

**PENGEMBANGAN SISTEM MONITORING LAN BERBASIS
SMS GATEWAY**

SKRIPSI



Disusun oleh :

Andy da Cruz

NIM. 0912916

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2011**



BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI


Nama : ANDY DA CRUZ
Nim : 09.12.916
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer S-1
Judul : **PENGEMBANGAN SISTEM MONITORING LAN
BERBASIS SMS GATEWAY**

Dipertahankan dihadapan Tim Pengujian Skripsi jenjang Program Strata Satu (S-1)

Pada Hari : Selasa
Tanggal : 9 Agustus 2011
Dengan Nilai : 77.6 (B+) *r*

Panitia Ujian Skripsi :

Ketua Majelis Penguji


Ir. Yusuf Ismail Nakhoda, MT.
NIP.Y.1018800189

Sekretaris Majelis Penguji



Dr. Eng. Aryuanto S, ST, MT.
NIP.P.1030800417

Anggota Penguji :

Penguji I


Irmalia S. Faradisa, ST, MT.
NIP.P. 1030000365

Penguji II


Ahmad Faisol, ST.
NIP.P.1030000431

LEMBAR PERSETUJUAN
PENGEMBANGAN SISTEM MONITORING LAN BERBASIS SMS
GATEWAY

SKRIPSI

Disusun dan diajukan untuk melengkapi dan memenuhi persyaratan
guna mencapai gelar sarjana teknik

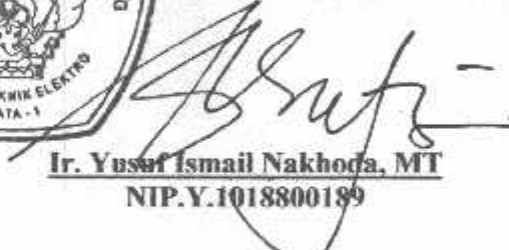
Disusun oleh :

Andy da Cruz
NIM. 0912916

Mengetahui,

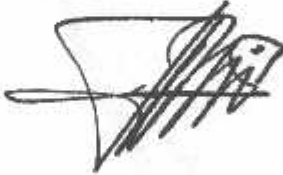


Ketua Program studi Teknik Elektro S-1


Ir. Yusuf Ismail Nakhoda, MT
NIP.Y.1018800189

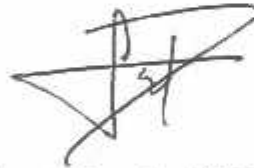
Diperiksa dan disetujui,

Dosen Pembimbing I



SotyoHadi, ST
NIP.Y.103970030

Dosen Pembimbing II



Sonny Prasetio, ST,MT
NIP.P.1031000433

JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2011

ABSTRAK

PENGEMBANGAN SISTEM MONITORING LAN BERBASIS SMS GATEWAY

Andy da Cruz, NIM 0912916

Dosen Pembimbing : SotyoHadi, ST dan Sonny Prasetyo, ST,MT

Monitoring jaringan merupakan sebuah proses untuk mengetahui bagaimana kondisi terkini suatu jaringan. Dengan mengetahui kondisi terkini suatu jaringan, seorang administrator dapat menentukan apa yang harus dilakukan jika suatu saat terjadi masalah. Untuk melakukan proses ini, umumnya seorang administrator jaringan harus berinteraksi dengan komputer server/ client secara langsung.

Dengan memanfaatkan teknologi website yang terus berkembang, dapat dibangun sebuah sistem monitoring realtime berbasis web untuk membantu proses ini secara otomatis. Administrator dapat mengakses aplikasi ini dari komputer manapun di dalam jaringan yang dikelolanya. Selain itu, aplikasi berbasis web dapat dengan mudah diintegrasikan dengan SMS Gateway, sehingga seorang administrator menerima informasi tentang kondisi terkini jaringan yang dikelolanya kapanpun dan dimanapun ia berada selama berada di dalam jangkauan operator seluler.

Kata Kunci : Client-Server, Website, Monitoring Jaringan, SMS Gateway

Abstract

Network Monitoring is a process which aims to find out the recent condition of a computer network. By knowing the recent condition of the network, administrator could determine what steps should be taken, if there's any problem within the network which he maintain. Usually, network administrator must interact with the server or workstations directly to do this process.

By the evolve of the webs technology, we could develop a realtime web-based network monitoring system to help this process automatically. Administrator can access the application from server or any workstations in the network which he maintain. It can be integrated with SMS Gateway easily, so the administrator receive the information about the recent condition of the network to his phone, anytime, anywhere as long as in the coverage of mobile operators.

Keyword : Client-Server, Website, Network Monitoring, SMS Gateway

KATA PENGANTAR

Puji dan syukur kehadirat Tuhan Yang Maha Esa yang telah memberikan RahmatNya, sehingga penulis dapat menyelesaikan penulisan skripsi yang berjudul **“PENGEMBANGAN SISTEM MONITORING LAN BERBASIS SMS GATEWAY”** ini dengan lancar dan tepat pada waktunya. Skripsi ini merupakan persyaratan kelulusan studi pada Jurusan Teknik Elektro S-1 konsentrasi Teknik Komputer ITN Malang dan untuk mencapai gelar Sarjana Teknik.

Keberhasilan dari penyelesaian skripsi ini tidak lepas dari dukungan dan bantuan dari semua pihak yang telah berperan aktif baik secara langsung maupun tidak langsung. Untuk itu penyusun menyampaikan terima kasih kepada:

1. Bapak Ir. Soeparno Djiwo, MT, selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. Sidik Noertjahjono, MT, selaku Dekan Fakultas Teknologi Industri
3. Bapak Ir. Yusuf Ismail Nakhoda, MT, selaku Ketua Jurusan Teknik Elektro S-1.
4. Bapak Eng. Aryuanto S. ST, MT, selaku sekretaris Jurusan Teknik Elektro S-1.
5. Bapak SotyoHadi ST, selaku Dosen Pembimbing I.
6. Bapak Sonny Prasetyo ST, MT, selaku Dosen Pembimbing II.
7. Orang tua dan keluarga kami yang telah memberikan doa restu, dorongan serta semangat.

8. Rekan – rekan kuliah khususnya jurusan Elektro S-1 ITN Malang.
9. Semua pihak yang telah membantu dalam penyelesaian penyusunan skripsi ini.

Penulis telah berusaha semaksimal mungkin dan menyadari sepenuhnya akan keterbatasan pengetahuan dalam menyelesaikan laporan skripsi ini. Untuk itu penulis mengharapkan saran dan kritik yang membangun dari pembaca demi kesempurnaan laporan skripsi ini.

Harapan penyusun smoga laporan skripsi ini dapat memberikan mamfaat perkembangan ilmu pengetahuan dari para pembaca.

Malang, Agustus 2011

Penyusun

DAFTAR ISI

LEMBAR PERSETUJUAN	i
ABSTRAKSI	ii
KATA PENGANTAR	iii
DAFTAR ISI	v
DAFTAR GAMBAR	ix
DAFTAR TABEL	xii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian	2
1.4 Batasan Masalah	2
1.5 Metode Penelitian	3
1.5.1. Metode pengumpulan data	3
1.5.2. Metode pengembangan sistem	3
1.6 Sistematika Penulisan	4
BAB II LANDASAN TEORI	5
2.1 Pengertian dasar sitem	5
2.2 pengertian database dan DBMS	6
2.2.1. Pengertian database	6

2.2.2. Database Managemen Sistem	7
2.3 MySql	8
2.4 PHP	11
2.5 Perangkat Analisis dan perancangan	12
2.5.1. Data Flow Diagram (DFD)	12
2.5.2. Entity Relationship Model	13
2.6 SMS gateway	16
2.7 Gammu	17
2.8 Jaringan Komputer	19
2.9 LAN (Local Area Network)	20
BAB III PERANCANGAN SISTEM	21
3.1 Gambaran umum sistem monitoring	21
3.2 Analisa kebutuhan sistem	21
3.3 Perancangan sistem monitoring	22
3.3.1. Perintah yang digunakan untuk memonitor	23
3.3.2. SMS gateway	24
3.3.3. Protokol yang digunakan dalam sistem monitoring	24
3.3.4. Prinsip kerja SMS	25
3.4 Desain sistem	25
3.4.1. <i>Flowchart</i>	27
3.4.1.1 <i>Flowchart</i> sistem monitoring	27
3.4.1.2 <i>Flowchart</i> SMS request	29

3.4.2 DFD (<i>Data Flow Diagram</i>)	30
3.4.2.1 Diagram Konteks	31
3.4.2.2 DFD level 1 sistem monitoring	33
3.4.2.3 ERD sistem monitoring	36
3.5 Perancangan Database	36
BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM	40
4.1 Implementasi sistem	40
4.1.1 Aplikasi monitoring LAN	40
4.1.1.1 Inisialisasi sistem	41
4.1.1.2 Monitoring	41
4.1.2 Administrasi data sistem monitoring	43
4.1.2.1 Autentifikasi user	44
4.1.2.2 Menu status	44
4.1.2.3 Menu laporan <i>gateway</i>	45
4.1.2.4 Menu laporan <i>user log</i>	47
4.1.2.5 Menu setting <i>gateway</i>	48
4.1.2.6 Menu setting <i>user list</i>	48
4.1.2.7 Menu setting <i>phone number</i>	49
4.1.2.8 Menu setting <i>general</i>	50
4.1.2.9 Menu setting <i>my account</i>	51
4.2 Pengujian sistem	52
4.2.1 pengujian pada topologi jaringan sederhana	52

4.2.2 pengujian secara otomatis	53
4.2.2.1 Pengujian jika seluruh klien aktif(<i>UP</i>)	58
4.2.2.2 Pengujian jika salah satu klien bermasalah	59
4.2.3 Pengujian sistem monitoring secara manual	60
4.2.3.1 Pengujian ping IP secara keseluruhan	61
4.2.3.2 Pengujian ping salah satu IP	62
4.2.4 Pengujian pada <i>browser</i>	66
4.2.5 Pengujian dengan menggunakan USB modem	68

BAB V PENUTUP

5.1 Kesimpulan	70
5.2 Saran	71

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

2.1	Model umum sebuah sistem	5
2.2	Tingkatan <i>database</i>	6
2.3	Relasi <i>one to one</i>	15
2.4	Relasi <i>one to many</i>	15
2.5	Relasi <i>many to many</i>	16
2.6	SMS <i>gateway</i> sebagai penghubung antar SMSC	16
2.7	SMS <i>gateway</i> sebagai penghubung antara aplikasi SMS dengan SMSC melalui protokol HTTP/HTTPS	17
3.1	Prinsip kerja SMS	25
3.2	Desain sistem monitoring LAN	26
3.3	<i>Flowchart</i> sistem monitoring	27
3.4	<i>Flowchart</i> proses <i>request</i> SMS	29
3.5	Diagram konteks sistem monitoring	31
3.6	DfD level 1 sistem monitoring LAN	33
3.7	ERD sistem monitoring	36
4.1	Tampilan sistem monitoring secara otomatis	41
4.2	Aplikasi memonitor jaringan secara otomatis	42
4.3	Tampilan halaman <i>login</i>	44
4.4	Tampilan menu status <i>gateway</i>	45
4.5	Tampilan report status <i>gateway</i>	46
4.6	Tampilan report IP <i>gateway</i> dalam bentuk pdf	46
4.7	Tampilan report <i>user log</i>	47

4.8	Tampilan report <i>user log</i> dalam bentuk pdf	47
4.9	Tampilan menu <i>gateway</i>	48
4.10	Tampilan menu <i>user list</i>	49
4.11	Tampilan menu <i>phone number</i>	50
4.12	Tampilan menu <i>setting general</i>	51
4.13	Tampilan menu <i>setting my account</i>	52
4.14	Topologi jaringan dalam pengujian	52
4.15	Tipe koneksi data dari HP	54
4.16	File <i>gammurc</i>	54
4.17	File <i>smsdrc</i>	55
4.18	Perintah identifikasi HP pada gammu	55
4.19	Perintah menjalankan gammu	56
4.20	Tampilan menginstall gammu pada <i>command prompt</i>	56
4.21	Tampilan sistem <i>services</i> gammu	57
4.22	Tampilan <i>command prompt</i> untuk <i>uninstall services</i> gammu	57
4.23	Tampilan SMS saat aplikasi <i>start up</i>	58
4.24	Tampilan aplikasi dalam keadaan <i>up</i>	58
4.25	Tampilan aplikasi bila salah satu jaringan <i>DOWN</i>	59
4.26	Tampilan SMS yang di terima administrator	59
4.27	Tampilan SMS bila jaringan kembali normal	60
4.28	Tampilan format untuk ping semua IP	61
4.29	Tampilan proses pada <i>command prompt</i> saat request semua IP	61
4.30	Bila semua IP dalam keadaan <i>UP</i>	62

4.31 Tampilan format request salah satu IP dengan menggunakan alias	62
4.32 Tampilan proses pada <i>command prompt</i> saat menerima request	63
4.33 Hasil balasan bila IP dalam keadaan <i>UP</i>	63
4.34 Hasil balasan bila IP dalam keadaan <i>DOWN</i>	63
4.35 Tampilan request dengan menggunakan nomor IP	64
4.36 Tampilan proses pada <i>command prompt</i> saat menerima request	64
4.37 Tampilan format kesalahan penulisan request	65
4.38 Hasil balasan SMS dari sistem aplikasi	65
4.39 Tampilan halaman admin pada <i>browser google chrome</i>	66
4.40 Tampilan halaman admin pada <i>browser Opera</i>	67
4.41 Tampilan halaman admin pada <i>browser internet explorer</i>	67
4.42 Tampilan halaman admin pada <i>browser mozilla firefox</i>	68
4.43 Tampilan Hasil <i>error</i> pada <i>command prompt</i>	69

DAFTAR TABEL

2.1	<i>Wild card</i> pada <i>key word</i>	10
2.2	Simbol DFD	13
2.3	Simbol E-R model	14
2.4	Merek <i>handphone</i> dan <i>modem</i> yang kompatibel dengan Gammu	18
3.1	<i>nms_useraccess</i>	36
3.2	<i>nms_users</i>	37
3.3	<i>nms_roles</i>	37
3.4	<i>nms_phonenumber</i>	37
3.5	<i>nms_gateway</i>	37
3.6	<i>status</i> (<i>nms_status</i>)	38
3.7	<i>detail status</i> (<i>nms_status_detail</i>)	38
3.8	<i>log user</i> (<i>nms_userlogs</i>)	38
3.9	<i>setting</i> (<i>nms_settings</i>)	39
3.10	<i>request</i> (<i>nms_request</i>)	39
4.1	Spesifikasi perlengkapan implementasi	40
4.2	perangkat untuk pengujian sistem	53

BAB I

PENDAHULUAN

1.1. Latar Belakang

Jaringan komputer mulai berkembang sekitar tahun 1960, diawali dengan sebuah proyek bernama ARPA (*Advanced Research Project Agency*). Dalam perkembangannya, penggunaan jaringan komputer yang semula hanya untuk kalangan tertentu, kini telah meluas mencakup hampir seluruh masyarakat dunia. Jaringan komputer kini telah menjadi basis komunikasi modern dan sangat membantu manusia dalam berbagai hal.

Salah satu contoh nyata yakni penggunaan jaringan komputer untuk mempermudah komunikasi dalam perusahaan – perusahaan besar atau instansi pemerintah dan pendidikan yang pada dasarnya terdiri dari beberapa bagian atau divisi yang saling berkaitan. Dengan membangun aplikasi yang memanfaatkan jaringan komputer sebagai media penghubung, maka efisiensi waktu dan tenaga dapat dicapai karena pegawai antar bagian tidak harus berinteraksi secara langsung.

Semakin besar skala suatu perusahaan, maka semakin besar pula ketergantungannya akan jaringan komputer. Hal ini menyebabkan semakin besarnya sumber daya yang diperlukan untuk membangun jaringan tersebut, serta kualitas jaringan yang mungkin menurun akibat *traffic* – lalu lintas data yang tinggi. Seorang *network* administrator harus melakukan memonitor jaringan yang di kelolanya setiap saat untuk mengetahui keadaan jaringan yang dikelolanya mengalami masalah atau tidak.

Untuk mempermudah pekerjaan seorang administrator jaringan, dapat dikembangkan sebuah sistem yang dapat melakukan monitoring jaringan secara otomatis dan *realtime*. Sistem ini sangat membantu administrator jaringan yang tidak mungkin siaga selama 24 jam di depan komputer untuk melakukan monitoring jaringan.

Dengan aplikasi ini, informasi mengenai permasalahan khususnya *error* koneksi dalam jaringan dapat disampaikan secara langsung ke handphone administrator dengan memanfaatkan SMS *Gateway* sehingga informasi yang diperoleh oleh administrator adalah informasi yang mewakili kondisi jaringan saat itu.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, muncul sebuah permasalahan yakni bagaimana mengembangkan sebuah sistem monitoring berbasis SMS Gateway pada jaringan LAN yang dapat mengirimkan informasi terbaru mengenai *error* koneksi jaringan ke HP administrator/user melalui handphone atau USB modem yang terkoneksi pada PC aplikasi.

1.3. Tujuan Penelitian

Menghasilkan sebuah sistem yang melakukan monitoring ke jaringan secara *realtime*, dan dapat menyampaikan informasi melalui SMS dengan akurat dan cepat ke Administrator jaringan dimanapun dia berada. Dengan begitu Administrator jaringan dapat dengan cepat mengambil langkah – langkah yang perlu untuk memperbaiki masalah yang ada pada jaringannya.

1.4. Batasan Masalah

Agar permasalahan mengarah sesuai dengan tujuan yang diharapkan, maka pembahasan dibatasi oleh hal – hal berikut ini :

1. Tidak membahas secara spesifik mengenai perangkat – perangkat keras komputer.
2. Aplikasi monitoring jaringan hanya terbatas pada jaringan LAN dan *sub network* pada LAN tersebut.
3. Sistem monitoring hanya memonitor kinerja jaringan dalam hal *error* koneksi pada *subnetwork* pada LAN.
4. Batas jumlah *subnet* yg dapat di monitor hanya di batasi 5 *subnet*.
5. IP yang di monitor oleh aplikasi hanya IP yang telah di *setting* dalam aplikasi.
6. Tidak membahas mengenai proses pengiriman pesan atau SMS melalui operator telekomunikasi (*carrier*).

1.5. Metodologi Penelitian

1.5.1. Metode Pengumpulan data

Data yang digunakan dalam pengembangan sistem informasi haruslah berkualitas dan bermanfaat. Oleh karena itu dalam pengambilan data harus dengan metode yang tepat. Dalam pengumpulan data penyusun menggunakan metode Pengumpulan data, Ini dilakukan dengan cara mencari bahan-bahan kepustakaan sebagai landasan teori yang ada hubungannya dengan permasalahan yang dijadikan sebagai obyek penelitian, serta mengumpulkan sumber berupa perintah – perintah yang digunakan dalam sistem monitoring jaringan.

1.5.2. Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan adalah sebagai berikut:

1. Analisis dan perancangan sistem

Pada tahap ini dilakukan analisa kebutuhan sistem yang digunakan sebagai acuan untuk melakukan perancangan sistem aplikasi. Perancangan sistem aplikasi dilakukan setelah diketahui melalui analisa kebutuhan sistem sebelumnya.

2. Implementasi dan pengujian sistem

Implementasi dilakukan dengan menempatkan hasil perancangan ke dalam kode-kode program yang akan diintegrasikan menjadi suatu unit program yang dapat dijalankan. Pengujian sistem dilakukan untuk mengetahui apakah spesifikasi sistem sudah terpenuhi.

3. Integrasi dan pengujian sistem

Integrasi sistem dilakukan untuk menggabungkan setiap unit program menjadi suatu sistem yang lengkap dan tahap selanjutnya adalah tahap pengujian yang bertujuan untuk mengetahui apakah sistem tersebut memenuhi persyaratan.

