**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNOLOGI INFORMASI**

**INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**USULAN TUGAS AKHIR**

# IDENTITAS PENGUSUL

**NAMA : AJI SETYO UTOMO**

**NRP : 5110100010**

**DOSEN WALI : Prof. Ir.Supeno Djanali, M.Sc.Ph.D.**

**DOSEN PEMBIMBING : 1. Waskitho Wibisono, S.Kom., M.Eng., Ph.D.**

1. **Royyana Muslim Ijtihadie, S.Kom., M.Kom., Ph.D.**

# JUDUL TUGAS AKHIR

“Manajemen Sumber Daya Untuk Meningkatkan Efisiensi Beban Kerja pada Komputasi Awan Menggunakan Kerangka Kerja OpenStack”

# LATAR BELAKANG

Perkembangan teknologi komputasi mengalami peningkatan yang sangat signifikan dari waktu ke waktu dan menjadi salah satu aspek yang paling penting dalam perkembangan teknologi informasi. Meningkatnya kebutuhan akan penggunaan teknologi komputasi yang didukung dengan sumber daya *server* dengan skalabilitas dan penyimpanan data yang semakin besar mendorong untuk beralih ke teknologi komputasi berbasis komputasi awan. Teknologi komputasi awan meyediakan layanan penyedia processor (*computing power),* penyimpanan (*storage*), jaringan (*network*) yang kemudian layanan tersebut diberikan kepada pengguna komputasi awan[1]. Dewasa ini banyak penyedia layanan awan yang sedang berkembang di antaranya penyedia infrastruktur sebagai layanan awan seperti OpenStack, Eucalyptus, CloudStack, Cloud Era, Puppet, dan Open nebula. Komputasi awan bisa dikatakan sebagai sumber daya untuk komputasi dan penyimpanan data dalam bentuk mesin virtual. Kehadiran komputasi awan dapat membantu ketersediaan *server* dan penyimpanan dalam membangun aplikasi yang dijalankan secara virtual.

OpenStack merupakan kerangka kerja *open-source* penyedia infrastuktur komputasi awan*.* OpenStack mempunyai kemampuan skalabilitas yang lebih besar dibandingkan kerangka kerja awan lainnya. Tujuan OpenStack adalah untuk memungkinkan setiap organisasi atau perusahaan membuat dan menyediakan layanan komputasi awan dengan menggunakan perangkat lunak *open-source* yang berjalan di atas perangkat keras yang standar. Di dalam OpenStack terdapat beberapa komponen yang memudahkan pengguna dalam melakukan konfigurasi infrastruktur awan, di antaranya adalah: OpenStack Compute, OpenStack Storage, dan OpenStack Image Service. OpenStack Compute adalah komponen yang berfungsi melakukan otomasi saat membuat ataupun mengelola VPS (*Virtual Private Server*) dalam jumlah besar. Sedangkan OpenStack Storage merupakan komponen dalam kerangka kerja OpenStack yang bertugas untuk membuat *object storage* yang bersifat *scalable* dengan menggunakan *cluster* untuk menyimpan data. Sedangkan OpenStack Image Service merupakan layanan OpenStack dalam *virtual image*. OpenStack saat ini telah digunakan perusahaan besar *hosting* seperti Rackspace Hosting dan NASA. Mereka menggunakan teknologi OpenStack untuk mengelola puluhan ribu *compute instance* dan *storage* dalam ukuran *petabyte*.

OpenStack memiliki keunggulan dengan banyaknya jumlah *instance* atau *virtual machine* yang dapat dibuat. Diperlukan alokasi sumber daya (*bandwidth, memory,* CPU (*Central Processing Unit*)*, storage*) di awal pembuatan sebuah *server instance* yang disesuaikan dengan kebutuhan. Tetapi alokasi sumber dayaseperti itu dirasa kurang maksimal, karena komputasi dari suatu *instance* bisa saja berubah-ubah sesuai tugas yang diberikan secara *real-time*. Sehingga jika suatu *instance* melakukan komputasi besar dengan kemampuan sumber daya kecil tentunya hal ini dapat menghambat komputasi, begitu juga sebaliknya jika suatu *instance* melakukan komputasi kecil dengan kemampuan sumber daya besar tentunya hal ini menjadikan sumber dayatidak digunakan maksimal.

