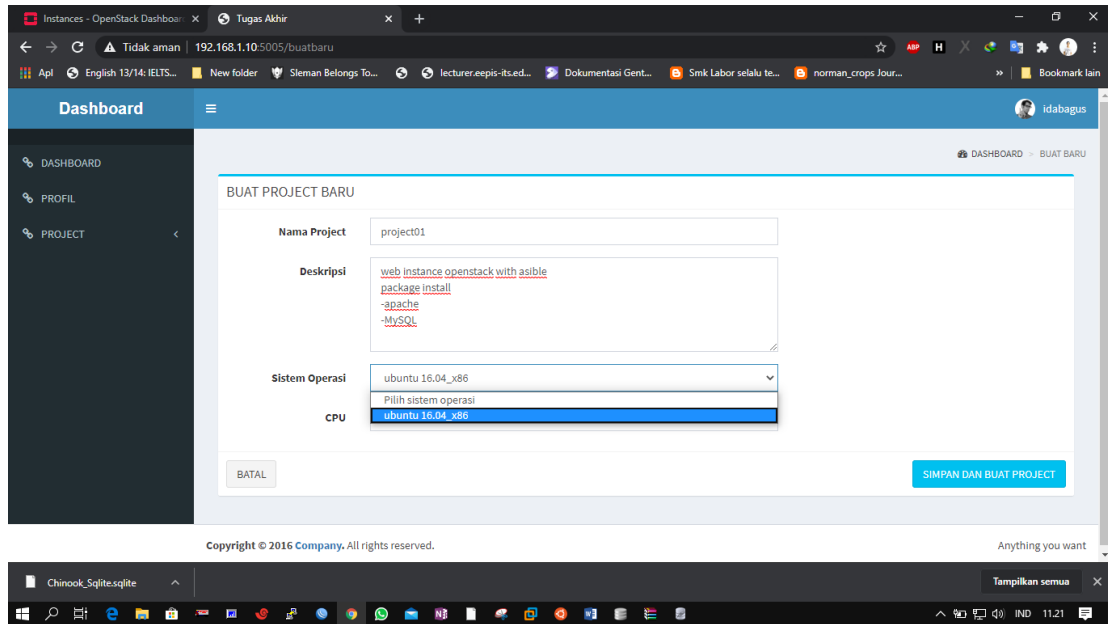
Pada penelitian ini, pengguna baru harus melakukan registrasi untuk dapat masuk ke dalam sistem, Setelah melakukan pendaftaran, pengguna akan diarahkan ke form login, selanjutnya pengguna dapat memasukkan username dan password yang sudah di buat pada form registrasi



Setelah proses login, pengguna akan diarahkan langsung ke dashboard menu. Pada dashboard menu, selanjutnya membuat virtual mesin baru pada menu “project”, kemudian pilih “buat baru”



Setelah dipilih akan muncul menu seperti pada gambar , selanjutnya pengguna memasukkan nama project. Nama project nantinya akan menjadi nama instance baru sebagai hostname, untuk deskripsi bersifat opsional, dapat ditambahkan keterangan ataupun tidak. Lalu pilih siste operasi yang akan digunakan sebagai server pada lingkungan aplikasi yang akan dibuat. Lalu pilih CPU, CPU disini berisi berapa core yang di butuhkan untuk lingkungan sistem yang akan dibangun oleh pengguna, seberapa besar penyimpanan dan memori RAM yang dibutuhkan untuk mengembangkan sistem yang akan dibuat. Lebih jelasnya dapat dilihat pada gambar berikut



Pada proses yang di ditampilkan pada gambar , pada backend dibuat fungsi untuk mendapatkan list CPU,RAM dan Storage didefinisikan dengan flavor, dengan mengauthentikasikan terlebih dahulu dengan API openstack menggunakan osconn. Dilanjutkan dengan fungsi yang akan ditampilkan, compute.flavors() untuk menampilkan list cpu ram dan storage, compute.images() untuk menampilkan list sistem operasi yang disediakan.

```
def getflavor():
    flavors = osconn.compute.flavors()
    for flavor in osconn.compute.flavors():
        print(flavor)

    json_data = [flavor]

    itr = True
    while itr:
        try:
            json_data.append(flavors.next().to_dict())
        except:
            itr = False
```

```

#     print(json_data)
    return json_data

# fungsi untuk melihat list sistem operasi yang tersedia
def getimages():
    images = osconn.compute.images()
    for image in osconn.compute.images():
        print(image)
    json_data = [image]

    itr = True
    while itr:
        try:
            json_data.append(images.next().to_dict())
        except:
            itr = False
    #     print(json_data)
    return json_data

```

Pada fungsi `getproject()` digunakan untuk mencetak semua list instance / virtual mesin yang sudah dibuat. Keluaran dari fungsi berikut ini berupa json selanjutnya dapat di olah kembali untuk ditampilkan di front-end.

```

def getproject():
    servers = osconn.compute.servers()
    for server in osconn.compute.servers():
        print(server)

    json_data = [server]

    itr = True
    while itr:

```

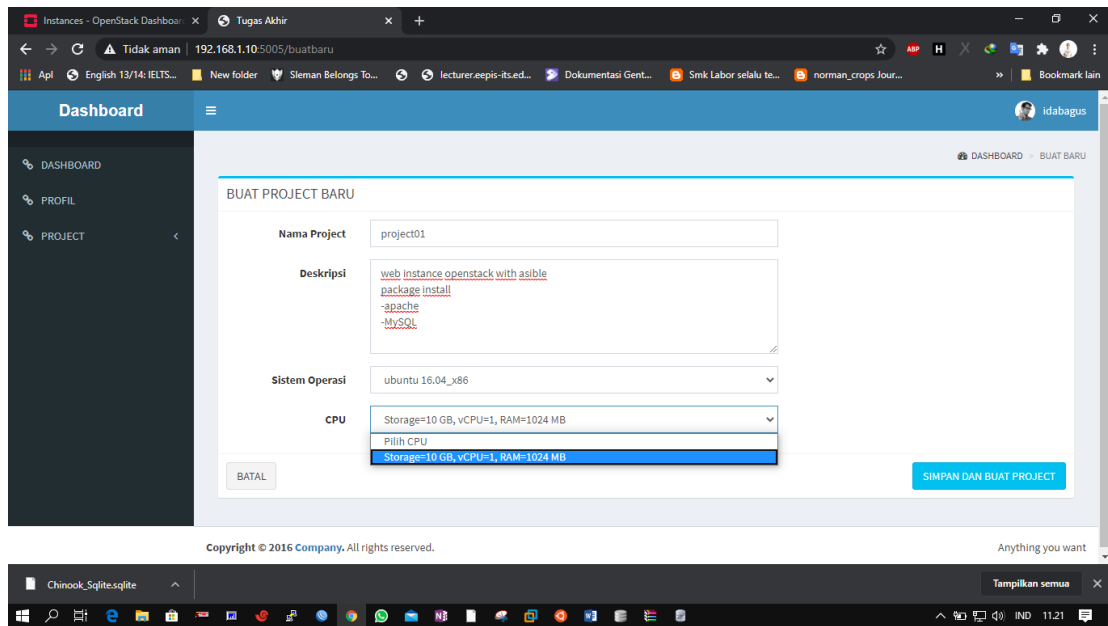
```
        try:
            json_data.append(servers.next().to_dict())
        except:
            itr = False

#    print(json_data)
    return json_data
```

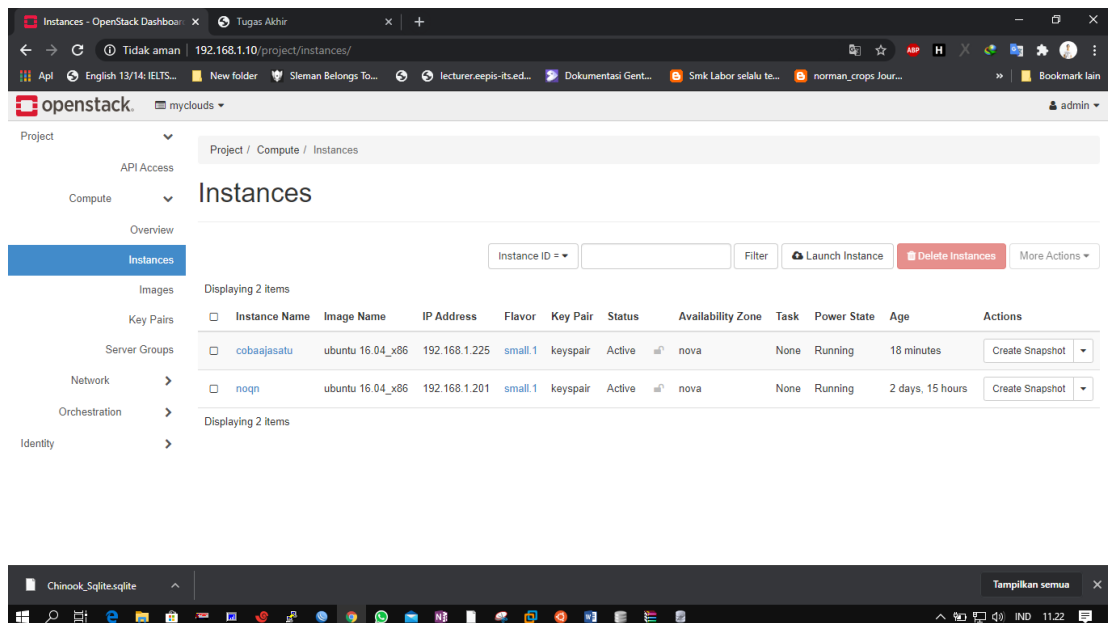
Pada fungsi di bawah ini merupakan fungsi untuk membuat instance baru dengan menampilkan kebutuhan flavor mesin dan sistem operasi yang akan di install pada instance lalu untuk menyimpan konfigurasi diberi nama sesuai user yang sedang login, dalam hal ini user dapat mengetahui berapa banyak instance yang sudah dibuatnya.

```
def buatbaru():
    json_data = {}
    json_data.update({"flavors": getflavor()})
    json_data.update({"images": getimages()})
    return render_template('buatbaru.html', json_data =
    json_data, level=level, name= current_user.username)
```

Pada segi antarmuka terlihat pada gambar dibawah tampilan proses dalam membuat instance baru. Pada sistem operasi hanya menampilkan satu sistem operasi saja dikarenakan pada server openstack hanya ada 1 sistem operasi yang dibuat. Begitu pula dengan CPU, RAM dan penyimpanan yang tersedia.



Pada gambar dibawah ini merupakan server openstack sebelum proses pembuatan instance baru,



Selanjutnya ketika melakukan klik pada tombol buat dan simpan project, sistem aplikasi akan mengirimkan perintah untuk membuat sebuah instance baru pada server

openstack dengan menggunakan API dan ansible playbook yang sudah dibuat pada tabel berikut ini.

```
def project_save():
    form=ProjectForm()

    osdesc = ""
    images = getimages()

    for image in images:
        if form.sistem_operasi.data == image["id"]:
            osdesc = image["name"]
            break

    cpudesc = ""
    flavors = getflavor()
    cduname = ""

    for flavor in flavors:
        if form.cpu.data == flavor["id"]:
            cpudesc = "Storage= "+str(flavor["disk"])+ " GB, vCPU=
"+str(flavor["vcpus"])+", RAM= "+str(flavor["ram"])+ " MB"
            cpuname = str(flavor["name"])
            break

    new_project = Project(
        project=form.namaproject.data,
        description=form.deskripsi.data,
        idos=form.sistem_operasi.data,
        osdesc=osdesc,
        idcpu=form.cpu.data,
        cpudesc=cpudesc,
        user=current_user.username
    )
    db.session.add(new_project)
```



```

db.session.commit()

fin = open("yaml/deploy.yaml", "rt")
data = fin.read()
data = data.replace('<name>', form.namaproject.data)
data = data.replace('<image>', osdesc)
data = data.replace('<flav>', cpuname)
fin.close()

fin = open("yaml/deploy_"+current_user.username+".yaml", "wt")
fin.write(data)
fin.close()

p = subprocess.Popen("ansible-playbook " +
"yaml/deploy_"+current_user.username+".yaml", shell=True,
stdout=subprocess.PIPE)
stdout, stderr = p.communicate()

project = None
if level == 0:
    project = Project.query.all();
else:
    project =
Project.query.filter_by(user=current_user.username).all();

return render_template('projectlist.html', project=project,
level=level)

```

API openstack digunakan untuk mengambil informasi dari nama sebuah project lalu selanjutnya dari nama tersebut selanjutnya akan dibuat file dengan ekstensi yaml (.yaml) dengan baris perintah

```

fin = open("yaml/deploy_"+current_user.username+".yaml", "wt")
fin.write(data).

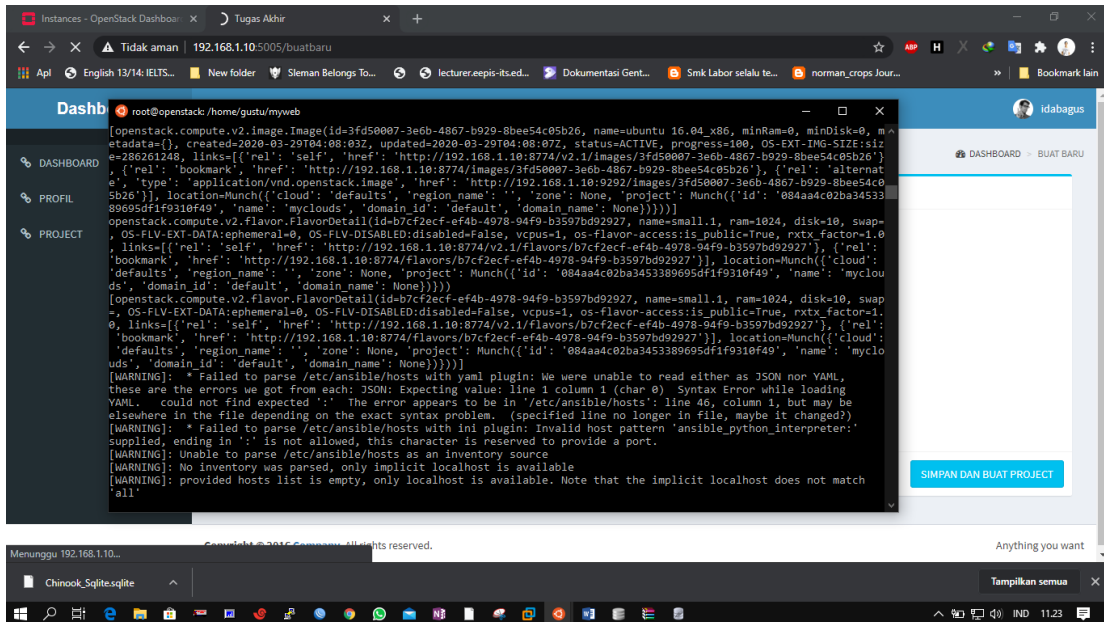
```

Setelah file yaml selesai dibuat, sistem akan langsung mengeksekusi file yaml menggunakan perintah ansible playbook, didefinisikan dengan p untuk memanggil perintah eksternal dari python.

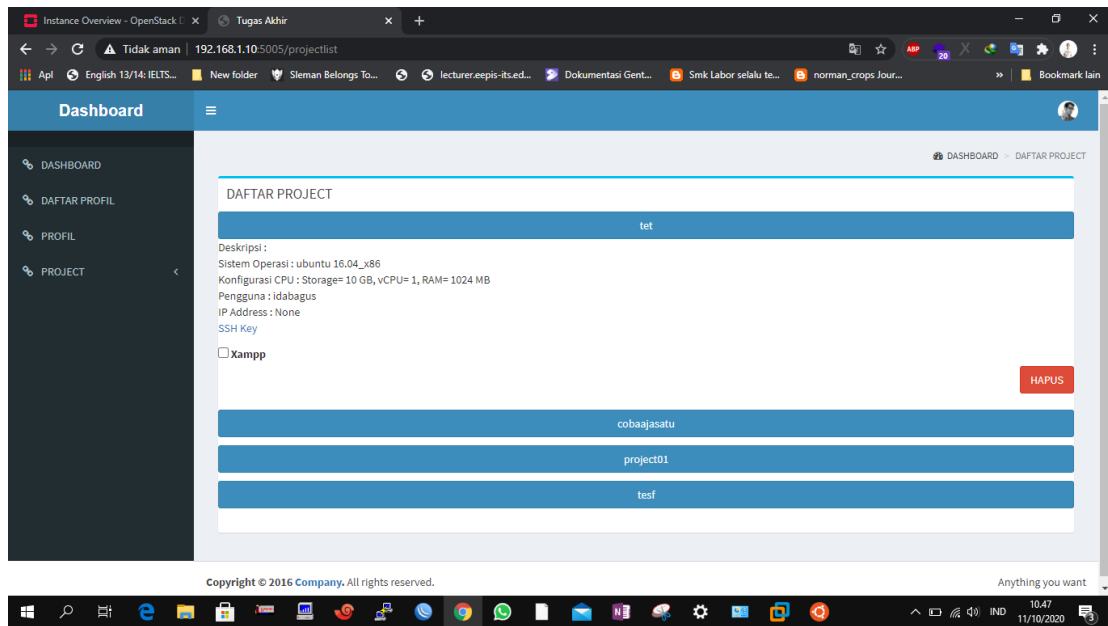
```
p = subprocess.Popen("ansible-playbook " +
"yaml/deploy_"+current_user.username+".yaml", shell=True,
stdout=subprocess.PIPE)

stdout, stderr = p.communicate()
```

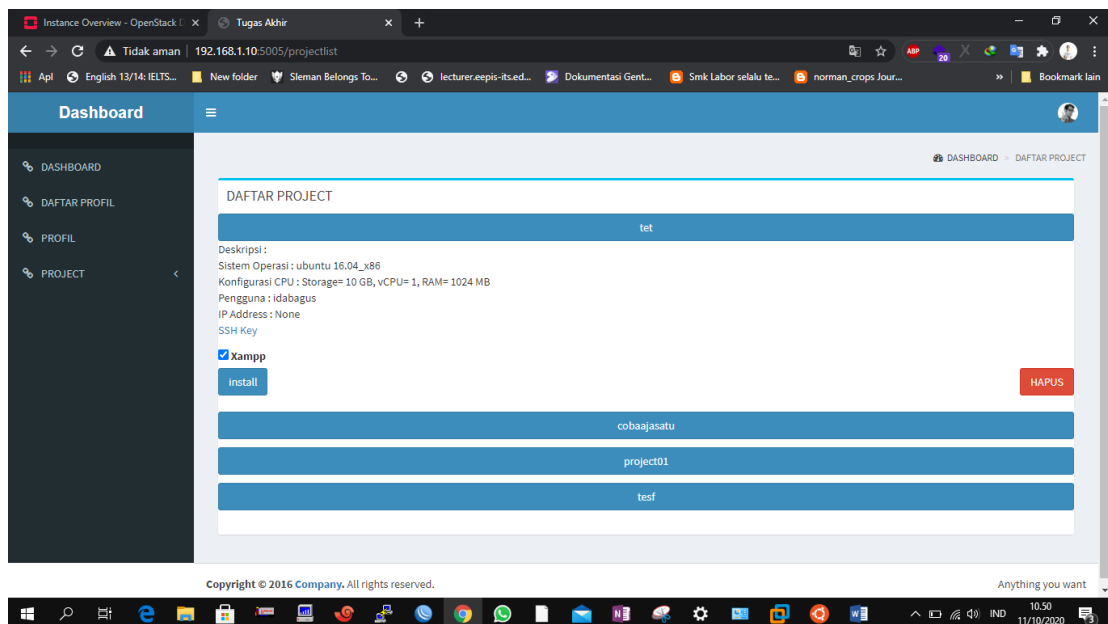
Setelah proses ansible dijalankan, proses debug akan muncul seperti gambar dibawah ini. Keluaran data yang ditampilkan oleh hasil debug pada terminal berupa json.