1.6. Sistematika Penulisan

Adapun sistematika penulisan laporan Skripsi ini terdiri atas :

BAB I : PENDAHULUAN

Berisi penjelasan tentang latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian serta sistematika penulisan.

BAB II : LANDASAN TEORI

Bab ini berisi tentang penjelasan teori - teori penunjang yang berhubungan dengan topik skripsi, meliputi hal-hal yang berhubungan dengan perancangan sistem, serta teori-teori dasar lainnya seperti *PHP*, *Data Base*, dan sebagainya.

BAB III : PERANCANGAN SISTEM

Bab ini berisi tentang deskripsi sistem serta desain rancangan sistem.

BAB IV : IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini menyajikan hasil implementasi serta pengujian mengenai cara kerja dari sistem.

BAB V : PENUTUP

Bab ini berisi tentang kesimpulan dan saran dari hasil pembahasan pada skripsi ini.

BAB II

LANDASAN TEORI

2.1. Pengertian Dasar Sistem

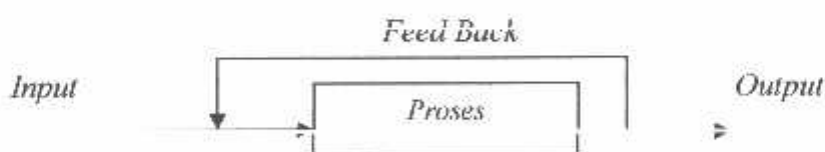
Berikut ini definisi tentang sistem yang merupakan pendapat dari beberapa ahli diantaranya adalah sebagai berikut :

Menurut **Steven A. Moscovice** dan **Mark G. Sinkin** yang diterjemahkan oleh **Jogiyanto H. M.** (1998 : 4), dalam buku *Accounting Information System* mengemukakan bahwa : "Sistem adalah suatu kesatuan yang terdiri dari bagian-bagian (disebut sebagai subsistem) yang saling berkaitan dan berusaha untuk mencapai tujuan (*goal*) yang sama".

Sedangkan menurut **Gordon B. Davis** (2000 : 784) sistem didefinisikan sebagai berikut : "Sistem adalah suatu jaringan kerja dari prosedur yang saling berhubungan, berkumpul dan bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran atau tujuan tertentu".

Teori sistem umum terutama menekankan perlunya memeriksa seluruh bagian sistem. Suatu sistem terdiri dari komponen-komponen, yaitu pekerjaan, kegiatan, misi atau bagian-bagian sistem yang dibentuk untuk mewujudkan tujuan. Untuk komponen misi atau tujuan, seringkali sukar untuk dilihat. Manajemen suatu sistem terdiri dari kegiatan-kegiatan yang diarahkan pada perencanaan dan pengendalian (*feedback*).

Dalam alur kerja sebuah sistem perlu ada masukkan (*inputan*), proses dan keluaran (*output*) serta *feedback* yang digunakan untuk dapat digunakan sebagai referensi untuk melihat kinerja dari sistem tersebut. Model umum dari sebuah sistem ditunjukkan dalam Gambar 2.1 berikut :



Gambar 2.1. Model Umum Sebuah Sistem

Dari beberapa definisi mengenai sistem diatas, maka dapat diambil kesimpulan bahwa suatu sistem pada dasarnya adalah suatu kesatuan jaringan kerja yang saling

berkaitan yang disusun untuk mempermudah dalam mencapai sasaran tertentu.

2.2. Pengertian Database dan DBMS

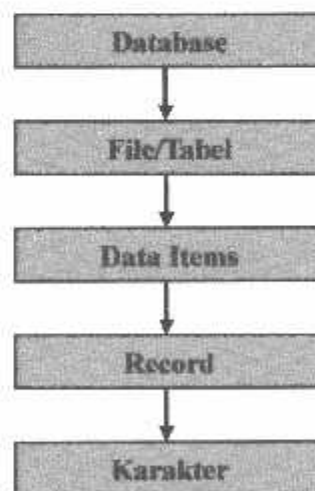
2.2.1. Pengertian Database

Berikut ini definisi tentang *database* yang merupakan pendapat dari beberapa ahli diantaranya adalah :

Menurut **Jogianto H. M.** (1990 : 217) pengertian *database* didefinisikan sebagai berikut : “*Database* merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, dan terdapat pada simpanan luar komputer serta digunakan perangkat lunak tertentu untuk memanipulasinya”.

Menurut **Hariato Kristanto** (1994 : 3) pengertian *database* didefinisikan sebagai berikut : “*Database* adalah kumpulan *file-file* yang mempunyai kaitan antara *file* yang satu dengan yang lainnya sehingga membentuk suatu bangunan data untuk menginformasikan satu perubahan, instansi dalam batasan tertentu”.

Dari beberapa definisi mengenai *database* diatas, maka dapat diambil kesimpulan bahwa *database* adalah kumpulan dari data yang terintegrasi satu sama lain dan apabila akan disimpan maka diletakkan pada media penyimpanan tertentu serta untuk mengambil dan memanipulasinya diperlukan prosedur yaitu dalam bentuk program yang berfungsi sebagai perintah. Dalam tingkatannya data diklasifikasikan dari tingkatan yang terbesar ke tingkatan yang terkecil adalah seperti digambarkan dalam Gambar 2.2



Gambar 2.2 Tingkatan *Database*

Keterangan :

1. Karakter

Karakter merupakan bagian data yang terkecil, dapat berupa karakter, numerik, huruf ataupun karakter- karakter khusus yang membentuk suatu *items* data(*fields*).

2. *Record*

Kumpulan dari data yang saling berhubungan membentuk suatu data. Jadi satu *record* mewakili satu data.

3. *Data Items*

Suatu *fields* menggambarkan suatu atribut dari *record-record* yang menunjukkan suatu item dari data misalnya Kode Pegawai, Nama Pegawai dan sebagainya.

4. *file/label*

Kumpulan dari *record-record* yang sejenis yang mempunyai panjang elemen yang sama, atribut yang sama namun berbeda data *value*-nya.

5. Database

Kumpulan *file-file* yang saling berkaitan satu dengan lainnya yang terpadu (*integrated*), terkontrol dan terpusat.

2.2.2. Database Management System

Menurut Abraham Silberschatz (1997 : 1) DBMS didefinisikan sebagai berikut : *"A Database Management System (DBMS) consist of a collection of interrelated data and a set of programs to access those data. The collection of data, usually referred to as the database, contains information about one particular enterprise. The primary goal of a DBMS is provide an environment that is most convenient and efficient to use in retrieving and storing database information"*.

Definisi tersebut menjelaskan bahwa DBMS terdiri dari kumpulan (koleksi) data yang saling terintegrasi antara satu dengan yang lain dan sebuah program untuk mengakses data-data tersebut. Kumpulan (koleksi) data tersebut biasanya ditunjukkan berupa *database*. *Database* sendiri merupakan komponen mendasar dan paling penting dalam suatu organisasi sistem informasi, yang di dalamnya terdiri dari komponen-

komponen berupa data yang terorganisasi. Data akan disimpan dalam satu atau lebih tabel, yang terdiri atas baris dan kolom.

Tujuan utama dari sebuah DBMS adalah sebagai penyedia lingkungan antara kemudahan, ketepatan, keakuratan serta efisiensi dalam pengelolaan dan penyimpanan sebagai database suatu informasi.

Sistem *database* didesain untuk manajemen sebuah bentuk informasi yang sangat besar. Manajemen data meliputi pendefinisian struktur dari penyimpanan suatu informasi serta mekanisme terhadap manipulasi suatu informasi. Sebagai tambahan penting adalah sebuah sistem *database* harus mampu memberikan keamanan atas informasi yang disimpan, walaupun sistem mengalami *crash* ataupun usaha yang tidak diberikan hak akses terhadap informasi tersebut. Apabila terdapat data yang dibagi (*share*) kepada banyak user, sistem harus dapat menghindari sebuah hasil informasi yang menyimpang (tidak sebenarnya).

Beberapa kemudahan dan keuntungan dalam menggunakan sistem *database* antara lain :

- Menghindari duplikasi data.
- Memudahkan atau mempercepat dalam pengaksesan dan pengolahan data, yaitu dengan data yang tersusun secara teratur dalam program.
- Memberikan keamanan data (*data security*).
- Menggunakan sedikit memory.
- Tidak memakan ruang dan tempat yang besar.
- Mudah dalam proses penyimpanan dan pemeliharaan
- Menghemat tenaga dalam mengolah dan menyimpan data

2.3. MYSQL

MySQL merupakan *database* yang paling digemari dikalangan *Programmer Web*, dengan alasan bahwa program ini merupakan *database* yang sangat kuat dan cukup stabil untuk digunakan sebagai media penyimpanan data. Sebagai sebuah *database Server* yang mampu untuk manajemen dengan baik, *MySQL* terhitung merupakan *database* yang paling digemari dan paling banyak digunakan dibanding *database* lainnya. Selain *MySQL* masih terdapat beberapa jenis *database server* yang juga

memiliki kemampuan yang juga tidak bisa dianggap sepele, *database* itu adalah Oracle, PostgreSQL dan lain-lain.

Kemampuan lain yang dimiliki MySQL adalah mampu mendukung *Relasional Database Manajemen System* (RDBMS), sehingga dengan kemampuan ini MySQL mampu menangani data-data perusahaan yang berukuran sangat besar hingga berukuran *Giga Byte*.

Hal lain yang perlu diketahui mengenai MySQL adalah bahwa MySQL merupakan sebuah software *database* yang dapat di gunakan dengan *Free* (Gratis) karena MySQL dilisensi dibawah *GNU General Public Licence* (GPL). Dengan adanya keadaan seperti itu maka siapa saja dapat menggunakan software *database* ini dengan bebas tanpa harus takut dengan hak cipta atau lisensinya.

Adapun beberapa perintah SQL yang didukung oleh MySQL adalah :

- **SELECT**

Untuk melihat data dari satu atau beberapa tabel.

```
select kolom-kolom  
from nama-tabel
```

- **INSERT INTO**

Untuk mengisi data pada suatu tabel atau menambah *record* pada tabel.

```
insert into nama-tabel (kolom1, kolom2,...)  
values (nilai1, nilai2,...);
```

- **DISTINCT**

Untuk menghilangkan *record record* yang sama.

```
select distinct kolom from nama-tabel;
```

- **SELECT ***

Untuk melihat isi kolom suatu tabel.

```
select * from nama-tabel;
```

- **WHERE**

Untuk menyaring / membatasi hasil *query* sehingga *record* yang dikeluarkan hanya *record* yang sesuai kriteria yang diinginkan.

```
select kolom, kolom ...from nama-tabel  
where criteria;
```

- **BETWEEN**

Untuk membatasi suatu kolom berada pada suatu baris nilai tertentu.

select kolom, kolom, ... from nama-tabel

where kriteria(salah satu kolom sebagai parameter) between .. and..;

- **LIKE**

Untuk mencari data yang memiliki pola tertentu.

Select kolom, kolom,..from nama-tabel

where kriteria(salah satu kolom) like '%win%';

Pada *keyword* ini digunakan tanda-tanda khusus (*wild card*). *Wild card* dalam *keyword* di tunjukkan dalam Tabel 2.1

Tabel 2.1. *Wild Card* Pada *Keyword*

Tanda	Arti
%	Sembarang nol/lebih karakter
-	Sembarang satu karakter
[]	Sembarang karakter dalam tanda kurung siku

- **ORDER BY**

Untuk mensortir data atau hasil *query*

select kolom, kolom from nama-tabel

order by kolom;

- **DESC**

Untuk mensortir data dengan urutan terbalik

select kolom, kolom from nama-tabel

order by kolom desc;

- **DELETE**

Untuk menghapus *record*

delete from nama-tabel;

Dengan kriteria tertentu,

delete from nama-tabel where kriteria;

- **UPDATE**

Untuk memodifikasi *record* nilai kolom secara keseluruhan.

update kolom set (kolom shg parameter)= 100;

Untuk memodifikasi nilai kolom dari suatu *record*

update nama-tabel

set nama-kolom1=nilai-baru1, nama-kolom2=nilai-baru2, ... where criteria;

2.4. PHP (PHP Hypertext Preprocessor)

PHP singkatan dari *PHP Hypertext Preprocessor*. Merupakan bahasa pemrograman berbasis *Web*. Yang memiliki kemampuan untuk memproses dan mengolah data secara dinamis, PHP ditempatkan dalam *server* dan diproses di dalam *server*. Hasilnya akan dikirimkan ke klien, pengguna mengakses dengan menggunakan *browser*.

Secara khusus, PHP dirancang untuk membentuk *web* dinamis, artinya ia dapat membentuk tampilan berdasarkan permintaan terkini. Misalnya, anda dapat membentuk suatu tampilan berdasarkan permintaan terkini.

Adapun beberapa keunggulan dari bahasa pemrograman PHP, sehingga banyak digunakan oleh para *web developer* adalah :

1. Gratis

PHP merupakan bahasa pemrograman *web* yang bersifat gratis sehingga banyak digunakan oleh *developer web* untuk membuat *website*.

2. *Cross Platform*

Artinya dapat digunakan di berbagai sistem operasi, mulai dari linux, windows, mac serta sistem operasi yang lainnya.

3. Mendukung Banyak *Database*

PHP telah mendukung banyak *database* seperti *MySQL*, *SQL*, *dBase*, *PostgreSQL*, *mSQL*, dan *database* lainnya.

4. *On The Fly*

PHP sudah mendukung *on the fly*, artinya dengan menggunakan PHP kita dapat membuat dokumen berupa teks, *Word*, *Excel*, *PDF*, menciptakan *image* dan

flash, juga menciptakan file seperti *zip*, *XML*, dan banyak lagi.

2.5. Perangkat Analisis dan Perancangan

2.5.1. *Data Flow Diagram* (DFD)

Berikut ini definisi tentang *Data Flow Diagram* (DFD) yang merupakan pendapat dari beberapa ahli diantaranya adalah :

Menurut **Raymond McLeod** (1998 : 610) *Data Flow Diagram* didefinisikan sebagai berikut : “*A data flow diagram (DFD) is a graphic representation of a system that uses a small number of symbol shapes to illustrate how data flows through interconnected processes*”.

Definisi tersebut menjelaskan bahwa *Data Flow Diagram* adalah suatu penyajian grafis dari suatu sistem yang menggunakan simbol-simbol untuk menggambarkan bagaimana aliran data saling berproses dan berhubungan.

Sedangkan menurut **Alton R. Kinkred** (1985 : 8) *Data Flow Diagram* didefinisikan sebagai berikut : “*A Data Flow Diagram describes the flow of data through a system as the processes that change or transform then data*”.



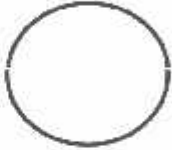

Definisi tersebut menjelaskan bahwa *Data Flow Diagram* adalah suatu gambaran tentang aliran data yang melewati suatu sistem dan beberapa proses yang bertujuan untuk mengubah data.

Jadi dapat disimpulkan *data flow diagram* adalah sebuah metode yang digunakan untuk menggambarkan arus data dari *input*, proses dan *output*-nya dalam suatu sistem yang disusun secara terstruktur dan jelas. Tujuan dari *data flow diagram* adalah menunjukkan proses data melalui sistem.

Berikut ini merupakan simbol-simbol yang digunakan di dalam DFD di tunjukkan dalam tabel 2.2

Tabel 2.2. Simbol DFD

Sumber : Raymond McLeod (1998 : 611)

Simbol	Keterangan
	<i>Entity</i> Kesatuan di lingkungan luar sistem yang berupa orang, organisasi atau system lainnya yang akan memberikan input atau
	<i>Arus</i> Arus data yang mengalir diantara proses, data store, dan entity.
	<i>Notasi Proses</i> Kegiatan yang dilakukan orang, komputer atau mesin dari hasil suatu arus data yang masuk kedalam proses untuk dihasilkan arus data yang akan keluar dari proses
	<i>Penyimpanan data atau file</i> Media penyimpanan data.



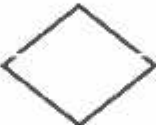
2.5.2. Entity Relationship Model (E-R Model)

Menurut Rames Elmasri dan Shamkant B. Navathe, Addison Wesley, Canada, 2000 menjelaskan bahwa E-R Model adalah suatu cara untuk mengidentifikasi tipe data *entity* dan relasi antar *entity* tersebut. Pada prinsipnya, membuat suatu E-R Model adalah mengelompokkan data yang sejenis dan membentuk relasi-relasi yang dapat menunjukkan suatu hubungan antara data dalam suatu sistem.

Berikut ini merupakan simbol-simbol yang digunakan di dalam E-R Model diantaranya dapat di tunjukkan dalam Tabel 2.3

Tabel 2.3 Simbol E-R Model

Sumber : Rames Elmasri dan Shamkant B. Navathe (2000)

Simbol	Keterangan
	Entity Menunjukkan entitas
	Attribute Adalah merupakan data elemen / data <i>item</i> , data <i>field</i> yang menggambarkan suatu <i>entity</i> . Atribut dibagi menjadi dua yaitu: 1. <i>Simple attribute</i> , misalnya kode tamu. 2. <i>Composite attribute</i> , misalnya nama tamu
	Relationship Menggambarkan hubungan antara dua atau lebih <i>entity</i>

Elemen – elemen dari ER-Model terdiri dari beberapa elemen diantaranya adalah sebagai berikut :

1. *Entity*

Adalah orang, tempat, kejadian atau konsep yang informasinya direkam atau diberikan, misalnya pada bidang kepegawaian atau perusahaan, *entity* yang terkandung adalah pegawai, absensi, gaji, cuti dan lain sebagainya.

2. *Attribute*

Merupakan data elemen, data *item*, atau data *field* yang menggambarkan dan mewakili suatu *entity*.

3. *Relationship*

Menggambarkan hubungan antara dua atau lebih *entity*. Bila data dinormalkan dan informasi dipindah dari satu tabel ke tabel yang lain, harus ada cara menghubungkan kedua tabel tersebut. Hubungan tersebut terbentuk dengan menggunakan kunci data yang bersifat unik. Dalam hubungan antar tabel dikenal ada dua kunci data

penghubung yaitu :

4. *Primary key* (kunci utama)

Yaitu suatu atribut/*field* atau satu set atribut yang mengidentifikasi secara unik suatu kejadian yang spesifik pada *entity*.

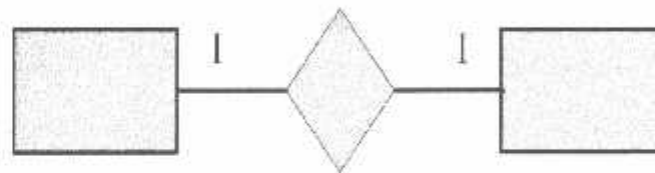
5. *Foreign key* (kunci tamu)

Yaitu suatu atribut atau set atribut yang melengkapi suatu hubungan yang menunjukkan ke *entity* induknya.