Pada Tugas Akhir ini akan dibuat sebuah sistem pengolahan sumber daya komputasi awan untuk mengatur efisiensi beban saat komputasi. Pada setiap *instance* yang memiliki kemampuan komputasi yang berbeda akan disesuiakan dengan beban komputasi sesuaisumber daya yang ada. Hal ini diperlukan agar kinerja dalam pembagian sumber daya komputasi awan menjadi lebih maksimal.

# RUMUSAN MASALAH

Rumusan masalah dari Tugas Akhir ini adalah sebagai berikut:

1. Membangun dan merancang OpenStack dengan sumber daya mandiri.
2. Membangun manajemen sumber daya di OpenStack.
3. Strategi pengolahan sumber daya pada *instance* atau *virtual machine*.

# BATASAN MASALAH

Batasan masalah pada Tugas Akhir ini adalah sebagai berikut:

1. Komputasi awan dengan kerangka kerja OpenStack*.*
2. Menggunakan dua komputer untuk infrastuktur awan.
3. Menggunakan Ubuntu Enterprise Cloud 64 bit sebgai sistem operasi pada *server*.

# TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan dari pengerjaan Tugas Akhir ini adalah membangun sistem komputasi awan berbasis *open-source* yang dapat melakukan pegawasan sumber daya dalam menentukan beban kerja komputasi.

# MANFAAT TUGAS AKHIR

Manfaat Tugas Akhir ini adalah pengguna dapat meningkatkan efisiensi sumber daya (*bandwidth, memory,* CPU (*Central Processing Unit*)*, storage*) pada *instance* dalam menentukan beban kerja komputasi.

# TINJAUAN PUSTAKA

* 1. **Komputasi awan**

Komputasi awanmerupakan sebuah model komputasi pada sumber daya seperti *processor* (*computing power),* penyimpanan (*storage*), jaringan (*network*) menjadi abstrak dan diberikan sebagai layanan di jaringan atau internet. Ketersediaan *on-demand* sesuai kebutuhkan, mudah untuk melakukan pengaturan, dinamik dan skalabilitas yang hampir tanpa limit adalah beberapa atribut penting dari komputasi awan**.** Terdapat tiga layanan yang disediakan oleh komputasi awan:

1. *Software as a Service* (SaaS)

SaaS adalah layanan dari komputasi awan dimana *pengguna* dapat menggunakan perangkat lunak yang telah disediakan oleh penyedia layanan awan.

1. *Platform as a Service* (PaaS)

Paas adalah layanan dari komputasi awan dimana pengguna dapat mengembangkan komputasi awan menjadi layanan SaaS. Biasanya sudah terdapat sistem operasi, *database*, *web server* dan kerangka kerja aplikasi agar dapat menjalankan aplikasi yang telah dibuat. Keuntungan layanan PaaS ini bagi pengembang adalah mereka dapat fokus pada aplikasi yang mereka buat tanpa memikirkan pemeliharaan dari *computing platform*.

1. *Infrastructure as a Service* (IaaS)

IaaS adalah layanan dasar dari komputasi awan, di layanan ini pengguna dapat merancang dan membangun sistem komputasi awan sendiri. Sebagai contoh, saat komputer virtual tersebut sudah kelebihan beban, kita bisa tambahkan CPU*, RAM,* dan *storage* [2].

Penggunaan awan semakin lama semakin berkembang. Saat ini banyak teknologi yang beralih ke komputasi awan karena sifatnya yangpraktis*.* Manfaat menggunakan komputasi awan sebagai berikut:

