

PENERAPAN SEO (*SEARCH ENGINE OPTIMIZATION*) MENGGUNAKAN TEKNIK *TOGETHER IN A SINGLE CONNECTION* (TSC) DALAM MENINGKATKAN KUNJUNGAN PADA SEBUAH BLOG

Devia Tito Setyaningsih, Hero Wintolo, Dwi Nugraheny
Teknik Informatika STTA Yogyakarta
informatika@stta.ac.id

Abstract

One of the internet service used by the people of Indonesia is a blog that can be accessed via <http://blogspot.com> or <http://wordpress.com> and others. Everyone can use this medium for the purpose of publication of information in the form of text, images, sounds, or video without having to master web programming languages. One of the obstacles a person who has a blog is a way of increasing number of visitors to his blog. In order for many visitors the blog owner needs to understand about SEO (Search Engine Optimization).

To allow users to blog in increasing the number of visitors and ranking his blog, they invented a system that utilizes parallel processing techniques and create a new method, named TSC (Together in a Single Connection) to increase the number of visitors, number of pages viewed and ranking.

Increase blog traffic system based on parallel processing can improve the effectiveness in increasing traffic to a blog. With this system the blog owner is expected to be easier to rise to visit his blog.

Keywords: *SEO, TSC, Blog Visitation.*

1. Latar Belakang Masalah

Salah satu layanan internet yang digunakan oleh masyarakat Indonesia adalah blog yang dapat diakses melalui <http://blogspot.com> atau <http://wordpress.com> dan lain-lain. Semua orang dapat menggunakan media ini untuk kepentingan publikasi berbagai informasi baik dalam bentuk teks, gambar, suara, atau video tanpa harus menguasai bahasa pemrograman web. Salah satu kendala seseorang yang memiliki blog adalah cara meningkatkan jumlah pengunjung blog-nya. Agar banyak pengunjungnya maka pemilik blog tersebut perlu memahami tentang SEO (*Search Engine Optimization*).

SEO merupakan serangkaian proses yang dilakukan secara sistematis yang bertujuan untuk meningkatkan volume dan kualitas trafik dari mesin pencari menuju alamat situs web tertentu dengan memanfaatkan mekanisme kerja alami algoritma mesin pencari tersebut. Tujuan SEO adalah menempatkan suatu alamat situs web pada posisi teratas (atau setidaknya pada halaman pertama) hasil pencarian berdasarkan subyek tertentu. Secara logis, alamat situs yang menempati posisi teratas hasil pencarian memiliki peluang lebih besar untuk mendapatkan banyak pengunjung. Teknik SEO sangat penting sekali dalam dunia internet karena tidak dapat dipungkiri, bahwa dengan menjadi urutan teratas sebuah *website* akan semakin diminati oleh para pengguna Internet. Pengunjung cenderung lebih tertarik dengan beberapa *website* yang ada di halaman awal karena dianggap lebih relevan dengan *keyword*

yang dicari. Dari kecenderungan tersebut, maka dapat menarik peminat para pengguna internet untuk berkunjung ke *website* tersebut. Teknik SEO terdapat beberapa macam, antara lain memasang *breadcrumb*, *dummy blog*, *on page* dan *off page*, memasang *meta tag*, dan membuat judul SEO yang *friendly*. *Breadcrumb* merupakan menu navigasi untuk pengunjung dalam pencarian artikel dan kembali lagi ke *home page* dengan mudah, sedangkan *dummy blog* adalah salah satu cara untuk optimasi blog agar bisa masuk dan punya peringkat bagus di mesin pencari. Istilah lain dari teknik SEO adalah memasang *meta tag*, maksudnya agar blog mudah dibaca oleh mesin pencari semacam google. Sedangkan, membuat judul SEO yang *friendly* adalah judul SEO dapat dengan mudah dicari oleh pengguna internet.

Selain teknik SEO diatas, dalam tugas akhir ini menggunakan teknik yang diberi nama *Together in a Single Connection* (atau disingkat dengan TSC), merupakan sebuah cara untuk meningkatkan kunjungan pada sebuah blog dengan memanfaatkan teknik pemrosesan paralel yang dikendalikan oleh komputer server dengan membuka blog setiap hari dengan alamat *internet protocol* (IP) yang berbeda-beda menggunakan media penghubung internet (modem). Dengan cara ini diharapkan jumlah *visitor* dan *page view* dari sebuah blog dapat meningkat, sehingga blog tersebut akan meningkat *ranking*-nya di *search engine*.

