

# PENCATATAN DAN PEMISAHAN DATA LOG PROXY SERVER KE DALAM DATABASE MENGGUNAKAN MODEL OTOMATA MESIN TURING DI STMIK PPKIA PRADNYA PARAMITA MALANG

Luqman Affandi \*)

## ABSTRACT

*Development of the Internet has brought a significant impact on human life. To share Internet resources and anticipated adverse effects and the convenience of Internet users is necessary to control Internet access. One of them is by installing a squid proxy server. To be able to help improve the ability of the proxy server can be done by analyzing the log proxy server, because the record contains a record of user's Internet access. But to be able to do further analysis is necessary to re-record data on the proxy log into a database. This is caused by the proxy server log file is recorded in a text file sequential so hard to do an analysis of the data. Needed a method to read and analyze the log file and the method can be used is the Turing machine model of automata. Turing machine is a deterministic finite state machine that has two-way tape. In the process of recording into the database, performed the separation of the information contained in the proxy log files. Separation is performed by the analysis of the proxy log file will be used for various controls such as site access, and who has access and others. Such information can be poured into a website, so that individuals who use Internet resources in a network can mutually control each other.*

*Key words: Listing, Proxy Server, automata, and proxy server log.*

## PENDAHULUAN

Keberadaan Internet sebagai sebuah jaringan raksasa yang berfungsi sebagai medium tukar-menukar informasi secara cepat dan murah telah berkembang pesat di berbagai bidang, seperti perdagangan, kurir, perbankan, ataupun pendidikan. Oleh sebab itu sumber daya Internet yang dibutuhkan semakin hari semakin bertambah baik dari segi *bandwidth* maupun informasi yang ada.

Untuk memenuhi kebutuhan terhadap akses Internet maka dilakukan pengaturan bagaimana pemakai Internet dalam satu lingkungan tertentu dapat menggunakan layanan Internet dengan nyaman. Hal ini dilakukan dengan

pembagian *bandwidth*, pengaturan hak akses, pembatasan jumlah koneksi dan sebagainya. Pengontrolan biasanya dilakukan dengan pemasangan sebuah alat atau perangkat lunak pada komputer yang secara langsung berhubungan dengan *backbone* Internet, salah satunya menggunakan *Proxy Server*.

*Proxy server* merupakan sebuah perangkat lunak berbasis server yang dapat mengatur pemanfaatan Internet, membagi sumber daya yang dimiliki dan meningkatkan kecepatan koneksi Internet dengan mengurangi permintaan akses secara langsung ke jaringan Internet dengan memanfaatkan *cache* Internet. Sehingga dalam sebuah kumpulan pengguna Internet tertentu

seperti di sebuah perusahaan atau organisasi yang lain, akses Internet dapat di kontrol dan dimonitoring secara baik.

Di dalam *proxy server* terdapat catatan (*log file*) yang mencatat berbagai kegiatan yang dilakukan pemakai Internet. Biasanya berisi alamat *website* yang dikunjungi pemakai, waktu akses, username dan IP komputer pemakai. Dari catatan *log file* ini dapat ditelusuri kebiasaan dari pemakaian Internet, sehingga dapat digunakan untuk membantu membuat keputusan terhadap langkah apa yang harus dilakukan terhadap akses ke jaringan Internet yang sedang dikelola. Permasalahannya, jika penelusuran *log file* dilakukan secara manual, maka akan menghabiskan waktu dan menyita banyak aktifitas. Karena *log file* selalu bertambah tiap detiknya sehingga untuk menghasilkan analisis yang terbaru dan ter-*update* sulit dilakukan.

Untuk dapat melakukan analisis lebih lanjut terhadap data di log proxy tersebut perlu dilakukan pencatatan ulang ke dalam sebuah database. Selain pencatatan perlu dilakukan pemisahan data karena data yang tersimpan hanya dipisah dengan spasi. Hal ini disebabkan file log proxy berupa file teks yang dicatat secara sequensial. Dengan dimasukkannya ke dalam database maka dapat dilakukan manipulasi data, misalnya menggunakan perintah-perintah sql untuk menghasilkan informasi yang dibutuhkan.