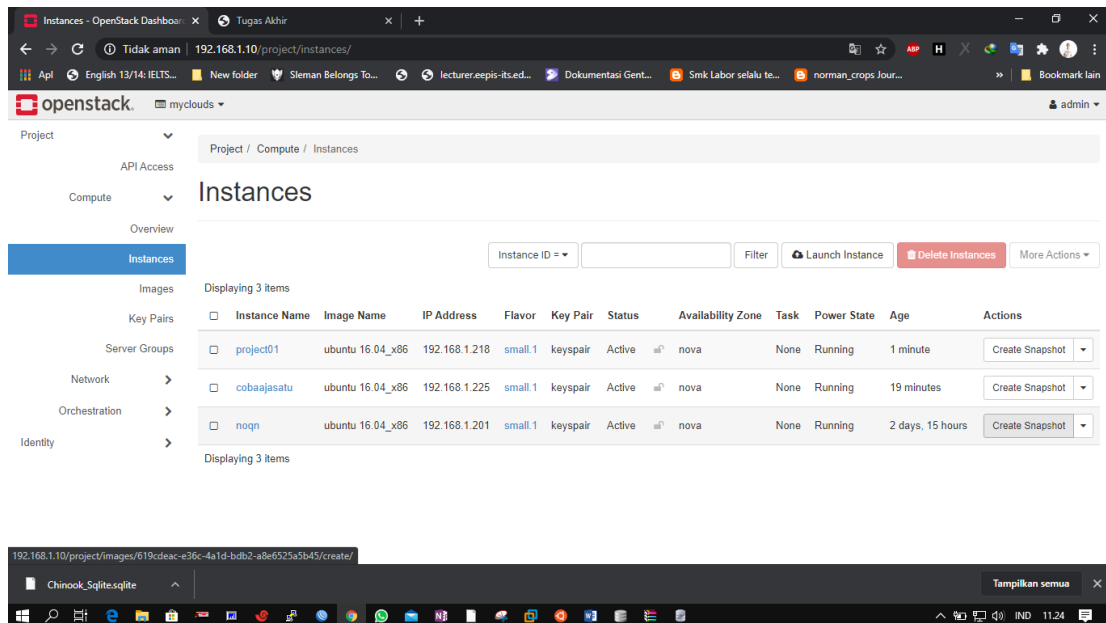


Ketika proses pembuatan instance selesai, sistem akan otomatis mengarahkan ke list instance yang sudah kita buat, seperti pada gambar berikut ini



Pada proses ini sistem hanya akan membuat instance baru dan menginstall sistem operasi saja, bila pengguna ingin menginstall paket XAMPP / LAMP, pengguna hanya perlu memberi centang pada menu xampp dan selanjutnya pilih menu install. Maka sistem akan melakukan instalasi XAMPP pada instance server pengguna secara otomatis dengan menggunakan ansible.





Pada gambar di atas tampilan dari server openstack ketika berhasil membuat sebuah instance melalui website yang digunakan, dibutuhkan waktu beberapa menit untuk membuat instance sampai instance tersebut dapat siap digunakan oleh para pengguna untuk mengembangkan aplikasinya. Untuk dapat mengakses instance yang dibuatnya, pengguna dapat melakukannya melalui SSH ke ip address instance yang dibuat.

```
def project_del():
    data = request.json['data']

    project = Project.query.filter_by(id=data["id"]).first()

    fin = open("yaml/delete.yaml", "rt")
    data = fin.read()
    data = data.replace('<name>', project.project)
    fin.close()
```

```

    fin = open("yaml/delete_"+current_user.username+".yaml", "wt")
    fin.write(data)
    fin.close()

    p = subprocess.Popen("ansible-playbook " +
        "yaml/delete_"+current_user.username+".yaml", shell=True,
        stdout=subprocess.PIPE)
    stdout, stderr = p.communicate()

    db.session.delete(project)
    db.session.commit()

    json_data = {"status": True}

    response = app.response_class(
        response=json.dumps(json_data),
        mimetype='application/json'
    )

    return response

```

Pada source code diatas digunakan untuk menghapus intance yang sudah di buat oleh pengguna, seluruh data yang ada pada instance akan di hapus secara menyeluruh. Penting untuk melakukan backup berkala dan melakukan pengecekan secara menyeluruh sebelum menghapus instance.

4.6 Implementasi kebutuhan Ansible

```

osconn = connection.Connection(auth_url=os.environ['OS_AUTH_URL'],

project_name=os.environ['OS_PROJECT_NAME'],
                                username=os.environ['OS_USERNAME'],
                                password=os.environ['OS_PASSWORD'],
                                user_domain_id="default",

```

```

project_id=os.environ['OS_PROJECT_ID'],

project_domain_id=os.environ['OS_PROJECT_DOMAIN_ID'],
                        compute_api_version='2.1',
                        verify=False)

```

Pada source code ini, merupakan fungsi untuk melakukan autentikasi ke openstack agar dapat menjalankan API yang disediakan pada server Openstack. Fungsi ini akan di eksekusi di awal sehingga sebelum menjalankan sistem yang dibuat, administartor server harus menjalankan file openrc.sh yang yang berisi data autentikasi yang dibutuhkan agar dapat berkomunikasi dengan server Openstack.

```

export                                OS_AUTH_URL=http://192.168.1.10:5000
# With the addition of Keystone we have standardized on the term
**project**
# as the entity that owns the resources.
export OS_PROJECT_ID=084aa4c02ba3453389695df1f9310f49
export OS_PROJECT_NAME="myclouds"
export OS_USER_DOMAIN_NAME="Default"
if [ -z "$OS_USER_DOMAIN_NAME" ]; then unset OS_USER_DOMAIN_NAME; fi
export OS_PROJECT_DOMAIN_ID="default"
if [ -z "$OS_PROJECT_DOMAIN_ID" ]; then unset OS_PROJECT_DOMAIN_ID;
fi
# unset v2.0 items in case set
unset OS_TENANT_ID
unset OS_TENANT_NAME
# In addition to the owning entity (tenant), OpenStack stores the
entity
# performing the action as the **user**.
export OS_USERNAME="admin"
# With Keystone you pass the keystone password.
echo  "Please enter your OpenStack Password for project
$OS_PROJECT_NAME as user $OS_USERNAME: "
read -sr OS_PASSWORD_INPUT

```

```
export OS_PASSWORD=$OS_PASSWORD_INPUT
# If your configuration has multiple regions, we set that information
here.
# OS_REGION_NAME is optional and only valid in certain environments.
export OS_REGION_NAME="RegionOne"
# Don't leave a blank variable, unset it if it was empty
if [ -z "$OS_REGION_NAME" ]; then unset OS_REGION_NAME; fi
export OS_INTERFACE=public
export OS_IDENTITY_API_VERSION=3
```

Pada baris kode diatas merupakan isi dari dari file myclouds-openr.sh. File myclouds-openr.sh diperoleh dari server openstack ketika admin ingin mengembangkan sistem openstack berjalan pada sistem pihak ketiga dengan menggunakan API yang tersedia di server openstack. File ini dijalankan sebelum program siap dijalankan agar API dapat digunakan.

```
- name: Deploy on OpenStack
  hosts: localhost
  gather_facts: false
  tasks:
    - name: Deploy an instance
      os_server:
        state: present
        name: <name>
        image: <image>
        key_name: keypair
        wait: yes
        flavor: <flav>
        auto_floating_ip: yes
        network: int-ext
        meta:
          hostname: webserver.localdomain
```

Pada source code diatas digunakan untuk membuat sebuah instance. Dikarenakan server openstack dan web aplikasi berjalan pada sistem yang sama host

yang dituju mengarah pada localhost. Localhost disini merupakan ip server openstack. Selanjutnya menambahkan perintah yang akan di eksekusi pada baris task, baris task berisi kebutuhan yang di perlukan dalam membuat instance baru dengan menambahkan baris API yang terhubung ke server openstack untuk mendeploy instance baru, perintah tersebut bernama os_server.

```
- name: Deploy on OpenStack
  hosts: localhost
  gather_facts: false
  tasks:
    - name: Deploy an instance
      os_server:
        state: present
        name: tesf
        image: ubuntu 16.04_x86
        key_name: keyspair
        wait: yes
        flavor: small.1
        auto_floating_ip: yes
        network: int-ext
        meta:
          hostname: webserver.localdomain
```

Pada baris kode ini merupakan contoh perintah yang dibuat secara otomatis dan langsung di eksekusi oleh ansible dengan mengirimkan perintah tersebut ke server openstack. Selanjutnya openstack membuat instance baru dengan nama instance dan spesifikasi server sesuai baris kode yang dikirimkan ansible ke API openstack.

```
- name: remove an instance
  hosts: localhost
  tasks:
    - name: remove an instance
      os_server:
```



```
name: <name>
state: absent
```

Pada baris kode diatas merupakan barisk kode yang digunakan untuk menghapus instance server yang dibuat oleh para pengguna bila sudah tidak dibutuhkan. Baris kode memanggil API `os_server` dan mencari nama instance yang akan dihapus, selanjutnya akan akan kembali di eksekusi oleh baris kode ... dan bari kode `db.session.delete(project)` langsung menghapus isi dalam database sesuai nama `project` yang dibuat.

File `apache.yaml`

```
- name: install apache & php
  remote_user: ubuntu
  hosts: all
  become: true
  become_user: root
  gather_facts: true
  tasks:
    - name: Update apt-get repo and cache
      apt: update_cache=yes force_apt_get=yes cache_valid_time=3600
    - name: "Install apache2"
      package: name=apache2 state=present
    - name: "Install apache2-php5"
      package: name=libapache2-mod-php state=present
    - name: "Install php-cli"
      package: name=php-cli state=present
    - name: "Install php-mcrypt"
      package: name=php-mcrypt state=present
    - name: "Install php-gd"
      package: name=php-gd state=present
```

Pada baris kode ... merupakan baris kode yang berguna untuk menginstall web server yang akan dieksekusi oleh ansible. Web server yang diinstall secara default akan menginstall Apache dan PHP untuk kebutuhan dalam membangun web secara mandiri pada server instance yang sudah dibuat. Untuk dapat dieksekusi ke instance yang dituju, pertama buat nama perintah dan username instance lalu `become_user: root` Merupakan baris kode untuk ansible mengeksekusi task yang diberikan dengan akses root. Selanjutnya dibuat tugas untuk melakukan instalasi paket `apache2` dan `PHP`

File `mysql.yaml`

```
- name: Install MySQL for production ready server
  user: ubuntu
  hosts: all
  become: True
  become_user: root
  vars:
    MySQL_root_pass: root
  tasks:
    - name: Update apt-get repo and cache
      apt: update_cache=yes force_apt_get=yes cache_valid_time=3600
    - name: Set MySQL root password before installing
      debconf:
        name="mysql-server"
        question="mysql-server/root_password"
        value="{{MySQL_root_pass | quote}}"
        vtype="password"
    - name: Confirm MySQL root password before installing
      debconf:
        name="mysql-server"
        question="mysql-server/root_password_again"
        value="{{MySQL_root_pass | quote}}"
        vtype="password"
    - name: test1
      apt:
        package={{ item }}
        state=present
        force=yes
      update_cache=yes
      cache_valid_time=3600
      when: ansible_os_family == "Debian"
      with_items:
```

```

- mysql-server
- mysql-client
- python-mysqldb

- name: Deletes anonymous MySQL server user for localhost
  mysql_user: user="" state="absent" login_password="{{
MySQL_root_pass }}" login_user=root
- name: Secures the MySQL root user
  mysql_user: user="root" password="{{ MySQL_root_pass }}"
host="{{ item }}" login_password="{{ MySQL_root_pass }}"
login_user=root
  with_items:
- 127.0.0.1
- localhost
- ::1
  - "{{ ansible_fqdn }}"
- name: Removes the MySQL test database
  mysql_db: db=test state=absent login_password="{{
MySQL_root_pass }}" login_user=root

```

Pada baris kode merupakan perintah berisi tugas untuk di eksekusi oleh ansible yang bertujuan untuk melakukan instalasi paket database MySQL pada instance, pada baris variabel, atau yang didefinisikan vars ditambahkan perintah password default untuk mengakses database, lalu ansible akan mengeksekusi tugas – tugas yang sudah dibuat. Dikarenakan di awal sudah melakukan instalasi web server sebelumnya maka pada task ini perlu ditambahkan perintah untuk melakukan update paket – paket yang ada pada instance, baru selanjutnya melakukan eksekusi menginstalasi paket basis data MySQL.

File lamp.yaml

```

- name: install LAMP Stack
  hosts: all
  remote_user: ubuntu
  become: true

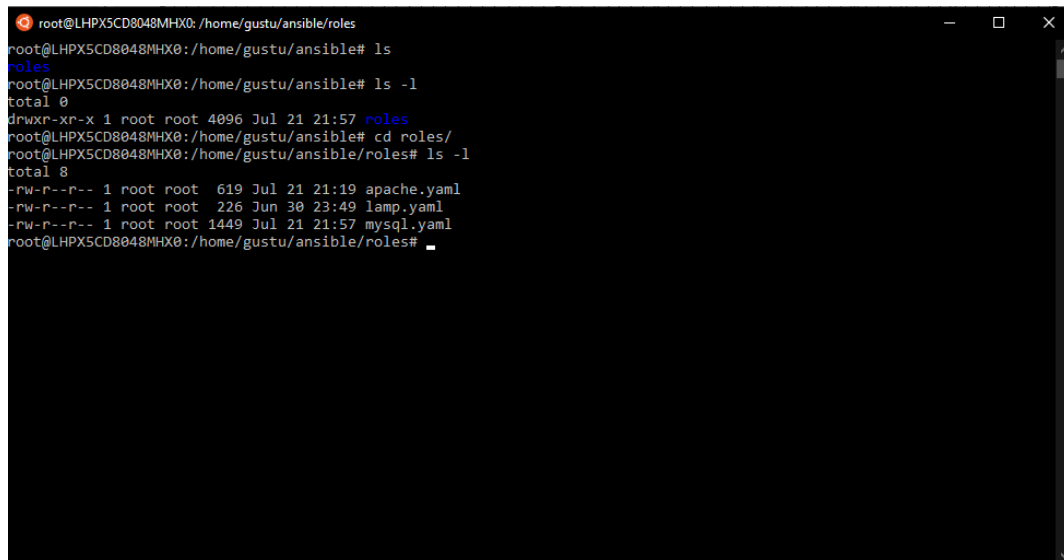
```

```
become_user: root
gather_facts: true

- name: Include Apache
  import_playbook: apache.yaml

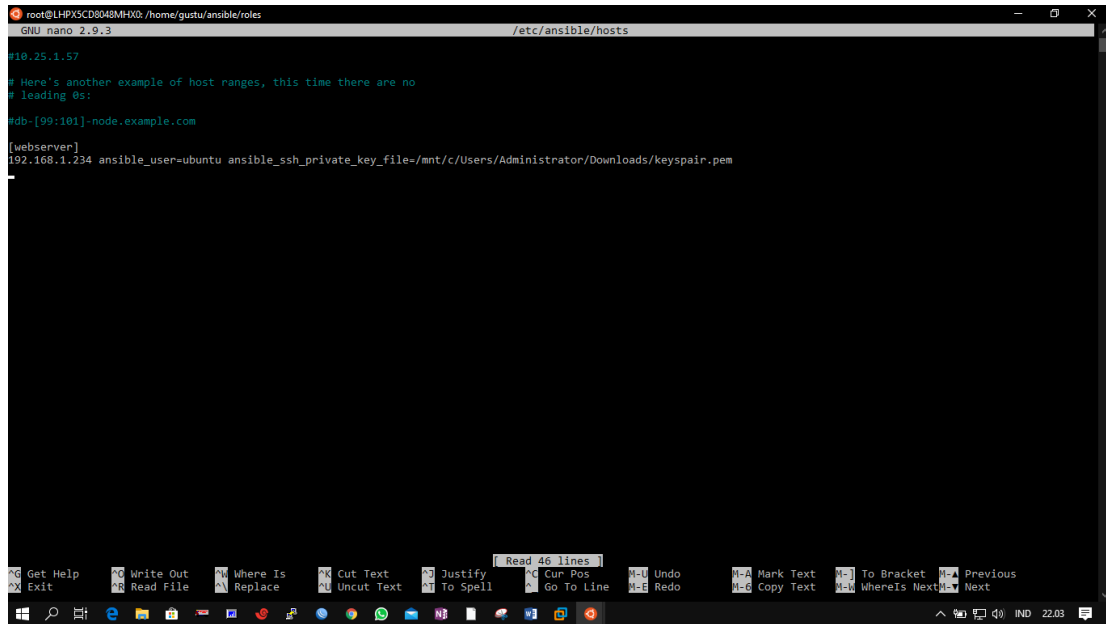
- name: Include MySQL
  import_playbook: mysql.yaml
```