1. [Skalabilitas](http://id.wikipedia.org/w/index.php?title=Skalabilitas&action=edit&redlink=1), yaitu dapat menambah kapasitas penyimpanan data tanpa harus membeli peralatan tambahan, misalnya [*hardisk*](http://id.wikipedia.org/wiki/Hardisk), *prosesor*,RAM, dan lain sebagainya. Cukup dengan menambah kapasitas yang disediakan oleh penyedia layanan komputasi awan.
2. [Aksesibilitas](http://id.wikipedia.org/wiki/Aksesibilitas), yaitu akses tidak terikat oleh waktu dan tempat dengan syarat terhubung dengan [internet](http://id.wikipedia.org/wiki/Internet), sehingga memudahkan mengakses data.
3. [Kreasi](http://id.wikipedia.org/w/index.php?title=Kreasi&action=edit&redlink=1), yaitu para pengguna bisa melakukan atau mengembangkan kreasi atau proyek mereka tanpa harus mengirimkan proyek mereka secara langsung ke perusahaan, tetapi pengguna dapat mengirimnya melalui penyedia layanan komputasi awan.
4. [Kecemasan](http://id.wikipedia.org/wiki/Kecemasan), ketika terjadi bencana alam data tersimpan aman di *cloud* meskipun *hardisk* atau [*gadget*](http://id.wikipedia.org/wiki/Gadget) rusak.
   1. **Ubuntu Enterprise Cloud**

Ubuntu Enterprise Cloud (UEC) adalah sebuah kumpulan aplikasi dari Canonical yang termasuk dalam Ubuntu Server. UEC versi terbaru menggunakan OpenStack bersama sejumlah *software open-source* lainnya. UEC dapat digunakan juga untuk sistem operasi awal dalam membangun komputasi awan[3].

* 1. **OpenStack**

OpenStack adalah sebuah kerangka kerja *open-source* dalam komputasi awan yang berorientasi di bidang IaaS. OpenStack mengendalikan proses komputasi dan sumber daya jaringan dalam sebuah *data center* melalui *dashboard* yang memberikan kontrol adminstrasi sekaligus memberikan hak akses pada pengguna melalui antarmuka *website*. OpenStack menyediakan sistem operasi awan untuk *public* dan *private cloud* di bawah Apache License. Adapun arsitektur utama OpenStack dapat dilihat pada Gambar 1. Saat ini Dalam membangun OpenStack terdiri dari tiga project:

1. OpenStack Compute (*code-name*: Nova)

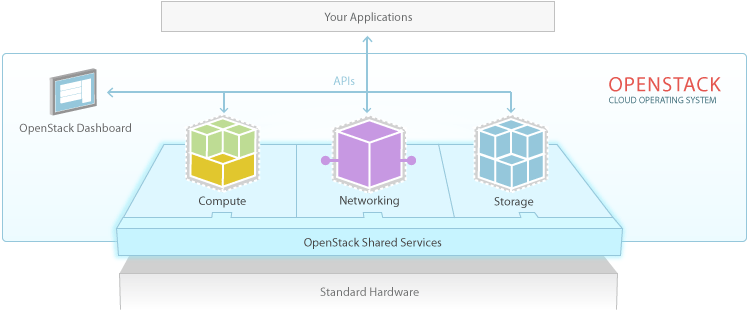
Adalah *open-source software* yang didesain untuk mengatur jaringan-jaringan dalam skala besar, virtual mesin serta menciptakan *platform* yang *scalable* untuk *cloud-computing*.

1. OpenStack Object Storage (*code-name*: Swift)

Adalah *open-source* *software* untuk membuat data *storage* yang *scalable* menggunakan *cluster server* untuk menyimpan data dengan skala *petabytes*.

1. OpenStack Image Service (*code-name*: Glance)

Adalah salah satu produk dari OpenStack yang digunakan untuk layanan virtual *disk images*[4].



Gambar 1. Arsitektur Utama OpenStack.

Dalam Tugas Akhir ini penulis menggunakan OpenStack sebagai kerangka kerja dalam membangun komputasi awan.