Berdasarkan hubungannya, hubungan antara *entity* dibedakan menjadi beberapa relasi diantaranya:

a. Relasi *One (1) to One (1)*

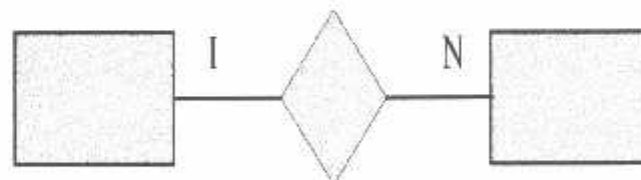
Menggambarkan hubungan satu ke satu, yaitu satu *record* pada *entity* pertama berhubungan dengan *satu record* pada *entity* kedua atau sebaliknya. Relasi *one (1) to one (1)* di tunjukkan dalam Gambar 2.3



Gambar 2.3 Relasi *One to one*

b. Relasi *One (1) to Many (M)*

Menggambarkan hubungan satu ke banyak, yaitu satu *record* pada *entity* pertama berhubungan dengan banyak *record* pada *entity* kedua atau sebaliknya. Relasi *one (1) to Many (M)* di tunjukkan dalam Gambar 2.4

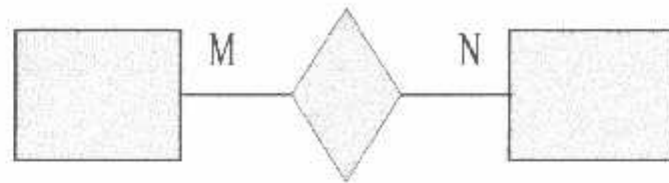


Gambar 2.4 Relasi *One to Many*

c. Relasi *Many (M) to Many (M)*

Menggambarkan hubungan banyak ke banyak, yaitu lebih dari satu *record* pada *entity* pertama berhubungan dengan lebih dari satu *record* pada *entity* kedua atau

sebaliknya. Dibutuhkan *associative* atau *entity perantara* atau tabel baru yang berisi *atribut / field* kunci dari masing-masing *entity*. Relasi *Many (M) to Many (N)* di tunjukkan dalam Gambar 2.5



Gambar 2.5 Relasi *Many to Many*

2.6. SMS Gateway

Salah satu kendala yang dihadapi dalam layanan SMS yakni perbedaan protokol yang digunakan oleh masing – masing SMS Center (SMSC). Kita tidak dapat terhubung ke SMSC lain, jika tidak menggunakan protokol yang sama. Untuk mengatasi hal tersebut, diperlukan sebuah SMS *gateway* yang bertindak sebagai *relay* atau *switch* antar SMSC.

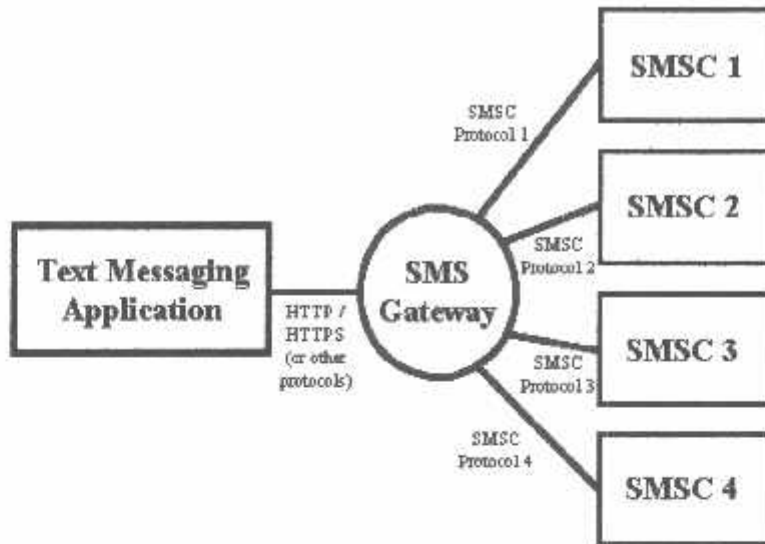
Berikut ini ilustrasi dari SMS *gateway* sebagai penghubung antar 2 SMS Center (SMSC) yang berbeda protokol, di tunjukkan dalam Gambar 2.6



Gambar 2.6 SMS *gateway* sebagai penghubung antar SMSC

SMS *gateway* dapat juga digunakan untuk menghubungkan antara sebuah aplikasi SMS dengan SMS Center (SMSC) tiap operator seluler melalui protokol HTTP/HTTPS.

Adapun hubungan yang dimaksud adalah kemampuan untuk mengirim dan menerima SMS. Berikut ilustrasinya di tunjukkan dalam Gambar 2.7



Gambar 2.1 SMS gateway sebagai penghubung antara aplikasi SMS dengan SMSC melalui protokol HTTP/HTTPS

Mekanisme inilah yang nantinya akan digunakan dalam pengembangan sistem monitoring jaringan berbasis SMS gateway.

2.7. Gammu

Gammu merupakan salah satu *tools* untuk mengembangkan aplikasi SMS Gateway yang cukup mudah diimplementasikan dan bebas biaya lisensi (*free*). Kelebihan Gammu dari tools SMS Gateway lainnya adalah :

1. Gammu bisa dijalankan di Windows maupun Linux
2. Gammu menggunakan *database MySQL*, bisa menggunakan *interface* berbasis web
3. Baik kabel data USB maupun SERIAL semua kompatibel di Gammu
4. Dapat membaca, menghapus, mengirim dan menerima SMS

Ada 2 mekanisme kerja Gammu yaitu sebagai aplikasi dan sebagai *daemon*. Gammu sebagai aplikasi akan bekerja ketika perintah Gammu dijalankan pada lingkungan *shell* beserta perintahnya disertakan sesuai fungsi yang diinginkan. Sedangkan sebagai *daemon*, Gammu ditandai dengan dijalankannya perintah *smgd* pada

shell. SMSD bukanlah perintah yang langsung terinstall melainkan perintah yang dijalankan pada *shell* atau MS-DOS *Prompt*. Gammu SMSD adalah program yang secara berkala scan modem GSM untuk pesan yang di terima, menyimpannya dalam database, serta mengirim pesan yang ada dalam database. program ini sangat cocok untuk mengelola sejumlah besar pesan yang di terima dan yang dikirim.

Pada dasarnya Gammu mampu didukung oleh hampir sebagian besar *handphone* dan *modem* yang ada di pasaran, namun ada beberapa spesifikasi *handphone* dan *modem* yang tidak dapat didukung sepenuhnya oleh Gammu, hal ini karena pada Gammu tidak mendukung semua jenis koneksi yang terdapat pada *handphone* dan *modem* tersebut.

Merek dari *Handphone* – *handphone* dan *modem* yang telah kompatibel dengan Gammu terdapat dalam Tabel 2.4

Tabel 2.4 Merek *handphone* dan *modem* yang kompatibel dengan Gammu

Merek hp dan USB modem	jumlah	Merek hp dan USB modem	Jumlah	Merek hp dan USB modem	jumlah
Alcatel	23	Apel	3	BenQ-Siemes	11
BlackBerry	3	CECT	7	Daxian	2
Elson	1	Emgeton	1	Falcon	3
Fly	4	Foston	1	Gionee	1
HTC	22	Huawei	46	Hughes	1
Jinpeng	1	Levono	4	LG	84
Mitsunichi	1	Mitsubisi	2	Motorola	123
Nokia	352	Onda	1	Openmoko	1
Option	15	PalmOne	2	Philips	8
Sagem	40	Samsung	209	Sandshine	1
Sanyo	2	SciPhone	9	Sharp	8
Siemens	57	Sierra Wireless	9	SimCom	2
Sonim	3	Sony Ericsson	256	Symphony	1
Teltonika	1	Toshiba	2	Vodafone	6
WaveCom	7	ZTE	19		

Agar suatu perangkat dalam jaringan dapat berkomunikasi dengan perangkat lainnya maka perlu di lakukan koneksi antara perangkat - perangkat tersebut.

Salah satu hal yang perlu di perhatikan dalam melakukan koneksi antara perangkat yaitu tipe koneksi dari perangkat - perangkat tersebut.

Gammu mendukung koneksi *serial* maupun USB. Dengan tipe – tipe koneksi sebagai berikut:

- at
- blucat
- dkuat

Pada program aplikasi yang dibuat, tipe koneksi yang di gunakan sebagai penghubung antara PC aplikasi dengan *Modem* GSM yaitu tipe koneksi *at19200*. Tipe koneksi *at* adalah sistem koneksi yang menggunakan *Modem* sebagai media penghubung antara dua perangkat jaringan, dengan kecepatan pengiriman data mencapai *19200bps (bit per second)*

2.8. Jaringan Komputer

Pada era globalisasi seperti sekarang ini, jutaan bahkan milyaran manusia saling berkomunikasi , dan kita tidak bisa lagi hanya bergantung pada komunikasi verbal untuk transfer informasi. Saat ini kita sudah menggunakan jaringan komputer untuk mengambil dan berbagi (*share*) informasi dengan cepat dan tepat. Dengan jaringan komputer inilah dapat ditingkatkan efisiensi dan efektifitas interaksi antar manusia dalam hal pemindahan informasi.

Jaringan komputer dapat didefinisikan sebagai hubungan antara dua atau lebih komputer beserta perangkat lainnya melalui media transmisi untuk melakukan komunikasi data satu dengan yang lain.

Adapun komunikasi data dapat diartikan sebagai pengiriman data secara elektronik dari satu tempat ke tempat lainnya melalui suatu media komunikasi, dan data yang dikirimkan tersebut merupakan hasil yang diproses oleh suatu sistem komputer.

4.9. LAN (*Local Area Network*)

Local Area Network (LAN) adalah sejumlah komputer yang saling dihubungkan bersama di dalam satu areal tertentu yang tidak begitu luas, seperti di dalam satu kantor atau gedung. Secara garis besar terdapat dua tipe jaringan atau LAN, yaitu jaringan *Peer-to-Peer* dan jaringan *client-server*. Pada jaringan *peer-to-peer*, setiap komputer yang terhubung ke jaringan dapat bertindak baik sebagai *workstation* maupun *server*. Sedangkan pada jaringan *client-server*, hanya satu komputer yang bertugas sebagai *server* dan komputer lain berperan sebagai *workstation*.

LAN mempunyai karakteristik sebagai berikut :

1. Mempunyai *transfer rate* data yang lebih tinggi
2. Meliputi wilayah geografi yang lebih sempit
3. Tidak membutuhkan jalur telekomunikasi yang disewa dari *operator* telekomunikasi.

BAB III PERANCANGAN SISTEM

3.1. Gambaran umum Sistem monitoring

Sistem monitoring LAN berbasis SMS *gateway* merupakan suatu sistem yang secara otomatis akan memonitor kondisi suatu jaringan LAN dan mencatat serta menyampaikan kondisi dari jaringan LAN saat itu, apakah terjadi masalah pada jaringan tersebut atau tidak. Dengan memanfaatkan SMS *gateway*, proses penyampaian informasi menjadi lebih cepat, karena dikirim melalui SMS ke administrator Jaringan di manapun dia berada sehingga administrator ataupun teknisi jaringan dapat dengan cepat dapat mengambil langkah - langkah yang perlu untuk mengatasi masalah yang terjadi pada jaringan tersebut.

3.2. Analisa Kebutuhan Sistem.

Analisa sistem ditujukan untuk menyediakan rancangan dengan pemahaman yang lebih menyeluruh terhadap kebutuhan – kebutuhan yang menjadi dasar perlunya dilakukan pengembangan Sistem monitoring LAN berbasis SMS *gateway*.

Analisa sistem monitoring berbasis SMS *gateway*, mempunyai beberapa prosedur yang mendasar, dalam melakukan pendekatan analisa sistem agar memperoleh pemecahan terhadap suatu masalah.

Pada sistem lama, seorang administrator jaringan harus berinteraksi secara langsung dengan salah satu *server* untuk melakukan monitoring ke masing – masing subnet pada jaringan yang dikelolanya. Dengan metode ini, administrator jaringan harus *standby* atau paling tidak selalu mengawasi *server* atau *router* dari jarak dekat.

Pada sistem baru yang dikembangkan, seorang administrator jaringan tidak perlu berinteraksi secara langsung dengan perangkat – perangkat jaringan seperti *router* atau *server* untuk memonitor jaringannya. Pada sistem ini sudah tersedia aplikasi yang secara otomatis melakukan monitoring ke jaringan dan administrator jaringan akan

menerima informasi tentang keadaan jaringan saat itu melalui SMS. Tentunya dengan sistem ini, seorang administrator jaringan tidak perlu *standby* atau berada di depan *server* atau *router* selama 24 jam. Administrator jaringan hanya berinteraksi dengan komputer *server* atau *router* jika terjadi masalah pada jaringan.

3.3 Perancangan fungsi sistem monitoring LAN

Sistem yang dikembangkan ini bersifat *internal* Artinya, pengguna program ini hanya kalangan tertentu yang memiliki hak akses ke dalam aplikasi program.

Program ini bekerja dalam sebuah *web server* dan dapat diakses melalui jaringan *intranet*. Oleh karena program ini berjalan dalam sebuah jaringan *intranet* maka dapat dijalankan secara *multiuser*, yaitu dapat diakses oleh beberapa *user/pengguna* secara bersamaan dari komputer yang berbeda-beda. *User* atau Pengguna yang dapat mengakses aplikasi program ini di antaranya adalah :

- Administrator

Administrator adalah seorang *user* yang memiliki hak paling tinggi, yaitu dapat mengubah seluruh data apabila ada kesalahan dalam pengisian data oleh dirinya sendiri, ataupun teknisi yang berhak melakukan pengisian data. Administrator juga dapat melihat laporan kondisi LAN yang dikelolanya. Selain itu, ia juga mempunyai hak untuk mengubah data dari *user/pengguna* yang lain serta memberikan hak akses kepada *user/pengguna* lain untuk mengakses aplikasi program.

- Teknisi

Teknisi adalah seorang *user* yang memiliki hak tertentu, tergantung akses yang diberikan oleh administrator. Seorang teknisi dapat saja memiliki semua akses, kecuali manajemen *user* atau pengaturan pengguna.

3.3.1. Perintah yang digunakan untuk memonitor

Perintah – perintah yang digunakan oleh administrator jaringan atau teknisi jaringan dalam memperoleh informasi atau mendeteksi kesalahan pada jaringan sebenarnya sudah di sediakan oleh tiap – tiap sistem operasi dimana aplikasi tersebut di jalankan. Perintah dasar yang digunakan tersebut adalah perintah *ping(packet internet groper)*. Dalam implementasi kali ini, kami menggunakan perintah *ping* tersebut untuk memonitor koneksi pada jaringan LAN yang di bangun.

PING singkatan dari *Packet Internet Gropher*, secara pengertian PING adalah sebuah program utilitas yang digunakan untuk memeriksa konektivitas jaringan berbasis teknologi *Transmission Control Protocol/Internet Protocol (TCP/IP)*. Dengan menggunakan *utilitas* ini, dapat diuji apakah sebuah komputer terhubung dengan komputer lainnya. Hal ini dilakukan dengan cara mengirim sebuah paket kepada alamat IP yang hendak diujicoba konektifitasnya dan menunggu *respons*/balasan darinya. Nama “PING” berasal dari sonar sebuah kapal selam yang sedang aktif, yang sering mengeluarkan bunyi PING ketika menemukan sebuah objek. Cara kerja ping yaitu dengan mengirimkan paket PING atau *ICMP(internet Control Message Protocol)* ke *host* atau komputer tujuan apabila mendapat *reply* atau balasan maka koneksi pada jaringan tersebut dikatakan bekerja atau terhubung, dan apabila tidak mendapat *reply* atau balasan yang berupa desitination *unreachable* atau *request time out* maka koneksi jaringan tersebut rusak atau tak terhubung. Aturan penulisan perintahnya adalah sebagai berikut:

- Ping (spasi) nomor IP yang mau di ping

Contoh : ping 192.168.1.1

Atau nama dari *DNS(Domain name sistem)*

Contoh: ping *www/google.com*

3.3.2. SMS Gateway

Merupakan bagian yang menghubungkan antara program aplikasi yang akan dibuat dengan jaringan *GSM (Global system for mobile communication)* atau *CDMA (Code Division Multiplexing Access)* sesuai dengan operator telekomunikasi yang kita gunakan. Bagian ini berfungsi untuk membaca SMS dari *MS(mobile stasion)* dan mengirim SMS kepengguna.

Seperti arti katanya, *gateway* berarti Pintu gerbang, sehingga dalam istilah ini *SMS gateway* berarti pintu gerbang atau jembatan antara dua buah perangkat atau lebih. Umumnya *SMS gateway* ini berupa sebuah komputer yang didalamnya telah terinstall aplikasi untuk menangani pengiriman SMS antar HP. Dalam hal ini, *SMS gateway* berfungsi sebagai aspek pusat yang menangani pengiriman surat atau informasi sesuai dengan alamat yang dituju.

3.3.3. Protokol yang digunakan dalam Sistem monitoring LAN

Notifikasi dikirim dan diterima melalui jaringan *wireless* atau tanpa kabel, sudah tentu sebuah jaringan mempunyai aturan atau protokol yang digunakan sebagai penunjangnya. Aturan atau protokol yang sering digunakan dalam aplikasi Sistem monitoring adalah sebagai berikut :

- *HTTP (Hypertext Transfer Protocol)*

HTTP merupakan protokol yang paling digunakan dalam internet saat ini. Tujuan utama *HTTP* pada mulanya adalah untuk menyediakan cara dalam menyajikan dan mengambil dari *HTML*. Saat ini *HTTP* tidak hanya terbatas dalam penyajian halaman *HTML* saja.

- *SMPP (Short Message peer to peer protocol)*

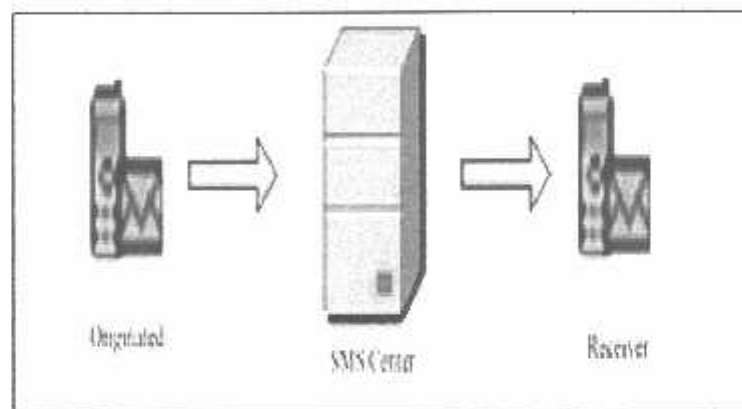
SMPP merupakan sebuah protokol standar industri yang digunakan dalam pertukaran short message antara *external short messaging entity (ESME)*, *routing Entity (RE)*, dan *Message Center (MC)*.

- SMS (*Short Message Services*)

SMS merupakan sebuah layanan pengiriman informasi antara perangkat yang berbasis *text*, layanan SMS biasanya di sediakan oleh *operator* telekomunikasi.

