

USULAN TUGAS AKHIR

1. IDENTITAS PENGUSUL

NAMA : Nandez Darras Dhiya'Ul Haq
NRP : 5110 100 088
DOSEN WALI : Ahmad Saikhu, S.SI., MT.
DOSEN PEMBIMBING : 1. Ary Mazharuddin Shiddiqi, S.Kom, M.Comp.Sc
2. Baskoro Adi Pratomo, S.Kom, M.Kom

2. JUDUL TUGAS AKHIR

“Penerapan Metode Round-Robin Adaptif Pada *Cluster* Server Berbasis Nginx”

3. LATAR BELAKANG

Perkembangan aplikasi web yang semakin pesat sejak munculnya teknologi internet sangat membantu dalam kemudahan serta kecepatan pengiriman, penyampaian dan penerimaan informasi. Dari segala aspek telah banyak memanfaatkan aplikasi web. Sejalan dengan ini maka beban kerja server-server akan meningkat secara signifikan. Terutama pada jam-jam tertentu yang membutuhkan kinerja maksimal.

Untuk mengatasi beban berlebih, dapat menggunakan sebuah server yang dapat di-*upgrade* sehingga dapat menangani masalah kelebihan beban tersebut. Namun solusi ini memiliki kelemahan. Antara lain biaya yang tinggi karena server diharuskan selalu untuk di-*upgrade* ketika mengalami kelebihan beban lagi dan server tidak selalu dapat di-*upgrade*. Sehingga dibutuhkan biaya lebih untuk membeli server yang baru.

Solusi lain adalah menggunakan sebuah sistem *cluster* yang dimanajemen oleh *load balancer* sehingga dapat membuat dua atau lebih server menjadi satu kesatuan yang dapat memperoleh *high availability* serta menutupi kemungkinan *failover*, sehingga

ketika beban sebuah server meningkat maka dapat ditambahkan sebuah server ke dalam *cluster* untuk menangani *request* tambahan tersebut.

Pada proposal tugas akhir ini, penulis akan membuahkan sebuah aplikasi *load balancer* menggunakan OpenResty dengan Lua *scripting* yang akan mengatur dan membagi hasil *request* dari klien ke server dan dengan algoritma round-robin adaptif, aplikasi akan membagi *request* berdasarkan beban server yang diterima. Semakin kecil beban yang diterima server, maka server tersebut yang akan menerima *request* pertama kali.

4. RUMUSAN MASALAH

Berikut beberapa hal yang menjadi rumusan masalah dalam tugas akhir ini:

- a. Bagaimana menerapkan algoritma round-robin adaptif untuk membagi *request* agar beban yang diterima masing-masing server sama besar?
- b. Bagaimana mendistribusikan kondisi masing-masing server pada algoritma round-robin adaptif?

5. BATASAN MASALAH

Dari permasalahan yang telah diuraikan di atas, terdapat beberapa batasan masalah pada tugas akhir ini, yaitu:

- a. Bahasa pemrograman yang digunakan adalah Lua versi 5.1.5, bahasa Lua adalah bahasa yang disupport oleh OpenResty.
- b. OpenResty yang digunakan adalah OpenResty versi 1.4.3.6.
- c. Nginx yang digunakan adalah versi 1.4.5.
- d. Kumpulan server yang di-*manage* oleh aplikasi harus berada dalam satu subnet dengan aplikasi.
- e. Server yang menjalankan nginx adalah berbasis linux.
- f. Analisa kemampuan server akan dilakukan menggunakan tool Apache Benchmark.

6. TUJUAN PEMBUATAN TUGAS AKHIR

Tujuan pembuatan tugas akhir ini adalah menciptakan sebuah sistem yang dapat membagi rata beban server pada web server Nginx sesuai algoritma round-robin adaptif dengan bantuan OpenResty.

7. MANFAAT TUGAS AKHIR

Dengan dibangunnya aplikasi ini, diharapkan pembagian beban server dalam menerima *request* dari klien dapat terbagi rata sesuai kondisi beban yang dialami server tersebut saat akan menerima *request*.