Pencatatan dan pemisahan data di dalam database dapat dilakukan dengan memanfaatkan model

otomata, yaitu sebuah model mesin abstrak yang mempunyai input dan output diskrit dari permasalahan yang sedang dihadapi. Model yang dapat dipakai adalah model mesin turing yang merupakan suatu *deterministic finite state machine* yang mempunyai *tape* dua arah. Tape dapat dilogikakan sebagai data *log proxy server* yang dicatat secara sequensial.

STMIK PPKIA Pradnya Paramita Malang adalah sebuah kampus komputer yang menggunakan *Proxy Server Squid* berbasis Linux untuk manajemen akses Internetnya. Data *log proxy server* di kampus tersebut dapat dikelola ke dalam sebuah database agar dapat digunakan untuk analisis lebih lanjut seperti pengontrolan, daftar akses, penanganan user dan sebagainya.

## TINJAUAN PUSTAKA

### 1. Jaringan Komputer

Jaringan komputer merupakan kumpulan komputer berjumlah banyak yang terpisah-pisah akan tetapi saling berhubungan dalam melakukan tugasnya (Marco Van Basten, 2009). Tujuan dibangunnya jaringan komputer yaitu:

- Berbagi sumber (*resource sharing*) yaitu program, peralatan atau data dapat dipakai bersama-sama oleh komputer yang terhubung di jaringan komputer.
- Keandalan tinggi (*high reliability*) yaitu tersedianya sumber-sumber alternatif kapanpun diperlukan. Misalnya pada aplikasi perbankan, jika salah satu mesin

tidak bekerja maka kinerja mesin yang lain tidak terganggu.

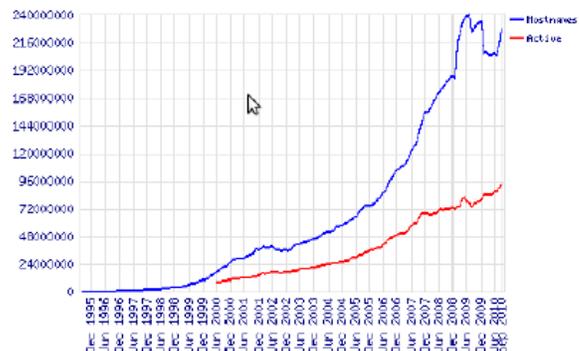
- Menghemat keuangan, yaitu membangun komputer-komputer personal lebih murah daripada menggunakan mainframe.
- Skalabilitas (*scalability*) yaitu meningkatkan kinerja dengan menambahkan komputer server atau client dengan mudah tanpa mengganggu kinerja komputer server atau komputer client yang sudah ada lebih dulu.
- Medium komunikasi yaitu memungkinkan kerjasama antar orang-orang yang saling berjauhan melalui jaringan komputer baik untuk bertukar data maupun berkomunikasi.
- Akses informasi luas yaitu dapat mengakses dan mendapatkan informasi dari jarak jauh.
- Komunikasi orang-ke-orang yaitu digunakan untuk berkomunikasi dari satu orang ke orang yang lain.
- Hiburan interaktif.

## 2. Internet

Menurut LaQuery (2006: 9) menyebutkan bahwa Internet adalah merupakan jaringan dari ribuan jaringan komputer yang menjangkau jutaan orang di seluruh dunia. Sedangkan menurut Firmansyah, pengertian internet secara umum yaitu sekumpulan jaringan komputer yang menghubungkan jaringan komputer seluruh dunia (2002:1). Dari beberapa pengertian tersebut dapat disimpulkan bahwa pengertian Internet adalah

jaringan komputer yang menghubungkan jaringan-jaringan komputer di seluruh dunia yang memungkinkan pengguna untuk bertukar informasi.

Perkembangan Internet saat ini sangat pesat, hal ini dapat di lihat dari perkembangan jumlah domain Internet yang terhubung dan jumlah pengguna Internet. Data terbaru dari jumlah situs di dunia yang aktif tercatat sebanyak 227.225.642. Berikut perkembangan terbaru dari total situs dan *hostname* di dunia dari Agustus 1995 sampai September 2010 menurut <http://news.netcraft.com> :



**Grafik Pertumbuhan Situs di Dunia Dari Agustus 1995 sampai September 2010**  
(Sumber : <http://news.netcraft.com>, tanggal akses 15 September 2010)

Data tersebut menunjukkan bahwa pertumbuhan Internet sangat pesat baik dari penyedia maupun pemakai Internet. Hal ini akan menimbulkan dampak baik maupun buruk.