3.3.4. Prinsip kerja SMS

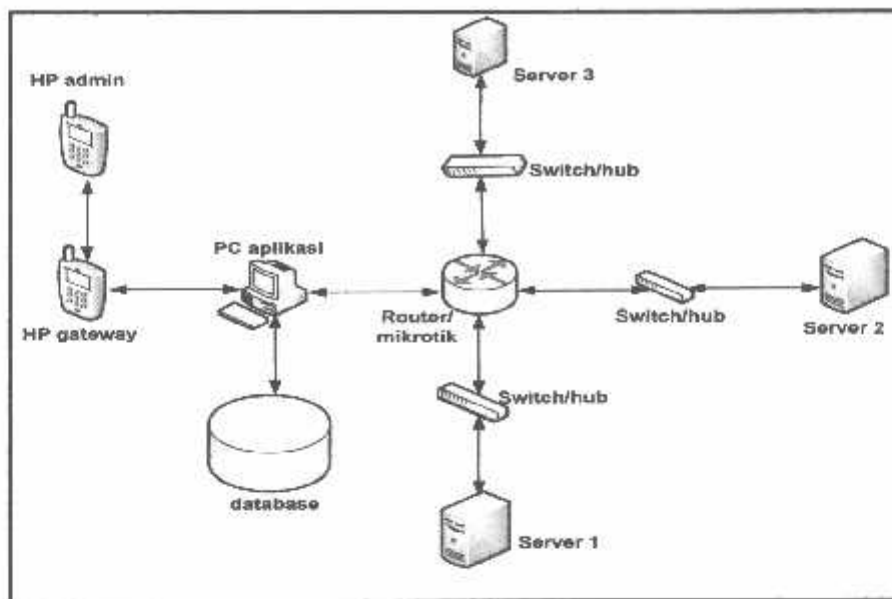
Prinsip kerja SMS adalah bahwa setiap jaringan mempunyai satu atau lebih *service center* (SC) yang berfungsi menyimpan dan meneruskan pesan dari pengirim ke tujuan seperti yang ditunjukkan dalam Gambar 3.1



Gambar 3.1 Prinsip kerja SMS

3.4 Desain Sistem

Sebuah proses untuk mendokumentasikan struktur dan aliran data melalui proses sistem, *logika*, kebijakan, dan prosedur yang akan di implementasikan oleh proses sistem. Dalam Gambar 3.2 di tunjukkan desain sistem dari Sistem monitoring LAN berbasis SMS *gateway*



Gambar 3.2 Desain Sistem monitoring LAN

Dalam sebuah jaringan LAN yang terdiri dari beberapa *subnet*, maka untuk menghubungkan tiap - tiap *subnet* tersebut dalam jaringan maka di perlukan sebuah *Router* atau *Switch* yang *manageable* untuk dapat menghubungkan *subnet – subnet* tersebut, pada Gambar 3.2 diatas diilustrasikan sebuah jaringan LAN yang terdiri dari 4 *subnet* yang masing – masing *subnet* tersebut terhubung ke sebuah *router* yang berfungsi sebagai jalur penghubung antar tiap *subnet*, *router* yang digunakan adalah *router mikrotik RB750*

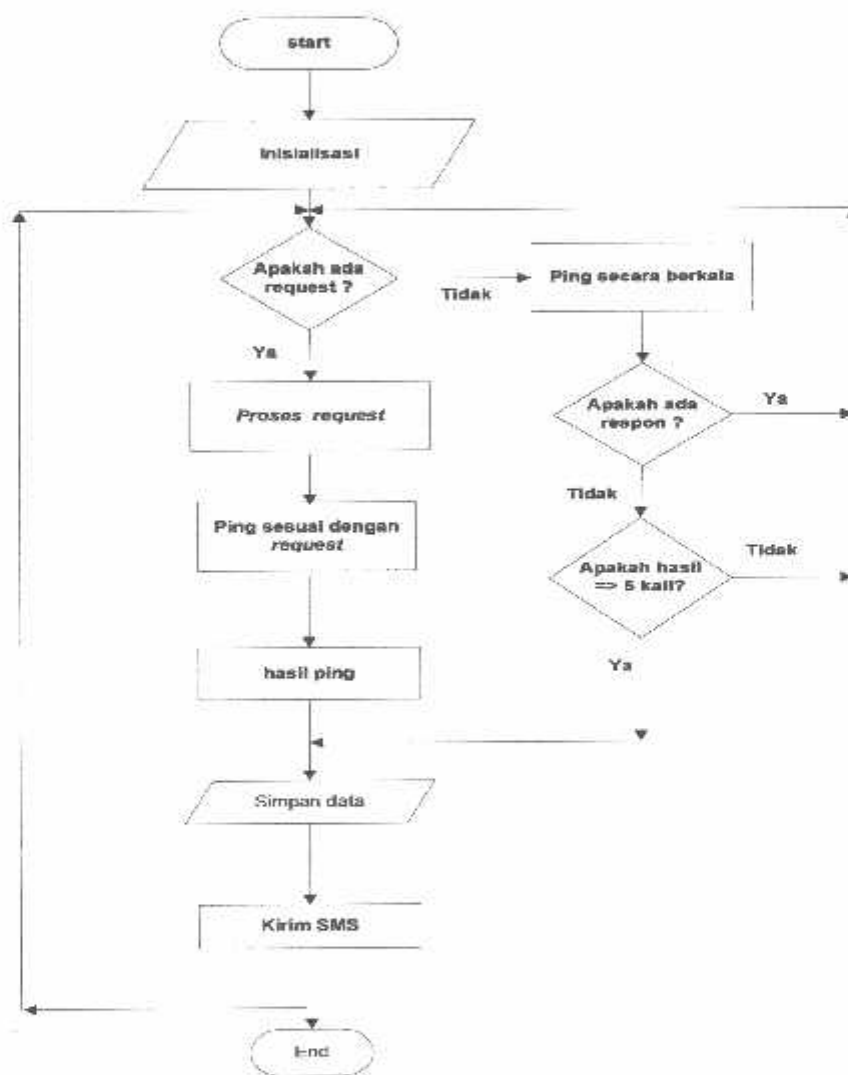
Aplikasi monitoring terinstall dalam satu *subnet* sendiri dan digunakan untuk memantau IP dari tiap *server* untuk memastikan *server – server* dalam jaringan tersebut bekerja secara optimal, Aplikasi tersebut akan melakukan *ping* ke IP dari tiap *server* untuk memantau koneksi dari *server – server* tersebut, bila salah satu dari *server* mengalami gangguan koneksi maka, aplikasi akan mengirim SMS melalui *handphone* atau USB *modem* yang terhubung dengan PC aplikasi tersebut.SMS dikirim ke nomor *handphone* dari teknisi atau administrator yang menangani jaringan agar dapat mengambil langkah – langkah yang perlu untuk cepat mengatasi masalah yang terjadi. Aplikasi juga menyimpan data – data monitoring ke sebuah *database* supaya data – data

tersebut bisa dipergunakan sebagai referensi bagi Administrator untuk menganalisa kinerja daripada jaringan yang di kelolanya.

3.4.1 Flowchart

3.4.1.1 Flowchart Sistem monitoring

Sesuai dengan desain Sistem yang telah dibuat, maka *flowchart* (diagram alir) dari sistem monitoring LAN berbasis SMS *gateway* ditunjukkan dalam Gambar 3.3 berikut ini:



Gambar 3.3 Flowchart Sistem monitoring

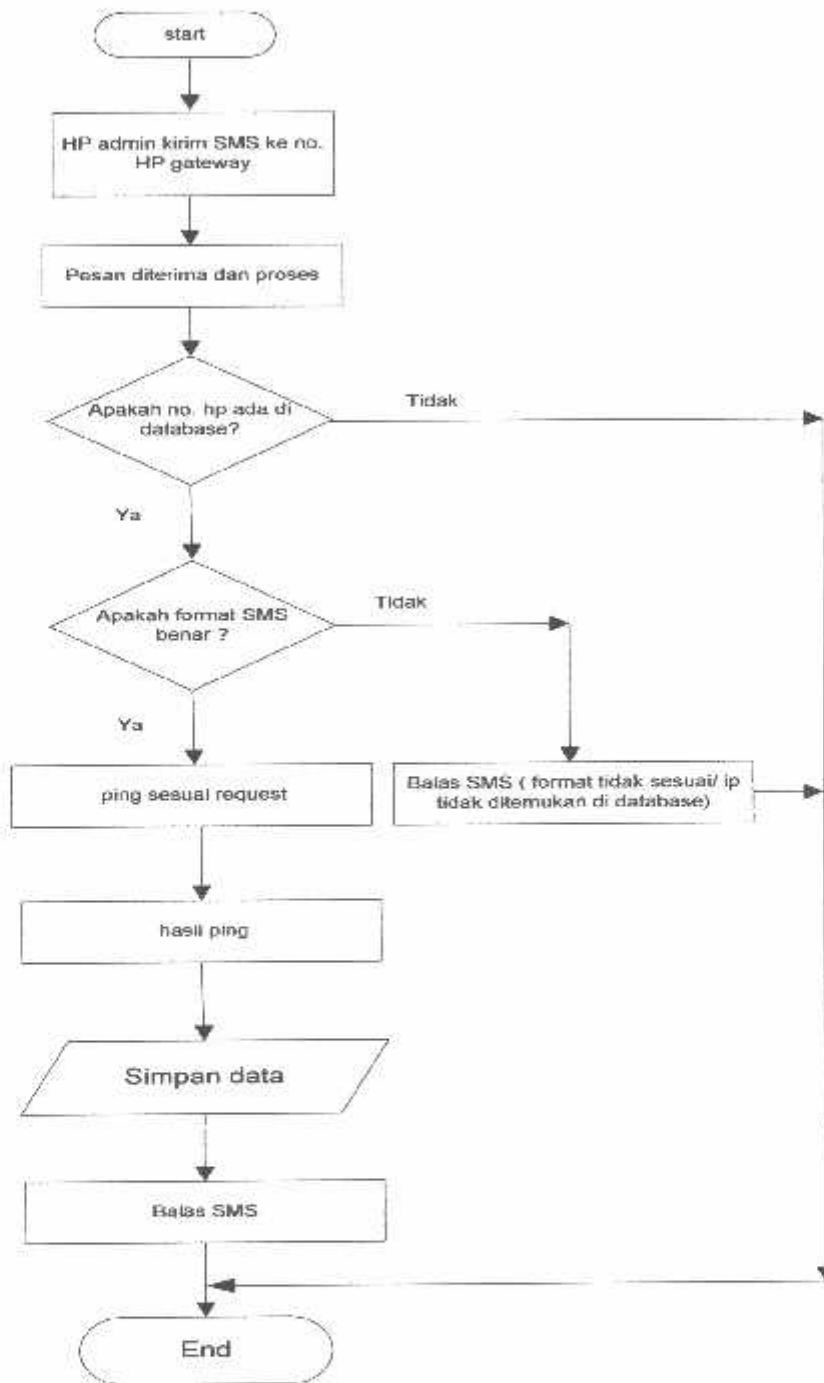
Proses kerja dari aplikasi sistem monitoring LAN berbasis SMS *gateway* secara umum dalam flowchart pada Gambar 3.3 diatas, Saat aplikasi *startup* maka program melakukan inialisasi antar perangkat yaitu inialisasi HP dengan PC aplikasi, lalu program mulai melakukan eksekusi, disini program akan melakukan pengecekan ke kotak *inbox* pada *database* untuk melihat apakah ada *request* atau permintaan secara manual dari admin jaringan atau tidak, bila ada maka program akan memproses permintaan tersebut lalu mengeksekusinya sesuai dengan permintaan admin. Eksekusi yang dilakukan dengan cara melakukan proses ping ke IP subnet sesuai permintaan, lalu akan menyimpan hasil ping ke dalam *database* dan kemudian hasil eksekusi tersebut akan di kirim ke Admin.

Bila pada saat program melakukan pengecekan di *inbox database* tidak ada *request*/permintaan maka program akan melakukan proses ping secara otomatis ke IP *gateway* tiap subnet dan IP *server* secara berkala sesuai dengan waktu yang ditentukan, bila terjadi error koneksi pada salah satu IP *gateway* atau IP *server* maka program akan melakukan proses ping sesuai dengan *setting*/pengaturan bila masih tidak ada *respon*/balasan maka program akan menyimpan hasil ping ke dalam *database* lalu hasil dari ping tersebut akan di kirim ke nomor *handphone* teknisi atau admin jaringan untuk memperingatkan admin jaringan bahwa salah satu koneksi *subnet* atau *server* pada jaringan yang di kelolahnya sedang terjadi masalah atau *error*.

Bila tidak ada kesalahan maka program akan terus melakukan ping secara berkala sesuai dengan pengaturan yang ada, selama program tersebut dijalankan.

3.4.1.2 Flowchart SMS request

Proses pengiriman SMS dari admin atau teknisi ke aplikasi untuk merequest keadaan jaringan di tunjukkan dalam flochart Gambar 3.4



Gambar 3.4 Flowchart proses request SMS

Penjelasan dari *flowchart* SMS request diatas, HP teknisi atau admin melakukan request/permintaan *ping* ke aplikasi, SMS tersebut akan diterima oleh HP/modem yang terkoneksi ke PC aplikasi, HP atau *modem* tersebut bertindak sebagai *gateway* penghubung antara aplikasi dengan HP admin atau teknisi.

Aplikasi akan melakukan pengecekan nomor HP dari pengirim SMS, bila nomor tersebut terdaftar dalam *database* maka aplikasi akan melakukan pengecekan berikutnya yaitu format dari SMS yang diterima, bila format sesuai dengan format pengaturan pada aplikasi, maka program akan melakukan proses *ping* sesuai dengan permintaan pengirim lalu hasil dari *ping* akan di simpan dalam *database* dan selanjutnya hasil *ping* juga di kirim kembali ke nomor HP pengirim, bila format SMS yang diterima tidak sesuai dengan format yang telah di atur maka aplikasi akan mengirim SMS balasan untuk mengingatkan pengirim kalau format dari SMS yang di terima aplikasi, tidak sesuai atau format tidak di temukan di *database*.

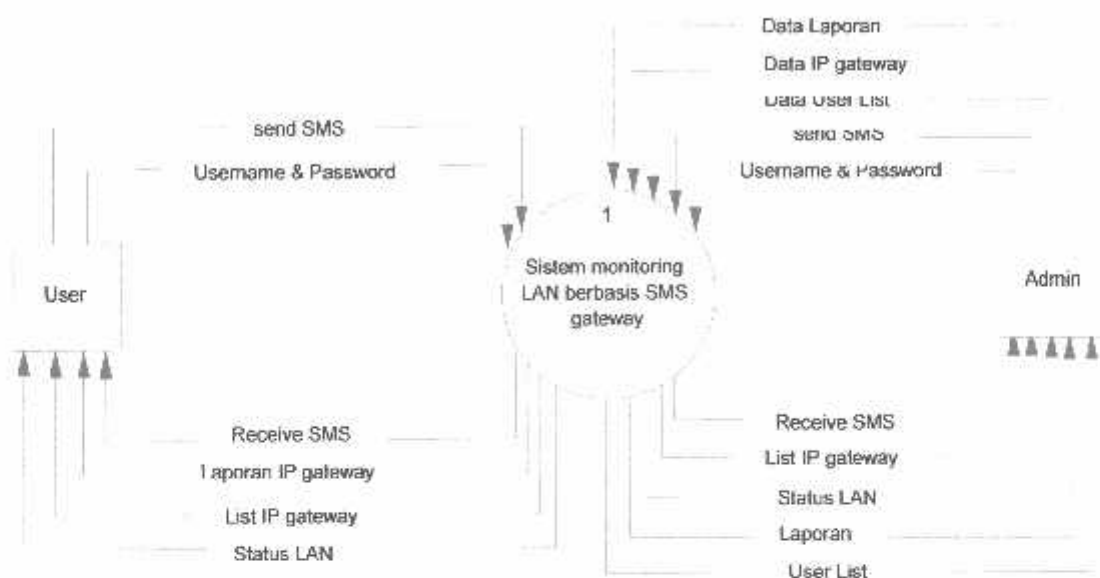
Bila nomor HP pengirim SMS tidak ada dalam *database* maka aplikasi tidak akan memproses SMS tersebut, dan aplikasi akan langsung menghapus SMS dari *inbox database*.

3.4.2 DFD (*Data Flow Diagram*)

Sebelum mengimplementasikan program, maka dilakukan pembuatan DFD atau *Data Flow Diagram*. DFD adalah suatu grafik yang menjelaskan sebuah sistem dengan menggunakan bentuk-bentuk atau simbol untuk menggambarkan aliran data dari proses-proses yang saling berhubungan. DFD menggambarkan input, proses, dan output yang terjadi dalam suatu sistem. DFD juga menggambarkan aliran data dalam sebuah sistem. Dalam mendokumentasikan sebuah sistem, DFD mempunyai level-level mulai dari yang terkecil ,yaitu level 0 atau sering disebut *context diagram*. *Context diagram* ini merupakan gambaran paling umum dari sistem, yang hanya memiliki satu proses saja untuk mewakili seluruh sistem. Semakin bertambahnya *level* dalam DFD maka akan semakin detail digambarkan proses-proses yang ada pada sistem, tetapi yang boleh bertambah hanya proses dan *data flow* saja. Sedangkan untuk *data source*, jumlahnya

harus tetap dengan yang ada pada *context diagram*. Khusus untuk *data store*, pada *context diagram* masih belum digambarkan, akan tampak pada level 1 dan konsisten jumlahnya sampai pada level berikutnya.

3.4.2.1 Diagram Konteks Sistem Monitoring LAN



Gambar 3.5 Diagram konteks sistem monitoring

Dalam sistem monitoring LAN berbasis SMS *gateway* ini, terdapat dua tipe pengguna dengan hak masing – masing. Administrator dapat mengakses dan melakukan manipulasi terhadap seluruh data yang ada di dalam sistem. Data – data yang dimaksud antara lain :

1. Data *user*

Data – data yang berhubungan dengan *user* atau pengguna dalam mengakses dan mengedit informasi dalam aplikasi.

2. Data *gateway*

Data – data yang berhubungan dengan pengaturan IP – IP yang akan di pantau atau di monitor oleh aplikasi monitoring

3. Data status *gateway*

Data – data hasil *ping* yang telah di simpan dalam *database* dan di tampilkan ke halaman status *gateway*, data ini hanya menampilkan hasil *ping* yang telah tersimpan di *database* dan tidak melakukan *ping* secara *real time*.

4. Data konfigurasi SMS

Data – data yang berhubungan dengan nomor – nomor *IP* yang bisa menerima dan mengirim SMS dari dan ke aplikasi monitoring.

5. Data setingan umum

Data – data yang berhubungan dengan pengaturan lamanya aplikasi melakukan *ping* ke tiap - tiap *IP*, dan *delay*/tenggang waktu proses *ping* antara satu *IP* dengan *IP* berikutnya.

6. Data laporan

Data – data yang berhubungan dengan laporan mengenai hasil *ping*. Sedangkan untuk tipe pengguna Teknisi, hanya dapat mengakses data – data yang terbatas sesuai dengan yang telah ditetapkan oleh Administrator. Data – data tersebut diantaranya adalah:

a. Data status Gateway

Data laporan mengenai status dari *IP – IP* yang di pantau oleh aplikasi, biasanya data - data tersebut digunakan sebagai bahan referensi oleh administrator jaringan untuk menilai kinerja dari jaringan yang di tangannya dan juga untuk lebih meningkatkan performa dari jaringan.

b. Data status *user*

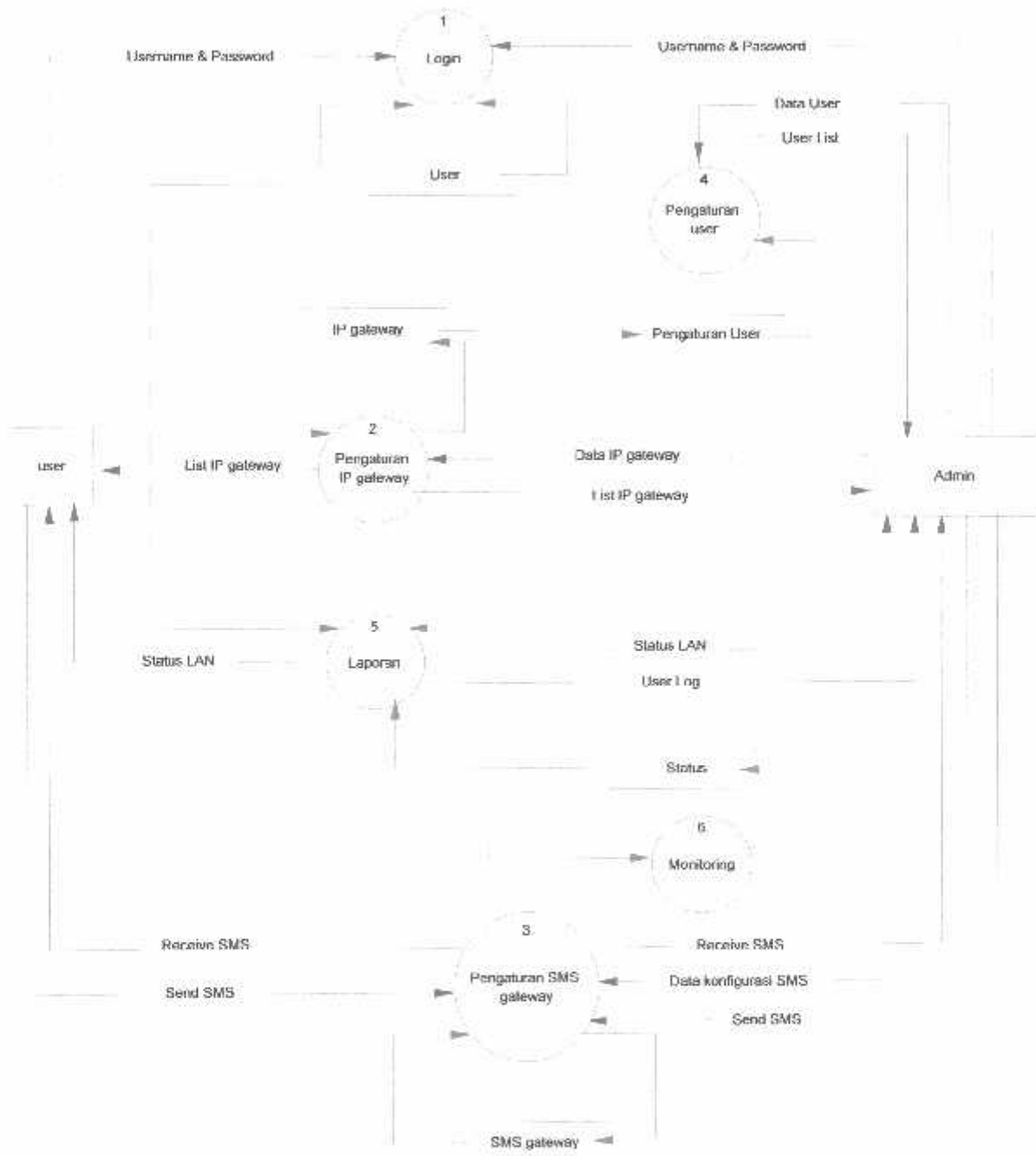
Data laporan mengenai *user*/pengguna yang mengakses aplikasi, baik itu sebatas memonitor maupun merubah *settingan – settingan* pada aplikasi, data *user* sangat penting karena bisa di gunakan untuk mengetahui *user – user* yang telah mengakses aplikasi, bila terjadi kesalahan monitoring pada aplikasi.

c. Data *account*

Data – data yang berhubungan dengan pengaturan *password*, pada data *account* ini, *user*/pengguna bisa mengganti akses *password* untuk *login* dengan *password* yang baru.