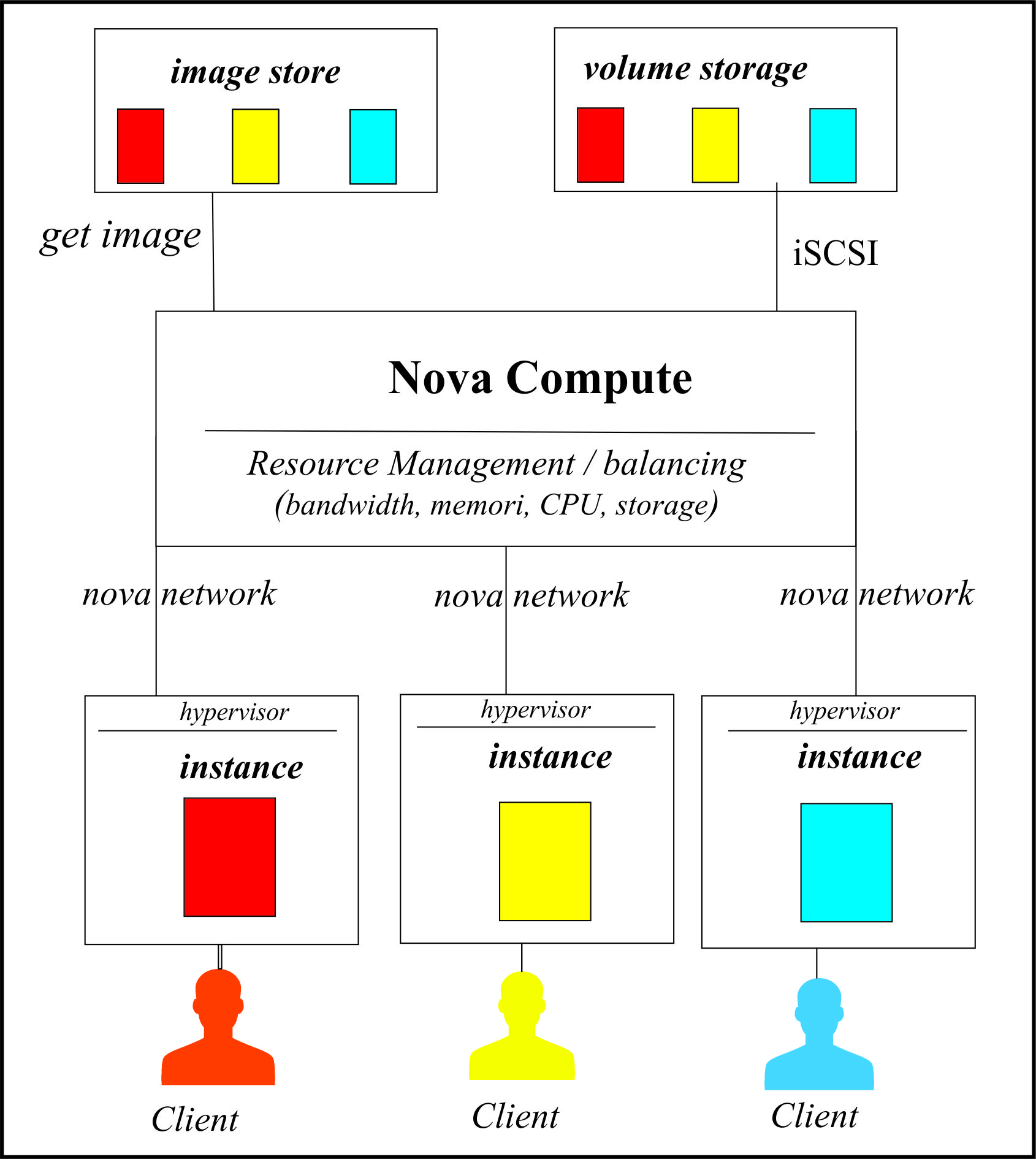
* 1. **OpenStack Neat**

OpenStack Neat adalah perluasan dalam menerapkan pengaturan VM (V*irtual Machine)* atau *instance* yang dinamis dengan menggunakan metode *live migration*. Tujuan utama dari dinamis *instance* adalah untuk meningkatkan pemanfaatan sumber daya *physical* dan mengurai konsumsi energi dengan mengalokasikan kembali *instance* *idle* menjadi *sleep mode*. Sebagai contoh, terdapat dua *instance* yang ditempatkan dengan *host* yang berbeda dengan kapasitas sumber daya dari salah satu *instance* menjalankan suatu proses dengan beban yang berat, kemudian salah satu *instance* yang lain dapat bermigrasi ke *instance* lainnya. *Instance* yang sedang *idle* dapatbermigrasi ke mode daya rendah untuk menghemat energi. Ketika sumber daya dari permintaan salah satu *instance* meningkat kembali maka akan ada perubahan *instance* lagi untuk menghindari penurunan kinerja. Proses ini dapat dilakukan secara dinamis dengan perpaduan OpenStack Neat [5]. Dalam Tugas Akhir ini penulis menggunakan OpenStack Neat sebagai acuan dalam mengatur dan mengelola *instance* yang terdapat pada OpenStack.