Pada tabel lamp.yaml merupakan baris kode untuk untuk melakukan eksekusi pada kedua file yaml yang sebelumnya dibuat. pada file ini tidak perlu di tambahkan task, dikarenakan file lampp.yaml akan memberi perintah untuk melakukan instalasi sesuai file yang sudah di buat pada tabel ... dan tabel ...



```
root@LHPX5CD8048MHX0: /home/gustu/ansible/roles
root@LHPX5CD8048MHX0:/home/gustu/ansible# ls
roles
root@LHPX5CD8048MHX0:/home/gustu/ansible# ls -l
total 0
drwxr-xr-x 1 root root 4096 Jul 21 21:57 roles
root@LHPX5CD8048MHX0:/home/gustu/ansible# cd roles/
root@LHPX5CD8048MHX0:/home/gustu/ansible/roles# ls -l
total 8
-rw-r--r-- 1 root root 619 Jul 21 21:19 apache.yaml
-rw-r--r-- 1 root root 226 Jun 30 23:49 lamp.yaml
-rw-r--r-- 1 root root 1449 Jul 21 21:57 mysql.yaml
root@LHPX5CD8048MHX0:/home/gustu/ansible/roles#
```

Pada gambar ... merupakan tampilan files yaml, pada direktori ansible ditambahkan sebuah folder bernama roles, jadi folder roles ini nantinya akan di eksekusi oleh lamp.yaml yang sudah berisikan perintah tugas yang akan di install yaitu Apache2, PHP dan MySQL.



```
root@LHPX5CD8048MHX0: /home/gustu/ansible/roles
GNU nano 2.9.3 /etc/ansible/hosts
#18.25.1.57
# Here's another example of host ranges, this time there are no
# leading 0s:
#db-[99:101]-node.example.com

[webserver]
192.168.1.234 ansible_user=ubuntu ansible_ssh_private_key_file=/mnt/c/Users/Administrator/Downloads/keyspair.pem
```

Pada gambar diatas, sistem membuat alamat ip address dibuat pada file hosts digunakan untuk mengenali server mana yang akan dieksekusi ketika file yaml dijalankan, serta menyertakan username dan password dari instance yang telah dibuat secara default melalui openstack.

```
[webserver]
192.168.1.234                                ansible_user=ubuntu
ansible_ssh_private_key_file=/mnt/c/Users/Administrator/Downloads/k
eyspair.pem
```

Pada baris kode merupakan langkah untuk ansible dapat meremote instance pengguna yang akan menginstall paket LAMPP yang selanjut akan akan di install dan di eksekusi oleh ansible. Baris kode ini ditambahkan di /etc/ansible/hosts, lalu menambahkan [webserver] sebagai grup jaringan yang akan di eksekusi. IP address 192.168.1.234 merupakan IP instance, dan untuk user secara default adalah ubuntu, bila sistem operasi yang dipilih adalah ubuntu. Dikarenakan ansible bekerja dengan sistem SSH maka membutuhkan kunci SSH dari setiap instance, maka ditambahkan lokasi file kunci ssh setiap instance.

```
root@LHPX5CD8048MHX0: /home/gustu/ansible/roles
TASK [Set MySQL root password before installing] *****
changed: [192.168.1.218]

TASK [Confirm MySQL root password before installing] *****
changed: [192.168.1.218]

TASK [test1] *****
[DEPRECATION WARNING]: Invoking "apt" only once while using a loop via squash_actions is deprecated. Instead of using a
loop to supply multiple items and specifying "package: "{{ item }}"", please use "package: ['mysql-server', 'mysql-
client', 'python-mysqldb']" and remove the loop. This feature will be removed in version 2.11. Deprecation warnings can
be disabled by setting deprecation_warnings=False in ansible.cfg.
ok: [192.168.1.218] => (item=['mysql-server', 'mysql-client', 'python-mysqldb'])

TASK [Deletes anonymous MySQL server user for localhost] *****
[WARNING]: Module did not set no_log for update_password
ok: [192.168.1.218]

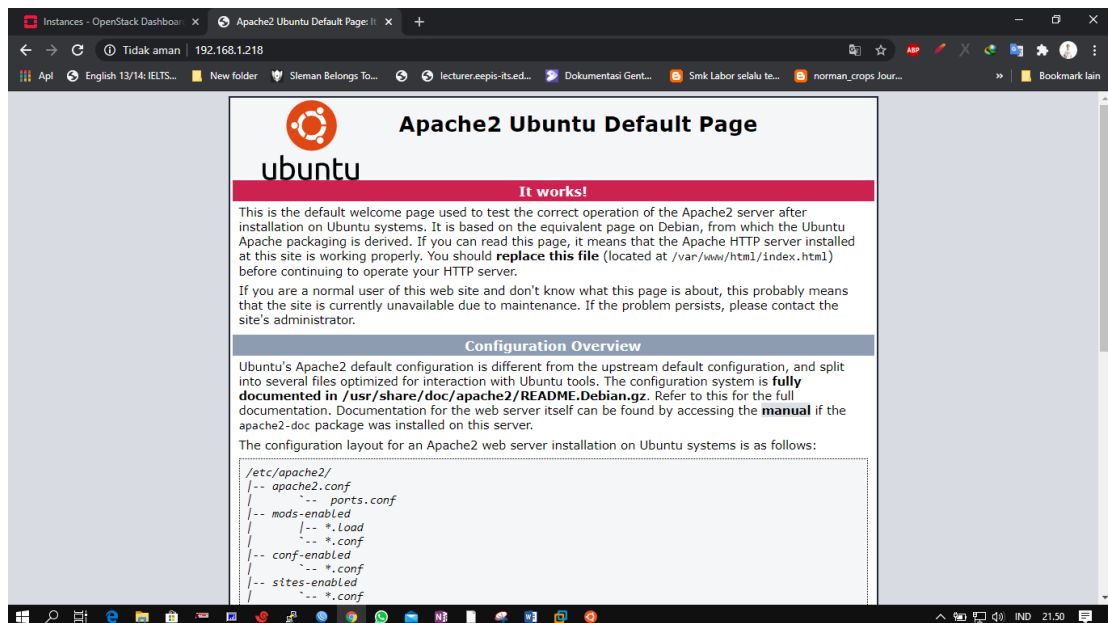
TASK [Secures the MySQL root user] *****
changed: [192.168.1.218] => (item=127.0.0.1)
ok: [192.168.1.218] => (item=localhost)
changed: [192.168.1.218] => (item=:1)
changed: [192.168.1.218] => (item=project01)

TASK [Removes the MySQL test database] *****
ok: [192.168.1.218]

PLAY RECAP *****
192.168.1.218      : ok=14  changed=3  unreachable=0  failed=0  skipped=0  rescued=0  ignored=0

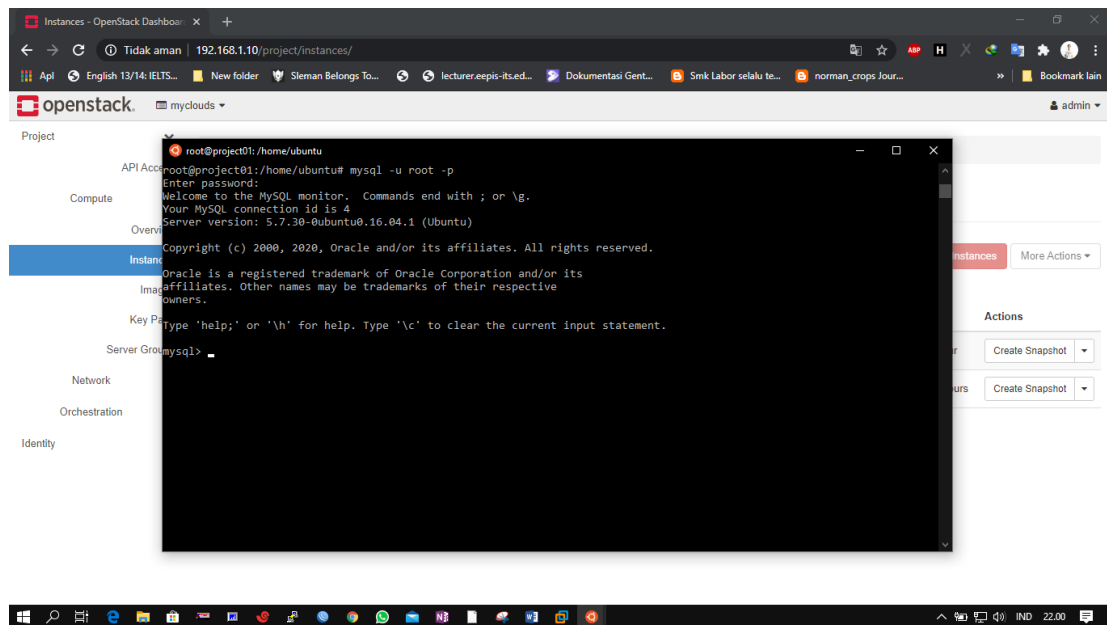
root@LHPX5CD8048MHX0: /home/gustu/ansible/roles#
```

Pada gambar merupakan hasil setelah menjalankan perintah untuk melakukan instalasi lamp.yaml. Bila sukses melakukan instalasi pada instance akan muncul pada gambar di atas semua hasil debug tidak ada kegagalan ketika instalasi.



Untuk menguji webserver yang telah kita buat dapat mengakses ip address yang di miliki instance melalui web browser, bila muncul seperti gambar di atas web server

yang di buat dinyatakan berhasil melakukan instalasi. Selanjutnya bila pengguna ingin menambahkan web yang dibuatnya dalam kasus ini diambil menggunakan bahasa pemrograman PHP, pengguna hanya perlu menyalin web yang sudah dibuat ke /var/www/html.



Pada gambar ... merupakan tampilan dari database yang sudah terinstall pada instance, pengguna dapat melakukan pengecekan kembali dengan meremote instance server yang sudah dibuat dengan menggunakan SSH. Serta melakukan perintah `mysql -u root -p` untuk mengecek ataupun untuk membuat project database di dalam instance server openstack.

4.7 Pengujian Sistem

Pada penelitian ini pengujian sistem “Sistem Manajemen Layanan Web Berbasis Platform as a Service (PaaS) Dengan API Openstack”, pengujian akan dilakukan dengan metode black box testing dan performance testing. Pengujian dilakukan dengan tujuan aplikasi yang dibuat dapat berjalan sesuai kebutuhan pengguna.

4.7.1 Black Box Testing

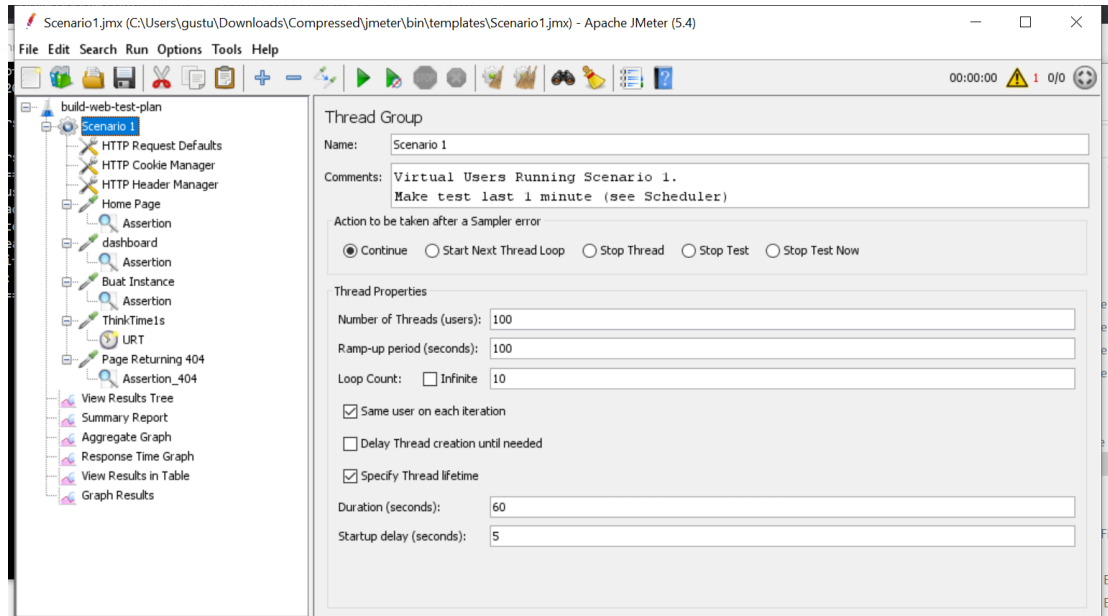
Black Box Testing merupakan pengujian fungsional dari sistem yang dibuat. Tujuan dari pengujian fungsional adalah untuk memvalidasi perilaku perangkat lunak yang dibuat terhadap fungsionalitas kebutuhan para pengguna. Berikut merupakan tabel pengujian menggunakan black box :

Kode Uji	Butir Uji	Teknik Pengujian	Hasil Pengujian
OS1	Pengguna baru melakukan pendaftaran untuk dapat masuk ke dalam sistem	Black Box	Diterima
OS2	Pengguna melakukan login ke dalam sistem	Black Box	Diterima
OS3	Admin dan pengguna memiliki antarmuka yang berbeda.	Black Box	Diterima
OS4	Pengguna dapat membuat instance (server virtual) baru untuk pertama kali,	Black Box	Diterima
OS5	Pengguna dapat membuat kembali instance (server virtual) baru.	Black Box	Diterima
OS6	Pengguna dapat memilih sendiri kebutuhan spesifikasi prosessor, RAM, penyimpanan dan sistem Operasi yang akan di buat	Black Box	Diterima
OS7	Pengguna dapat melihat list instance yang telah dibuat.	Black Box	Diterima
OS8	Pengguna dapat mendownload SSH Key, untuk mengakses instance melalui SSH.	Black Box	Diterima
OS9	Pengguna dapat mengakses instance melalui SSH dengan menggunakan SSHkey.	Black Box	Diterima
OS10	Pengguna dapat melihat IP publik dari instance melalui web	Black Box	Tidak diterima

OS11	Pengguna dapat melakukan instalasi paket Apache, PHP dan MySQL (LAMPP) secara otomatis.	Black Box	Tidak diterima
OS12	Pengguna dapat mengganti kata sandi untuk masuk ke dalam sistem	Black Box	Diterima
	Penggu tidak dapat masuk ke dashboard bila tidak melakukan login terlebih dahulu atau belum terdaftar dalam web admin	Black Box	Diterima
OS13	Pengguna dapat keluar atau logout dari sistem pada menu yang sudah disediakan.	Black Box	Diterima

4.7.2 Performance Testing

Pada pengujian *Performance Testing* merupakan lanjutan dari pengujian fungsionalitas sistem yang sudah dibuat. Performance Testing bertujuan memvalidasi “kecepatan” sistem yang dibuat. Kecepatan dalam hal ini peneliti melakukan pengujian dengan mengukur aspek waktu respon sistem. Peneliti melakukan pengujian kinerja dalam ruang lingkup yang mendekati ruang lingkup sebenarnya. Pengujian kinerja sistem ini dicapai dengan melakukan simulasi menggunakan perangkat lunak aplikasi *Jmeter*.



Pada gambar merupakan tampilan aplikasi untuk melakukan pengujian performa pada web server, dengan skenario web server akan di akses oleh 100 pengguna per detiknya dengan total jumlah sample sebanyak 1000.

Hasil tersebut di dapat dari :

$$delay = \frac{seconds}{users}$$

Jadi dalam waktu 1 detik terdapat 100 pengguna melakukan akses ke web server. Pada Ramp-up period memberitahu jmeter berapa panjang jeda waktu sebelum pengguna selanjutnya memulai request ke web server. Sehingga bila ada 100 pengguna dan peningkatan waktu selama 100 detik maka didapat jeda antara pengguna awal dengan pengguna selanjutnya ada 1 detik.

$$total\ sample = user \times loop\ count$$

Sedangkan untuk sampel yang dibutuhkan dalam pengujian diperoleh dari banyaknya pengguna dikali dengan jumlah request yang di lakukan oleh setiap pengguna. sehingga setiap pengguna melakukan 10 kali request maka didapat hasil data sampel sebanyak 1000 data.