8. TINJAUAN PUSTAKA

a. OpenResty

OpenResty adalah sebuah aplikasi web server yang terdiri dari web server Nginx, modul-modul untuk Nginx, serta sebagian dari dependensi eksternal mereka.

Dengan banyaknya keuntungan yang dapat diambil dari penggunaan modul nginx, membuat OpenResty menjadi web server yang sangat efektif. Dimana para pengembang web dapat menggunakan bahasa pemrograman Lua sebagai modul Nginx C dan dapat membangun sebuah aplikasi yang dapat menangani lebih dari 10000 koneksi [1].

b. Nginx

Nginx (dibaca “*engine x*”) adalah http dan reverse proxy, serta mail proxy yang ditulis oleh Igor Sysoev. Nginx sejak lama telah digunakan oleh berbagai situs Rusia yang memiliki *traffic* sangat tinggi. Menurut Netcraft, Nginx melayani 17.65% situs tersibuk pada bulan Maret 2014 [2].

Igor Sysoev memulai pengembangan Nginx pada tahun 2002, dengan perilisannya ke publik pertama kali pada tahun 2004. Nginx saat ini melayani 12,18% (22,2 juta) dari semua domain yang aktif. Topologi nginx berdasarkan event *asynchronous* yang memiliki skalabilitas lebih tinggi. Berikut beberapa situs ternama yang menggunakan nginx, seperti Netflix, Hulu, Pinterest, CloudFlare, Airbnb, WordPress.com, GitHub, SoundCloud, Zynga, Eventbrite, Zappos, Media Temple, Heroku, RightScale, Engine Yard, dan MaxCDN [3].

c. Algoritma Round-Robin Adaptif

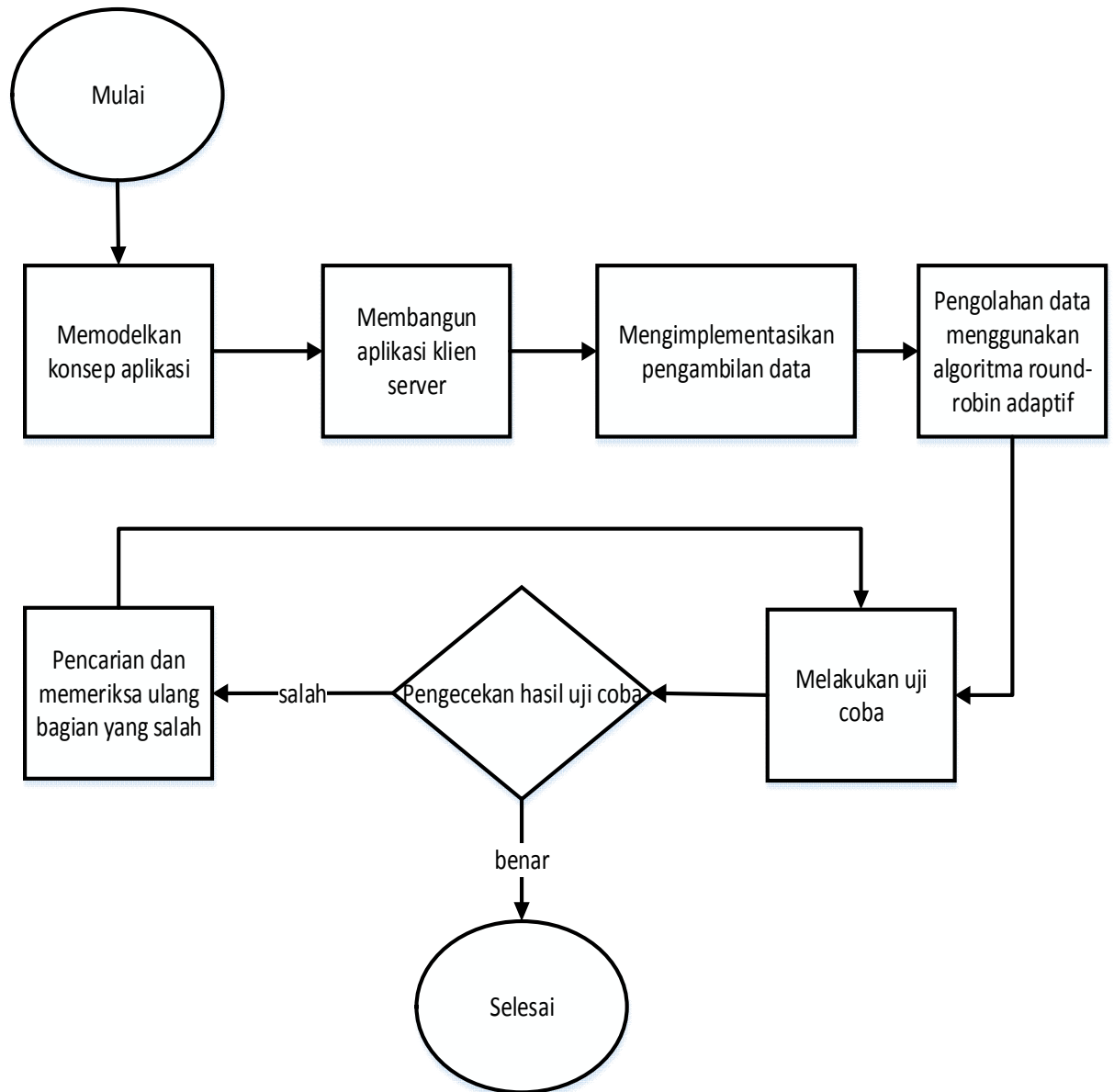
Algoritma round-robin adaptif digunakan sebagai pengembangan dari algoritma *weighted* round-robin. Algoritma ini melakukan pemantauan terhadap sistem *load* dan kemudian menyesuaikan bobot pada algoritma *weighted* round-robin. Di dalam algoritma ini terdapat definisi bobot yang merupakan nilai yang telah didefinisikan. Nilai bobot ini mempresentasikan suatu penilaian kecenderungan beban pada faktor-faktor yang telah ditentukan seperti CPU dan memori. Server dengan bobot paling rendah berarti mengalami beban paling sedikit [4].