2. Blog

Istilah blog merupakan bentuk singkat dari weblog atau web log. Blog pada dasarnya adalah sebuah jurnal *online* yang tersedia atau disediakan di web (blog). Menurut pakar blog, *a blog is a type of website that is usually arranged in chronological order from the most recent 'post' (or entry) at the top of the main page to the older entries towards the bottom* (Darren Rowse). Jadi, sebuah blog adalah sejenis *website* yang biasanya diatur dalam urutan kronologis dari 'post' paling baru (atau *entry*) di bagian atas halaman utama ke entri yang lebih lama ke arah bawah. Orang yang menulis atau melakukan entri data dalam sebuah blog disebut dengan "blogger", sedangkan aktifitas yang dilakukan untuk memelihara sebuah blog atau menambah artikel (*content*) ke sebuah blog disebut "blogging". Sebuah blog memiliki banyak manfaat, diantaranya adalah untuk bisnis, untuk menyalurkan kegemaran menulis, untuk pendidikan, untuk menerbitkan jurnal, dan lain sebagainya.

a. SEO (*Search Engine Optimization*)

SEO merupakan sebuah proses yang digunakan untuk mengoptimalkan dan memaksimalkan halaman blog untuk pencarian dalam mesin pencari. Proses tersebut merupakan sebuah teknik pemasaran internet yang ditujukan untuk meningkatkan trafik kunjungan dalam sebuah konten jaringan dan meningkatkan kualitas trafik sendiri. Beberapa teknik SEO sudah banyak yang digunakan oleh pemilik blog. Sedangkan di tugas akhir ini memperkenalkan teknik baru SEO, yang diberi nama teknik TSC (*Together in a Single Connection*). Teknik ini akan dibangun dan dibahas dalam penyusunan tugas akhir ini. Pembukaan alamat web akan dilakukan secara berulang-ulang sampai halaman web mencapai 10 halaman web. Satu halaman web diberi *timer* selama 20 detik untuk membuka utuh halaman web tersebut. Apabila sudah mencapai 20 detik, maka *client* akan membuka halaman web yang lain setiap 20 detik.

b. Layanan Hitstats

Hitstats merupakan sebuah situs yang menyediakan dan menampilkan informasi lengkap tentang pengunjung dan tampilan halaman, serta menampilkan seluruh informasi rinci tentang lalu lintas situs web dan hasilnya digunakan untuk mengembangkan rencana pemasaran yang tepat. Layanan *hitstats* juga menawarkan layanan gratis yang lengkap dan *real time*. Selain itu, *hitstats* dalam sebuah blog menampilkan jumlah pengunjung yang *online*, jumlah pengunjung dalam sehari, jumlah pengunjung yang aktif membuka blog, menampilkan halaman populer di blog tersebut, dan informasi lainnya.

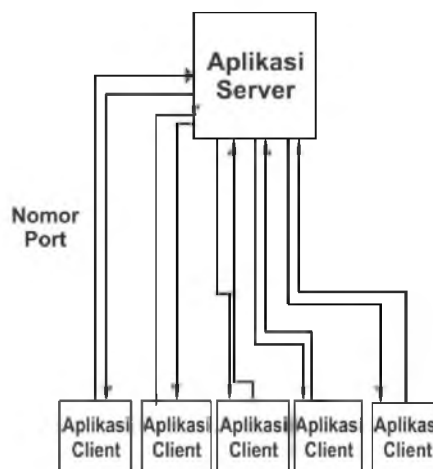
c. Pemrograman Socket Delphi

Socket adalah mekanisme komunikasi yang memungkinkan terjadinya pertukaran data antar program atau proses baik dalam satu mesin maupun antar mesin. Salah satu elemen penting yang digunakan dalam aplikasi socket adalah *port*. *Port* merupakan sebuah koneksi data virtual yang digunakan oleh sebuah aplikasi untuk bertukar data secara langsung. Terdapat banyak *port* di dalam sebuah sistem komputer dengan fungsinya masing-masing.