# RINGKASAN ISI TUGAS AKHIR

Tugas Akhir yang diusulkan adalah sebuah sistem komputasi awan berbasis *open-source* yangdibangun menggunakan kerangka kerja OpenStackdengan sistem manajemen sumber daya komputasi awan yang dapat memaksimalkan penggunaan sumber daya pada *instance* atau *virtual machine* secara *real-time*. Sistem manajemen sumber daya dibangun dengan tujuan untuk dapat mengatur segala aktivitas komputasi pada *instance*. Baik lalu lintas jaringan (*bandwidth*), proses komputasi (CPU*)*, proses penyimpanan (*storage*), dan banyaknya proses pada setiap *instance* (*memory*). Sistem juga bertugas untuk mengatur ketersediaan sumber daya ketika *client* melakukan permintaan komputasi dengan beban *instance server* yang berat maka sistemakan menggunakan alokasi sumber daya dari *instance* lain yang komputasinya lebih ringan.

Dalam OpenStack pengolahan sumber daya *instance* dilakukan di OpenStack Compute (NOVA). Adapun contoh arsitektur sistem dapat dilihat pada Gambar 2.

**

Gambar 2. Arsitektur Sistem OpenStack denganmanajemen sumber daya*.*

# METODOLOGI

## Penyusunan proposal Tugas Akhir

Proposal Tugas Akhir ini berisi tentang sistem manajemen sumber daya komputasi awan yang dapat memaksimalkan penggunaan sumber daya pada *instance* secara *real-time*. Sistem ini dapat mengatur beban komputasi pada setiap *instance*, dimana kemampuan kinerja komputasi yang berbeda pada setiap *instance* dapat menyesuaikan dengan alokasi sumber daya yang ada. Hal ini diperlukan agar alokasi sumber daya komputasi awan menjadi lebih optimal.

## Studi literatur

Pada tahap ini dilakukan pencarian data dan pembelajaran literatur antara lain sebagai berikut:

1. Perancangan komputasi awan dengan sumber daya mandiri*.*
2. Perancangan IaaS (*Infrastructure as Service*) dengan menggunakan kerangka kerja OpenStack.
3. Pengembangan sistem dalam manajemen sumber daya *instance* atau *virtual machine* dengan acuan OpenStack Neat.

## Analisis dan desain perangkat lunak

Pada Gambar 2 merupakan gambaran tentang arsitektur sistem yang akan penulis rancang. Sistem ini dibagun engan menggunakan kerangka layanan OpenStack Computeatau *code-name* Nova. Sistem yang dibangun adalah sistem yang dapat mengatur beban komputasi dari sumber daya yang tersedia. Ada beberapa parameter untuk menentukan beban dalam komputasi di antaranya adalah:

* + 1. *Bandwidth*, hal ini berpengaruh pada cepat servis yang akan diperoleh *client.* Secara garis besar, semakin besar *bandwidth* yang disediakan oleh *server* maka akan cepat pula respon yang diterima oleh *client.*
    2. *Memory*, hal ini sangat berpengaruh pada jumlah pekerjaan yang dapat dilakukan per satuan waktu. Memoriyang besar akan mendukung *instance* dan *server* dalam memproses data yang besar pula. Memori pada *instance* ditentukan dan tidak boleh melewati batas dari *server cloud*, karena *server cloud* juga membutuhkan memori dalam menjalankan banyak *instance* yang lain dalam satu satuan waktu.
    3. CPU*,* hal ini merupakan komponen terpenting alam menjalankan proses pekerjaan suatu *instance*.
    4. *Storage*, hal ini merupakan komponen dalam penyimpanan [6].