Dampak yang ditimbulkan dari perkembangan ini antara lain :

- Semakin banyak konten Internet yang dapat diakses.
- Semakin banyak informasi yang tersedia di Internet.
- Dapat menunjang kegiatan bisnis.
- Sebagai sumber ilmu pengetahuan untuk semua bidang.

Sedangkan dampak buruknya antara lain:

- Karena Internet bersifat bebas, maka banyak konten-konten negatif (pornografi, sara, dan lain-lain) yang dapat diakses dengan mudah.
- Semakin mudah orang berbuat jahat menggunakan jaringan internet, misalkan melakukan penyebaran virus ataupun trojan.
- Semakin besar sumber daya komputer yang dibutuhkan untuk akses Internet, misalnya *bandwidth*.

Dari dampak yang ditimbulkan tersebut perlu dilakukan usaha pengontrolan koneksi Internet agar informasi yang begitu luas dari Internet dapat dimanfaatkan dengan sebaik-baiknya sesuai dengan kebutuhan.

### 3. **Proteksi Internet (*Internet Security*)**

Menurut Pungky Sulistyono dalam jurnal "Membangun Infrastruktur IT menggunakan SQUID sebagai *Proxy Server*" (2007, 2), beberapa

model keamanan yang dapat dipakai seperti berikut :

- Mengatur Akses.

Cara yang umum digunakan untuk mengatur akses yaitu dengan menggunakan mekanisme *authentication* dan *access control*. Implementasi mekanisme *authentication* menggunakan pencocokan *userid* dan *password*. Sedangkan *access control* biasanya dilakukan dengan melakukan pengelompokan pemakai dalam sebuah kelompok (*group*) dengan memberikan hak-hak tertentu misalnya kapan mereka dapat melakukan akses.

- Menutup *service*.

Pengamanan model ini dapat dilakukan dengan menutup layanan-layanan yang tidak diperlukan di server atau komputer. Beberapa layanan yang sering dijalankan seperti *telnet*, *ftp*, *smtp*, *tcp*, dan lain sebagainya.

- Memasang Proteksi.

Proteksi secara umum dapat berupa *filter* terhadap komunikasi yang terjadi. Secara lebih spesifik, biasanya dengan memasang sebuah *firewall*. Filter dapat digunakan untuk menyaring e-mail, informasi, akses, atau bahkan pada level packet.

Pengamanan di jaringan komputer dapat dilakukan dengan dua cara yaitu dengan memasang sebuah perangkat keras atau memasang perangkat lunak. Contoh penggunaan perangkat keras salah satunya menggunakan *Cisco*. Contoh pemasangan perangkat lunak untuk pengamanan komunikasi di jaringan komputer dengan

memasang sebuah *firewall*. Dengan *firewall* informasi dari luar dan dalam dapat disaring sesuai dengan kebutuhan. Pada pengamanan Internet sendiri, banyak pengguna memakai *proxy server* untuk mengatur dan membagi koneksi Internet.

Menurut Pungky Sulistyو dalam jurnal “Membangun Infrastruktur IT menggunakan SQUID sebagai *Proxy Server*” (2007, 2), *Proxy Server* mempunyai 3 fungsi utama yaitu:

- *Connection Sharing*

Sebagai perantara antara pengguna dan server-server di Internet. *Proxy Server* bekerja dengan cara menerima permintaan layanan dari *user* dan kemudian sebagai gantinya *proxy server* akan mewakili permintaan pengguna ke server-server di Internet yang dimaksudkan.

- *Filtering*

Bekerja pada layer aplikasi sehingga berfungsi sebagai *firewall packing filtering* yang digunakan untuk melindungi jaringan lokal dari serangan atau gangguan yang berasal dari jaringan Internet dengan cara melakukan *filtering* atas paket yang lewat dari dan ke jaringan-jaringan yang dihubungkan dan dapat dikonfigurasi untuk menolak akses ke website tertentu pada waktu-waktu tertentu.

- *Caching*

*Proxy server* memiliki mekanisme penyimpanan obyek-obyek yang sudah pernah diminta dari server-server di Internet, biasanya disebut *caching*. *Caching Proxy* menyimpan data yang

sering diminta oleh *client* ke suatu tempat penyimpanan di server.