Dalam tugas akhir ini, komponen socket yang dipakai adalah *ServerSocket* dan *ClientSocket* yang terdapat di *library* Internet. Masing-masing socket memiliki nomor *port* yang berbeda-beda, sedangkan untuk menghubungkan koneksi *serversocket* dan *clientsocket* mempunyai nomor *port* yang sama.

d. Together in a Single Connection (TSC)

TSC merupakan salah satu teknik dalam meningkatkan *page view* dan *visitor* sebuah blog. Teknik TSC memanfaatkan 2 (dua) atau lebih komputer *client* yang merupakan sebuah perangkat yang menerima yang akan menampilkan dan menjalankan aplikasi (*software* komputer), sedangkan sebuah komputer *server* adalah perangkat yang menyediakan dan bertindak sebagai pengelola aplikasi, data, dan keamanannya. Komputer *server* dan *client* terhubung dengan teknik pemrosesan paralel, yaitu dengan mengirim alamat blog dari komputer *server* ke komputer *client* setiap hari menggunakan sebuah koneksi (modem), sehingga akan mempunyai alamat IP yang berbeda-beda setiap koneksi dihubungkan. Gambar 1 merupakan gambaran dari aplikasi yang akan dibangun menggunakan 5 (lima) aplikasi *client*.



Gambar 1 Gambaran Aplikasi Menggunakan 5 Aplikasi Client

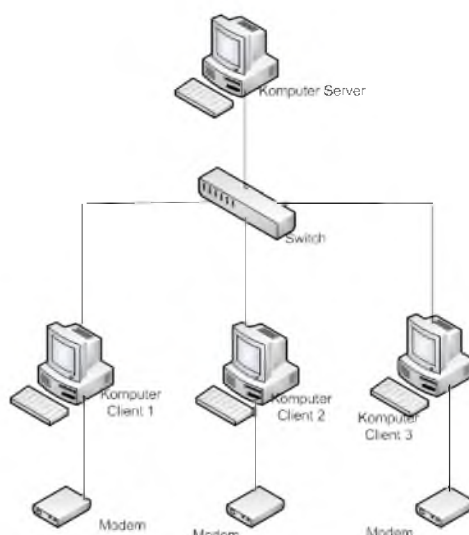
3. Penjelasan Aplikasi Menggunakan Metode TSC (*Together in a Single Connection*)

Aplikasi yang dibangun ini menggunakan metode TSC. Metode TSC merupakan sebuah metode yang berjalan atau melakukan proses dalam sebuah koneksi yang sama. Proses yang berjalan dalam satu koneksi dan berjalan secara bersama-sama dan berulang-ulang.

Secara umum, aplikasi ini merupakan aplikasi *client-server*. Aplikasi yang terdiri dari aplikasi *client* dan aplikasi *server*. Aplikasi ini merupakan aplikasi yang digunakan untuk mengirim data dari *server* ke *client*. Data yang dikirimkan berupa halaman web yang berjumlah 10. Sepuluh halaman web tersebut dikirim secara bersama-sama dalam sebuah koneksi yang sama dan dikirim 20 detik sekali setiap halaman web-nya.

Konfigurasi Jaringan

Untuk menjalankan aplikasi ini harus dilakukan konfigurasi jaringan terlebih dahulu. Peralatan yang dibutuhkan yaitu sebuah komputer yang digunakan sebagai *server* dan 3 unit komputer yang digunakan sebagai *client*. Digunakan *switch* untuk menyatukan kabel-kabel yang terhubung antara komputer *server* dan beberapa komputer *client*. Modem dihubungkan dengan komputer *client* sebagai pembuka pesan. Gambaran dari konfigurasi jaringan yang digunakan pada aplikasi ini dapat dilihat pada gambar 2.