HTTP Header Manager

Name: HTTP Header Manager

Comments:

Name:	Value
User-Agent	Mozilla/5.0 (Macintosh; Intel Mac OS X 10.9; rv:48.0) Gecko/20100101 Firefox/48.0
Accept	text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language	fr,en-US;q=0.7,en;q=0.3
Accept-Encoding	gzip, deflate

Setelah menentukan jumlah pengguna yang akan melakukan pengujian. Selanjutnya dibuat HTTP Header manager yang digunakan sebagai bentuk permintaan dan respon yang akan di lakukan oleh banyaknya pengguna.

HTTP Request

Name: Home Page

Comments:

Basic Advanced

Web Server

Protocol [http]: http Server Name or IP: 192.168.1.10 Port Number: 5005

HTTP Request

GET Path: / Content encoding:

☐ Redirect Automatically ☒ Follow Redirects ☒ Use KeepAlive ☐ Use multipart/form-data ☐ Browser-compatible headers

Parameters Body Data Files Upload

Send Parameters With the Request:

Name:	Value	URL Encode?	Content-Type	Include Equals?
-------	-------	-------------	--------------	-----------------

Detail Add Add from Clipboard Delete Up Down

Pada gambar .. dilakukan request pada web server yang akan dilakukan pengujian dengan menambahkan alamat web server yang akan diuji. Pada kolom web server ditambahkan alamat website atau bila menggunakan alamat IP dimasukkan alamat IP Address website dan port yang di gunakan. Selanjutnya pada kolom HTTP Request ditambahkan halaman yang dituju, pada penelitian ini halaman yang di uji adalah :

1. http://192.168.1.10:5005/
2. http://192.168.1.10:5005/dashboard

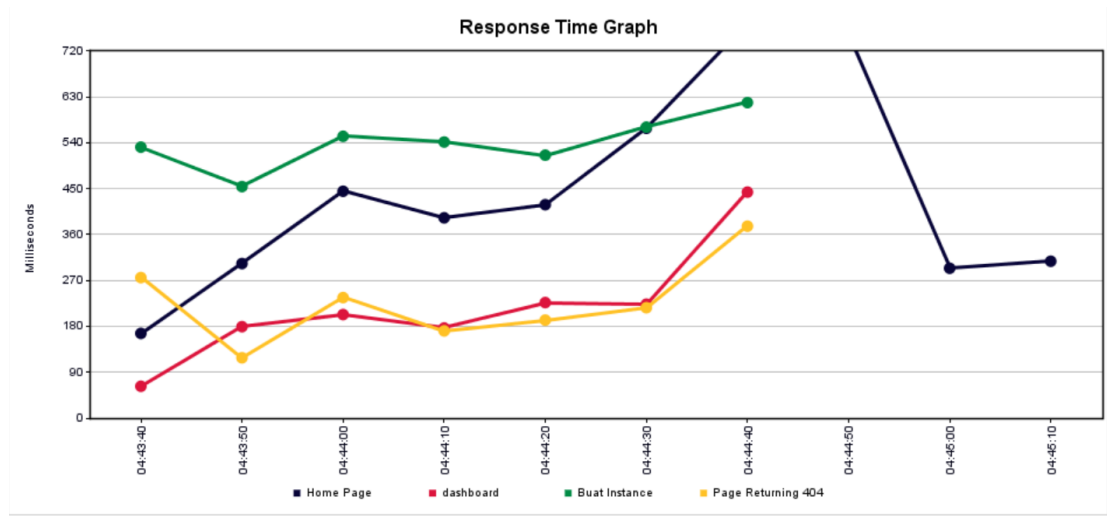
3. <http://192.168.1.10:5005/buatbaru>

4. <http://192.168.1.10:5005/test1>

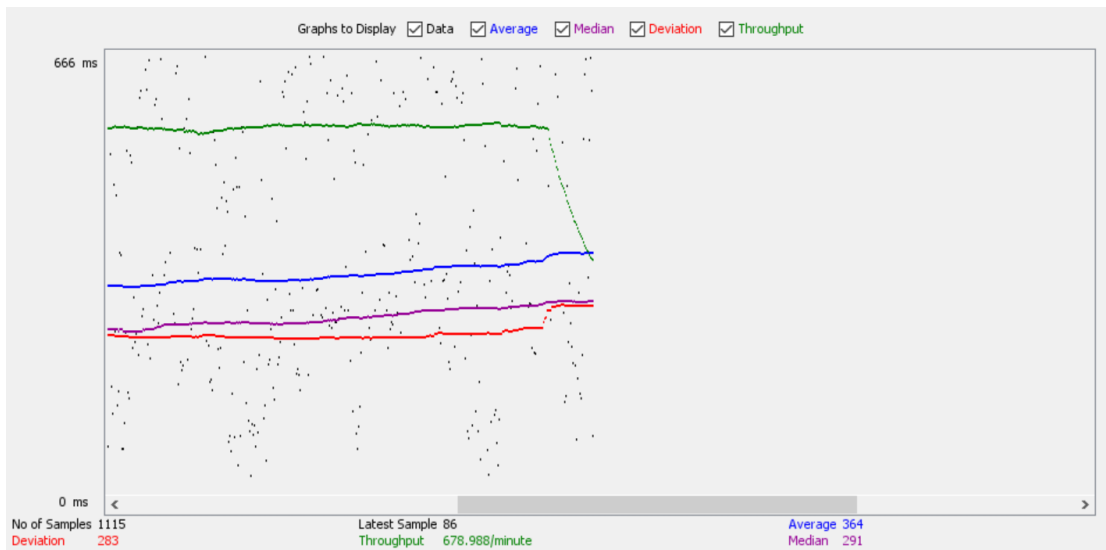
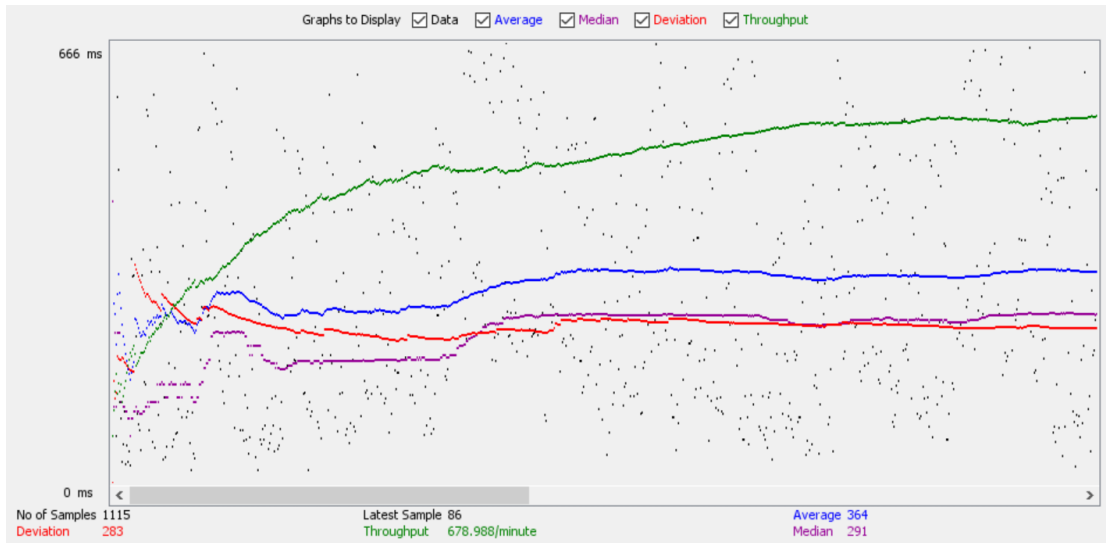
pada link <http://192.168.1.10:5005/test1> digunakan untuk menguji error handling 404 pada web server

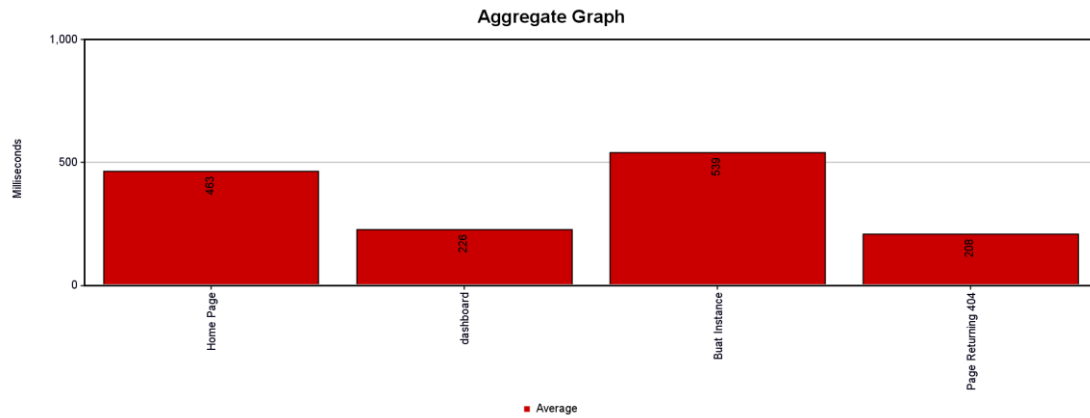
Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
Home Page	314	463	49	2174	292.54	0.00%	3.2/sec	159.19	0.97	51151.0
dashboard	272	226	27	2971	262.31	0.00%	4.5/sec	54.81	1.42	12429.0
Buat Instance	268	539	241	1511	171.88	0.00%	4.5/sec	68.40	1.41	15626.0
Page Returnin...	261	208	20	2659	227.84	0.00%	4.5/sec	1.72	1.39	393.0
TOTAL	1115	364	20	2971	283.84	0.00%	11.3/sec	235.22	3.51	21284.7

Pada gambar tabel merupakan hasil pengujian performa web server yang dibuat,



Pada gambar merupakan hasil dari pengujian waktu respon web server, dapat dilihat pada gambar waktu yang dibutuhkan pada pembuatan instance cukup lebih lama dibandingkan dalam mengakses alamat home page, dashboard atau page error, dikarenakan dalam alamat buat instance ini melakukan autentikasi pada server openstack. Sehingga membuat waktu respon yang dibutuhkan dalam menampilkan halaman membuat sebuah instance baru menjadi sedikit lebih lama melebihi 720 milliseconds dalam suatu waktu.





Pada gambar aggregate graph menunjukkan hasil dari rata – rata waktu yang dibutuhkan dalam mengakses setiap halaman yang di uji. Dari hasil yang ditampilkan rata – rata waktu dalam membuat sebuah instance mendapatkan waktu rata – rata yang cukup tinggi melebihi 500 Milliseconds dikarenakan setiap kali mengakses halaman buat baru, sistem akan selalu melakukan autentikasi API dan melakukan pengecekan menjadikan pada halaman buat instance menjadi sedikit lebih lama.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, dapat diambil beberapa kesimpulan sebagai berikut :

1. Sistem yang dibangun dapat diimplementasikan dengan Sistem Cloud computing secara otomatisasi dengan menggunakan ansible dengan prinsip kerja sistem membuat virtual server secara otomatis dan melakukan instalasi paket LAMP secara otomatis.
2. Berdasarkan hasil pengujian yang dilakukan didapatkan hasil bahwa membangun sebuah server dengan didukung oleh teknologi otomatisasi membuat waktu yang dibutuhkan lebih efisien dalam membuat sebuah server yang siap digunakan serta secara otomatis terinstal web server beserta sebuah database.
3. Dari hasil pengujian performa mendapatkan hasil yang sangat baik, server berjalan dengan baik ketika dikases oleh 100 pengguna per detik. Namun terjadi sedikit penurunan performa per 200 milliseconds ketika web managemen akan melakukan pengambilan data pada server openstack, tetapi tidak mempengaruhi performa server secara keseluruhan.

5.2 Saran

Untuk pengembangan sistem lebih lanjut, penulis ingin memberikan beberapa saran sebagai berikut :

1. Dikarenakan sistem ini berbasis cloud computing ada baiknya bila dikembangkan lebih lanjut dapat menambahkan fitur keamanan data pengguna, serta dapat dikembangkan secara dinamis untuk kebutuhan penyimpanan, jaringan, maupun untuk komputasinya.
2. Dapat menambahkan monitoring instance untuk dapat melihat kesehatan setiap instance / virtual server untuk pengguna. Jadi pengguna pun dapat memantau

keadaan virtual servernya sendiri. Hal tersebut dapat menjadi pertimbangan untuk penggunaanya.

DAFTAR PUSTAKA

- Xia, Q., Lan, Y., & Xiao, L. (2015). A Heuristic Adaptive Threshold Algorithm on IaaS Clouds.
- Anggeriana, H. (2011). *Cloud Computing*.
- Ansible. (t.thn.). *ansible.com*. Diambil kembali dari <http://docs.ansible.com/ansible/latest/index.html>:
<http://docs.ansible.com/ansible/latest/index.html>
- Collings, T., & Kurt, W. (2015). *Duties of the System Administrator. In Red Hat Linux Networking System Administrator (chap. 1)*.
- Corporation, Exabyte. (2004). *The Basic Backup Guide*. cororado.
- Doty, S. (2008). *Python Basics*.
- Gerald D. Everett, R. M. (2007). *Software Testing: Testing Across the Entire Software Development Life Cycle 1st Edition*. Canada: IEEE Press.
- Lei Xiaojiang, S. Y. (2013). The Design and Implementation of Resource Monitoring for Cloud Computing Service Platform.
- Liyun Zuo, L. S. (2015). A Dynamic Self-adaptive Resource-Load Evaluation Method in Cloud Computing.
- Masse, M. (2012). REST API. Dalam *REST API Design Rulebook* (hal. 5). America: O'Reilly Media, Inc.
- Mulyana, E. (2017, 10 05). *Pengantar Openstack*. Diambil kembali dari <https://eueung.gitbooks.io>: <https://eueung.gitbooks.io/buku-komunitas-sdn-rg/content/index.html>
- Nicole Ng, H. C. (2011). An Adaptive Threshold Method to Address Routing Issues in Delay-Tolerant Networks.
- Saleh, Y. W. (2013). Adaptive Resource Management for Service Workflows in Cloud Environments.
- Sosinsky, B. (2011). *Cloud Computing Bible*. Canada: Wiley Publishing, Inc.

LAMPIRAN

URL	Latency	IdleTime	Connect	responseCode	success
http://192.168.1.10:5005/	221	0	117	200	true
http://192.168.1.10:5005/dashboard	178	0	89	200	true
http://192.168.1.10:5005/	161	0	48	200	true
http://192.168.1.10:5005/dashboard	62	0	32	200	true
http://192.168.1.10:5005/buatbaru	240	0	8	200	true
http://192.168.1.10:5005/buatbaru	319	0	49	200	true
http://192.168.1.10:5005/	67	0	21	200	true
http://192.168.1.10:5005/dashboard	40	0	9	200	true
http://192.168.1.10:5005/buatbaru	245	0	33	200	true
http://192.168.1.10:5005/test	112	0	64	404	true
http://192.168.1.10:5005/test	58	0	27	404	true
http://192.168.1.10:5005/	87	0	35	200	true
http://192.168.1.10:5005/	111	0	44	200	true
http://192.168.1.10:5005/dashboard	153	0	83	200	true
http://192.168.1.10:5005/dashboard	128	0	70	200	true
http://192.168.1.10:5005/	201	0	116	200	true
http://192.168.1.10:5005/dashboard	65	0	19	200	true
http://192.168.1.10:5005/buatbaru	313	0	52	200	true
http://192.168.1.10:5005/test	90	0	49	404	true
http://192.168.1.10:5005/buatbaru	304	0	27	200	true
http://192.168.1.10:5005/	122	0	47	200	true
http://192.168.1.10:5005/dashboard	62	0	17	200	true
http://192.168.1.10:5005/	91	0	41	200	true
http://192.168.1.10:5005/buatbaru	289	0	46	200	true
http://192.168.1.10:5005/dashboard	68	0	19	200	true
http://192.168.1.10:5005/buatbaru	1107	0	23	200	true
http://192.168.1.10:5005/buatbaru	267	0	36	200	true
http://192.168.1.10:5005/test	65	0	24	404	true
http://192.168.1.10:5005/test	90	0	49	404	true
http://192.168.1.10:5005/	92	0	45	200	true
http://192.168.1.10:5005/	91	0	43	200	true

http://192.168.1.10:5005/dashboard	267	0	220	200	true
http://192.168.1.10:5005/	236	0	6	200	true
http://192.168.1.10:5005/dashboard	54	0	12	200	true
http://192.168.1.10:5005/test	69	0	22	404	true
http://192.168.1.10:5005/	50	0	31	200	true
http://192.168.1.10:5005/dashboard	47	0	19	200	true
http://192.168.1.10:5005/buatbaru	250	0	33	200	true
http://192.168.1.10:5005/test	65	0	19	404	true
http://192.168.1.10:5005/test	111	0	50	404	true
http://192.168.1.10:5005/	63	0	27	200	true
http://192.168.1.10:5005/buatbaru	264	0	23	200	true
http://192.168.1.10:5005/	50	0	12	200	true
http://192.168.1.10:5005/dashboard	75	0	26	200	true
http://192.168.1.10:5005/buatbaru	326	0	42	200	true
http://192.168.1.10:5005/dashboard	1079	0	1048	200	true
http://192.168.1.10:5005/dashboard	78	0	48	200	true
http://192.168.1.10:5005/	67	0	12	200	true
http://192.168.1.10:5005/dashboard	59	0	41	200	true
http://192.168.1.10:5005/buatbaru	268	0	20	200	true
http://192.168.1.10:5005/buatbaru	373	0	28	200	true
http://192.168.1.10:5005/buatbaru	355	0	29	200	true
http://192.168.1.10:5005/buatbaru	265	0	43	200	true
http://192.168.1.10:5005/test	27	0	16	404	true
http://192.168.1.10:5005/	51	0	33	200	true
http://192.168.1.10:5005/dashboard	46	0	26	200	true
http://192.168.1.10:5005/test	28	0	11	404	true
http://192.168.1.10:5005/	27	0	5	200	true
http://192.168.1.10:5005/test	51	0	29	404	true
http://192.168.1.10:5005/dashboard	77	0	21	200	true
http://192.168.1.10:5005/	52	0	26	200	true
http://192.168.1.10:5005/	78	0	20	200	true
http://192.168.1.10:5005/dashboard	23	0	8	200	true
http://192.168.1.10:5005/dashboard	109	0	46	200	true
http://192.168.1.10:5005/test	78	0	46	404	true
http://192.168.1.10:5005/test	88	0	6	404	true
http://192.168.1.10:5005/buatbaru	544	0	8	200	true