Pengembangan pengontrolan dan performa terhadap koneksi Internet banyak dilakukan dengan memanfaatkan *proxy server*. Umumnya pengembangan tersebut dilakukan dengan menganalisis *log* atau *cache proxy server*. Karena di dalam file tersebut tersimpan informasi kejadian yang dilakukan oleh pengguna Internet.

Beberapa vendor juga telah mengembangkan perangkat lunak untuk mengontrol kegiatan *client proxy server*, diantaranya yaitu *sqstat*, *sarg*, dan *srg*. Umumnya analisis yang dilakukan ditampilkan ke dalam bentuk web agar mudah dioleh sesuai dengan kebutuhan. Tetapi program tersebut hanya memberikan informasi terhadap kegiatan pengguna saja, belum ada imbal balik dari informasi tersebut ke *proxy server* agar dapat dikonfigurasi kembali sesuai dengan informasi yang didapat. Jika ingin mengubah konfigurasi, harus dilakukan secara manual. Selain itu, program tersebut juga tidak sepenuhnya dapat dimanfaatkan di lingkungan kampus atau pendidikan, karena fitur yang disediakan bersifat standar.

## METODE PENELITIAN

Untuk membangun sebuah sistem pencatatan dan pembagian data *log proxy server* maka diperlukan tahapan-tahapan penelitian. Tahapan-tahapan yang digunakan mengacu pada

*system development life cycle* (SDLC). Langkah-langkah penelitian yang dilakukan meliputi :

a. Analisis masalah dan pengumpulan data, terdiri dari :

- Menganalisis masalah yang terjadi di UPT Laboratorium/ Pusat Komputer STMIK PPKIA Pradnya Paramita.
- Menyusun daftar permasalahan yang dihadapi.
- Mengumpulkan data yang terkait.
- Menganalisis data yang diperlukan.

b. Mendesain alur sistem pengontrolan dan mendesain database.

- Menentukan dan merancang model otomata yang digunakan untuk memecahkan beberapa masalah yang telah disusun.
- Mendesain sistem menggunakan metode UML (*Unified Modeling language*).
- Mendesain database yang diperlukan.
- Merancang tatap muka program.
  - c. Pengkodean program.
  - d. Pengimplementasian program.
  - e. Pengujian program dan perbaikan.
- Pengujian menggunakan metode *whitebox*.
- Pengujian menggunakan metode *blackbox*.

## HASIL DAN PEMBAHASAN

Program pencatatan dan pembagian data *proxy server squid* ini digunakan untuk mencatat  
Jurnal Teknologi Informasi Vol 3. No.1

kembali dan pemilahan data *log proxy server squid* agar dapat disimpan ke dalam sebuah database untuk digunakan sebagai bahan analisis selanjutnya. Hal ini diperlukan karena file log proxy server dicatat dalam file teks secara sequensial sehingga akan sulit untuk diolah menjadi sebuah informasi yang bermanfaat.

### 1. Analisis Masalah dan Pengumpulan Data

Analisis dilakukan dengan melakukan survey langsung ke bagian yang terkait di kampus STMIK PPKIA Pradnya Paramita Malang yaitu administrator Internet dan sub bagian data. Dari hasil survei disusun daftar permasalahan yang dihadapi sebagai berikut:

- Proses pembacaan *log proxy server, log* yang dipakai hanya memanfaatkan file *access.log*. Data dicatat kembali kedalam sebuah database agar dapat lebih mudah proses.
- Pembuatan laporan mengenai akses Internet, diantaranya laporan kunjungan *user*, grafik pemakaian tiap jurusan, grafik pemakaian Internet dan peringakt situs yang sering dikunjungi.

Data yang diperlukan untuk analisis tersebut yaitu :

- Data *access.log* dari *proxy server squid*.
- Data mahasiswa beserta konfigurasi IP komputer yang dipakai (misalnya laptop).