Gambar 2 Konfigurasi Jaringan

Pengujian Aplikasi

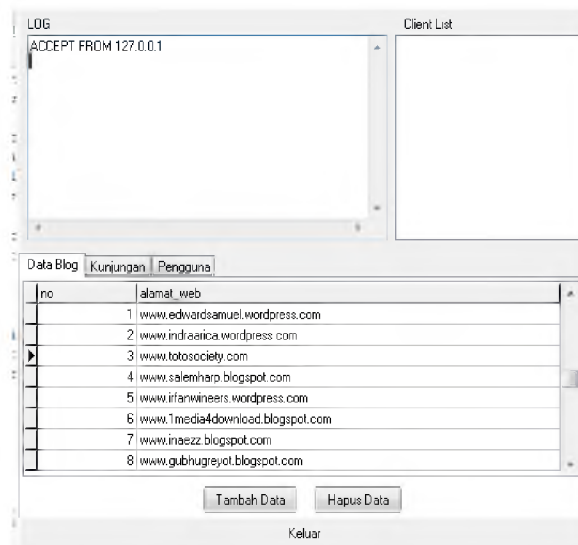
Uji coba aplikasi ini dilakukan di laboratorium komputer STTA Yogyakarta. Hal ini dikarenakan karena STTA mempunyai komputer dalam jumlah yang banyak yang dapat mendukung jalannya aplikasi ini. Uji coba dilakukan menggunakan 4 buah unit komputer yang telah terhubung dalam suatu jaringan dengan bentuk topologi seperti yang tergambar dalam konfigurasi jaringan. Dari beberapa unit komputer tersebut, satu unit komputer bertindak sebagai *server* dan yang lain bertindak sebagai *client*. Tabel 1 menggambarkan beberapa nama komputer dan alamat IP yang terdapat di STTA Yogyakarta yang dapat digunakan sebagai komputer yang bertindak sebagai *client*.

Tabel 1 Daftar Nama Komputer untuk Uji Coba

Nama Komputer	Alamat IP
PC Server	198.168.5.250
PC Client 1	198.168.5.251
PC Client 2	198.168.5.252
PC Client 3	198.168.5.253

Uji Coba Penerimaan Pesan Oleh Client

Dalam uji coba penerimaan pesan oleh *client*, aplikasi *server* dan aplikasi *client* harus terkoneksi terlebih dahulu, dengan menyamakan nomor *port* untuk kedua aplikasi tersebut sebagai jalur komunikasi dan aplikasi *client* harus mengisikan nomor IP komputer *server*. Setelah kedua aplikasi terkoneksi, maka pesan yang dikirimkan oleh seseorang yang ditujukan kepada aplikasi *client* dapat ditampilkan, sebelum pesan tersebut tampil di aplikasi *client*, pesan melewati aplikasi *server*. Pesan yang ditampilkan dalam aplikasi *client* merupakan sebuah halaman web yang dibuka di *browser* yang dibangun di aplikasi *client*. Pesan yang dikirim ke *client* telah melewati *server* dengan membaca *string* pesan sebelum spasi yang menunjukkan tujuan *client*. Seperti ditunjukkan pada gambar 4.10 bahwa pesan tersebut dikirimkan ke *client* yang bernama PC Client, sedangkan string pesan setelah spasi merupakan isi dari pesan. Dalam uji coba ini digunakan beberapa buah komputer *client* yang dikirim pesan yang sama, dan aplikasi *client* dapat menerima semua pesan yang dikirimkan. Hal ini digambarkan pada tabel 2. Gambar dari aplikasi *client* yang menerima pesan ditunjukkan seperti pada gambar 3.

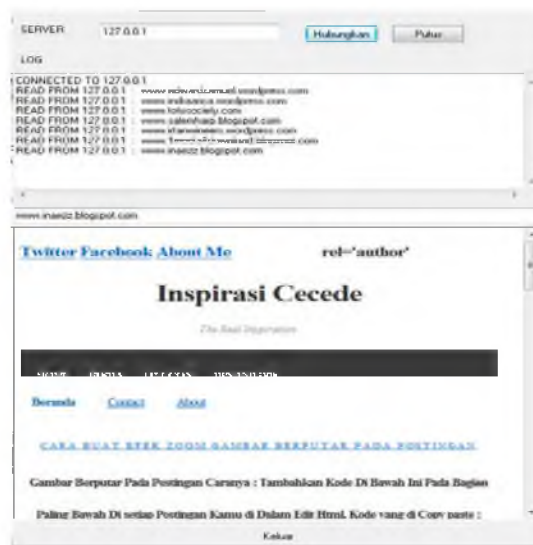


Gambar 3 Pesan yang Dikirim ke PC Client

Tabel 2 Uji Menerima Pesan

No	Nama Komputer Client	Diterima	Tidak Diterima
1	PC Client 1	√	—
2	PC Client 2	√	—
3	PC Client 3	√	—