Dari empat kategori beban yang telah dijelaskan dapat yang dapat dijadikan acuan utama dalam menentukan proses pengaturan penyesuaian sumber daya pada *instance* adalahCPUdanmemori, dimana dalam menentukan *balancing* terdapat proses pengaturan dalam menentukan *instance* mana yang memiliki beban komputasi lebih besar dan *instance* mana yang memiliki beban komputasi lebih kecil. Setelah proses *sorting*, didapatkan data beban CPU dan memori dari komputasi semua *instance*. Sehingga sistem dapat menentukan *instance* mana yang akan digunakan untuk menambah sumber dayaCPUdanmemori yang lain. Diagram alur proses manajemen sumber daya komputasi dapat dilihat di Gambar 3.

* 1. **Implementasi perangkat lunak**

Rencana pembuatan perangkat lunak ini akan diimplementasikan dengan menggunakan:

1. Sistem operasi *server* mengunakan Ubuntu Enterprise Cloud.
2. Menggunakan OpenStack dalam membangun layanan *Infrastructure as Service*.
3. Menngunakan OpenStack Neat untuk acuan dalam mengembangkan *instance*.

## Pengujian dan evaluasi

Pada tahap ini pengujian sistem manajemen sumber dayauntuk meningkatkan efisiensi komputasi dilakukan dengan beberapa pengujian sebagai berikut:

1. Melakukan pengujian untuk *instance* yang memiliki lalu-lintas komputasi padat dan tidak.
2. Melakukan pengujian terhadap efisiensi pada pembagian sumber daya.



Gambar 3. Diagaram alir sistem manajemen sumber daya*.*

## Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam membangun sistem manajemensumber daya pada komputasi awan serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku Tugas Akhir secara garis besar antara lain:

1. Pendahuluan
   1. Latar Belakang
   2. Rumusan Masalah
   3. Batasan Tugas Akhir
   4. Tujuan
   5. Metodologi
   6. Sistematika Penulisan
2. Tinjauan Pustaka
3. Desain dan Implementasi
4. Pengujian dan Evaluasi
5. Kesimpulan dan Saran
6. Daftar Pustaka

# JADWAL KEGIATAN

Jadwal pengerjaan Tugas Akhir ini akan dilakukan sesuai dengan rencana pengerjaan seperti ditunjukkan pada Tabel 1.

Tabel 1. Jadwal Pengerjaan Tugas Akhir

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Tahapan** | **2014** | | | | | | | | | | | | | | | | |
| **Maret** | | | | **April** | | | |  | **Mei** | | | | **Juni** | | | |
| Penyusunan Proposal |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Studi Kepustakaan |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Desain Sistem |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Implementasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Uji Coba dan Evaluasi |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Penyusunan Buku Tugas Akhir |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1. **DAFTAR PUSTAKA**

|  |  |
| --- | --- |
| [1] | onnopurbo. [Online]. Available: http://kambing.ui.ac.id/onnopurbo/ebook/ebook-voip/OWP-20110701-petunjuk-praktis-cloud-computing-menggunakan-opensource.pdf. [Diakses 5 Maret 2014]. |
| [2] | A. Budiyanto, “Pengantar Cloud Computing,” Cloudindonesia, [Online]. Available: http://www.cloudindonesia.or.id/wp-content/uploads/2012/05/E-Book-Pengantar-Cloud-Computing-R1.pdf. [Diakses 10 Maret 2014]. |
| [3] | A. Jha, Johnson D, K. Murari, M. Raju, V. Cherian and Y. Girikumar, in *OpenStack Beginner's Guide for Ubuntu - Precis*, vol. 3.0, CSS CROP, 2012, pp. 10-12. |
| [4] | OpenStack Cloud Software, “OpenStack: The Open Source Cloud Operating System,” [Online]. Available: http://www.openstack.org/software/. [Diakses 5 Maret 2013]. |
| [5] | beloglazov, “Dynamic Consolidation of Virtual Machines on OpenStack,” OpenStack Neat, [Online]. Available: http://openstack-neat.org/. [Diakses 12 Maret 2014]. |
| [6] | I. Sofana, Teori dan Praktik Cloud Computing (OpenNebula, VMware, dan amazon AWS), Bandung: Penerbit Informatika, 2012, pp. 108-114. |