### 2. Desain Sistem

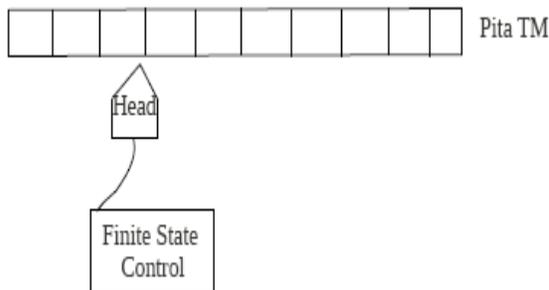
Langkah-langkah desain sistem dilakukan dengan cara:

---

<sup>\*)</sup> Dosen STMIK PPKIA Pradnya Paramita Malang

- Model otomata.

Model otomata ini digunakan untuk memecahkan permasalahan pembacaan *file access.log* dan menentukan *file access.log* yang sedang dibaca. Pembacaan *file access.log* menggunakan metode otomata mesin Turing yang merupakan suatu *deterministic finite state machine* yang mempunyai *tape* dua arah. Fungsi *tape* adalah sumber dari input mesin, tempat untuk menyimpan hasil berikutnya dan merupakan suatu perangkat dimana output dikirim. Mesin dilengkapi dengan *head* yang dapat membaca satu simbol masukan pada *tape* (Swingly Purba : 115). Penggambaran dari mesin Turing seperti gambar berikut :



Mesin Turing digunakan untuk memilah data dari isi file *access.log*. Pita mesin Turing mewakili teks yang akan diteliti. *Head* mewakili variabel yang menampung posisi awal simbol spasi atau *slash* sebagai awal data. Kemudian *head* akan digeser menelusuri pita mesin Turing ke kanan sampai bertemu dengan spasi atau *slash*. *String* sepanjang perjalanan *head* merupakan data yang dicari. Proses selanjutnya yaitu memindah variabel awal data ketempat *head* berada. Setelah

itu proses akan dilanjutkan kembali sampai bertemu dengan simbol pemisah tersebut atau sampai akhir dari teks yang ditelusuri. Algoritma tersebut merupakan isi dari *finite state control* yang menggerakkan *head*.

Algoritma tersebut dapat dituliskan sebagai berikut :

$$L = \{a^n b^n \mid n \geq 0\}$$

$n$   
 $a$  : awal pencarian yaitu deteksi spasi pertama  
 $n$   $n$   $n$   
 $b$  : spasi pasangan dari  $a$ . Dan  $b$  merupakan awal  $a$  berikutnya.

Algoritma di atas lebih diperinci lagi sebagai berikut :

1. Mulai dari S, head membaca simbol a.
2. Overwrite a tersebut dengan suatu simbol (misalkan A) untuk menandakan bahwa tersebut sudah dibaca. Selanjutnya head harus bergerak ke kanan untuk mencari sebuah b sebagai pasangan a yang sudah dibaca tersebut.

- Jika yang ditemukan adalah simbol a maka a tersebut harus dilewati (tidak boleh dioverwrite), dengan kata lain a dioverwrite dengan a juga dan head bergerak ke kanan.
- Jika TM pernah membaca simbol b ada kemungkinan ditemukan simbol B. Simbol B tersebut harus dilewati (tidak boleh dioverwrite), artinya B dioverwrite

dengan B juga dan head bergerak ke kanan.

3. Head membaca simbol b, maka b tersebut harus dioverwrite dengan simbol lain (misalnya B) untuk menandakan bahwa b tersebut (sebagai pasangan dari a) telah dibaca, dan head bergerak ke kiri untuk mencari simbol A.

- Jika ditemukan B maka B tersebut harus dilewati (tidak boleh dioverwrite), dengan kata lain B dioverwrite dengan B juga dan head bergerak ke kiri.
- Jika ditemukan a maka a tersebut harus dilewati (tidak boleh dioverwrite), dengan kata lain a dioverwrite dengan a juga dan head bergerak ke kiri.

4. Head membaca simbol A, maka A tersebut harus dilewati (tidak boleh dioverwrite), dengan kata lain A dioverwrite dengan A juga dan head bergerak ke kanan.

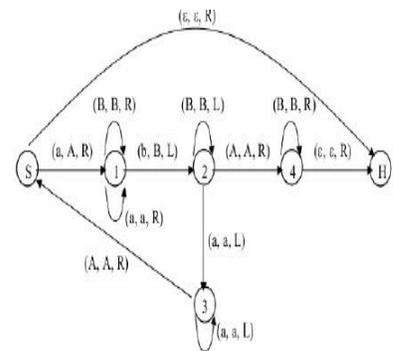
5. Head membaca simbol a, ulangi langkah 2 dan 3.