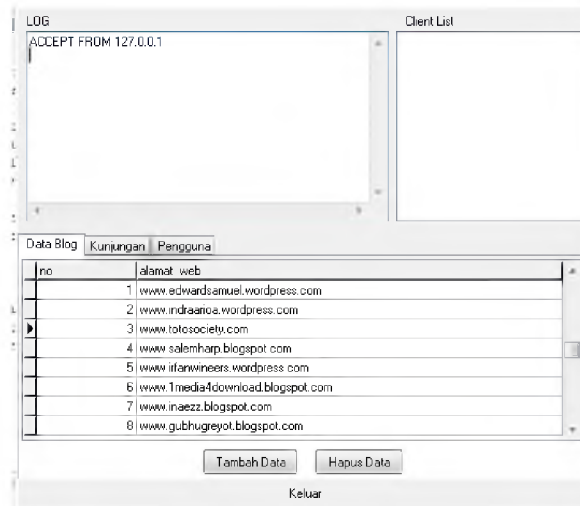
Dari tabel 2 uji menerima pesan, dapat diketahui bahwa komputer *client* yang telah dilakukan uji coba, yaitu PC *Client* dan lainnya berhasil menerima pesan yang dikirimkan oleh pengguna. Dari beberapa kali uji coba yang dilakukan di beberapa komputer *client*, semua pesan yang dikirimkan dapat diterima, dan tidak ada pesan yang gagal diterima oleh aplikasi *client*. Tabel 2 menjelaskan bahwa tanda centang pada tabel diterima menggambarkan bahwa PC *Client* 1, 2 dan 3 sudah menerima data dari *server*. Gambar 4 merupakan gambar yang berkaitan dengan gambar 3 sebelumnya, yaitu menjelaskan aplikasi *client* saat menerima pesan dari *server*. Selanjutnya pesan tersebut akan tertampil di edit text dan pesan yang berupa halaman web tersebut akan dibuka di *browser* milik aplikasi *client*.



Gambar 4 Tampilan Komputer *Client* pada PC *Client* Saat Menerima Pesan

Uji Mengirim Pesan

Dalam uji mengirim pesan ini, dilakukan uji coba ke beberapa komputer yang terdapat di STTA Yogyakarta. Apabila ingin mengirim pesan, pengguna harus memasukkan alamat IP komputer *server* dari penerima pesan yaitu komputer *client*. Agar komputer *server* dan *client* terhubung, sehingga pesan akan dapat dikirim dari aplikasi *server* ke aplikasi *client*. Setelah pesan berhasil dikirim, maka aplikasi *server* akan berhenti dan keluar dari aplikasi. Tabel 3 memuat data uji coba pengiriman pesan yang dikirimkan ke komputer PC *Client* dan lainnya.



Gambar 5 Aplikasi *Server* Mengirim Pesan

Tabel 3 Uji Pengiriman Pesan

No	Nama Komputer <i>Client</i>	Dikirim	Tidak Dikirim
1	PC <i>Client</i> 1	√	—
2	PC <i>Client</i> 2	√	—
3	PC <i>Client</i> 3	√	—

Dari tabel 3 uji coba pengiriman pesan dapat diketahui bahwa semua pesan yang dikirimkan oleh komputer *client*, dalam hal ini komputer PC *Client* dan yang lain berhasil mengirimkan pesan ke alamat IP tujuan. Tabel 3 menjelaskan bahwa tanda centang (√) pada tabel dikirim menggambarkan bahwa *server* sudah mengirim data ke PC *Client* 1, 2 dan 3.

Uji Fungsi

Dalam uji fungsi ini dimaksudkan untuk melakukan pengujian terhadap sistem yang dibangun apakah sudah berjalan dengan baik atau belum. Pengujian uji fungsi ini meliputi uji operator dan halaman web yang digunakan sebagai bahan uji coba.