3.4.2.2. DFD Level 1 Sistem Monitoring

Pengembangan dari diagram konteks /DFD level 0 ke DFD level 1 di mana pada DFD level 1 akan di jelaskan secara rinci mengenai proses – proses yang berjalan pada sistem yang sedang di kembangkan. DFD level 1 di tunjukkan dalam Gambar 3.6



Gambar 3.6 DFD level 1 Sistem monitoring LAN

Dari context diagram (level 0), dapat dirumuskan beberapa proses inti pada sistem monitoring LAN yang disusun menjadi DFD Level 1 sesuai dalam Gambar 3.6 Berikut merupakan dari penjelasan masing – masing proses yang berjalan pada DFD level 1 adalah:

1. Proses *login* (1.1)

Proses *login* merupakan suatu upaya *user* untuk masuk ke dalam aplikasi administrasi sistem monitoring LAN. *User* yang terlibat di dalam proses ini adalah *user* administrator, dan *user* lainnya (teknisi). Adapun data – data yang diperlukan adalah *username* serta *password* yang valid. Autentifikasi melibatkan data dari tabel *user*.

2. Proses pengaturan IP *gateway* (1.2)

Proses ini merupakan proses menambah, mengurangi, serta memperbaharui daftar IP *gateway/ Server* yang akan dimonitoring oleh sistem. Proses ini dapat diakses oleh semua jenis *user*, dengan hak untuk menambah, mengurangi, serta memperbaharui. Tabel yang digunakan dalam proses ini adalah tabel data *gateway*, sedangkan datanya sendiri adalah detail *gateway* (IP, serta *Alias*).

3. Proses pengaturan SMS *gateway* (1.3)

Proses pengaturan sms *gateway* mencakup pengaturan nomor *handphone* yang akan menerima notifikasi jika terjadi masalah pada salah satu *gateway/ server*. Selain itu juga, termasuk pengaturan *setting* umum dari aplikasi, misalnya *delay* untuk melakukan *ping*, dan lain sebagainya. Semuanya berhubungan dengan data *handphone*, serta data *settingan* umum sistem.

4. Proses pengaturan *user* (1.4)

Pengaturan *user* tentu saja merupakan proses untuk menambah, mengurangi, serta memperbaharui daftar dari *user* yang berhak untuk mengakses aplikasi

administrasi monitoring LAN. Melibatkan data *user*, serta tabel *user* selama melaksanakan proses ini.

5. Proses pelaporan (1.5)

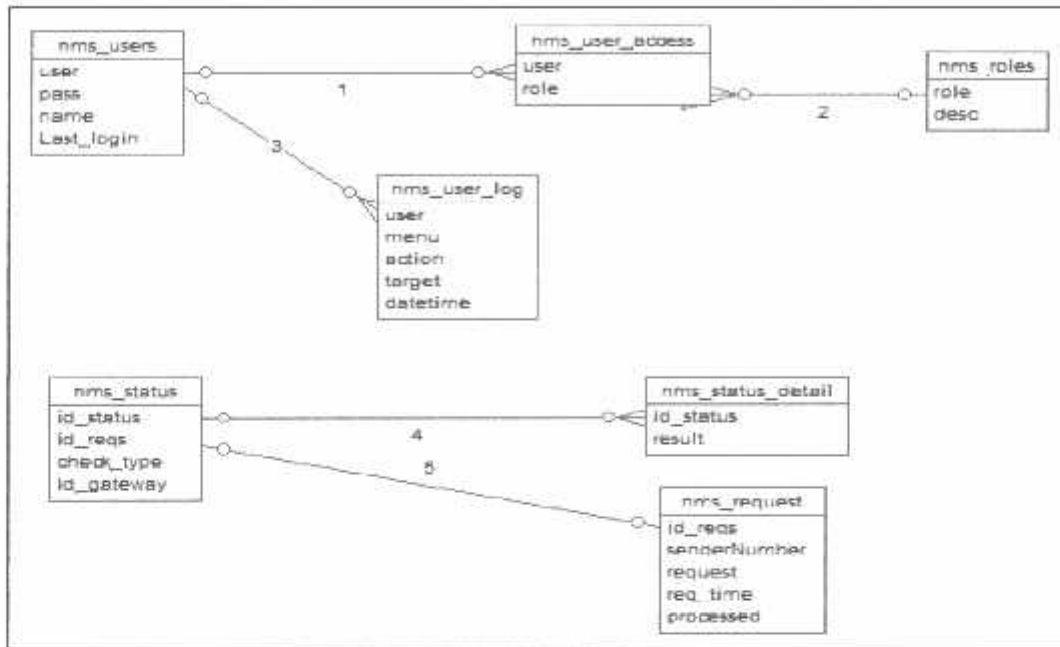
Pelaporan yakni suatu proses untuk melihat kondisi sistem secara berkala. Periode yang digunakan yakni per 1 bulan. Ada 2 jenis pelaporan, yakni laporan tentang kondisi *down* dari IP *gateway/server*, serta laporan aktifitas *user* yang telah *login* ke dalam sistem administrasi. Melibatkan data *user* (untuk autentifikasi) , data status *gateway*, serta data aktifitas *user* selama *login*.

6. Proses monitoring (1.6)

Proses monitoring merupakan proses inti, yakni dimana aplikasi melakukan monitoring secara otomatis, dan juga secara manual. Otomatis dijalankan oleh system setiap kali komputer aplikasi aktif, sedangkan monitoring manual dilakukan hanya jika terdapat request dari *user*. Data yang dilibatkan dalam proses ini adalah data *gateway* (ID atau IP), serta data hasil *ping (result)*. Semua data akan terhubung ke tabel status *gateway* dan tabel data *gateway*.

3.4.2.3. ERD sistem monitoring

ERD menggambarkan bagaimana hubungan antara tiap tabel dalam *database* yang di gunakan dalam membangun program aplikasi monitoring. ERD di tunjukkan dalam Gambar 3.7



Gambar 3.7 ERD sistem monitoring

3.5. Perancangan Database

Model sistem memainkan peranan penting dalam pengembangan sistem yang akan diimplementasikan sebagai sebuah database. Dalam perancangan sistem ini, ada beberapa tabel yang digunakan adalah sebagai berikut :

Tabel 3.1 Tabel nms_useraccess

No.	Nama Field	Tipe	Keterangan
1	User	varchar(15)	
2	Role	varchar(15)	

Tabel 3.2 Tabel nms_users

No.	Nama Field	Tipe	Keterangan
1	User	varchar(30)	primary_key
2	Pass	varchar(30)	
3	Name	varchar(64)	
4	last_login	datetime()	

Tabel 3.3 Tabel nms_roles

No.	Nama Field	Tipe	Keterangan
1	Role	varchar(15)	
2	Desc	varchar(15)	

Tabel 3.4 Tabel nms_phonenumber

No.	Nama Field	Tipe	Keterangan
1	Id_phone	Int(3)	Primary_key, auto_increment
2	Number	varchar(15)	
3	Active	tinyint(1)	

Tabel 3.5 Tabel nms_gateway

No.	Nama Field	Tipe	Keterangan
1	id_gateway	Int(5)	primary_key, auto_increment
2	ip_gateway	varchar(15)	primary_key
3	Alias	varchar(32)	
4	Desc	varchar(128)	

Tabel 3.6 Tabel status (nms_status)

No.	Nama Field	Tipe	Keterangan
1	id_status	Int(5)	Primary_key, auto_increment
2	id_reqs	Int(5)	
3	check_type	Tinyint(1)	
4	id_gateway	varchar(15)	
5	Status	tinyint(2)	
6	last_check	datetime()	

Tabel 3.7 Tabel detail status (nms_status_detail)

No.	Nama Field	Tipe	Keterangan
1	id_status	int(5)	primary_key
2	Result	text	

Tabel 3.8 Tabel *log user* (nms_userlogs)

No.	Nama Field	Tipe	Keterangan
1	User	varchar(15)	
2	Menu	varchar(15)	
3	Action	varchar(128)	
4	Target	Text	
5	Datetime	datetime	

Tabel 3.9 Tabel *setting* (nms_settings)

No.	Nama Field	Tipe	Keterangan
1	autoPingDelay	Int(5)	
2	delayPingEachGateway	Int(5)	
3	sendSMSonStart	Tinyint(1)	
3	maxDowntime	int(6)	

Tabel 3.10 Tabel *request* (nms_request)

No.	Nama Field	Tipe	Keterangan
1	id_reqs	Int(5)	Primary_key, auto_increment
2	senderNumber	varchar(17)	
3	Request	varchar(160)	
4	Reqtime	Datetime	
5	Processed	tinyint(1)	

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Implementasi dilakukan dengan menerapkan hasil desain yang telah dibuat ke dalam bahasa pemrograman (*coding*) PHP, sehingga prosedur – prosedur yang telah dibuat dapat dimengerti oleh mesin dan menghasilkan keluaran seperti apa yang diharapkan. Berikut ini adalah perlengkapan yang digunakan dalam implementasi sistem :

Table 4.1 Spesifikasi perlengkapan implementasi

No	Perlengkapan	Spesifikasi	Keterangan
1	Software PC	Sistem Operasi	Windows Service Pack 3
		Bahasa pemrograman	PHP 5
		Koneksi <i>interface</i>	Gammu 1.7.4
		<i>Web server</i>	Apache 2
		<i>Database server</i>	MySQL 5
2	USB Modem	Huawei	U220
3	HP	Sony Eriesson	K530i
4	Laptop	<i>Processor</i>	Intel centrino duo
		<i>Hard disk</i>	250 GB
		<i>Memori</i>	1 GB
5	Router Mikrotik RB750	<i>Fast Ethernet</i>	5 port
6	Tampilan	<i>Web browser</i>	Mozilla firefox 5.0

4.1.1 Aplikasi Monitoring LAN

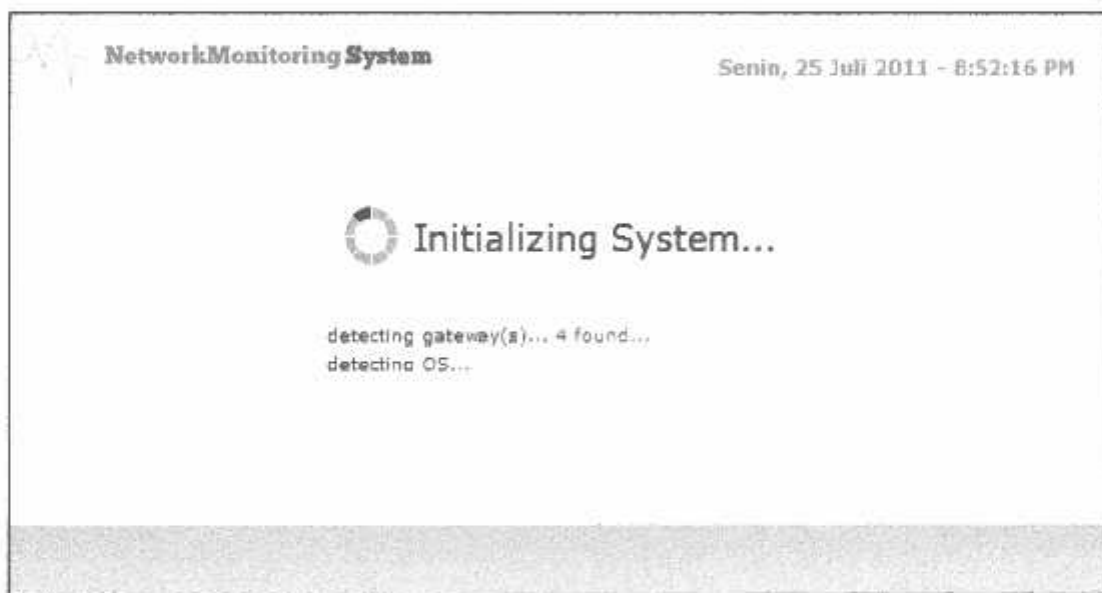
Aplikasi ini adalah inti dari sistem yang melakukan Monitoring secara berkala sesuai konfigurasi yang tersimpan di dalam *database*. Sebagian besar kode program menggunakan teknologi AJAX dengan bahasa pemrograman PHP sehingga aplikasi ini

dapat melakukan update data dan monitoring secara realtime dan otomatis tanpa perlu *me-refresh* halaman *web browser*.

4.1.1.1 Inisialisasi Sistem

Inisialisasi sistem merupakan proses pendeteksian konfigurasi dan Sistem Operasi dimana aplikasi ini berjalan. Jika tidak terdapat *gateway* yang terdeteksi, maka sistem tidak akan berjalan. Sebaliknya, jika seluruh konfigurasi telah *valid*, maka sistem akan mengirimkan pesan bahwa sistem sedang *UP* dan *RUNNING*

Berikut adalah tampilan halaman utama aplikasi monitoring LAN *gateway* secara otomatis dalam Gambar 4.1



Gambar 4.1 Tampilan sistem monitoring secara otomatis

4.1.1.2 Monitoring

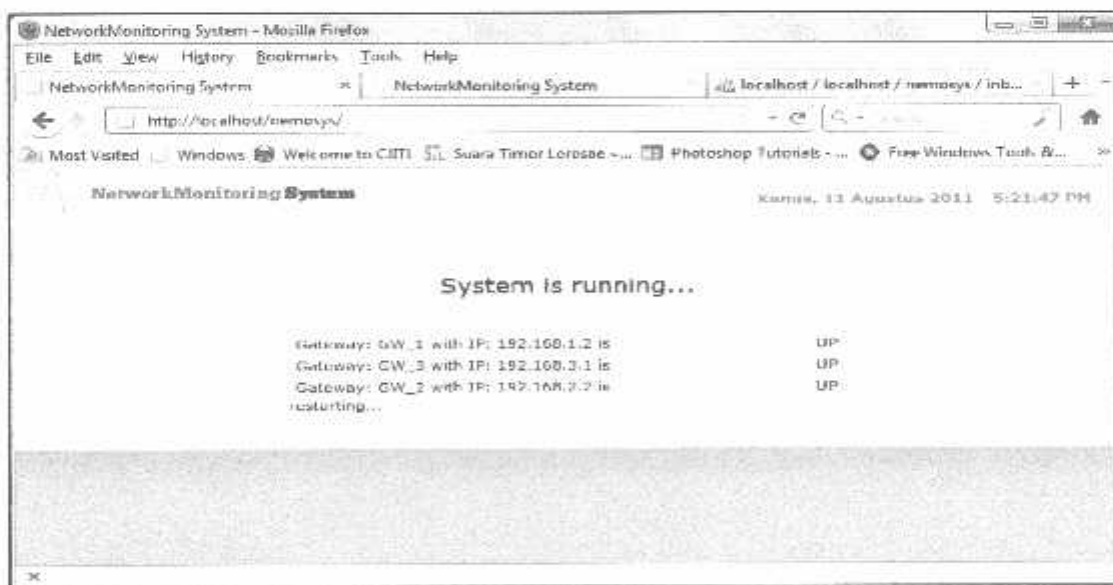
Monitoring merupakan proses untuk mendeteksi keadaan IP dari *gateway/server* yang telah didaftarkan di sistem. Monitoring IP *gateway/server* dibedakan menjadi 2, yakni proses yang dilakukan secara otomatis dan proses yang dijalankan secara manual. Monitoring secara otomatis dilakukan oleh sistem setelah proses inisialisasi selesai.

Sedangkan monitoring manual dilakukan hanya jika sistem mendeteksi adanya *request* dari administrator atau teknisi.

a. Monitoring Otomatis

Sebelum melakukan monitoring otomatis, terlebih dahulu aplikasi membaca setting yang tersimpan di dalam database, termasuk jumlah IP *gateway/server* yang terdaftar, dan harus dimonitor. Untuk melakukan monitoring, aplikasi menjalankan perintah *ping* sebanyak 5 kali untuk setiap IP *gateway/server* yang ada. Hasil perintah *ping* serta kesimpulan apakah *gateway/server* tersebut *UP* atau *DOWN*, disimpan ke dalam *database*.

Jika suatu *gateway/server DOWN* dan sebelumnya *gateway* tersebut berstatus *UP*, maka sistem akan secara otomatis mengirimkan pesan ke nomor HP yang telah terdaftar di sistem. Jika terus terjadi *DOWN* selama batas tertentu (sesuai setting administrator), maka sistem akan kembali mengirimkan notifikasi ke HP administrator untuk mengingatkannya sampai jaringan tersebut di perbaiki oleh teknisi atau administrator. Berikut ini di tunjukkan sisem yang berjalan secara otomatis setelah sistem melakukan inialisasi dengan perangkat – perangkat yang terhubung, dalam Gambar 4.2



Gambar 4.2 Aplikasi memonitor jaringan secara otomatis

Teknik yang digunakan untuk melakukan monitoring secara otomatis yakni dengan menggunakan *timer* pada *Javascript*. Timer diatur sesuai jeda waktu yang telah di atur oleh *user*, dan memanggil fungsi untuk melakukan request secara *realtime* ke aplikasi PHP yang berada di server. Selanjutnya aplikasi PHP tersebut menjalankan perintah untuk melakukan ping satu per satu terhadap IP *gateway/server* yang terdaftar

b. Monitoring Manual

Monitoring manual hanya dilakukan jika sistem mendeteksi adanya request dari nomor HP yang telah terdaftar di sistem. *Request* yang valid adalah sebuah SMS dengan format yang sesuai dengan aturan yang telah ditetapkan.