9. RINGKASAN ISI TUGAS AKHIR

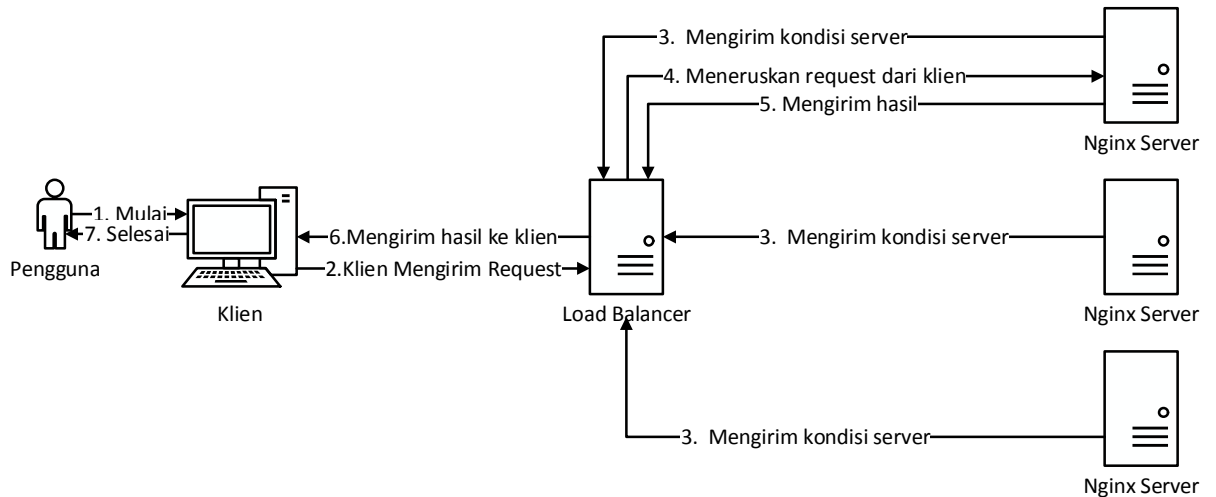
Nginx merupakan salah satu aplikasi web server yang cukup menjadi tren di tengah para administrator jaringan saat ini. Seiring dengan semakin bertambahnya pengguna internet, maka permintaan data akan meningkat secara signifikan, sehingga beban kerja server juga akan mengalami peningkatan. Oleh karena itu diperlukan suatu cara yang mampu mendistribusikan *request* ke masing-masing server pada satu sistem yang dilakukan oleh sebuah aplikasi *load balancer*.