Uji Operator

Dalam melakukan uji operator ini menggunakan 3 buah *sim card* dari operator telepon seluler yang sama, yaitu tri (3). Tujuan dari uji operator ini yaitu untuk mengetahui bahwa biaya yang diperlukan untuk mengirim pesan menggunakan operator three merupakan yang termurah (“Setyoadi, Wangsit. 2012”). Tabel 4 menunjukkan perbandingan biaya dari 3 *provider* yang berbeda.

Tabel 4 Uji Operator

No	Nama Operator	Biaya (Rp.)
1	Tri	10.000
2	Telkomsel	50.000
3	Axis	10.000

Dari tabel 4 uji operator dapat diketahui bahwa biaya pengiriman pesan untuk operator merupakan yang termurah. *Provider* tri (3) merupakan yang termurah karena dengan biaya Rp.10.000,- mendapatkan kuota 200Mb.

Uji Peringkat Halaman Web

Dalam melakukan uji peringkat ini menggunakan sebuah situs web yang berfungsi mengukur sebuah situs yang mempunyai jumlah pengunjung dan jumlah halaman yang dibuka oleh seorang pengunjung yaitu *histats.com*. Selain itu, uji peringkat juga menggunakan *alexa.com* untuk mengetahui di peringkat berapa situs web yang digunakan untuk uji program yang telah dibangun ini. Hal ini ditunjukkan pada gambar 6



Gambar 6 Menunjukkan Situs Web Sebelum Uji Coba

Gambar 6 menunjukkan situs web sebelum di uji coba. Berhasil atau tidaknya hasil dari uji coba ini dilihat dan dipantau dari *website Histats.com* yang digunakan untuk memantau *visitor* dan *page view*, dan *Alexa.com* yang digunakan untuk memantau peringkat yang didapatkan setelah uji coba. Kedua *website* tersebut sebagai indikator untuk hasil uji coba program yang dibangun.



Gambar 7. Menunjukkan Hasil Uji Coba

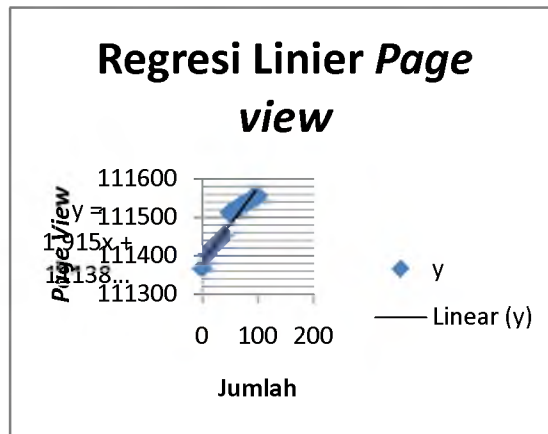
Gambar 7 merupakan hasil dari uji coba program, hasil diatas menunjukkan bahwa setelah dibuka random 10kali untuk *visitor* naik 1 dari 106,034 menjadi 106,035 dan untuk *page*

view naik 112,294 menjadi 112,312. Visitor hanya naik 1, hal ini kemungkinan *histats.com* tidak dapat mendeteksi alamat IP yang berbeda-beda.

Ditinjau dari *website alexa.com*, terdapat situs web yang dapat naik setelah dibuka 30kali dalam sehari dan terdapat beberapa situs web yang tidak naik. Hal ini dikarenakan situs web tersebut terdapat *script* yang tidak dapat dibuka / diproses oleh *browser* yang dibangun sendiri.