http://192.168.1.10:5005/test	107	0	33	404	true
http://192.168.1.10:5005/buatbaru	348	0	40	200	true
http://192.168.1.10:5005/buatbaru	355	0	46	200	true
http://192.168.1.10:5005/	95	0	47	200	true
http://192.168.1.10:5005/test	111	0	57	404	true
http://192.168.1.10:5005/buatbaru	325	0	30	200	true
http://192.168.1.10:5005/dashboard	101	0	46	200	true
http://192.168.1.10:5005/	108	0	50	200	true
http://192.168.1.10:5005/	111	0	50	200	true
http://192.168.1.10:5005/dashboard	192	0	66	200	true
http://192.168.1.10:5005/dashboard	226	0	104	200	true
http://192.168.1.10:5005/buatbaru	455	0	70	200	true
http://192.168.1.10:5005/	180	0	115	200	true
http://192.168.1.10:5005/buatbaru	422	0	126	200	true
http://192.168.1.10:5005/buatbaru	374	0	80	200	true
http://192.168.1.10:5005/dashboard	166	0	108	200	true
http://192.168.1.10:5005/test	178	0	56	404	true
http://192.168.1.10:5005/	154	0	72	200	true
http://192.168.1.10:5005/test	215	0	118	404	true
http://192.168.1.10:5005/test	210	0	118	404	true
http://192.168.1.10:5005/test	188	0	98	404	true
http://192.168.1.10:5005/dashboard	100	0	48	200	true
http://192.168.1.10:5005/buatbaru	598	0	119	200	true
http://192.168.1.10:5005/	149	0	95	200	true
http://192.168.1.10:5005/	118	0	45	200	true
http://192.168.1.10:5005/	132	0	60	200	true
http://192.168.1.10:5005/	107	0	53	200	true
http://192.168.1.10:5005/	237	0	84	200	true
http://192.168.1.10:5005/dashboard	98	0	70	200	true
http://192.168.1.10:5005/dashboard	113	0	73	200	true
http://192.168.1.10:5005/dashboard	86	0	50	200	true
http://192.168.1.10:5005/test	70	0	33	404	true
http://192.168.1.10:5005/dashboard	36	0	3	200	true
http://192.168.1.10:5005/dashboard	42	0	3	200	true
http://192.168.1.10:5005/buatbaru	474	0	58	200	true
http://192.168.1.10:5005/	136	0	51	200	true

http://192.168.1.10:5005/dashboard	25	0	14	200	true
http://192.168.1.10:5005/test	40	0	14	404	true
http://192.168.1.10:5005/test	78	0	40	404	true
http://192.168.1.10:5005/	14	0	4	200	true
http://192.168.1.10:5005/buatbaru	432	0	44	200	true
http://192.168.1.10:5005/dashboard	77	0	21	200	true
http://192.168.1.10:5005/	37	0	15	200	true
http://192.168.1.10:5005/buatbaru	508	0	52	200	true
http://192.168.1.10:5005/buatbaru	546	0	47	200	true
http://192.168.1.10:5005/dashboard	88	0	26	200	true
http://192.168.1.10:5005/buatbaru	598	0	52	200	true
http://192.168.1.10:5005/buatbaru	706	0	52	200	true
http://192.168.1.10:5005/buatbaru	450	0	24	200	true
http://192.168.1.10:5005/	61	0	11	200	true
http://192.168.1.10:5005/buatbaru	317	0	27	200	true
http://192.168.1.10:5005/buatbaru	275	0	13	200	true
http://192.168.1.10:5005/dashboard	137	0	88	200	true
http://192.168.1.10:5005/test	96	0	40	404	true
http://192.168.1.10:5005/	28	0	16	200	true
http://192.168.1.10:5005/buatbaru	309	0	54	200	true
http://192.168.1.10:5005/test	89	0	57	404	true
http://192.168.1.10:5005/dashboard	34	0	12	200	true
http://192.168.1.10:5005/	36	0	8	200	true
http://192.168.1.10:5005/dashboard	72	0	37	200	true
http://192.168.1.10:5005/buatbaru	279	0	22	200	true
http://192.168.1.10:5005/test	130	0	55	404	true
http://192.168.1.10:5005/test	152	0	68	404	true
http://192.168.1.10:5005/test	193	0	79	404	true
http://192.168.1.10:5005/buatbaru	274	0	9	200	true
http://192.168.1.10:5005/test	241	0	58	404	true
http://192.168.1.10:5005/test	230	0	120	404	true
http://192.168.1.10:5005/test	243	0	109	404	true
http://192.168.1.10:5005/test	279	0	138	404	true
http://192.168.1.10:5005/test	314	0	174	404	true
http://192.168.1.10:5005/	243	0	76	200	true
http://192.168.1.10:5005/	248	0	74	200	true

http://192.168.1.10:5005/	249	0	140	200	true
http://192.168.1.10:5005/	243	0	120	200	true
http://192.168.1.10:5005/dashboard	219	0	99	200	true
http://192.168.1.10:5005/test	252	0	142	404	true
http://192.168.1.10:5005/	283	0	161	200	true
http://192.168.1.10:5005/dashboard	222	0	118	200	true
http://192.168.1.10:5005/dashboard	223	0	99	200	true
http://192.168.1.10:5005/dashboard	227	0	116	200	true
http://192.168.1.10:5005/	223	0	124	200	true
http://192.168.1.10:5005/buatbaru	363	0	105	200	true
http://192.168.1.10:5005/	306	0	112	200	true
http://192.168.1.10:5005/test	296	0	231	404	true
http://192.168.1.10:5005/dashboard	354	0	258	200	true
http://192.168.1.10:5005/dashboard	128	0	61	200	true
http://192.168.1.10:5005/test	101	0	29	404	true
http://192.168.1.10:5005/dashboard	140	0	62	200	true
http://192.168.1.10:5005/	350	0	254	200	true
http://192.168.1.10:5005/dashboard	89	0	34	200	true
http://192.168.1.10:5005/	135	0	57	200	true
http://192.168.1.10:5005/	97	0	40	200	true
http://192.168.1.10:5005/dashboard	127	0	78	200	true
http://192.168.1.10:5005/buatbaru	702	0	258	200	true
http://192.168.1.10:5005/dashboard	144	0	51	200	true
http://192.168.1.10:5005/buatbaru	750	0	258	200	true
http://192.168.1.10:5005/buatbaru	795	0	257	200	true
http://192.168.1.10:5005/buatbaru	459	0	36	200	true
http://192.168.1.10:5005/buatbaru	440	0	30	200	true
http://192.168.1.10:5005/buatbaru	422	0	36	200	true
http://192.168.1.10:5005/buatbaru	377	0	45	200	true
http://192.168.1.10:5005/buatbaru	294	0	8	200	true
http://192.168.1.10:5005/buatbaru	325	0	43	200	true
http://192.168.1.10:5005/	60	0	30	200	true
http://192.168.1.10:5005/dashboard	46	0	10	200	true
http://192.168.1.10:5005/test	35	0	10	404	true
http://192.168.1.10:5005/buatbaru	258	0	20	200	true
http://192.168.1.10:5005/	35	0	7	200	true

http://192.168.1.10:5005/dashboard	65	0	37	200	true
http://192.168.1.10:5005/test	57	0	18	404	true
http://192.168.1.10:5005/test	50	0	16	404	true
http://192.168.1.10:5005/buatbaru	235	0	18	200	true
http://192.168.1.10:5005/test	28	0	11	404	true
http://192.168.1.10:5005/test	46	0	14	404	true
http://192.168.1.10:5005/test	40	0	15	404	true
http://192.168.1.10:5005/	76	0	57	200	true
http://192.168.1.10:5005/	67	0	43	200	true
http://192.168.1.10:5005/test	75	0	18	404	true
http://192.168.1.10:5005/dashboard	85	0	25	200	true
http://192.168.1.10:5005/	68	0	30	200	true
http://192.168.1.10:5005/dashboard	89	0	34	200	true
http://192.168.1.10:5005/	79	0	28	200	true
http://192.168.1.10:5005/test	98	0	33	404	true
http://192.168.1.10:5005/dashboard	64	0	36	200	true
http://192.168.1.10:5005/	85	0	33	200	true
http://192.168.1.10:5005/test	98	0	32	404	true
http://192.168.1.10:5005/dashboard	149	0	88	200	true
http://192.168.1.10:5005/dashboard	140	0	51	200	true
http://192.168.1.10:5005/	151	0	88	200	true
http://192.168.1.10:5005/test	389	0	38	404	true
http://192.168.1.10:5005/buatbaru	386	0	22	200	true
http://192.168.1.10:5005/buatbaru	356	0	39	200	true
http://192.168.1.10:5005/	112	0	29	200	true
http://192.168.1.10:5005/test	94	0	50	404	true
http://192.168.1.10:5005/dashboard	66	0	42	200	true
http://192.168.1.10:5005/buatbaru	376	0	85	200	true
http://192.168.1.10:5005/dashboard	99	0	24	200	true
http://192.168.1.10:5005/buatbaru	319	0	43	200	true
http://192.168.1.10:5005/	81	0	43	200	true
http://192.168.1.10:5005/buatbaru	327	0	58	200	true
http://192.168.1.10:5005/	94	0	18	200	true
http://192.168.1.10:5005/dashboard	93	0	48	200	true
http://192.168.1.10:5005/dashboard	74	0	32	200	true
http://192.168.1.10:5005/buatbaru	302	0	53	200	true

http://192.168.1.10:5005/buatbaru	286	0	34	200	true
http://192.168.1.10:5005/test	29	0	21	404	true
http://192.168.1.10:5005/buatbaru	265	0	36	200	true
http://192.168.1.10:5005/	50	0	6	200	true
http://192.168.1.10:5005/buatbaru	288	0	39	200	true
http://192.168.1.10:5005/	70	0	30	200	true
http://192.168.1.10:5005/dashboard	79	0	39	200	true
http://192.168.1.10:5005/dashboard	286	0	38	200	true
http://192.168.1.10:5005/test	64	0	35	404	true
http://192.168.1.10:5005/test	41	0	29	404	true
http://192.168.1.10:5005/test	56	0	11	404	true
http://192.168.1.10:5005/	30	0	8	200	true
http://192.168.1.10:5005/buatbaru	285	0	66	200	true
http://192.168.1.10:5005/test	70	0	50	404	true
http://192.168.1.10:5005/buatbaru	319	0	65	200	true
http://192.168.1.10:5005/dashboard	68	0	25	200	true
http://192.168.1.10:5005/test	88	0	27	404	true
http://192.168.1.10:5005/	61	0	35	200	true
http://192.168.1.10:5005/	69	0	20	200	true
http://192.168.1.10:5005/dashboard	85	0	39	200	true
http://192.168.1.10:5005/test	257	0	34	404	true
http://192.168.1.10:5005/dashboard	245	0	38	200	true
http://192.168.1.10:5005/test	210	0	12	404	true
http://192.168.1.10:5005/buatbaru	368	0	41	200	true
http://192.168.1.10:5005/test	68	0	37	404	true
http://192.168.1.10:5005/test	56	0	6	404	true
http://192.168.1.10:5005/	216	0	197	200	true
http://192.168.1.10:5005/buatbaru	304	0	16	200	true
http://192.168.1.10:5005/	64	0	27	200	true
http://192.168.1.10:5005/dashboard	91	0	29	200	true
http://192.168.1.10:5005/	69	0	16	200	true
http://192.168.1.10:5005/	64	0	33	200	true
http://192.168.1.10:5005/dashboard	114	0	45	200	true
http://192.168.1.10:5005/	100	0	44	200	true
http://192.168.1.10:5005/dashboard	104	0	49	200	true
http://192.168.1.10:5005/dashboard	94	0	50	200	true

http://192.168.1.10:5005/buatbaru	289	0	16	200	true
http://192.168.1.10:5005/dashboard	130	0	75	200	true
http://192.168.1.10:5005/buatbaru	323	0	42	200	true
http://192.168.1.10:5005/buatbaru	331	0	55	200	true
http://192.168.1.10:5005/buatbaru	370	0	66	200	true
http://192.168.1.10:5005/test	44	0	16	404	true
http://192.168.1.10:5005/test	38	0	25	404	true
http://192.168.1.10:5005/buatbaru	371	0	45	200	true
http://192.168.1.10:5005/buatbaru	355	0	46	200	true
http://192.168.1.10:5005/	52	0	18	200	true
http://192.168.1.10:5005/	107	0	14	200	true
http://192.168.1.10:5005/	77	0	33	200	true
http://192.168.1.10:5005/dashboard	80	0	44	200	true
http://192.168.1.10:5005/dashboard	94	0	38	200	true
http://192.168.1.10:5005/dashboard	77	0	43	200	true
http://192.168.1.10:5005/test	120	0	42	404	true
http://192.168.1.10:5005/test	65	0	34	404	true
http://192.168.1.10:5005/buatbaru	339	0	48	200	true
http://192.168.1.10:5005/buatbaru	350	0	32	200	true
http://192.168.1.10:5005/	98	0	52	200	true
http://192.168.1.10:5005/buatbaru	328	0	46	200	true
http://192.168.1.10:5005/test	202	0	161	404	true
http://192.168.1.10:5005/dashboard	219	0	198	200	true
http://192.168.1.10:5005/test	65	0	40	404	true
http://192.168.1.10:5005/	30	0	17	200	true
http://192.168.1.10:5005/test	82	0	17	404	true
http://192.168.1.10:5005/test	94	0	29	404	true
http://192.168.1.10:5005/test	64	0	20	404	true
http://192.168.1.10:5005/	71	0	18	200	true
http://192.168.1.10:5005/dashboard	69	0	30	200	true
http://192.168.1.10:5005/dashboard	54	0	6	200	true
http://192.168.1.10:5005/test	148	0	87	404	true
http://192.168.1.10:5005/	66	0	24	200	true
http://192.168.1.10:5005/	66	0	24	200	true
http://192.168.1.10:5005/buatbaru	328	0	19	200	true
http://192.168.1.10:5005/	61	0	18	200	true

http://192.168.1.10:5005/dashboard	159	0	71	200	true
http://192.168.1.10:5005/dashboard	152	0	54	200	true
http://192.168.1.10:5005/dashboard	182	0	71	200	true
http://192.168.1.10:5005/buatbaru	309	0	36	200	true
http://192.168.1.10:5005/buatbaru	345	0	85	200	true
http://192.168.1.10:5005/	140	0	79	200	true
http://192.168.1.10:5005/dashboard	73	0	53	200	true
http://192.168.1.10:5005/test	17	0	7	404	true
http://192.168.1.10:5005/buatbaru	355	0	48	200	true
http://192.168.1.10:5005/test	72	0	39	404	true
http://192.168.1.10:5005/buatbaru	320	0	28	200	true
http://192.168.1.10:5005/	39	0	5	200	true
http://192.168.1.10:5005/buatbaru	404	0	31	200	true
http://192.168.1.10:5005/dashboard	91	0	28	200	true
http://192.168.1.10:5005/test	86	0	59	404	true
http://192.168.1.10:5005/buatbaru	335	0	11	200	true
http://192.168.1.10:5005/	91	0	62	200	true
http://192.168.1.10:5005/dashboard	56	0	16	200	true
http://192.168.1.10:5005/	71	0	35	200	true
http://192.168.1.10:5005/	68	0	31	200	true
http://192.168.1.10:5005/buatbaru	258	0	27	200	true
http://192.168.1.10:5005/buatbaru	346	0	34	200	true
http://192.168.1.10:5005/dashboard	258	0	130	200	true
http://192.168.1.10:5005/test	251	0	132	404	true
http://192.168.1.10:5005/test	214	0	118	404	true
http://192.168.1.10:5005/test	177	0	98	404	true
http://192.168.1.10:5005/dashboard	105	0	44	200	true
http://192.168.1.10:5005/buatbaru	3219	0	103	200	true
http://192.168.1.10:5005/buatbaru	3486	0	96	200	true
http://192.168.1.10:5005/test	3191	0	102	404	true
http://192.168.1.10:5005/	222	0	59	200	true
http://192.168.1.10:5005/	218	0	59	200	true
http://192.168.1.10:5005/test	3213	0	3149	404	true
http://192.168.1.10:5005/test	3052	0	2974	404	true
http://192.168.1.10:5005/test	3140	0	3063	404	true
http://192.168.1.10:5005/test	115	0	65	404	true