Tugas akhir ini akan membuat sebuah aplikasi *load balancer* berdasarkan algoritma round-robin adaptif, sehingga suatu beban yang didistribusikan akan

mempertimbangkan hasil *response* dari suatu server. Berikut akan dijelaskan langkah-langkah pengerjaan tugas akhir pada Gambar 1.



Gambar 1. Alur pengerjaan proposal tugas akhir



Gambar 2. Alur kerja aplikasi

Pada Gambar 2 dapat dilihat bahwa aplikasi *load balancer* menerima *request* dari klien. Aplikasi akan mengecek beban tiap server berdasarkan algoritma round-robin adaptif. Pengumpulan data setiap server dilakukan oleh sebuah aplikasi *load collector* yang akan diteruskan untuk digunakan aplikasi *load balancer*. Data yang diambil oleh aplikasi *load collector* adalah *memory*, *cpu*, *swap*, *network i/o*, dan *disk i/o*.

10. METODOLOGI

a. Penyusunan proposal tugas akhir

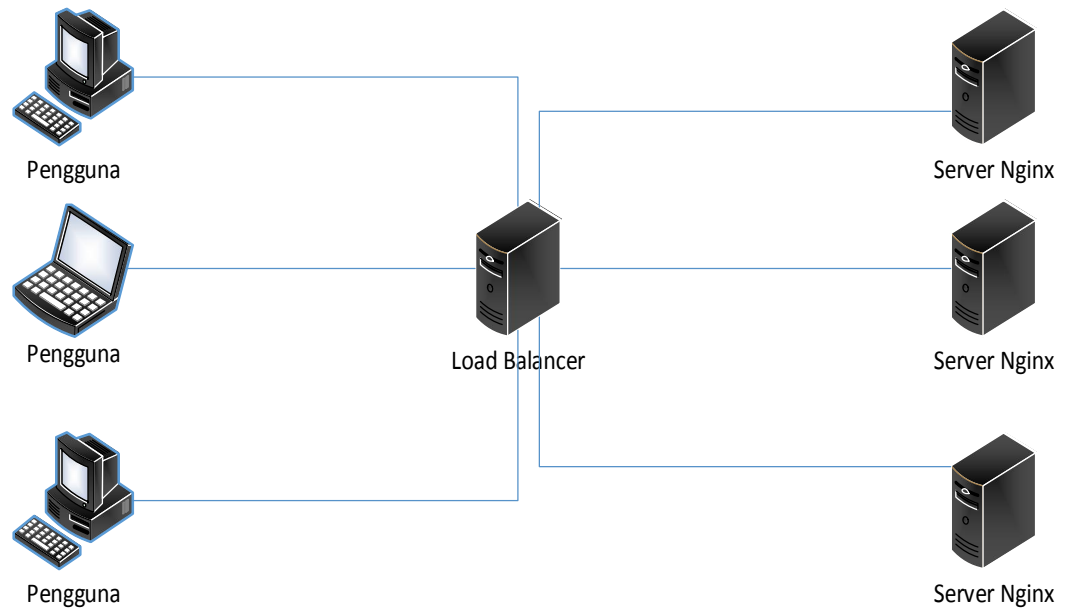
Proposal tugas akhir ini berisikan mengenai pembangunan aplikasi yang dibuat untuk melakukan proses *load balancing* menggunakan algoritma round-robin adaptif. Sehingga dapat mengurangi kelebihan beban pada server yang memiliki permintaan data tinggi.