Untuk melakukan monitoring terhadap satu IP *gateway/server* format penulisannya adalah sebagai berikut:

ping#IP/alias

Contoh : ping#192.168.3.5 atau ping#server_2

Untuk melakukan monitoring terhadap semua IP *gateway/server* format penulisannya adalah sebagai berikut:

ping#all

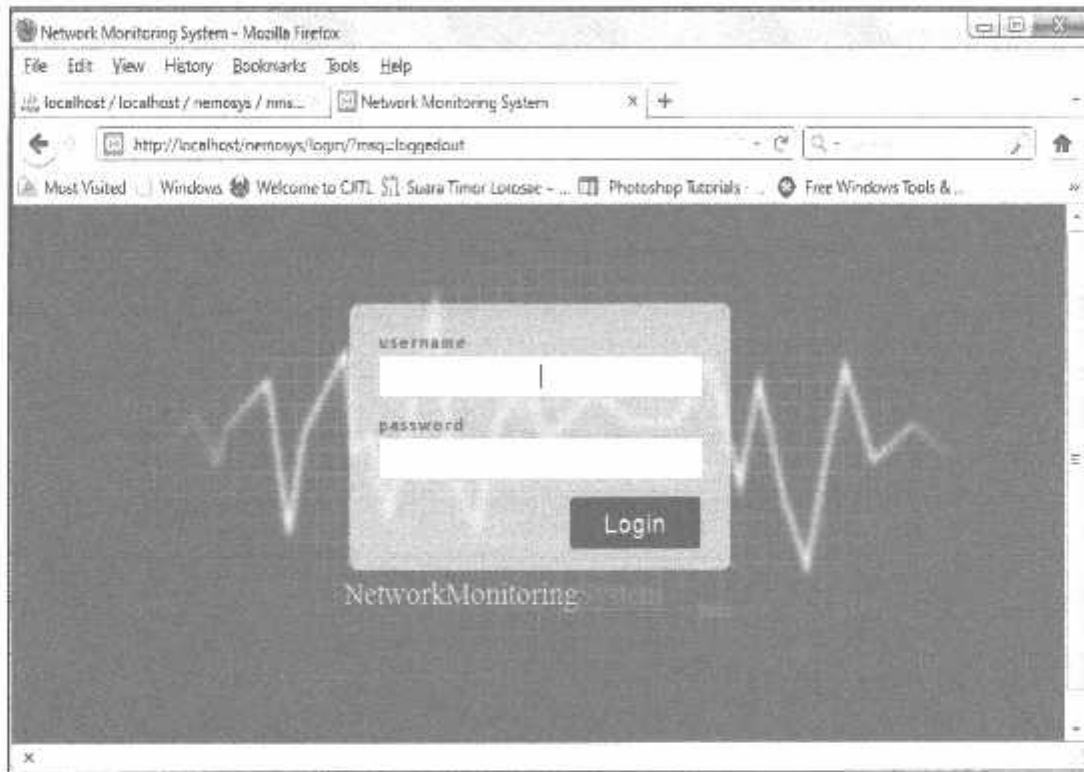
Dalam melakukan monitoring terhadap seluruh IP *gateway/server* yang ada di dalam database. Tentunya jika melakukan request ALL, waktu *reply* SMS akan relatif lebih lama.

4.1.2 Administrasi data sistem monitoring

Aplikasi administrasi ini adalah kumpulan modul – modul yang dapat digunakan untuk menambah dan memanipulasi data serta *setting* untuk sistem monitoring LAN. Aplikasi ini terdiri sebuah halaman autentifikasi dan 3 buah modul yang direpresentasikan ke dalam 3 menu utama. Berikut ini penjelasan masing – masing halaman.

4.1.2.1 Autentifikasi *user*

Autentifikasi *user* merupakan halaman untuk masuk ke dalam aplikasi administrasi. Membutuhkan *username* serta *password* yang valid agar dapat mengakses menu – menu utama yang ada di halaman sistem administrasi. Tampilan halaman *login* untuk masuk ke dalam sistem ditunjukkan dalam gambar 4.3



Gambar 4.3 tampilan halaman *login*

4.1.2.2 Menu Status

Merupakan menu untuk melihat status terakhir dari *gateway/server* yang terdaftar di dalam sistem. Modul ini hanya melakukan *update* data yang diambil dari *database* secara berkala, dan tidak melakukan proses monitoring secara otomatis. Namun demikian, terdapat tombol untuk melakukan ping secara manual jika dirasa perlu.

Terdapat beberapa informasi yang ditampilkan, antara lain alias dari IP *gateway/server* serta status terakhir IP *gateway/server* tersebut. Tombol *checkk* di gunakan untuk melakukan ping secara manual pada tiap IP *gateway/server*, dan tombol *refresh* digunakan untuk memanggil mengambil data terbaru dari *database* untuk di tampilkan. Tampilan dari menu status di tunjukkan dalam gambar 4.4



Gambar 4.4 Tampilan menu status *gateway*

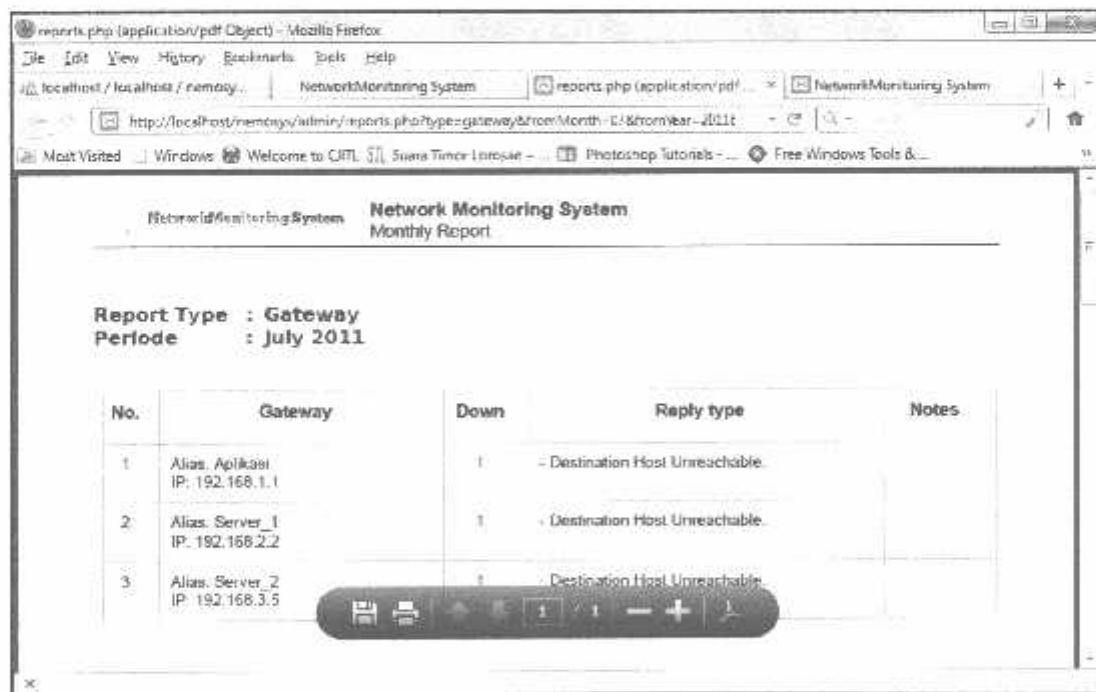
4.1.2.3 Menu laporan *gateway*

Menu laporan yang pertama adalah laporan IP *gateway/server*. Laporan ini berisi informasi IP *gateway/server* yang mengalami *DOWN* dalam periode 1 bulan tertentu. Detail yang ditampilkan yakni IP *gateway/server*, jumlah *DOWN* dalam 1 bulan, serta detail jenis masalah yang terjadi (penyebab IP *gateway/server* *DOWN*). Detail dari jenis masalah yang di tampilkan biasanya berupa *destination host unreachable*, *request time out*, *general failure* dan lain – lain yang berhubungan dengan

proses ping. Sub-menu *reports gateway* dapat diakses oleh seluruh *user* yang memiliki akses ke modul Laporan/Report. Menu laporan/*reports gateway* di tunjukkan dalam Gambar 4.5 dan 4.6



Gambar 4.5 Tampilan *reports* status IP



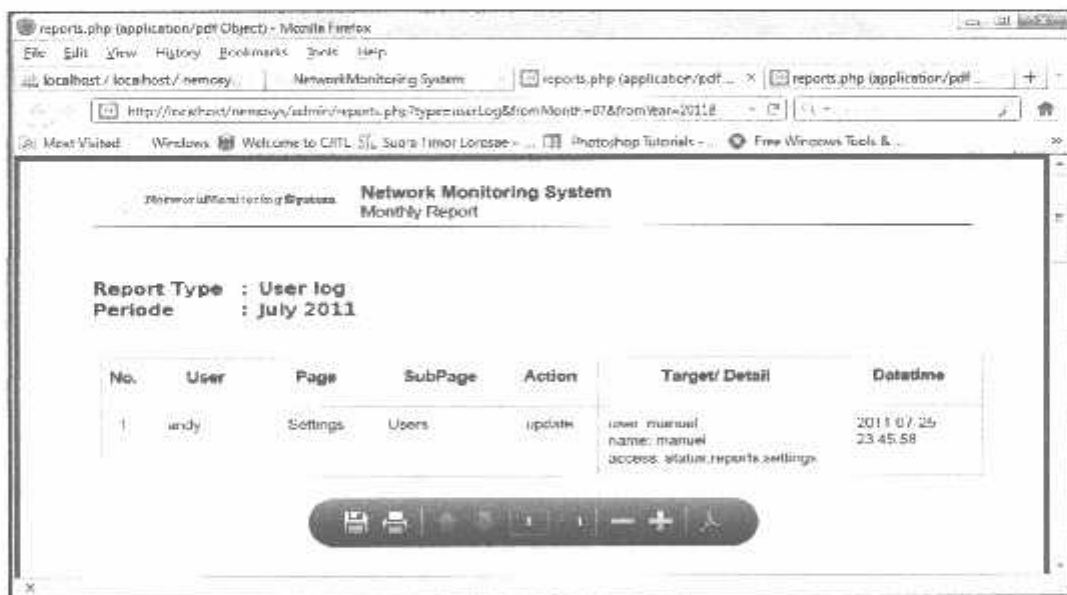
Gambar 4.6 Tampilan *reports* status IP dalam bentuk pdf

4.1.2.4 Menu laporan *user log*

Laporan *user log* yakni laporan yang berisi informasi aktifitas yang dilakukan oleh setiap user yang login ke dalam aplikasi administrasi monitoring LAN. Kategori aktifitas (*action*) yang dilakukan adalah menambah data , memperbaharui data (*update*), dan menghapus data (*delete*). Semua aktifitas ini direkam beserta target atau objek yang diubah. Menu *reports user log* di tunjukkan dalam Gambar 4.7 dan 4.8



Gambar 4.7 Tampilan *report user log*



Gambar 4.8 Tampilan *report user log* dalam bentuk pdf

4.1.2.5 Menu *setting gateway*

Merupakan sub-menu dari *Setting*, yang digunakan untuk melihat, memperbaharui, serta menghapus IP *gateway/server* yang akan dimonitor oleh sistem. Data yang harus diisi jika menambah atau memperbaharui data adalah Alias dan IP dari *gateway/server*. Tampilan dari menu *gateway* di tunjukkan dalam Gambar 4.9

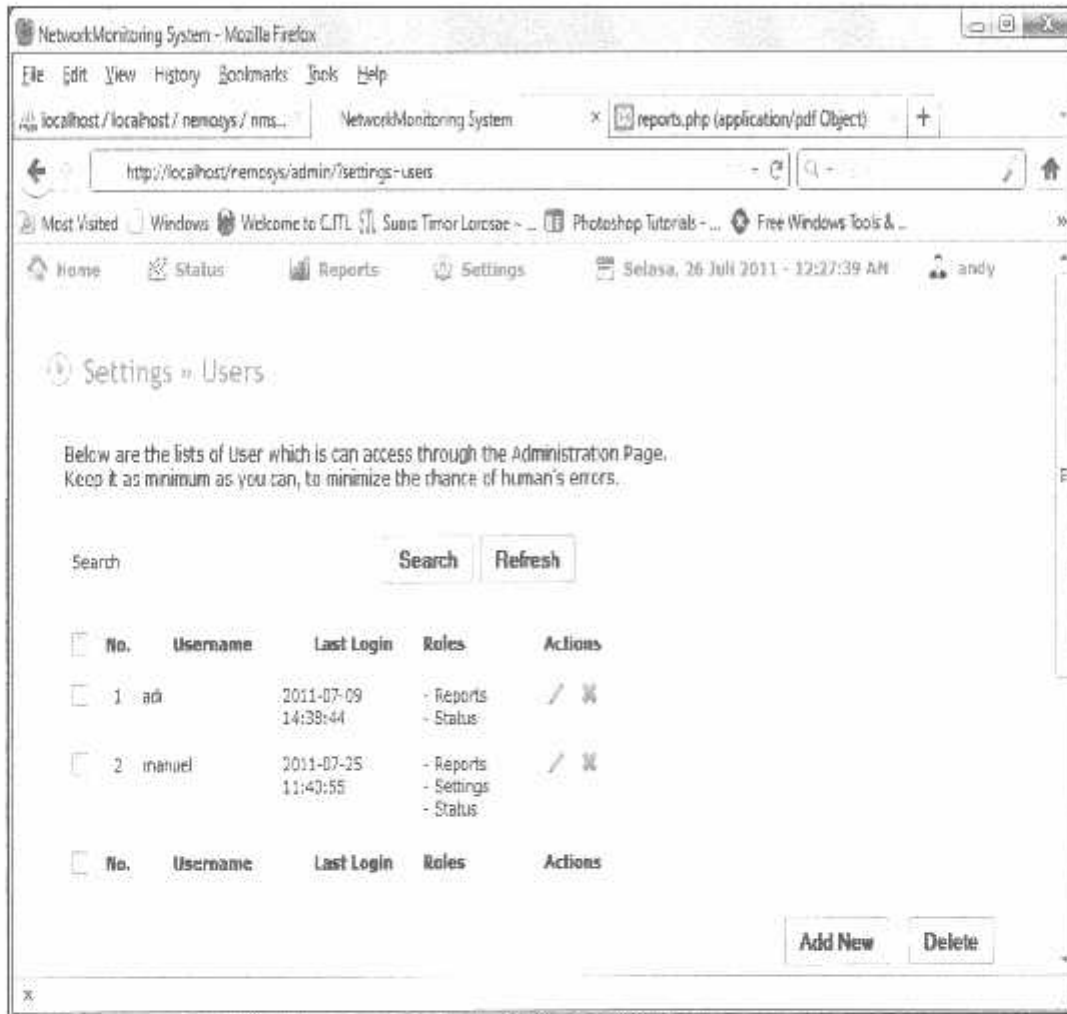


Gambar 4.9 Tampilan menu *gateway*

4.1.2.6 Menu *Setting user list*

Merupakan sub-menu dari *Setting*, yang digunakan untuk melihat, memperbaharui, serta menghapus *user* yang terdaftar di sistem. Untuk menambahkan *user* baru, harus mengisi beberapa data antara lain *username*, *password*, nama asli calon *user*, serta hak akses terhadap modul/ menu yang ada. Setting ini dapat diakses oleh administrator serta

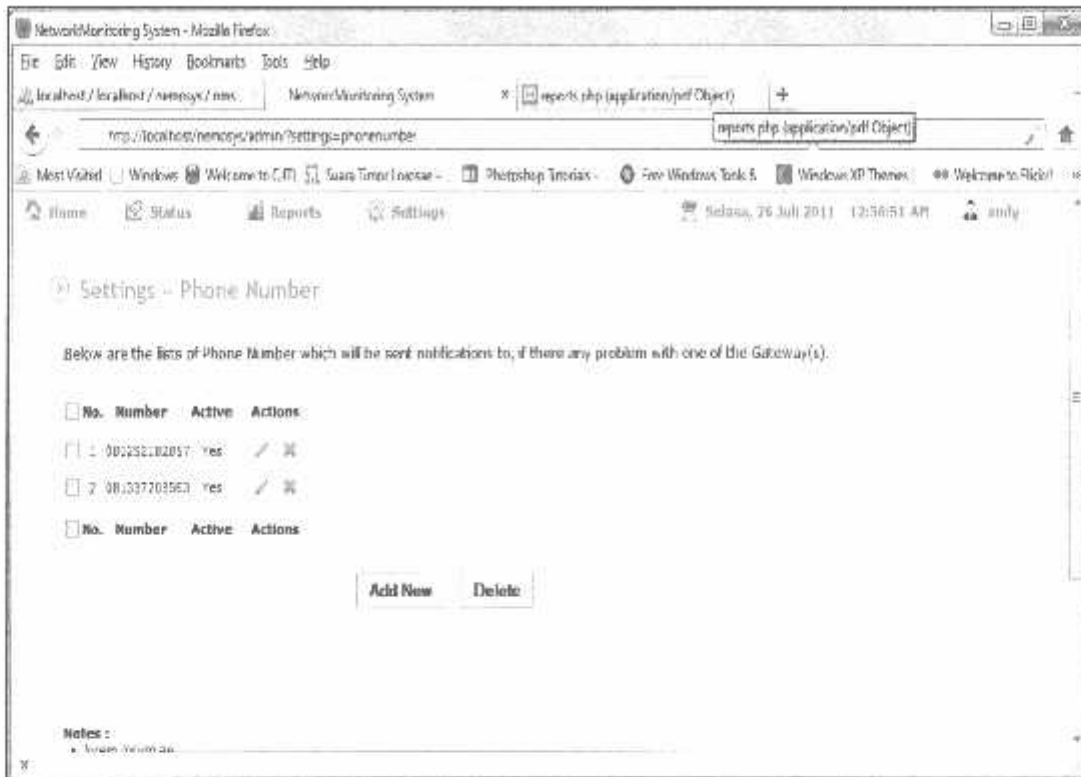
user biasa yang telah diberikan hak untuk mengakses sub-menu ini. Tampilan dari menu *gateway* di tunjukkan dalam Gambar 4.10



Gambar 4.10 Tampilan menu *user list*

4.1.2.7 Menu *setting phone number*

Merupakan sub-menu dari *Setting* yang digunakan untuk melihat, memperbaharui, serta menghapus nomor HP. Notifikasi akan dikirimkan ke nomor – nomor yang telah didaftarkan , nomor HP yang terdaftar juga dapat mengirim *request* secara manual berupa SMS ke aplikasi untuk mengetahui kondisi jaringan. Semua *user* yang mempunyai hak akses terhadap modul *Setting* dapat menggunakan sub-menu ini.tampilan menu *phone number* di tunjukkan dalam Gambar 4.11



Gambar 4.11 Tampilan menu *phone number*

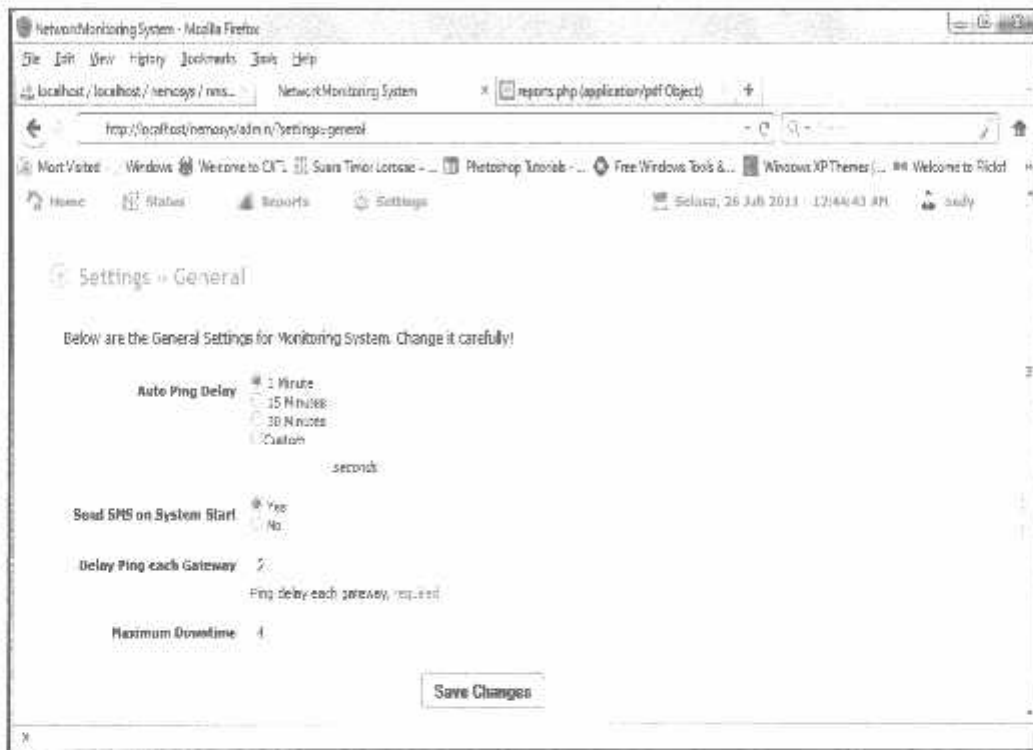
4.1.2.8 Menu *setting general*

Merupakan sub-menu dari *setting* yang digunakan untuk memperbaharui settingan umum sistem monitoring LAN. Adapun settingan umum yang dimaksudkan yakni :

1. tenggang waktu untuk kembali melakukan *ping* terhadap seluruh IP *gateway/server* yang ada dalam sistem.
2. Tenggang waktu *ping IP gateway* antara satu IP dengan IP berikutnya.
3. Jumlah maksimum status *DOWN* pada IP *gateway /server* sebelum mengirim ulang notifikasi kembali ke admin/user.