http://192.168.1.10:5005/test	153	0	58	404	true
http://192.168.1.10:5005/dashboard	147	0	52	200	true
http://192.168.1.10:5005/dashboard	151	0	58	200	true
http://192.168.1.10:5005/	3031	0	2966	200	true
http://192.168.1.10:5005/	112	0	63	200	true
http://192.168.1.10:5005/	110	0	59	200	true
http://192.168.1.10:5005/	138	0	54	200	true
http://192.168.1.10:5005/dashboard	157	0	63	200	true
http://192.168.1.10:5005/	116	0	43	200	true
http://192.168.1.10:5005/	118	0	60	200	true
http://192.168.1.10:5005/dashboard	167	0	81	200	true
http://192.168.1.10:5005/	115	0	61	200	true
http://192.168.1.10:5005/dashboard	143	0	78	200	true
http://192.168.1.10:5005/dashboard	161	0	73	200	true
http://192.168.1.10:5005/buatbaru	1301	0	69	200	true
http://192.168.1.10:5005/buatbaru	1242	0	66	200	true
http://192.168.1.10:5005/dashboard	1244	0	67	200	true
http://192.168.1.10:5005/	196	0	86	200	true
http://192.168.1.10:5005/buatbaru	1551	0	85	200	true
http://192.168.1.10:5005/dashboard	1300	0	62	200	true
http://192.168.1.10:5005/buatbaru	1542	0	75	200	true
http://192.168.1.10:5005/buatbaru	2896	0	1193	200	true
http://192.168.1.10:5005/dashboard	2941	0	1240	200	true
http://192.168.1.10:5005/buatbaru	2932	0	1230	200	true
http://192.168.1.10:5005/test	1789	0	398	404	true
http://192.168.1.10:5005/test	1789	0	397	404	true
http://192.168.1.10:5005/test	85	0	44	404	true
http://192.168.1.10:5005/test	85	0	44	404	true
http://192.168.1.10:5005/test	84	0	44	404	true
http://192.168.1.10:5005/dashboard	1785	0	390	200	true
http://192.168.1.10:5005/test	126	0	43	404	true
http://192.168.1.10:5005/	1776	0	385	200	true
http://192.168.1.10:5005/	121	0	38	200	true
http://192.168.1.10:5005/	125	0	39	200	true
http://192.168.1.10:5005/dashboard	247	0	128	200	true
http://192.168.1.10:5005/buatbaru	2115	0	392	200	true

http://192.168.1.10:5005/	107	0	17	200	true
http://192.168.1.10:5005/	108	0	18	200	true
http://192.168.1.10:5005/buatbaru	2147	0	388	200	true
http://192.168.1.10:5005/buatbaru	493	0	42	200	true
http://192.168.1.10:5005/buatbaru	425	0	50	200	true
http://192.168.1.10:5005/	200	0	50	200	true
http://192.168.1.10:5005/	200	0	51	200	true
http://192.168.1.10:5005/	175	0	51	200	true
http://192.168.1.10:5005/dashboard	125	0	101	200	true
http://192.168.1.10:5005/	229	0	106	200	true
http://192.168.1.10:5005/dashboard	136	0	57	200	true
http://192.168.1.10:5005/dashboard	104	0	46	200	true
http://192.168.1.10:5005/dashboard	136	0	46	200	true
http://192.168.1.10:5005/dashboard	98	0	39	200	true
http://192.168.1.10:5005/dashboard	120	0	67	200	true
http://192.168.1.10:5005/dashboard	128	0	66	200	true
http://192.168.1.10:5005/dashboard	164	0	67	200	true
http://192.168.1.10:5005/	118	0	48	200	true
http://192.168.1.10:5005/buatbaru	474	0	42	200	true
http://192.168.1.10:5005/buatbaru	582	0	40	200	true
http://192.168.1.10:5005/test	254	0	130	404	true
http://192.168.1.10:5005/buatbaru	643	0	64	200	true
http://192.168.1.10:5005/buatbaru	764	0	87	200	true
http://192.168.1.10:5005/buatbaru	810	0	56	200	true
http://192.168.1.10:5005/buatbaru	745	0	100	200	true
http://192.168.1.10:5005/buatbaru	885	0	62	200	true
http://192.168.1.10:5005/buatbaru	812	0	80	200	true
http://192.168.1.10:5005/buatbaru	569	0	53	200	true
http://192.168.1.10:5005/test	135	0	96	404	true
http://192.168.1.10:5005/test	161	0	65	404	true
http://192.168.1.10:5005/test	145	0	50	404	true
http://192.168.1.10:5005/test	151	0	55	404	true
http://192.168.1.10:5005/test	803	0	113	404	true
http://192.168.1.10:5005/	150	0	54	200	true
http://192.168.1.10:5005/	145	0	89	200	true
http://192.168.1.10:5005/	161	0	89	200	true

http://192.168.1.10:5005/	181	0	89	200	true
http://192.168.1.10:5005/dashboard	257	0	117	200	true
http://192.168.1.10:5005/dashboard	973	0	111	200	true
http://192.168.1.10:5005/test	212	0	104	404	true
http://192.168.1.10:5005/	118	0	57	200	true
http://192.168.1.10:5005/dashboard	219	0	107	200	true
http://192.168.1.10:5005/dashboard	218	0	107	200	true
http://192.168.1.10:5005/test	190	0	80	404	true
http://192.168.1.10:5005/dashboard	225	0	107	200	true
http://192.168.1.10:5005/dashboard	130	0	51	200	true
http://192.168.1.10:5005/test	156	0	82	404	true
http://192.168.1.10:5005/	153	0	69	200	true
http://192.168.1.10:5005/test	117	0	50	404	true
http://192.168.1.10:5005/test	127	0	60	404	true
http://192.168.1.10:5005/test	85	0	34	404	true
http://192.168.1.10:5005/buatbaru	420	0	92	200	true
http://192.168.1.10:5005/dashboard	125	0	42	200	true
http://192.168.1.10:5005/	160	0	90	200	true
http://192.168.1.10:5005/test	102	0	51	404	true
http://192.168.1.10:5005/test	103	0	51	404	true
http://192.168.1.10:5005/test	132	0	44	404	true
http://192.168.1.10:5005/buatbaru	499	0	91	200	true
http://192.168.1.10:5005/	91	0	37	200	true
http://192.168.1.10:5005/dashboard	221	0	58	200	true
http://192.168.1.10:5005/buatbaru	642	0	92	200	true
http://192.168.1.10:5005/	133	0	66	200	true
http://192.168.1.10:5005/dashboard	279	0	142	200	true
http://192.168.1.10:5005/buatbaru	653	0	92	200	true
http://192.168.1.10:5005/buatbaru	659	0	56	200	true
http://192.168.1.10:5005/buatbaru	712	0	87	200	true
http://192.168.1.10:5005/	128	0	44	200	true
http://192.168.1.10:5005/	124	0	44	200	true
http://192.168.1.10:5005/buatbaru	501	0	61	200	true
http://192.168.1.10:5005/	260	0	70	200	true
http://192.168.1.10:5005/	113	0	55	200	true
http://192.168.1.10:5005/	270	0	107	200	true

http://192.168.1.10:5005/dashboard	237	0	100	200	true
http://192.168.1.10:5005/dashboard	202	0	101	200	true
http://192.168.1.10:5005/dashboard	216	0	116	200	true
http://192.168.1.10:5005/dashboard	171	0	73	200	true
http://192.168.1.10:5005/dashboard	170	0	76	200	true
http://192.168.1.10:5005/dashboard	159	0	87	200	true
http://192.168.1.10:5005/buatbaru	587	0	127	200	true
http://192.168.1.10:5005/buatbaru	633	0	126	200	true
http://192.168.1.10:5005/test	230	0	113	404	true
http://192.168.1.10:5005/buatbaru	463	0	83	200	true
http://192.168.1.10:5005/	232	0	91	200	true
http://192.168.1.10:5005/test	196	0	76	404	true
http://192.168.1.10:5005/buatbaru	438	0	98	200	true
http://192.168.1.10:5005/buatbaru	471	0	96	200	true
http://192.168.1.10:5005/buatbaru	653	0	101	200	true
http://192.168.1.10:5005/dashboard	321	0	133	200	true
http://192.168.1.10:5005/buatbaru	784	0	95	200	true
http://192.168.1.10:5005/buatbaru	790	0	80	200	true
http://192.168.1.10:5005/test	361	0	182	404	true
http://192.168.1.10:5005/test	357	0	189	404	true
http://192.168.1.10:5005/test	338	0	186	404	true
http://192.168.1.10:5005/test	314	0	196	404	true
http://192.168.1.10:5005/	265	0	120	200	true
http://192.168.1.10:5005/test	294	0	164	404	true
http://192.168.1.10:5005/	319	0	135	200	true
http://192.168.1.10:5005/dashboard	319	0	134	200	true
http://192.168.1.10:5005/buatbaru	502	0	174	200	true
http://192.168.1.10:5005/dashboard	319	0	146	200	true
http://192.168.1.10:5005/test	360	0	178	404	true
http://192.168.1.10:5005/test	351	0	172	404	true
http://192.168.1.10:5005/	265	0	125	200	true
http://192.168.1.10:5005/test	319	0	170	404	true
http://192.168.1.10:5005/	241	0	117	200	true
http://192.168.1.10:5005/test	400	0	175	404	true
http://192.168.1.10:5005/	297	0	139	200	true
http://192.168.1.10:5005/test	466	0	227	404	true

http://192.168.1.10:5005/dashboard	494	0	230	200	true
http://192.168.1.10:5005/buatbaru	605	0	163	200	true
http://192.168.1.10:5005/test	468	0	206	404	true
http://192.168.1.10:5005/buatbaru	588	0	167	200	true
http://192.168.1.10:5005/test	437	0	225	404	true
http://192.168.1.10:5005/test	428	0	216	404	true
http://192.168.1.10:5005/	266	0	130	200	true
http://192.168.1.10:5005/dashboard	496	0	256	200	true
http://192.168.1.10:5005/dashboard	447	0	229	200	true
http://192.168.1.10:5005/	409	0	158	200	true
http://192.168.1.10:5005/dashboard	336	0	155	200	true
http://192.168.1.10:5005/	246	0	112	200	true
http://192.168.1.10:5005/	495	0	259	200	true
http://192.168.1.10:5005/test	397	0	174	404	true
http://192.168.1.10:5005/	458	0	242	200	true
http://192.168.1.10:5005/dashboard	384	0	179	200	true
http://192.168.1.10:5005/buatbaru	644	0	203	200	true
http://192.168.1.10:5005/dashboard	487	0	200	200	true
http://192.168.1.10:5005/dashboard	509	0	228	200	true
http://192.168.1.10:5005/buatbaru	651	0	170	200	true
http://192.168.1.10:5005/buatbaru	706	0	160	200	true
http://192.168.1.10:5005/	356	0	175	200	true
http://192.168.1.10:5005/	374	0	175	200	true
http://192.168.1.10:5005/	344	0	178	200	true
http://192.168.1.10:5005/buatbaru	679	0	202	200	true
http://192.168.1.10:5005/	564	0	212	200	true
http://192.168.1.10:5005/test	336	0	211	404	true
http://192.168.1.10:5005/	1446	0	1203	200	true
http://192.168.1.10:5005/dashboard	274	0	136	200	true
http://192.168.1.10:5005/buatbaru	593	0	276	200	true
http://192.168.1.10:5005/dashboard	279	0	136	200	true
http://192.168.1.10:5005/test	255	0	114	404	true
http://192.168.1.10:5005/	501	0	227	200	true
http://192.168.1.10:5005/dashboard	249	0	117	200	true
http://192.168.1.10:5005/dashboard	667	0	275	200	true
http://192.168.1.10:5005/	466	0	247	200	true

http://192.168.1.10:5005/dashboard	338	0	111	200	true
http://192.168.1.10:5005/buatbaru	624	0	217	200	true
http://192.168.1.10:5005/buatbaru	618	0	229	200	true
http://192.168.1.10:5005/dashboard	351	0	154	200	true
http://192.168.1.10:5005/dashboard	499	0	177	200	true
http://192.168.1.10:5005/test	488	0	237	404	true
http://192.168.1.10:5005/buatbaru	818	0	153	200	true
http://192.168.1.10:5005/buatbaru	861	0	148	200	true
http://192.168.1.10:5005/buatbaru	811	0	149	200	true
http://192.168.1.10:5005/	385	0	146	200	true
http://192.168.1.10:5005/test	499	0	262	404	true
http://192.168.1.10:5005/test	482	0	256	404	true
http://192.168.1.10:5005/buatbaru	771	0	176	200	true
http://192.168.1.10:5005/buatbaru	785	0	176	200	true
http://192.168.1.10:5005/buatbaru	696	0	257	200	true
http://192.168.1.10:5005/test	430	0	232	404	true
http://192.168.1.10:5005/dashboard	248	0	129	200	true
http://192.168.1.10:5005/	441	0	174	200	true
http://192.168.1.10:5005/test	455	0	212	404	true
http://192.168.1.10:5005/	561	0	125	200	true
http://192.168.1.10:5005/dashboard	477	0	208	200	true
http://192.168.1.10:5005/buatbaru	782	0	264	200	true
http://192.168.1.10:5005/dashboard	420	0	224	200	true
http://192.168.1.10:5005/test	420	0	233	404	true
http://192.168.1.10:5005/test	402	0	233	404	true
http://192.168.1.10:5005/dashboard	409	0	192	200	true
http://192.168.1.10:5005/buatbaru	613	0	239	200	true
http://192.168.1.10:5005/	451	0	229	200	true
http://192.168.1.10:5005/	501	0	220	200	true
http://192.168.1.10:5005/	425	0	235	200	true
http://192.168.1.10:5005/test	432	0	233	404	true
http://192.168.1.10:5005/test	413	0	222	404	true
http://192.168.1.10:5005/	453	0	257	200	true
http://192.168.1.10:5005/buatbaru	617	0	202	200	true
http://192.168.1.10:5005/dashboard	393	0	191	200	true
http://192.168.1.10:5005/test	414	0	175	404	true

http://192.168.1.10:5005/	401	0	232	200	true
http://192.168.1.10:5005/test	432	0	201	404	true
http://192.168.1.10:5005/dashboard	440	0	198	200	true
http://192.168.1.10:5005/test	443	0	176	404	true
http://192.168.1.10:5005/dashboard	438	0	173	200	true
http://192.168.1.10:5005/buatbaru	834	0	247	200	true
http://192.168.1.10:5005/buatbaru	663	0	230	200	true
http://192.168.1.10:5005/	464	0	234	200	true
http://192.168.1.10:5005/dashboard	464	0	237	200	true
http://192.168.1.10:5005/	434	0	189	200	true
http://192.168.1.10:5005/test	426	0	224	404	true
http://192.168.1.10:5005/	671	0	247	200	true
http://192.168.1.10:5005/dashboard	444	0	251	200	true
http://192.168.1.10:5005/buatbaru	629	0	229	200	true
http://192.168.1.10:5005/	482	0	257	200	true
http://192.168.1.10:5005/	460	0	224	200	true
http://192.168.1.10:5005/	417	0	197	200	true
http://192.168.1.10:5005/test	1529	0	222	404	true
http://192.168.1.10:5005/dashboard	380	0	176	200	true
http://192.168.1.10:5005/buatbaru	707	0	195	200	true
http://192.168.1.10:5005/test	402	0	222	404	true
http://192.168.1.10:5005/buatbaru	591	0	193	200	true
http://192.168.1.10:5005/	422	0	220	200	true
http://192.168.1.10:5005/dashboard	395	0	227	200	true
http://192.168.1.10:5005/dashboard	357	0	156	200	true
http://192.168.1.10:5005/dashboard	416	0	164	200	true
http://192.168.1.10:5005/test	428	0	186	404	true
http://192.168.1.10:5005/dashboard	442	0	193	200	true
http://192.168.1.10:5005/dashboard	1431	0	1191	200	true
http://192.168.1.10:5005/	425	0	227	200	true
http://192.168.1.10:5005/buatbaru	638	0	214	200	true
http://192.168.1.10:5005/dashboard	477	0	250	200	true
http://192.168.1.10:5005/buatbaru	720	0	243	200	true
http://192.168.1.10:5005/	383	0	176	200	true
http://192.168.1.10:5005/	413	0	191	200	true
http://192.168.1.10:5005/test	451	0	232	404	true

http://192.168.1.10:5005/test	451	0	233	404	true
http://192.168.1.10:5005/buatbaru	675	0	185	200	true
http://192.168.1.10:5005/dashboard	445	0	215	200	true
http://192.168.1.10:5005/buatbaru	681	0	248	200	true
http://192.168.1.10:5005/	1421	0	1188	200	true
http://192.168.1.10:5005/dashboard	454	0	229	200	true
http://192.168.1.10:5005/dashboard	477	0	228	200	true
http://192.168.1.10:5005/buatbaru	800	0	262	200	true
http://192.168.1.10:5005/buatbaru	747	0	223	200	true
http://192.168.1.10:5005/test	467	0	201	404	true
http://192.168.1.10:5005/buatbaru	741	0	264	200	true
http://192.168.1.10:5005/buatbaru	700	0	215	200	true
http://192.168.1.10:5005/	430	0	200	200	true
http://192.168.1.10:5005/	463	0	185	200	true
http://192.168.1.10:5005/test	427	0	268	404	true
http://192.168.1.10:5005/test	361	0	237	404	true
http://192.168.1.10:5005/	384	0	198	200	true
http://192.168.1.10:5005/dashboard	423	0	267	200	true
http://192.168.1.10:5005/buatbaru	571	0	194	200	true
http://192.168.1.10:5005/buatbaru	946	0	205	200	true
http://192.168.1.10:5005/	466	0	209	200	true
http://192.168.1.10:5005/	434	0	190	200	true
http://192.168.1.10:5005/dashboard	198	0	86	200	true
http://192.168.1.10:5005/dashboard	157	0	77	200	true
http://192.168.1.10:5005/dashboard	220	0	67	200	true
http://192.168.1.10:5005/buatbaru	544	0	242	200	true
http://192.168.1.10:5005/dashboard	183	0	74	200	true
http://192.168.1.10:5005/test	187	0	74	404	true
http://192.168.1.10:5005/buatbaru	532	0	237	200	true
http://192.168.1.10:5005/dashboard	188	0	74	200	true
http://192.168.1.10:5005/test	187	0	94	404	true
http://192.168.1.10:5005/	218	0	66	200	true
http://192.168.1.10:5005/	197	0	101	200	true
http://192.168.1.10:5005/	220	0	67	200	true
http://192.168.1.10:5005/test	270	0	129	404	true
http://192.168.1.10:5005/	312	0	159	200	true

