



giving and caring the world

PENDAHULUAN

1.1 Latar Belakang

Web server merupakan seperangkat piranti lunak yang memberikan layanan data, dan berfungsi menerima permintaan HTTP atau HTTPS dari klien dan kemudian mengirimkan kembali hasilnya dalam bentuk halaman *web*. Pada saat *browser* meminta data ke *web server* maka instruksi permintaan data oleh *browser* tersebut dikemas dalam TCP yang merupakan protocol transport dan di kirim ke alamat HTTP. Data yang di-*parsing* dari *browser* ke *web server* disebut sebagai HTTP *request*. Kemudian *web server* akan mencari data dan dikemas dalam TCP protocol untuk kemudian dikirim lagi ke *browser*.

Semakin banyak *user* yang mengakses halaman *web* semakin banyak *request* ke *web server*, dan hal ini menyebabkan *web server* harus memberikan *respons* dari permintaan *user*. Ini menjadikan *traffic* pada *web server* menjadi semakin meningkat. Guna mengantisipasi peningkatan akses atau koneksi ke web server, diterapkan teknik *load balancing*.

Traffic maupun *load* yang diterima di sistem *server* yang menerapkan teknik *load balancing* harus dimonitor tingkat keseimbangannya. Oleh karena itu perlu bangun sebuah aplikasi yang menggambarkan tingkat *load* yang diterima masing-masing server, *data transfered* yang menggambarkan HTTP *response* dan *data rate* yang menggambarkan HTTP *request*.

Server untuk aplikasi yang vital harus dapat diandalkan. Misalnya, *server* yang menjalankan aplikasi Web Based General Ledger System, yang merupakan sebuah aplikasi keuangan. *Server* tersebut harus cepat mendapatkan perbaikan saat mengalami kegagalan atau *down*. Untuk itu dibutuhkan suatu aplikasi yang bisa memberikan pemberitahuan kepada administrator sistem ketika *server*. Aplikasi notifikasi ini senantiasa mengecek kondisi server. Saat ada *server* yang tidak dapat



giving and caring the world

diakses, maka dengan segera aplikasi ini memberikan notifikasi melalui SMS kepada sistem administrator sistem.



giving and caring the world

1.2 Rumusan Masalah

Dari uraian di atas dapat dirumuskan bahwa rumusan masalah pada proyek akhir ini adalah:

1. Bagaimana menerapkan *load balancing* untuk menyeimbangkan *load* antar server?
2. Bagaimana membangun aplikasi yang bisa memonitor aktivitas *load balancer*?
3. Bagaimana membuat sistem notifikasi untuk memberitahukan kegagalan sistem?

1.3 Tujuan

Tujuan pembuatan sistem *server* dan aplikasi ini adalah :

1. Mengimplementasikan `mod_jk` sebagai *load balancer* untuk aplikasi berbasis web.
2. Membangun aplikasi yang mampu memonitor aktivitas *load balancer* dan memberitahukan kegagalan sistem dengan menggunakan Java.

1.4 Batasan Masalah

Batasan masalah dalam pembuatan sistem ini adalah :

1. Menggunakan aplikasi Web Based General Ledger System sebagai studi kasus.
2. Konfigurasi untuk demonstrasi hanya di lakukan di media virtual.



giving and caring the world

1.5 Metodologi Penelitian

Metodologi penelitian dalam penulisan laporan ini adalah :

1. Analisis Kebutuhan

Pada tahap ini dilakukan pencarian bahan-bahan referensi untuk melakukan pembangunan sistem server ini, baik itu referensi dari buku maupun referensi dari Internet.

2. Perancangan Sistem

Pada tahap ini dilakukan perancangan sistem yang akan dibangun, meliputi perancangan pembuatan aplikasi monitoring dan notifikasi, dan perancangan jaringan computer yang akan dibangun.

3. Pembuatan Kode

Tahap ini merupakan tahap pembangun aplikasi.

4. Pengujian Sistem

Tahap ini menjelaskan bagaimana pengujian terhadap sistem yang telah dibangun.

5. Implementasi

Tahap ini menjelaskan bagaimana pengimplementasian sistem yang dibangun.



giving and caring the world

1.6 Jadwal Pengerjaan

	Jadwal pengerjaan Proyek											
	Mei				Juni				Juli			
	1	2	3	4	1	2	3	4	1	2	3	4
Analisis kebutuhan	■	■	■	■	■	■						
Perancangan		■	■									
Pembuatan kode			■	■	■							
Pengujian			■	■	■							
Implementasi				■	■	■						
Dokumentasi	■	■	■	■	■	■	■	■				

Table 1.1 : Jadwal pengerjaan