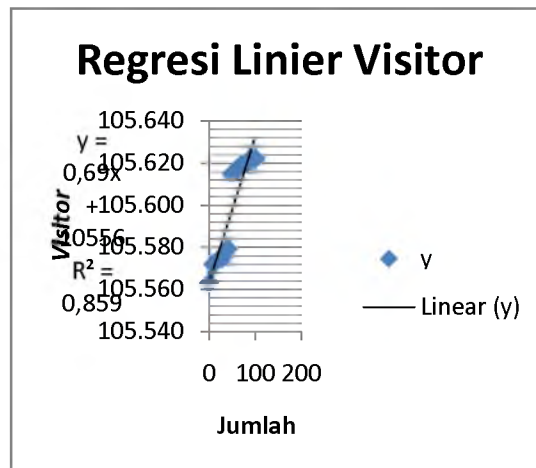
Analisa Regresi Linier

Analisa regresi dalam statistika adalah salah satu metode untuk menentukan hubungan sebab-akibat antara satu variabel dengan variabel yang lain. Dalam hal ini, sumbu x sebagai jumlah membuka situs web (secara *random*) dan sumbu y sebagai *page view* yaitu untuk melihat jumlah halaman sebuah blog yang sudah dibuka oleh pengunjung. Regresi linier ini dibuat menggunakan Microsoft Excel seperti ditunjukkan pada gambar 8.



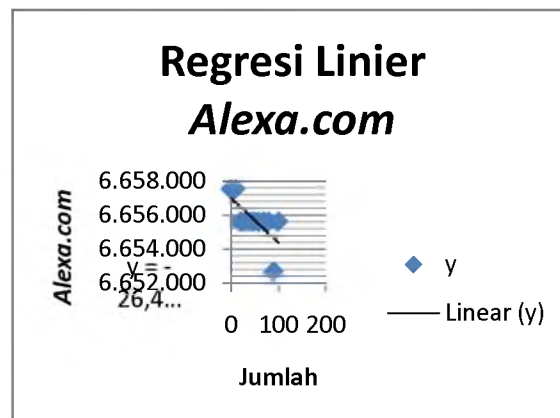
Gambar 8 Hasil Analisa antara Jumlah dan Page View

Gambar 8 menjelaskan bahwa menunjukkan grafik keberhasilan kenaikan dalam uji coba aplikasi *client* dan *server* untuk mengirim, menerima dan membuka situs web. Pengujian ini dilakukan secara *random* dalam memonitornya. Ada beberapa halaman web yang menunjukkan kenaikan yang tidak terlalu terlihat. Hal ini disebabkan karena *website histats.com* tidak mencatat situs web yang dibuka dengan *provider* yang sama walaupun dengan alamat IP yang berbeda tetapi. *Histats* hanya mencatat *page view*-nya dan untuk *visitor* hanya beberapa saja yang tercatat. Hal ini disebabkan karena *histats.com* hanya mencatat alamat IP yang berbeda saja untuk *page view*. Sedangkan untuk *visitor*, *histats.com* hanya dapat mencatat dan melihat dari *provider* yang berbeda walaupun alamat IP berbeda. Regresi linier yang kedua adalah analisa antara jumlah membuka situs web dan *visitor* seperti ditunjukkan pada gambar 9.



Gambar 9 Hasil Analisa antara Jumlah dan Visitor

Dari gambar 9 menunjukkan bahwa antara jumlah dalam membuka situs web dan jumlah *visitor* kurang berhasil dalam hasil uji coba. Hal ini terjadi kemungkinan *website histats.com* tidak dapat mendeteksi alamat ip yang berbeda dalam *provider* yang sama. Regresi linier yang ketiga adalah analisa antara jumlah membuka situs web dan *alexa.com* seperti ditunjukkan pada gambar 10.



Gambar 10 Hasil Analisa antara Jumlah dan Alexa.com

Dari gambar 10 menunjukkan bahwa antara jumlah dalam membuka situs web dan *ranking* di *website alexa.com* kurang berhasil dalam hasil uji coba. Hal ini terjadi *website alexa.com* akan naik apabila dalam uji coba menggunakan *provider* yang berbeda-beda dan dalam sehari dibuka 30kali seperti yang ditunjukkan gambar 10.

Hasil Analisa

Hasil dari semua percobaan dapat dianalisa dari implementasi dan simulasi percobaan adalah tentang proses mekanisme kerja dari sistem secara keseluruhan. Secara keseluruhan hasil analisa terdiri dari konsep dasar yang dibangun untuk menjadi sebuah sistem dengan apa yang telah sebelumnya.