http://192.168.1.10:5005/buatbaru	610	0	66	200	true
http://192.168.1.10:5005/test	311	0	180	404	true
http://192.168.1.10:5005/buatbaru	508	0	89	200	true
http://192.168.1.10:5005/dashboard	256	0	145	200	true
http://192.168.1.10:5005/dashboard	234	0	115	200	true
http://192.168.1.10:5005/buatbaru	523	0	92	200	true
http://192.168.1.10:5005/dashboard	265	0	127	200	true
http://192.168.1.10:5005/dashboard	255	0	108	200	true
http://192.168.1.10:5005/test	221	0	105	404	true
http://192.168.1.10:5005/buatbaru	736	0	99	200	true
http://192.168.1.10:5005/buatbaru	796	0	89	200	true
http://192.168.1.10:5005/buatbaru	711	0	96	200	true
http://192.168.1.10:5005/test	329	0	148	404	true
http://192.168.1.10:5005/test	326	0	144	404	true
http://192.168.1.10:5005/	222	0	127	200	true
http://192.168.1.10:5005/test	448	0	229	404	true
http://192.168.1.10:5005/	211	0	102	200	true
http://192.168.1.10:5005/buatbaru	656	0	108	200	true
http://192.168.1.10:5005/buatbaru	698	0	108	200	true
http://192.168.1.10:5005/buatbaru	620	0	172	200	true
http://192.168.1.10:5005/test	482	0	256	404	true
http://192.168.1.10:5005/test	434	0	216	404	true
http://192.168.1.10:5005/buatbaru	640	0	172	200	true
http://192.168.1.10:5005/dashboard	422	0	224	200	true
http://192.168.1.10:5005/	401	0	167	200	true
http://192.168.1.10:5005/dashboard	439	0	221	200	true
http://192.168.1.10:5005/test	1213	0	148	404	true
http://192.168.1.10:5005/test	507	0	223	404	true
http://192.168.1.10:5005/test	531	0	236	404	true
http://192.168.1.10:5005/	450	0	231	200	true
http://192.168.1.10:5005/	461	0	232	200	true
http://192.168.1.10:5005/test	573	0	256	404	true
http://192.168.1.10:5005/dashboard	577	0	273	200	true
http://192.168.1.10:5005/test	1610	0	1325	404	true
http://192.168.1.10:5005/test	585	0	342	404	true
http://192.168.1.10:5005/test	569	0	326	404	true

http://192.168.1.10:5005/test	531	0	301	404	true
http://192.168.1.10:5005/buatbaru	713	0	236	200	true
http://192.168.1.10:5005/dashboard	516	0	275	200	true
http://192.168.1.10:5005/dashboard	508	0	239	200	true
http://192.168.1.10:5005/buatbaru	741	0	320	200	true
http://192.168.1.10:5005/	502	0	227	200	true
http://192.168.1.10:5005/	501	0	229	200	true
http://192.168.1.10:5005/test	490	0	238	404	true
http://192.168.1.10:5005/test	537	0	260	404	true
http://192.168.1.10:5005/test	524	0	238	404	true
http://192.168.1.10:5005/test	515	0	265	404	true
http://192.168.1.10:5005/buatbaru	719	0	228	200	true
http://192.168.1.10:5005/	594	0	349	200	true
http://192.168.1.10:5005/	580	0	285	200	true
http://192.168.1.10:5005/dashboard	497	0	242	200	true
http://192.168.1.10:5005/	563	0	284	200	true
http://192.168.1.10:5005/	527	0	284	200	true
http://192.168.1.10:5005/	511	0	233	200	true
http://192.168.1.10:5005/dashboard	625	0	249	200	true
http://192.168.1.10:5005/buatbaru	654	0	239	200	true
http://192.168.1.10:5005/	505	0	260	200	true
http://192.168.1.10:5005/	523	0	267	200	true
http://192.168.1.10:5005/dashboard	333	0	145	200	true
http://192.168.1.10:5005/dashboard	335	0	123	200	true
http://192.168.1.10:5005/dashboard	336	0	116	200	true
http://192.168.1.10:5005/	464	0	226	200	true
http://192.168.1.10:5005/	368	0	164	200	true
http://192.168.1.10:5005/dashboard	369	0	140	200	true
http://192.168.1.10:5005/dashboard	380	0	152	200	true
http://192.168.1.10:5005/	541	0	264	200	true
http://192.168.1.10:5005/dashboard	374	0	226	200	true
http://192.168.1.10:5005/dashboard	377	0	220	200	true
http://192.168.1.10:5005/test	409	0	235	404	true
http://192.168.1.10:5005/	400	0	252	200	true
http://192.168.1.10:5005/buatbaru	601	0	120	200	true
http://192.168.1.10:5005/dashboard	404	0	198	200	true

http://192.168.1.10:5005/dashboard	453	0	189	200	true
http://192.168.1.10:5005/buatbaru	586	0	202	200	true
http://192.168.1.10:5005/test	379	0	153	404	true
http://192.168.1.10:5005/dashboard	410	0	147	200	true
http://192.168.1.10:5005/dashboard	371	0	184	200	true
http://192.168.1.10:5005/	499	0	303	200	true
http://192.168.1.10:5005/buatbaru	909	0	223	200	true
http://192.168.1.10:5005/test	515	0	224	404	true
http://192.168.1.10:5005/	422	0	186	200	true
http://192.168.1.10:5005/test	516	0	244	404	true
http://192.168.1.10:5005/buatbaru	1112	0	186	200	true
http://192.168.1.10:5005/buatbaru	1244	0	232	200	true
http://192.168.1.10:5005/	374	0	176	200	true
http://192.168.1.10:5005/buatbaru	1158	0	147	200	true
http://192.168.1.10:5005/dashboard	370	0	262	200	true
http://192.168.1.10:5005/dashboard	525	0	254	200	true
http://192.168.1.10:5005/	516	0	204	200	true
http://192.168.1.10:5005/buatbaru	1503	0	203	200	true
http://192.168.1.10:5005/buatbaru	2520	0	1217	200	true
http://192.168.1.10:5005/buatbaru	988	0	199	200	true
http://192.168.1.10:5005/buatbaru	1061	0	185	200	true
http://192.168.1.10:5005/test	148	0	77	404	true
http://192.168.1.10:5005/dashboard	148	0	44	200	true
http://192.168.1.10:5005/buatbaru	1325	0	201	200	true
http://192.168.1.10:5005/buatbaru	1327	0	207	200	true
http://192.168.1.10:5005/buatbaru	879	0	266	200	true
http://192.168.1.10:5005/buatbaru	1089	0	182	200	true
http://192.168.1.10:5005/test	165	0	79	404	true
http://192.168.1.10:5005/	447	0	263	200	true
http://192.168.1.10:5005/dashboard	153	0	100	200	true
http://192.168.1.10:5005/	170	0	86	200	true
http://192.168.1.10:5005/	79	0	25	200	true
http://192.168.1.10:5005/buatbaru	561	0	70	200	true
http://192.168.1.10:5005/buatbaru	475	0	46	200	true
http://192.168.1.10:5005/buatbaru	611	0	70	200	true
http://192.168.1.10:5005/buatbaru	463	0	54	200	true

http://192.168.1.10:5005/dashboard	256	0	98	200	true
http://192.168.1.10:5005/dashboard	232	0	137	200	true
http://192.168.1.10:5005/test	229	0	119	404	true
http://192.168.1.10:5005/dashboard	142	0	57	200	true
http://192.168.1.10:5005/	236	0	94	200	true
http://192.168.1.10:5005/test	215	0	110	404	true
http://192.168.1.10:5005/test	216	0	129	404	true
http://192.168.1.10:5005/test	226	0	134	404	true
http://192.168.1.10:5005/dashboard	219	0	104	200	true
http://192.168.1.10:5005/test	207	0	95	404	true
http://192.168.1.10:5005/test	201	0	99	404	true
http://192.168.1.10:5005/test	201	0	89	404	true
http://192.168.1.10:5005/test	222	0	90	404	true
http://192.168.1.10:5005/test	211	0	96	404	true
http://192.168.1.10:5005/test	231	0	102	404	true
http://192.168.1.10:5005/test	223	0	100	404	true
http://192.168.1.10:5005/buatbaru	604	0	106	200	true
http://192.168.1.10:5005/buatbaru	545	0	79	200	true
http://192.168.1.10:5005/buatbaru	464	0	111	200	true
http://192.168.1.10:5005/	213	0	95	200	true
http://192.168.1.10:5005/buatbaru	472	0	101	200	true
http://192.168.1.10:5005/test	760	0	105	404	true
http://192.168.1.10:5005/	230	0	99	200	true
http://192.168.1.10:5005/	251	0	97	200	true
http://192.168.1.10:5005/test	326	0	169	404	true
http://192.168.1.10:5005/test	309	0	175	404	true
http://192.168.1.10:5005/	277	0	128	200	true
http://192.168.1.10:5005/test	275	0	145	404	true
http://192.168.1.10:5005/	252	0	110	200	true
http://192.168.1.10:5005/dashboard	276	0	157	200	true
http://192.168.1.10:5005/	272	0	110	200	true
http://192.168.1.10:5005/	275	0	112	200	true
http://192.168.1.10:5005/dashboard	239	0	109	200	true
http://192.168.1.10:5005/dashboard	258	0	118	200	true
http://192.168.1.10:5005/	244	0	129	200	true
http://192.168.1.10:5005/dashboard	315	0	126	200	true

http://192.168.1.10:5005/dashboard	306	0	119	200	true
http://192.168.1.10:5005/dashboard	310	0	120	200	true
http://192.168.1.10:5005/dashboard	301	0	160	200	true
http://192.168.1.10:5005/	261	0	123	200	true
http://192.168.1.10:5005/	235	0	103	200	true
http://192.168.1.10:5005/	237	0	94	200	true
http://192.168.1.10:5005/	304	0	131	200	true
http://192.168.1.10:5005/test	188	0	94	404	true
http://192.168.1.10:5005/test	171	0	74	404	true
http://192.168.1.10:5005/dashboard	174	0	55	200	true
http://192.168.1.10:5005/test	171	0	66	404	true
http://192.168.1.10:5005/	247	0	109	200	true
http://192.168.1.10:5005/	200	0	99	200	true
http://192.168.1.10:5005/buatbaru	707	0	133	200	true
http://192.168.1.10:5005/dashboard	208	0	63	200	true
http://192.168.1.10:5005/dashboard	235	0	58	200	true
http://192.168.1.10:5005/dashboard	212	0	63	200	true
http://192.168.1.10:5005/test	1218	0	1023	404	true
http://192.168.1.10:5005/dashboard	283	0	18	200	true
http://192.168.1.10:5005/dashboard	287	0	18	200	true
http://192.168.1.10:5005/buatbaru	666	0	162	200	true
http://192.168.1.10:5005/buatbaru	655	0	126	200	true
http://192.168.1.10:5005/	197	0	23	200	true
http://192.168.1.10:5005/buatbaru	842	0	134	200	true
http://192.168.1.10:5005/buatbaru	561	0	85	200	true
http://192.168.1.10:5005/	212	0	36	200	true
http://192.168.1.10:5005/	302	0	17	200	true
http://192.168.1.10:5005/dashboard	1095	0	1066	200	true
http://192.168.1.10:5005/test	144	0	48	404	true
http://192.168.1.10:5005/dashboard	154	0	38	200	true
http://192.168.1.10:5005/dashboard	131	0	16	200	true
http://192.168.1.10:5005/buatbaru	784	0	102	200	true
http://192.168.1.10:5005/dashboard	170	0	16	200	true
http://192.168.1.10:5005/buatbaru	873	0	105	200	true
http://192.168.1.10:5005/	193	0	138	200	true
http://192.168.1.10:5005/buatbaru	565	0	17	200	true

http://192.168.1.10:5005/dashboard	128	0	50	200	true
http://192.168.1.10:5005/	146	0	68	200	true
http://192.168.1.10:5005/buatbaru	579	0	8	200	true
http://192.168.1.10:5005/buatbaru	570	0	43	200	true
http://192.168.1.10:5005/buatbaru	666	0	72	200	true
http://192.168.1.10:5005/dashboard	67	0	39	200	true
http://192.168.1.10:5005/buatbaru	406	0	51	200	true
http://192.168.1.10:5005/	76	0	35	200	true
http://192.168.1.10:5005/buatbaru	397	0	48	200	true
http://192.168.1.10:5005/buatbaru	709	0	44	200	true
http://192.168.1.10:5005/dashboard	106	0	52	200	true
http://192.168.1.10:5005/buatbaru	675	0	62	200	true
http://192.168.1.10:5005/buatbaru	802	0	45	200	true
http://192.168.1.10:5005/test	100	0	48	404	true
http://192.168.1.10:5005/buatbaru	377	0	49	200	true
http://192.168.1.10:5005/buatbaru	749	0	53	200	true
http://192.168.1.10:5005/buatbaru	552	0	48	200	true
http://192.168.1.10:5005/	80	0	31	200	true
http://192.168.1.10:5005/dashboard	23	0	6	200	true
http://192.168.1.10:5005/buatbaru	323	0	40	200	true
http://192.168.1.10:5005/test	83	0	48	404	true
http://192.168.1.10:5005/test	42	0	34	404	true
http://192.168.1.10:5005/	33	0	9	200	true
http://192.168.1.10:5005/buatbaru	273	0	30	200	true
http://192.168.1.10:5005/dashboard	49	0	22	200	true
http://192.168.1.10:5005/test	16	0	6	404	true
http://192.168.1.10:5005/test	42	0	29	404	true
http://192.168.1.10:5005/	44	0	9	200	true
http://192.168.1.10:5005/test	38	0	24	404	true
http://192.168.1.10:5005/dashboard	52	0	11	200	true
http://192.168.1.10:5005/test	444	0	46	404	true
http://192.168.1.10:5005/test	454	0	49	404	true
http://192.168.1.10:5005/	36	0	25	200	true
http://192.168.1.10:5005/test	71	0	22	404	true
http://192.168.1.10:5005/test	82	0	24	404	true
http://192.168.1.10:5005/dashboard	81	0	21	200	true

http://192.168.1.10:5005/	70	0	39	200	true
http://192.168.1.10:5005/	77	0	31	200	true
http://192.168.1.10:5005/test	99	0	41	404	true
http://192.168.1.10:5005/	43	0	12	200	true
http://192.168.1.10:5005/buatbaru	308	0	10	200	true
http://192.168.1.10:5005/dashboard	86	0	40	200	true
http://192.168.1.10:5005/	83	0	28	200	true
http://192.168.1.10:5005/dashboard	100	0	52	200	true
http://192.168.1.10:5005/	99	0	40	200	true
http://192.168.1.10:5005/dashboard	100	0	44	200	true
http://192.168.1.10:5005/test	98	0	45	404	true
http://192.168.1.10:5005/dashboard	103	0	27	200	true
http://192.168.1.10:5005/test	87	0	34	404	true
http://192.168.1.10:5005/test	70	0	26	404	true
http://192.168.1.10:5005/dashboard	131	0	74	200	true
http://192.168.1.10:5005/	102	0	52	200	true
http://192.168.1.10:5005/test	110	0	38	404	true
http://192.168.1.10:5005/	102	0	45	200	true
http://192.168.1.10:5005/buatbaru	307	0	38	200	true
http://192.168.1.10:5005/dashboard	173	0	37	200	true
http://192.168.1.10:5005/test	135	0	45	404	true
http://192.168.1.10:5005/test	157	0	43	404	true
http://192.168.1.10:5005/	120	0	38	200	true
http://192.168.1.10:5005/test	111	0	19	404	true
http://192.168.1.10:5005/dashboard	115	0	54	200	true
http://192.168.1.10:5005/test	72	0	21	404	true
http://192.168.1.10:5005/buatbaru	669	0	25	200	true
http://192.168.1.10:5005/buatbaru	375	0	45	200	true
http://192.168.1.10:5005/	132	0	35	200	true
http://192.168.1.10:5005/buatbaru	592	0	39	200	true
http://192.168.1.10:5005/dashboard	116	0	28	200	true
http://192.168.1.10:5005/	67	0	19	200	true
http://192.168.1.10:5005/	82	0	38	200	true
http://192.168.1.10:5005/test	96	0	13	404	true
http://192.168.1.10:5005/dashboard	95	0	12	200	true
http://192.168.1.10:5005/dashboard	66	0	15	200	true