- Pilihan untuk mengirim SMS notifikasi atau tidak ke admin/user, pada saat sistem *startup*.

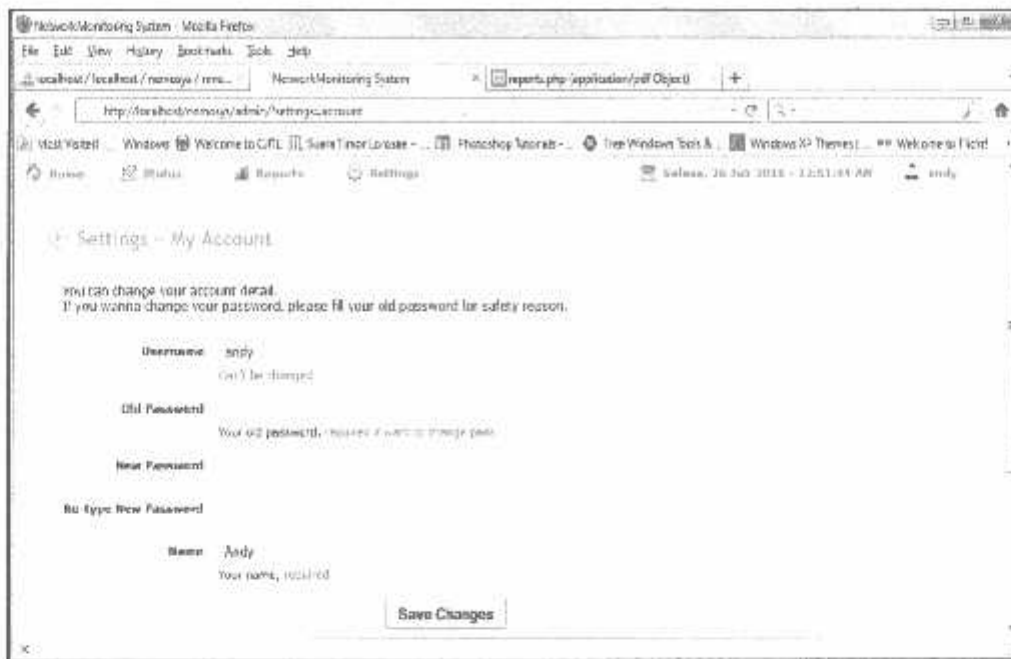
Seluruh *user* dapat mengakses jika memiliki hak terhadap modul *setting*. Tampilan dari menu *setting general* di tunjukkan dalam Gambar 4.12



Gambar 4.12 Tampilan menu *setting general*

4.1.2.9 Menu *setting my account*

Sub-menu terakhir yakni *my account*, digunakan untuk memperbaharui data *user* yang sedang aktif (*login*) saat itu. Data yang dapat diubah hanya *password* dan nama asli. *Username* yg digunakan untuk *login* tidak dapat diubah. Jika ingin mengganti *password*, *user* harus memasukkan *password* lama sebagai salah satu upaya keamanan dan mencegah hal – hal yg tidak diinginkan. Tampilan dari menu *setting my account* di tunjukkan dalam Gambar 4.13

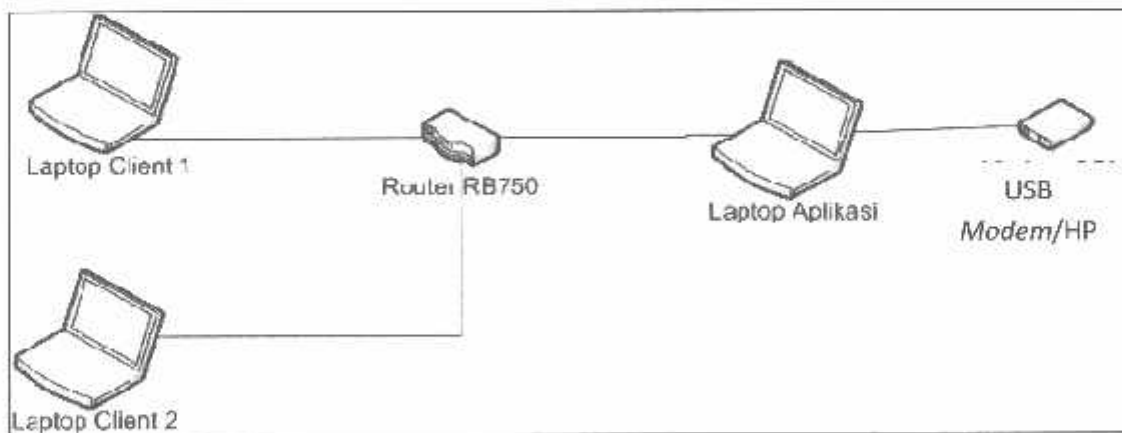


Gambar 4.13 Tampilan menu *setting my account*

4.2 Pengujian Sistem

4.2.1 Pengujian pada Topologi Jaringan yg Sederhana

Pengujian sistem dilakukan pada jaringan sederhana, seperti yang di tunjukkan dalam Gambar 4.14



Gambar 4.14 Topologi jaringan dalam pengujian sistem

Berikut perangkat – perangkat yang digunakan dalam pengujian sistem monitoring seperti di tunjukkan dalam Tabel 4.2

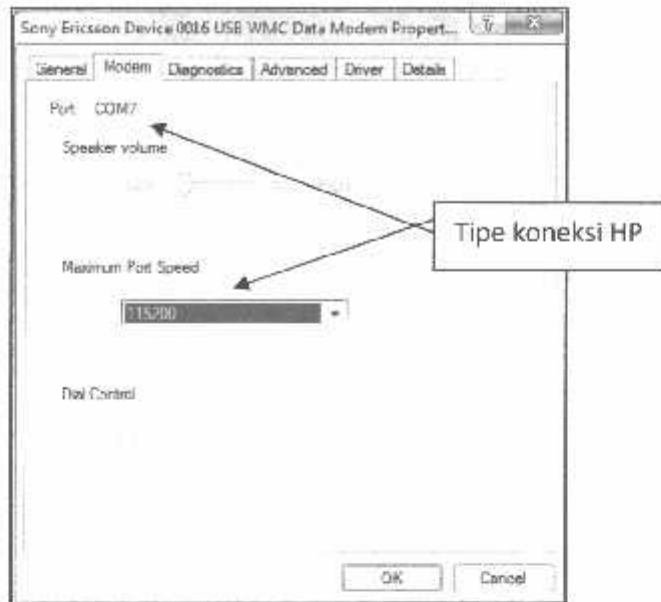
Tabel 4.2 perangkat untuk pengujian sistem

No	Nama	IP	Keterangan
1	Aplikasi	192.168.1.1	Laptop tempat aplikasi
2	PC client 1	192.168.2.2	Mewakili 1 tipe jaringan
3	Laptop Client 2	192.168.3.5	Mewakili 1 tipe jaringan
4	Router RB750		RouterBoard Mikrotik
5	USB Modem GSM		Huawei E220 – (<i>Sim card axis</i>)
6	HP Sony Ericsson		K530i – (<i>Sim card Simpati</i>)

4.2.2 Pengujian sistem monitoring secara Otomatis

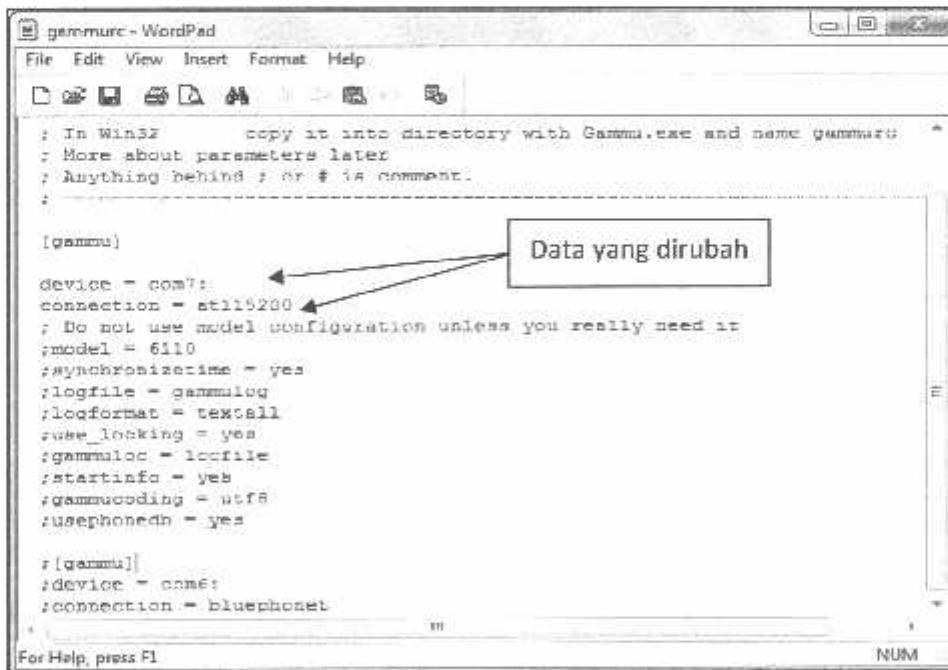
Sebelum menjalankan aplikasi perlu membuat koneksi gammu dengan aplikasi database yang ada pada sistem, langkah – langkah untuk melakukan koneksi gammu adalah sebagai berikut:

- a. Mengatur tipe koneksi gammu dengan HP/USB *modem* yaitu dengan mengatur *COM port* yang digunakan dan kecepatan transfer data. Cek koneksi modem Pada *device manager* di *My computer*, Berikut di tunjukkan tipe HP dan koneksinya, dalam Gambar 4.15

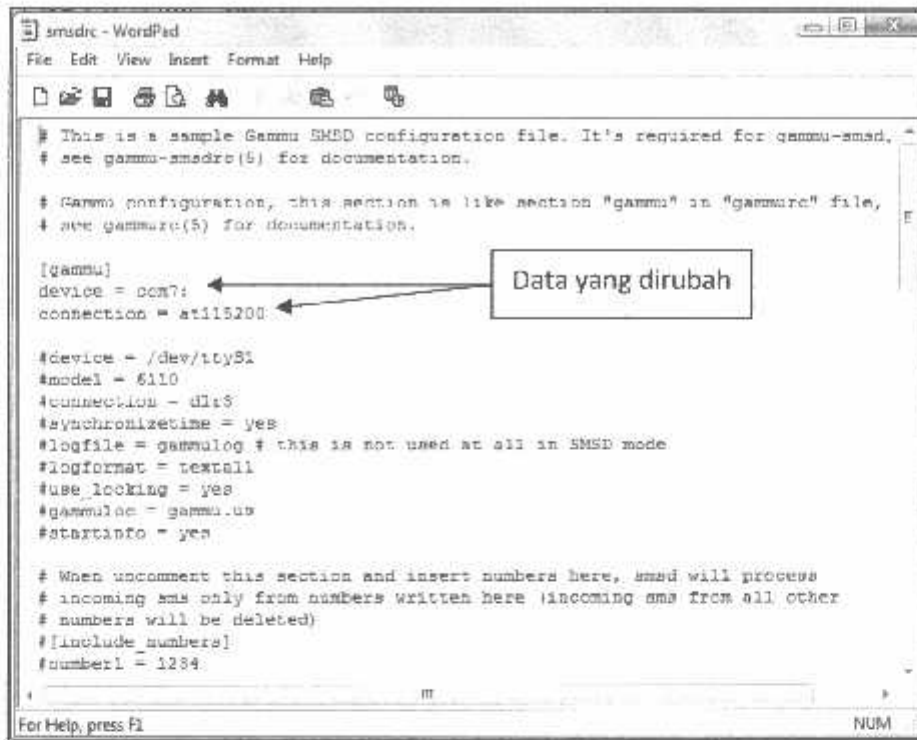


Gambar 4.15 tipe koneksi data dari IIP

- b. Mengubah koneksi di gammu dengan cara mengubah file *gammurc* dan *smsdrsrc* pada file yang terletak di *C:\gammu\bin*, di tunjukkan dalam Gambar 4.16 dan Gambar 4.17



Gambar 4.16 File *gammurc*

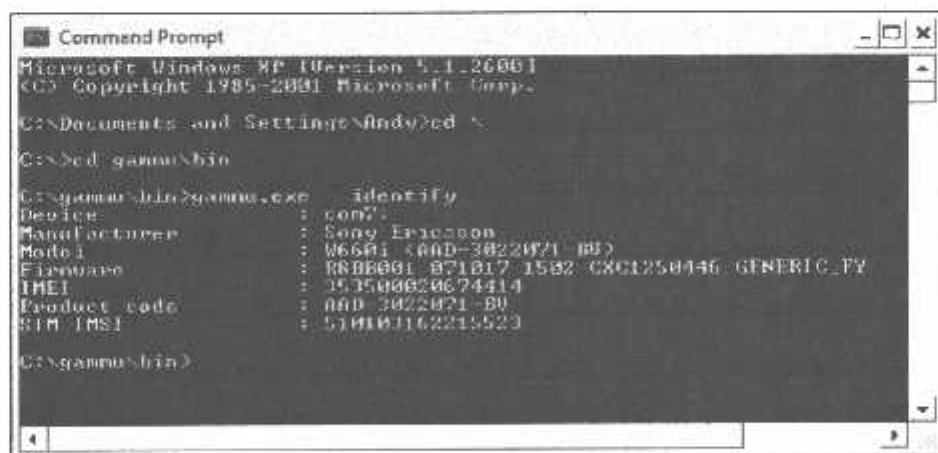


Gambar 4.17 File *smsdrc*

- c. Mengidentifikasi tipe HP/USB *modem* pada *gammu* dengan cara menjalankan *command prompt*. Dengan menuliskan perintah sebagai berikut:

Gammu.exe -identify

Tampilan pada *command prompt* di tunjukkan dalam Gambar 4.18

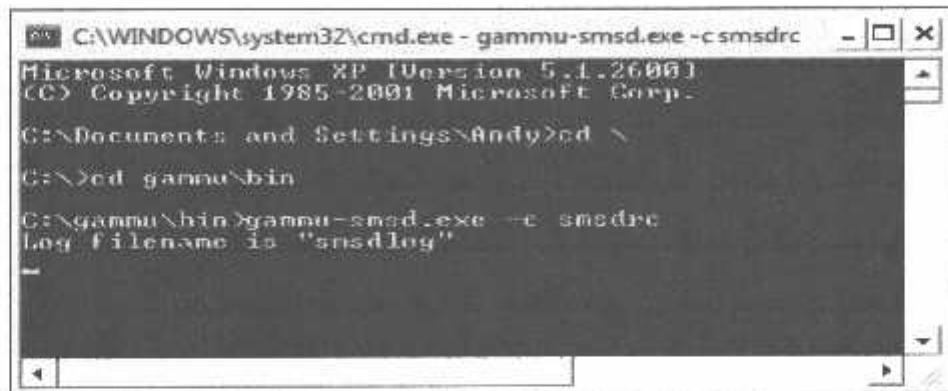


Gambar 4.18 perintah identifikasi HP pada *gammu*

- d. Menjalankan *gammu* agar mengakses data baik dari HP/USB *modem* maupun dari *database* yaitu lewat *command prompt* dengan perintah sebagai berikut:

```
gammu-smsd.exe -c smsdrc
```

tampilan pada *command prompt* di tunjukkan dalam Gambar 4.19



```
C:\WINDOWS\system32\cmd.exe - gammu-smsd.exe -c smsdrc
Microsoft Windows XP [Version 5.1.2600.1]
(C) Copyright 1985-2001 Microsoft Corp.

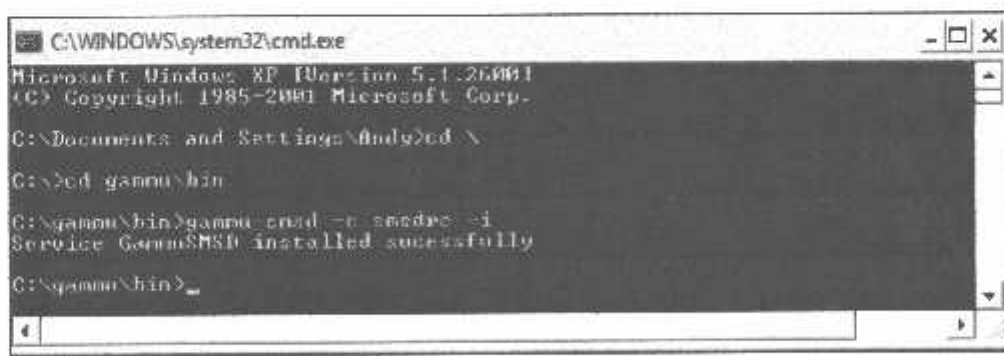
C:\Documents and Settings\Andy>cd \
C:\>cd gammu\bin
C:\gammu\bin>gammu-smsd.exe -c smsdrc
log filename is "smsdlog"
_
```

Gambar 4.19 perintah menjalankan *gammu*

- e. Untuk menginstall *gammu* agar dapat berjalan secara otomatis pada sistem yaitu dengan menjalankan perintah pada *command prompt* dengan format penulisan sebagai berikut:

```
gammu-smsd -c smsdrc -i
```

berikut tampilan untuk menginstall *gammu* agar berjalan secara otomatis dalam sistem di tunjukkan dalam Gambar 4.20