Langkah-langkah analisa dari hasil percobaan adalah sebagai berikut:

1. Aplikasi *server* dan aplikasi *client* dapat saling berkomunikasi dengan menggunakan *port* yang telah ditentukan oleh aplikasi *server*.
2. Komputer *client* terlebih dahulu harus terhubung dengan modem.

3. Pesan akan dikirimkan ke komputer *client* yang terhubung, dan apabila ada 2 komputer *client* yang terhubung maka pesan yang dikirim akan sama dengan pesan yang dikirimkan ke komputer *client* yang terlebih dahulu terhubung.
4. Pesan yang diterima di aplikasi *client* akan dibuka di *browser* yang dibangun di aplikasi *client*.

Sedangkan analisa dari hasil percobaan adalah sebagai berikut:

1. Aplikasi *server* berhasil mengirimkan pesan yang dikirim ke aplikasi *client*.
2. *Browser* yang dibangun oleh aplikasi *client* tidak mendukung *javascript*, sehingga akan terjadi *error* untuk halaman web yang mengandung *javascript*. Tetapi tetap dapat diproses oleh *browser* oleh aplikasi *client* dengan meng-klik OK.
3. Hasil analisa regresi linier *page view* menunjukkan bahwa dengan metode TSC halaman web yang dibuka menunjukkan kenaikan jumlah halaman yang telah terbuka.
4. Hasil analisa regresi linier *visitor* menunjukkan bahwa dengan metode TSC halaman web yang dibuka menunjukkan kenaikan jumlah pengunjung yang tidak terlalu terlihat, hal ini dikarenakan *histats.com* tidak mencatat *provider* yang sama walaupun alamat IP-nya berbeda-beda.
5. Hasil analisa regresi linier *alexa.com* menunjukkan bahwa dengan metode TSC halaman web yang dibuka menunjukkan kenaikan *ranking* yang tidak terlalu terlihat, hal ini dikarenakan *alexa.com* tidak mencatat *provider* yang sama walaupun alamat IP-nya berbeda-beda.

5. Kesimpulan

Dari hasil uji coba pada yang dilakukan dapat ditarik beberapa kesimpulan:

1. Aplikasi yang dirancang dan diujicobakan sudah berjalan sesuai dengan tujuan dengan indicator kenaikan jumlah *visitor* dan *page view* melalui *histats.com*.
2. Metode TSC (*Together Single Connection*) yang digunakan dalam aplikasi sudah berhasil menaikkan *ranking* blog pada *alexa.com* dengan minimal jumlah kunjungan 30kali.
3. Jumlah *visitor* sebuah situs akan bertambah besar saat dikunjungi dengan *IP address* yang berbeda.
4. Analisa statistik dengan metode regresi linier *histats.com* dan *alexa.com* menunjukkan bahwa hasil uji coba berhasil menaikkan kunjungan dan *ranking* blog.

Selain kesimpulan, penelitian ini juga memiliki saran yang dapat digunakan untuk pengembangan dan penelitian selanjutnya yang terkait dengan penelitian ini. Saran tersebut antara lain:

1. Menggunakan *browser* yang memiliki fasilitas *update javascript* dan *Macromedia Flash*.
2. Penggunaan *provider* yang berbeda pada aplikasi *client* diharapkan dapat bertambah kuantitasnya

Referensi

- [1] Kristanto, Andri, Rekayasa Perangkat Lunak, Gava Media Yogyakarta, 2004.
- [2] Madcoms, Pemrograman Borland Delphi 7, Andi Offset, Yogyakarta, 2003.
- [3] Sutedjo, Budi, Konsep & Aplikasi Pemrograman Client Server dan Sistem Terdistribusi, Andi Offset, 2006.

- [4] Setyoadi, Wangsit, Otomatisasi Penerimaan Dan Pengiriman Pesan Dengan Sistem Terdistribusi Untuk Mendukung Penyebaran Informasi Akademik, Compiler, Volume 1 Nomor 1. Mei 2012.
- [5] (22 April 2012) Histats, <http://www.histats.com>.
- [6] (30 April 2012) Master Komputer, <http://master-komputer.com/tutorial/41-programming/86-socket-programming-delphi.pdf>.
- [7] (2011) Dhany, <http://dhany.web.id/panduan-seo>