b. Studi literatur

Tugas akhir ini menggunakan literatur *paper* beserta artikel dari internet. *Paper* yang digunakan adalah "*Load Balancing for Heterogeneous Web Servers*" [5]. *Paper* tersebut menjadi salah satu acuan dalam pengerjaan tugas akhir ini.

c. Analisis dan Desain Perangkat Lunak

Dalam aplikasi ini digunakan arsitektur klien-server, sehingga dibutuhkan server yang bertugas sebagai penyedia dan pengolah data, serta beberapa klien. Gambar 3 menunjukkan diagram arsitektur jaringan yang dibutuhkan oleh aplikasi.



Gambar 3. Arsitektur jaringan

d. Implementasi perangkat lunak

Dalam pembuatan aplikasi, digunakan beberapa teknologi untuk dapat mengaplikasikan rancangan yang sudah ada, diantaranya:

- a. OpenResty
OpenResty adalah sebuah aplikasi web server yang terdiri dari web server Nginx, modul-modul untuk Nginx, serta sebagian dari dependensi eksternal.
- b. Bahasa Pemrograman Aplikasi
Aplikasi ini dibangun dengan menggunakan bahasa pemrograman Lua. Penggunaan bahasa pemrograman diharapkan dapat membantu menangani kebutuhan aplikasi terutama kemudahan untuk konektivitas dengan basis data dan kebutuhan lainnya.
- c. Modeling Tools
Beberapa *modeling tools* yang digunakan untuk mengembangkan aplikasi ini adalah Power Designer 15.00, StarUML, dan Microsoft Visio 2013.

e. Pengujian dan Evaluasi

Pada tahap ini akan dilakukan pengujian terhadap fungsionalitas dan performa untuk mengetahui apakah masing-masing proses pada sistem sesuai dengan kebutuhan sistem dan performa sistem berdasarkan jumlah klien dan banyaknya *request*. Pada tahap evaluasi akan dilakukan pengujian terhadap hasil dari evaluasi akan perhitungan komputasi yang telah dibuat pada tahap implementasi.

f. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang menjelaskan dasar teori dan metode yang digunakan dalam tugas akhir ini serta hasil dari implementasi aplikasi perangkat lunak yang telah dibuat. Sistematika penulisan buku tugas akhir secara garis besar antara lain:

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

11. JADWAL KEGIATAN

Jadwal pengerjaan tugas akhir dapat dilihat pada Tabel 1 yang dimulai dari tahapan penyusunan proposal, studi literatur, perancangan sistem, implementasi, pengujian dan evaluasi, dan penyusunan buku.

Tabel 1. Tabel jadwal pengerjaan tugas akhir

Tahapan	2014																		
	Maret				April				Mei				Juni				Juli		
Penyusunan Proposal	■	■																	
Studi Literatur		■	■	■	■	■	■												
Perancangan sistem					■	■	■	■	■										
Implementasi								■	■	■	■	■	■	■	■	■	■		
Pengujian dan evaluasi										■	■	■	■	■	■	■	■		
Penyusunan buku															■	■	■		

12. DAFTAR PUSTAKA

- [1] "OpenResty - a fast web app server by extending nginx," [Online]. Available: <http://openresty.org/>. [Accessed 7 March 2014].
- [2] "Nginx," [Online]. Available: <http://nginx.org/en/>. [Accessed 7 March 2014].
- [3] "Nginx Community," [Online]. Available: <http://wiki.nginx.org/Main>. [Accessed 7 March 2014].
- [4] R. M. A. Baskoro, Implementasi Algoritma Round-Robin Adaptif Untuk Load Balancing pada Mysql Proxy, Surabaya: Institut Teknologi Sepuluh Nopember, 2010.
- [5] A. Piorkowski, A. Kempny, A. Hajduk and J. Strzelczyk, "Load Balancing for Heterogonous Web Servers," vol. 79, pp. 189-198, 2010.
- [6] R. Ierusalimschy , Programming in Lua, Third Edition, lua.org, 2013.