http://192.168.1.10:5005/	119	0	31	200	true
http://192.168.1.10:5005/buatbaru	555	0	50	200	true
http://192.168.1.10:5005/dashboard	139	0	12	200	true
http://192.168.1.10:5005/buatbaru	537	0	38	200	true
http://192.168.1.10:5005/dashboard	87	0	27	200	true
http://192.168.1.10:5005/buatbaru	687	0	55	200	true
http://192.168.1.10:5005/buatbaru	443	0	25	200	true
http://192.168.1.10:5005/test	155	0	72	404	true
http://192.168.1.10:5005/	150	0	22	200	true
http://192.168.1.10:5005/buatbaru	467	0	52	200	true
http://192.168.1.10:5005/buatbaru	371	0	16	200	true
http://192.168.1.10:5005/dashboard	93	0	22	200	true
http://192.168.1.10:5005/	118	0	53	200	true
http://192.168.1.10:5005/buatbaru	405	0	15	200	true
http://192.168.1.10:5005/	74	0	48	200	true
http://192.168.1.10:5005/dashboard	88	0	43	200	true
http://192.168.1.10:5005/test	99	0	34	404	true
http://192.168.1.10:5005/dashboard	79	0	35	200	true
http://192.168.1.10:5005/buatbaru	512	0	18	200	true
http://192.168.1.10:5005/buatbaru	387	0	61	200	true
http://192.168.1.10:5005/buatbaru	397	0	47	200	true
http://192.168.1.10:5005/	48	0	22	200	true
http://192.168.1.10:5005/dashboard	85	0	27	200	true
http://192.168.1.10:5005/buatbaru	323	0	30	200	true
http://192.168.1.10:5005/test	56	0	13	404	true
http://192.168.1.10:5005/buatbaru	373	0	6	200	true
http://192.168.1.10:5005/test	100	0	29	404	true
http://192.168.1.10:5005/	35	0	4	200	true
http://192.168.1.10:5005/test	112	0	55	404	true
http://192.168.1.10:5005/buatbaru	290	0	16	200	true
http://192.168.1.10:5005/test	95	0	65	404	true
http://192.168.1.10:5005/buatbaru	323	0	41	200	true
http://192.168.1.10:5005/dashboard	213	0	27	200	true
http://192.168.1.10:5005/test	259	0	196	404	true
http://192.168.1.10:5005/test	162	0	68	404	true
http://192.168.1.10:5005/test	172	0	84	404	true

http://192.168.1.10:5005/	216	0	184	200	true
http://192.168.1.10:5005/test	199	0	97	404	true
http://192.168.1.10:5005/test	174	0	74	404	true
http://192.168.1.10:5005/test	174	0	98	404	true
http://192.168.1.10:5005/	170	0	78	200	true
http://192.168.1.10:5005/test	136	0	69	404	true
http://192.168.1.10:5005/buatbaru	339	0	56	200	true
http://192.168.1.10:5005/	177	0	75	200	true
http://192.168.1.10:5005/dashboard	141	0	65	200	true
http://192.168.1.10:5005/test	100	0	59	404	true
http://192.168.1.10:5005/test	93	0	43	404	true
http://192.168.1.10:5005/dashboard	99	0	50	200	true
http://192.168.1.10:5005/dashboard	78	0	34	200	true
http://192.168.1.10:5005/	133	0	60	200	true
http://192.168.1.10:5005/	175	0	99	200	true
http://192.168.1.10:5005/test	110	0	48	404	true
http://192.168.1.10:5005/	86	0	41	200	true
http://192.168.1.10:5005/	93	0	53	200	true
http://192.168.1.10:5005/dashboard	68	0	34	200	true
http://192.168.1.10:5005/dashboard	81	0	35	200	true
http://192.168.1.10:5005/test	73	0	26	404	true
http://192.168.1.10:5005/dashboard	75	0	25	200	true
http://192.168.1.10:5005/	121	0	62	200	true
http://192.168.1.10:5005/dashboard	116	0	37	200	true
http://192.168.1.10:5005/dashboard	202	0	86	200	true
http://192.168.1.10:5005/test	208	0	84	404	true
http://192.168.1.10:5005/	134	0	39	200	true
http://192.168.1.10:5005/buatbaru	564	0	38	200	true
http://192.168.1.10:5005/test	175	0	109	404	true
http://192.168.1.10:5005/buatbaru	584	0	62	200	true
http://192.168.1.10:5005/	158	0	46	200	true
http://192.168.1.10:5005/test	137	0	65	404	true
http://192.168.1.10:5005/dashboard	139	0	83	200	true
http://192.168.1.10:5005/	147	0	97	200	true
http://192.168.1.10:5005/buatbaru	732	0	39	200	true
http://192.168.1.10:5005/buatbaru	526	0	48	200	true

http://192.168.1.10:5005/	150	0	92	200	true
http://192.168.1.10:5005/dashboard	153	0	79	200	true
http://192.168.1.10:5005/buatbaru	685	0	77	200	true
http://192.168.1.10:5005/buatbaru	673	0	71	200	true
http://192.168.1.10:5005/	135	0	53	200	true
http://192.168.1.10:5005/dashboard	76	0	43	200	true
http://192.168.1.10:5005/buatbaru	499	0	117	200	true
http://192.168.1.10:5005/dashboard	90	0	41	200	true
http://192.168.1.10:5005/	135	0	74	200	true
http://192.168.1.10:5005/buatbaru	521	0	117	200	true
http://192.168.1.10:5005/dashboard	273	0	37	200	true
http://192.168.1.10:5005/dashboard	235	0	191	200	true
http://192.168.1.10:5005/test	160	0	116	404	true
http://192.168.1.10:5005/buatbaru	425	0	87	200	true
http://192.168.1.10:5005/buatbaru	328	0	24	200	true
http://192.168.1.10:5005/buatbaru	366	0	27	200	true
http://192.168.1.10:5005/buatbaru	436	0	11	200	true
http://192.168.1.10:5005/buatbaru	682	0	192	200	true
http://192.168.1.10:5005/buatbaru	445	0	11	200	true
http://192.168.1.10:5005/test	109	0	51	404	true
http://192.168.1.10:5005/test	74	0	17	404	true
http://192.168.1.10:5005/	92	0	62	200	true
http://192.168.1.10:5005/test	126	0	79	404	true
http://192.168.1.10:5005/test	124	0	63	404	true
http://192.168.1.10:5005/dashboard	132	0	69	200	true
http://192.168.1.10:5005/test	123	0	55	404	true
http://192.168.1.10:5005/	127	0	43	200	true
http://192.168.1.10:5005/	124	0	41	200	true
http://192.168.1.10:5005/test	100	0	56	404	true
http://192.168.1.10:5005/test	126	0	78	404	true
http://192.168.1.10:5005/test	103	0	63	404	true
http://192.168.1.10:5005/	112	0	59	200	true
http://192.168.1.10:5005/dashboard	106	0	15	200	true
http://192.168.1.10:5005/dashboard	104	0	15	200	true
http://192.168.1.10:5005/dashboard	83	0	15	200	true
http://192.168.1.10:5005/test	183	0	80	404	true

http://192.168.1.10:5005/	108	0	22	200	true
http://192.168.1.10:5005/	101	0	25	200	true
http://192.168.1.10:5005/	104	0	12	200	true
http://192.168.1.10:5005/test	168	0	77	404	true
http://192.168.1.10:5005/	117	0	33	200	true
http://192.168.1.10:5005/test	180	0	90	404	true
http://192.168.1.10:5005/buatbaru	397	0	32	200	true
http://192.168.1.10:5005/dashboard	134	0	55	200	true
http://192.168.1.10:5005/dashboard	138	0	54	200	true
http://192.168.1.10:5005/dashboard	139	0	89	200	true
http://192.168.1.10:5005/buatbaru	406	0	72	200	true
http://192.168.1.10:5005/dashboard	117	0	47	200	true
http://192.168.1.10:5005/	130	0	55	200	true
http://192.168.1.10:5005/buatbaru	527	0	72	200	true
http://192.168.1.10:5005/buatbaru	512	0	79	200	true
http://192.168.1.10:5005/	127	0	56	200	true
http://192.168.1.10:5005/	132	0	49	200	true
http://192.168.1.10:5005/test	211	0	120	404	true
http://192.168.1.10:5005/test	199	0	108	404	true
http://192.168.1.10:5005/test	197	0	104	404	true
http://192.168.1.10:5005/	100	0	45	200	true
http://192.168.1.10:5005/dashboard	225	0	110	200	true
http://192.168.1.10:5005/buatbaru	397	0	76	200	true
http://192.168.1.10:5005/dashboard	179	0	109	200	true
http://192.168.1.10:5005/dashboard	156	0	90	200	true
http://192.168.1.10:5005/buatbaru	458	0	44	200	true
http://192.168.1.10:5005/dashboard	131	0	80	200	true
http://192.168.1.10:5005/buatbaru	473	0	42	200	true
http://192.168.1.10:5005/buatbaru	351	0	111	200	true
http://192.168.1.10:5005/	158	0	83	200	true
http://192.168.1.10:5005/dashboard	66	0	14	200	true
http://192.168.1.10:5005/buatbaru	324	0	42	200	true
http://192.168.1.10:5005/buatbaru	356	0	39	200	true
http://192.168.1.10:5005/buatbaru	489	0	67	200	true
http://192.168.1.10:5005/buatbaru	435	0	47	200	true
http://192.168.1.10:5005/buatbaru	306	0	38	200	true

http://192.168.1.10:5005/	57	0	27	200	true
http://192.168.1.10:5005/test	39	0	15	404	true
http://192.168.1.10:5005/dashboard	52	0	8	200	true
http://192.168.1.10:5005/	80	0	8	200	true
http://192.168.1.10:5005/test	54	0	27	404	true
http://192.168.1.10:5005/dashboard	52	0	13	200	true
http://192.168.1.10:5005/	39	0	5	200	true
http://192.168.1.10:5005/dashboard	50	0	3	200	true
http://192.168.1.10:5005/test	77	0	35	404	true
http://192.168.1.10:5005/test	91	0	33	404	true
http://192.168.1.10:5005/buatbaru	369	0	42	200	true
http://192.168.1.10:5005/	50	0	13	200	true
http://192.168.1.10:5005/dashboard	28	0	7	200	true
http://192.168.1.10:5005/test	35	0	15	404	true
http://192.168.1.10:5005/test	38	0	13	404	true
http://192.168.1.10:5005/test	37	0	15	404	true
http://192.168.1.10:5005/test	41	0	8	404	true
http://192.168.1.10:5005/	43	0	16	200	true
http://192.168.1.10:5005/buatbaru	374	0	43	200	true
http://192.168.1.10:5005/	65	0	22	200	true
http://192.168.1.10:5005/buatbaru	499	0	38	200	true
http://192.168.1.10:5005/dashboard	114	0	26	200	true
http://192.168.1.10:5005/buatbaru	283	0	16	200	true
http://192.168.1.10:5005/dashboard	219	0	48	200	true
http://192.168.1.10:5005/test	91	0	50	404	true
http://192.168.1.10:5005/test	116	0	51	404	true
http://192.168.1.10:5005/test	115	0	49	404	true
http://192.168.1.10:5005/test	163	0	45	404	true
http://192.168.1.10:5005/test	110	0	15	404	true
http://192.168.1.10:5005/	154	0	37	200	true
http://192.168.1.10:5005/	82	0	50	200	true
http://192.168.1.10:5005/	45	0	17	200	true
http://192.168.1.10:5005/	79	0	51	200	true
http://192.168.1.10:5005/dashboard	63	0	36	200	true
http://192.168.1.10:5005/	82	0	33	200	true
http://192.168.1.10:5005/dashboard	37	0	26	200	true

http://192.168.1.10:5005/dashboard	47	0	23	200	true
http://192.168.1.10:5005/dashboard	116	0	23	200	true
http://192.168.1.10:5005/dashboard	92	0	7	200	true
http://192.168.1.10:5005/buatbaru	472	0	48	200	true
http://192.168.1.10:5005/buatbaru	510	0	41	200	true
http://192.168.1.10:5005/test	120	0	58	404	true
http://192.168.1.10:5005/buatbaru	263	0	18	200	true
http://192.168.1.10:5005/	45	0	9	200	true
http://192.168.1.10:5005/dashboard	108	0	22	200	true
http://192.168.1.10:5005/buatbaru	414	0	52	200	true
http://192.168.1.10:5005/buatbaru	502	0	76	200	true
http://192.168.1.10:5005/test	137	0	51	404	true
http://192.168.1.10:5005/buatbaru	480	0	50	200	true
http://192.168.1.10:5005/buatbaru	482	0	51	200	true
http://192.168.1.10:5005/test	133	0	100	404	true
http://192.168.1.10:5005/	96	0	64	200	true
http://192.168.1.10:5005/	52	0	22	200	true
http://192.168.1.10:5005/test	98	0	52	404	true
http://192.168.1.10:5005/dashboard	53	0	9	200	true
http://192.168.1.10:5005/	83	0	40	200	true
http://192.168.1.10:5005/dashboard	54	0	8	200	true
http://192.168.1.10:5005/buatbaru	343	0	63	200	true
http://192.168.1.10:5005/dashboard	46	0	8	200	true
http://192.168.1.10:5005/	79	0	46	200	true
http://192.168.1.10:5005/dashboard	40	0	21	200	true
http://192.168.1.10:5005/buatbaru	331	0	14	200	true
http://192.168.1.10:5005/test	147	0	94	404	true
http://192.168.1.10:5005/test	128	0	82	404	true
http://192.168.1.10:5005/buatbaru	507	0	8	200	true
http://192.168.1.10:5005/buatbaru	460	0	13	200	true
http://192.168.1.10:5005/test	54	0	15	404	true
http://192.168.1.10:5005/buatbaru	288	0	8	200	true
http://192.168.1.10:5005/	75	0	30	200	true
http://192.168.1.10:5005/test	88	0	35	404	true
http://192.168.1.10:5005/test	95	0	46	404	true
http://192.168.1.10:5005/test	103	0	49	404	true

http://192.168.1.10:5005/test	104	0	50	404	true
http://192.168.1.10:5005/dashboard	100	0	33	200	true
http://192.168.1.10:5005/	94	0	53	200	true
http://192.168.1.10:5005/dashboard	219	0	111	200	true
http://192.168.1.10:5005/	108	0	31	200	true
http://192.168.1.10:5005/test	235	0	75	404	true
http://192.168.1.10:5005/	113	0	28	200	true
http://192.168.1.10:5005/	204	0	33	200	true
http://192.168.1.10:5005/buatbaru	428	0	88	200	true
http://192.168.1.10:5005/dashboard	291	0	109	200	true
http://192.168.1.10:5005/dashboard	323	0	154	200	true
http://192.168.1.10:5005/dashboard	255	0	134	200	true
http://192.168.1.10:5005/test	248	0	144	404	true
http://192.168.1.10:5005/buatbaru	548	0	146	200	true
http://192.168.1.10:5005/test	286	0	176	404	true
http://192.168.1.10:5005/test	249	0	142	404	true
http://192.168.1.10:5005/test	182	0	62	404	true
http://192.168.1.10:5005/	323	0	157	200	true
http://192.168.1.10:5005/dashboard	214	0	82	200	true
http://192.168.1.10:5005/	177	0	91	200	true
http://192.168.1.10:5005/	165	0	78	200	true
http://192.168.1.10:5005/buatbaru	478	0	156	200	true
http://192.168.1.10:5005/	171	0	87	200	true
http://192.168.1.10:5005/buatbaru	619	0	90	200	true
http://192.168.1.10:5005/buatbaru	610	0	95	200	true
http://192.168.1.10:5005/	208	0	105	200	true
http://192.168.1.10:5005/dashboard	109	0	63	200	true
http://192.168.1.10:5005/dashboard	123	0	67	200	true
http://192.168.1.10:5005/dashboard	112	0	53	200	true
http://192.168.1.10:5005/dashboard	113	0	27	200	true
http://192.168.1.10:5005/buatbaru	512	0	248	200	true
http://192.168.1.10:5005/test	140	0	46	404	true
http://192.168.1.10:5005/	41	0	23	200	true
http://192.168.1.10:5005/	65	0	39	200	true
http://192.168.1.10:5005/buatbaru	356	0	92	200	true
http://192.168.1.10:5005/buatbaru	393	0	80	200	true

http://192.168.1.10:5005/buatbaru	460	0	61	200	true
http://192.168.1.10:5005/buatbaru	397	0	80	200	true
http://192.168.1.10:5005/	37	0	4	200	true
http://192.168.1.10:5005/	61	0	34	200	true
http://192.168.1.10:5005/	37	0	3	200	true
http://192.168.1.10:5005/	99	0	64	200	true
http://192.168.1.10:5005/	51	0	8	200	true
http://192.168.1.10:5005/	55	0	22	200	true
http://192.168.1.10:5005/	37	0	5	200	true
http://192.168.1.10:5005/	20	0	7	200	true
http://192.168.1.10:5005/	84	0	67	200	true
http://192.168.1.10:5005/	21	0	14	200	true
http://192.168.1.10:5005/	48	0	35	200	true
http://192.168.1.10:5005/	38	0	19	200	true
http://192.168.1.10:5005/	177	0	12	200	true
http://192.168.1.10:5005/	67	0	40	200	true
http://192.168.1.10:5005/	76	0	38	200	true
http://192.168.1.10:5005/	11	0	3	200	true
http://192.168.1.10:5005/	27	0	8	200	true
http://192.168.1.10:5005/	23	0	7	200	true
http://192.168.1.10:5005/	31	0	12	200	true
http://192.168.1.10:5005/	549	0	238	200	true
http://192.168.1.10:5005/	106	0	85	200	true
http://192.168.1.10:5005/	51	0	30	200	true
http://192.168.1.10:5005/	32	0	19	200	true
http://192.168.1.10:5005/	131	0	19	200	true
http://192.168.1.10:5005/	73	0	27	200	true
http://192.168.1.10:5005/	85	0	42	200	true
http://192.168.1.10:5005/	58	0	39	200	true
http://192.168.1.10:5005/	89	0	63	200	true
http://192.168.1.10:5005/	87	0	60	200	true
http://192.168.1.10:5005/	39	0	4	200	true
http://192.168.1.10:5005/	56	0	5	200	true
http://192.168.1.10:5005/	56	0	5	200	true
http://192.168.1.10:5005/	56	0	5	200	true
http://192.168.1.10:5005/	60	0	5	200	true

http://192.168.1.10:5005/	47	0	37	200	true
http://192.168.1.10:5005/	80	0	55	200	true
http://192.168.1.10:5005/	45	0	14	200	true
http://192.168.1.10:5005/	46	0	22	200	true
http://192.168.1.10:5005/	47	0	24	200	true