6. (Setelah langkah 3) head membaca simbol A, maka A tersebut harus dilewati (tidak boleh dioverwrite), dengan kata lain A dioverwrite dengan A juga dan head bergerak ke kanan.

7. Head membaca simbol B, maka B tersebut harus dilewati (tidak boleh dioverwrite), dengan kata lain B dioverwrite dengan A juga dan head bergerak ke kanan.

8. Head membaca simbol  $\epsilon$ , maka  $\epsilon$  dioverwrite dengan  $\epsilon$  dan head bergerak ke kanan menuju stata H

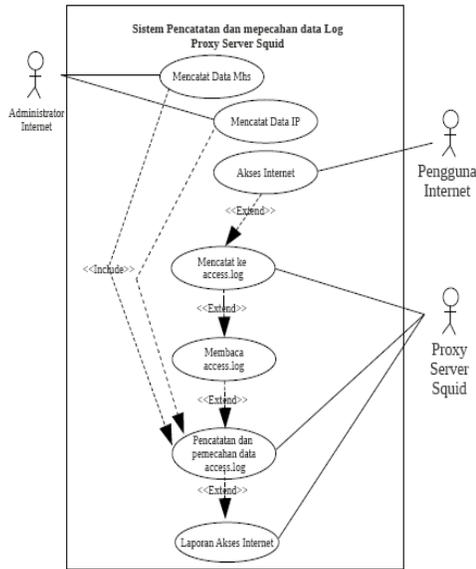
Skema graf mesin turing sebagai berikut :



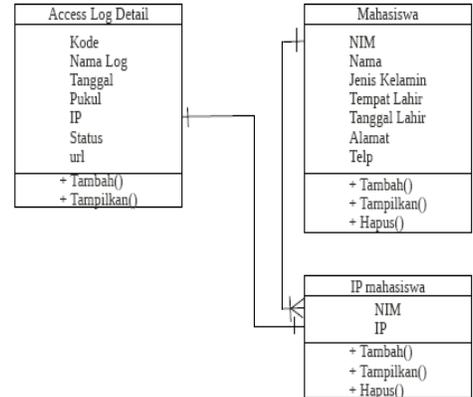
- Desain sistem dengan UML

Desain sistem yang digunakan adalah: metode *Object Oriented Analysis*, yaitu UML (*Unified Modeling Language*). Adapun metode UML yang digunakan dalam desain aplikasi sistem pengontrolan *proxy server squid* ini adalah : *Use-Case Diagram*, *Activity Diagram*, *Class Diagram* dan *Sequence Diagram*. Berikut rincian desain sistem:

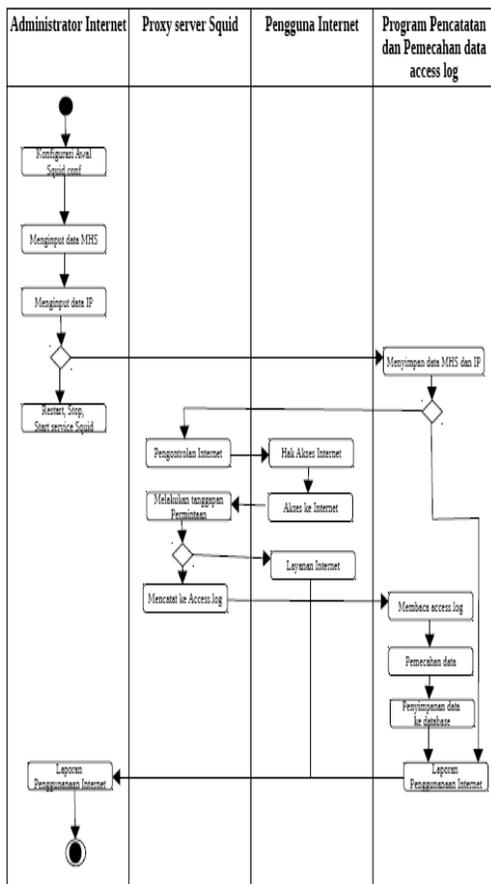
- *Use-Case Diagram*



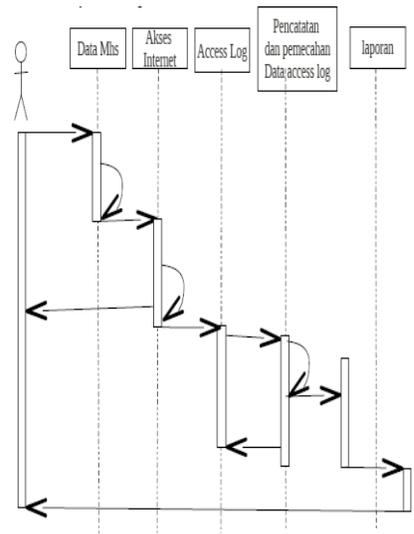
◦ *Class Diagram*



◦ *Activity Diagram*



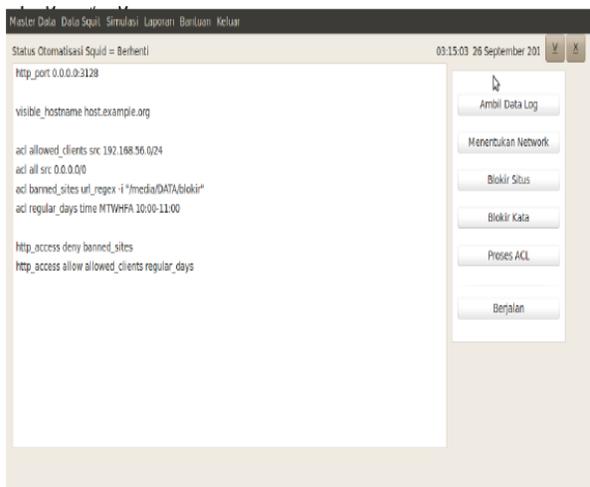
◦ *Sequence Diagram*



• Hasil Implementasi

Dari desain tersebut kemudian dibuat ke dalam bahasa Gambas dan database MySQL untuk membentuk sebuah sistem pencatatan dan

pemecahan data *proxy server squid*. Berikut tampilan program yang dibuat :



## KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian, dapat mengambil kesimpulan sebagai berikut:

- Pembacaan *log proxy server squid* khususnya pada *file access.log* dapat dilakukan dengan metode otomata mesin turing, begitu pula pemecahan data dapat dilakukan dengan metode otomata mesin turing.

Adapun saran yang dapat diberikan penulis untuk penelitian berikutnya adalah sebagai berikut :

- Dalam proses pembuatan *query* untuk menganalisis data *access.log* dan data pendukung kegiatan belajar mengajar, perlu diadakan penyederhanaan untuk mempercepat proses otomatisasi pengontrolan dan pelaporan.

- Diperlukan algoritma untuk mencatat temuan situs-situs yang ditutup agar menambah daftar alamat Internet yang yang ditutup.
- Pemanfaatan *file cache.log* dan *store.log* untuk pengembangan pembacaan *log proxy server* perlu dilakukan untuk menambah kehandalan program.

## DAFTAR PUSTAKA

- D.S. Oberol. 2002. *Journal Setting up a Squid Proxy Server*. Auugn, Australian.
- Pungky, Sulistyo. Didiek S. Wiyono, Bambang Harjito. 2007. *Jurnal Membangun Infrastruktur IT Menggunakan Squid sebagai Proxy Server* . Universitas Sahid, Surakarta.
- Purba, Swingly. 2008. *Otomata dan Bahasa Formal*. Graha Ilmu, Yogyakarta.
- Rafiudin, Ahmad. 2003. *Panduan Membangun Jaringan Komputer Untuk Pemula*. PT. Elex Media Komputindo, Jakarta.
- Rousskov, Alex. 2007. *Journal A Performance Study of the Squid Proxy on HTTP/1.0*. Inktomi Corporation, San Mateo.
- Syahroni, Nanang. Titon Dutono, Supeno Djanali. 2003. *Journal A Performance Comparison of Normal Query and Adaptive Query in Web-Client Monitoring System* . Fukui University, Japan.

Vakali, Athena. 2002. *Journal Proxy Cache Replacement Algorithms: A History-Based Approach* . Aristotle University, Greece.

Vakali, Athena. 2005. *Journal A Web-based evolutionary model for Internet Data Caching* . Aristotle University, Greece  
<http://news.netcraft.com>