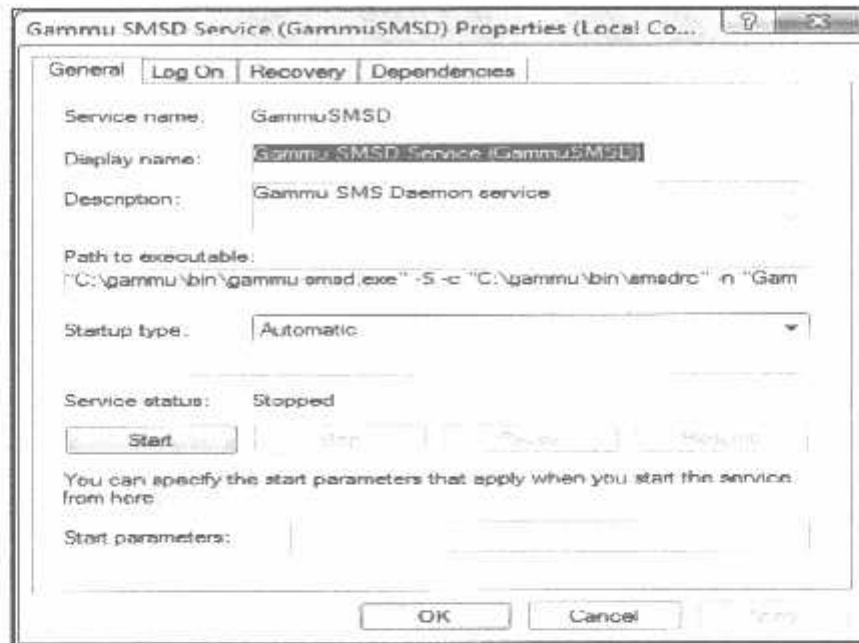


```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600.1]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Andy>cd \
C:\>cd gammu\bin
C:\gammu\bin>gammu-smsd -c smsdrc -i
Service GammuSMSD installed successfully
C:\gammu\bin>_
```

Gambar 4.20 Tampilan menginstall *gammu* pada *command prompt*

Untuk melihat apakah *gammu* sudah terinstall dengan baik pada sistem, yaitu dengan membuka *services.msc* pada sistem *run* atau pada *control panel* . tampilan sistem *services* di tunjukkan dalam Gambar 4.21

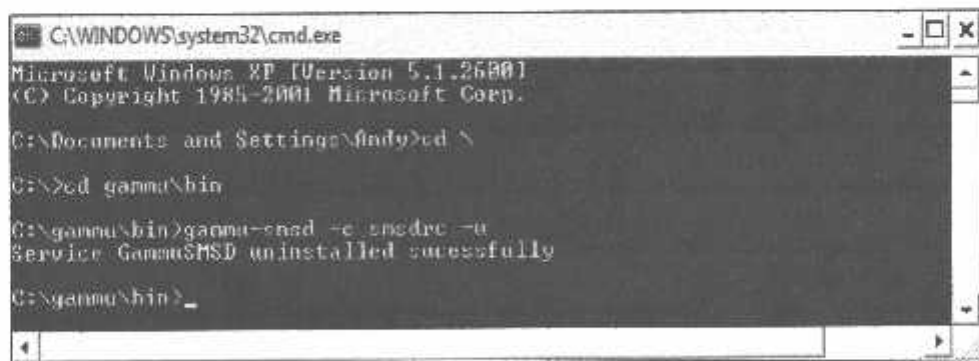


Gambar 4.21 Tampilan sistem *services* *gammu*

- f. Untuk *uninstall service* *gammu* dari sistem, yaitu dengan menjalankan perintah pada *command prompt* dengan format penulisan sebagai berikut:

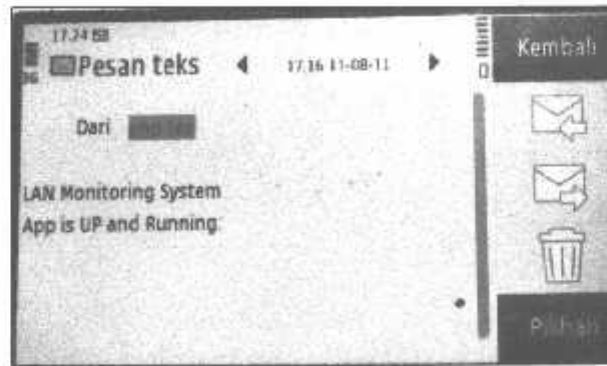
```
gammu-smad -e smadrc -u
```

berikut tampilan untuk *uninstall service* *gammu* pada sistem di tunjukkan dalam Gambar 4.22



Gambar 4.22 Tampilan *command prompt* untuk *uninstall service* *gammu*

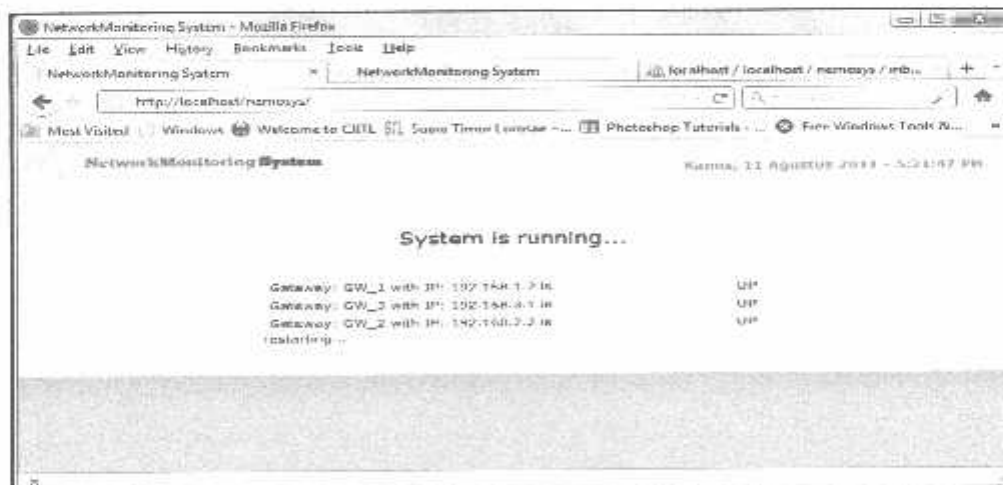
Pengujian monitoring secara otomatis dilakukan dengan menjalankan *web browser* dan mengarahkan alamat *URL* ke *http://localhost/nemosys/*. Terdapat 2 buah *client* yang dimonitor oleh sistem. Masing – masing *client* mewakili satu tipe jaringan berbeda. Sistem akan mengirimkan pesan SMS secara otomatis ke administrator jaringan menginformasikan setiap kali aplikasi berjalan baik itu pada saat program di jalankan maupun pada saat start up atau komputer aplikasi *restart*. Hasil dari pengiriman SMS di tunjukkan dalam Gambar 4.23



Gambar 4.23 Tampilan SMS saat aplikasi *start up*

4.2.2.1 Pengujian Jika Seluruh Client Aktif (UP)

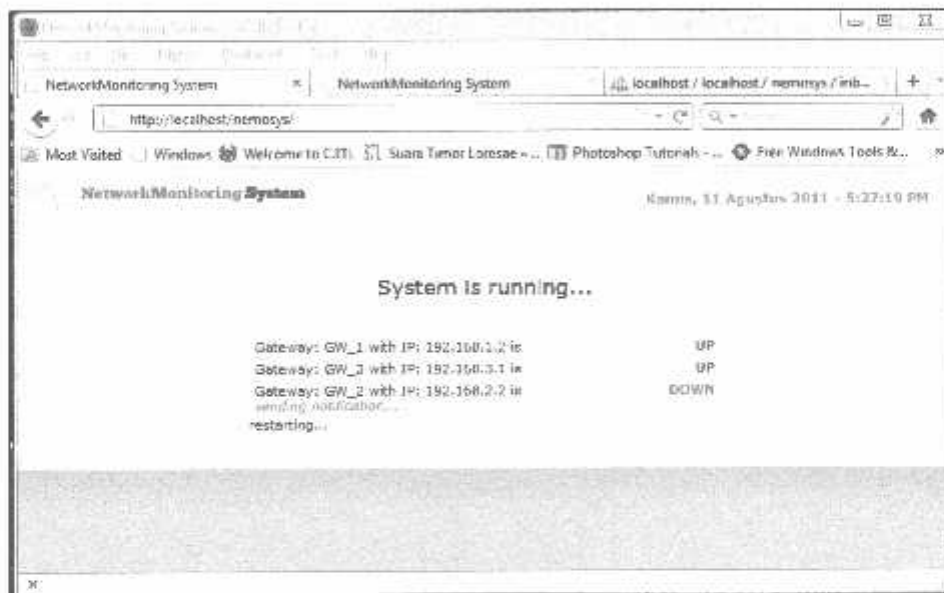
Pada pengujian pertama, disimulasikan bahwa seluruh *client* yang dimonitor berstatus aktif (*UP*). Hasil pengujian ditunjukkan dalam Gambar 4.24



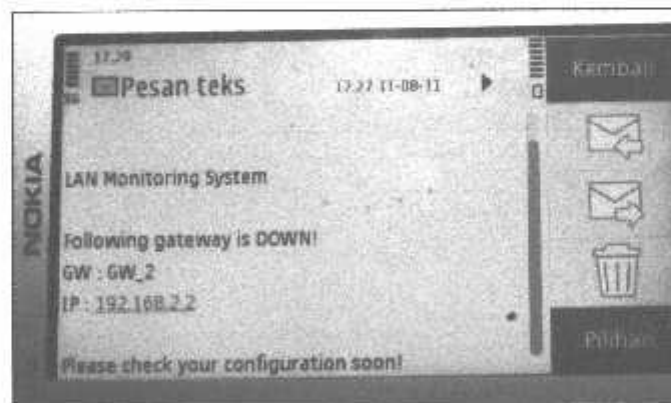
Gambar 4.24 Tampilan aplikasi dalam keadaan *UP*

4.2.2.2 Pengujian jika salah satu klien Bermasalah

Untuk mensimulasikan suatu gateway mengalami masalah, dapat dilakukan dengan beberapa cara. Pada pengujian ini, dilakukan dengan mencabut salah satu kabel yang terhubung ke salah satu *gateway*. Hasil pengujian menunjukkan bahwa sistem mendeteksi bahwa *client* yang tidak terhubung berubah status menjadi *DOWN*. Sistem akan langsung mengirimkan SMS notifikasi yang menginformasikan bahwa client dengan IP tersebut dalam keadaan *DOWN*. Berikut ditunjukkan dalam Gambar 4.25 bila salah satu IP yang di monitor *DOWN* dan Gambar 4.26 SMS notifikasi yang di terima oleh administrator jaringan.

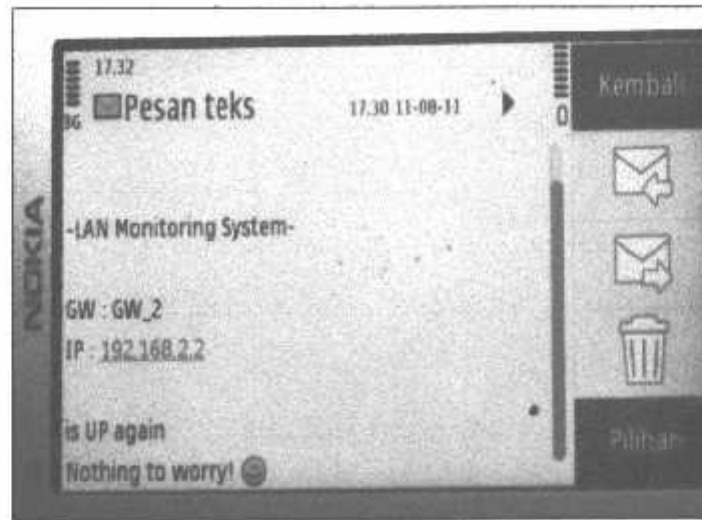


Gambar 4.25 Tampilan aplikasi bila salah satu jaringan *DOWN*



Gambar 4.26 Tampilan SMS yang di terima administrator

Bila IP dari *subnet* yang di monitoring kembali *UP* maka aplikasi akan mengirimkan notifikasi ke administrator jaringan untuk menginformasikannya. SMS yang di terima oleh administrator jaringan di tunjukkan dalam Gambar 4.27



Gambar 4.27 Tampilan SMS bila jaringan kembali normal (*UP*)

4.2.3 Pengujian sistem monitoring secara manual

Berbeda dengan sistem monitoring secara otomatis yang hanya akan mengirim hasil monitoring ke admin/user bila salah satu *subnet* pada jaringan *DOWN*, dalam sistem monitoring secara manual, untuk mengetahui kondisi dari jaringan, admin/user harus mengirimkan pesan *request* lewat SMS ke sistem aplikasi, Setelah menerima pesan SMS, sistem akan mengecek nomor pengirim SMS bila nomor pengirim ada dalam *database* maka sistem akan memproses SMS tersebut dan hasil dari proses ping akan disimpan dalam *database* dan juga sistem akan membalas SMS sesuai dengan hasil yang telah di proses

Format yang digunakan untuk merequest ke sistem untuk melakukan ping pada salah satu IP *gateway/server* pada jaringan maka formatnya adalah sebagai berikut:

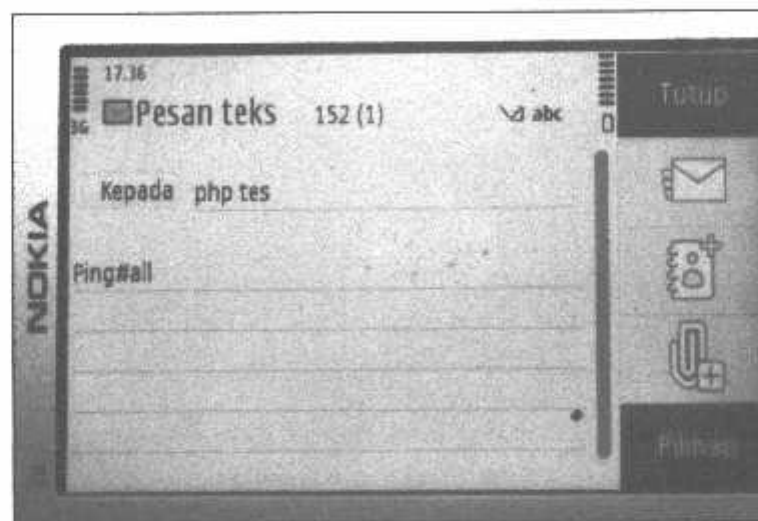
PING#IP/ALIAS

Untuk melakukan ping terhadap seluruh IP *gateway/server* dalam jaringan maka format yang di gunakan untuk merequest adalah sebagai berikut:

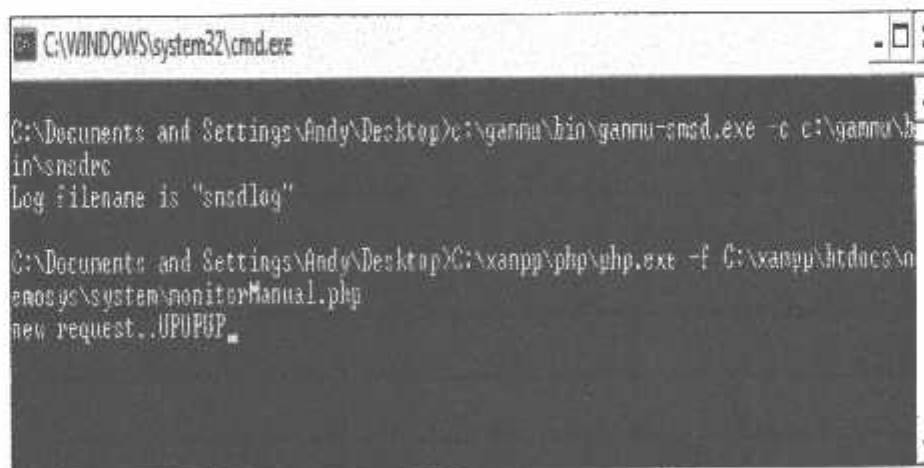
PING#ALL

4.2.3.1 Pengujian ping IP secara keseluruhan

Berikut ini adalah isi SMS yang dikirim oleh admin/*user* ke nomor HP/USB *modem* yang terkoneksi pada PC aplikasi. Untuk merequest kondisi jaringan secara keseluruhan, di tunjukkan dalam Gambar 4.28 dan Gambar 4.29

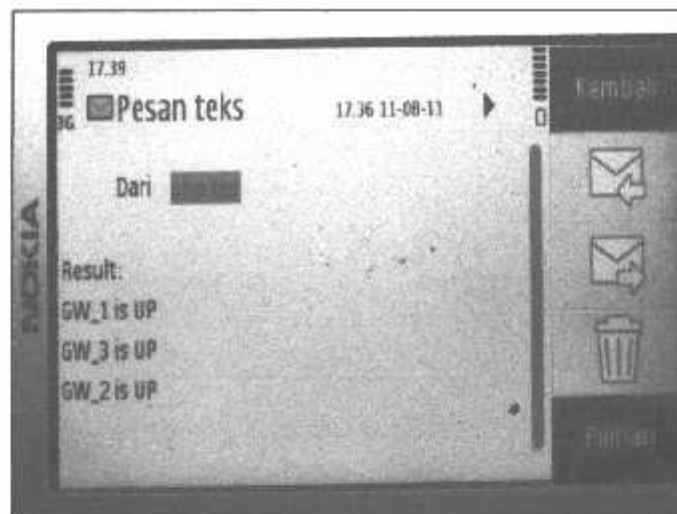


Gambar 4.28 Tampilan format SMS untuk ping semua IP



Gambar 4.29 Tampilan proses pada *command prompt* saat *request* semua IP

Hasil balasan SMS yang diterima admin/user bila kondisi jaringan semua dalam keadaan UP, di tunjukkan dalam Gambar 4.30

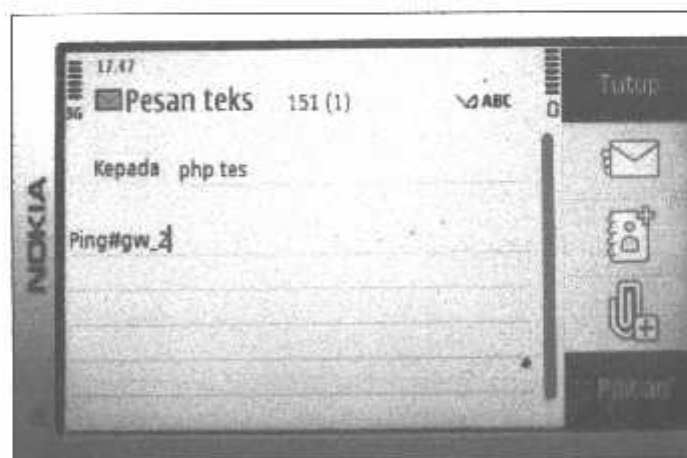


Gambar 4.30 Bila semua IP dalam keadaan UP

4.2.3.2 Pengujian ping salah satu IP

Pengujian salah satu client dilakukan dengan mengirimkan *request* dengan isi SMS sebagai berikut :

PING#GW_2 untuk *request* ke sistem dengan menggunakan alias di tunjukkan dalam Gambar 4.31 dan tampilan pada *command prompt* di tunjukkan dalam Gambar 4.32



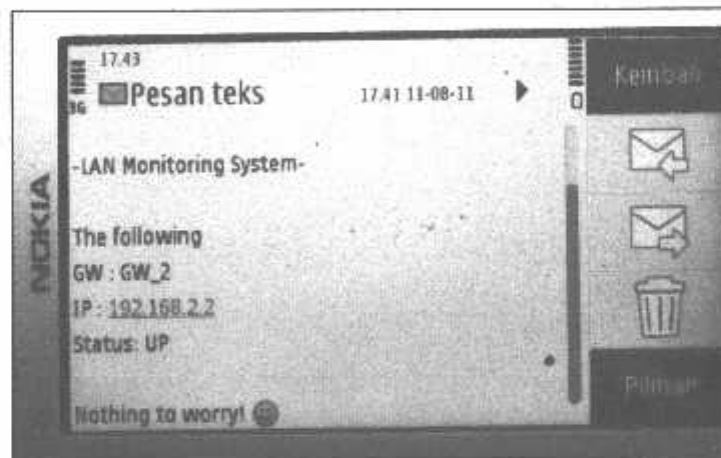
Gambar 4.31 Tampilan format *request* salah satu IP dengan menggunakan alias

```
C:\Documents and Settings\Andy\Desktop>cd %xampp%\bin\gammu-mod.exe -c c:\xampp\bin\gammu\bin\index
Log filename is "modlog"

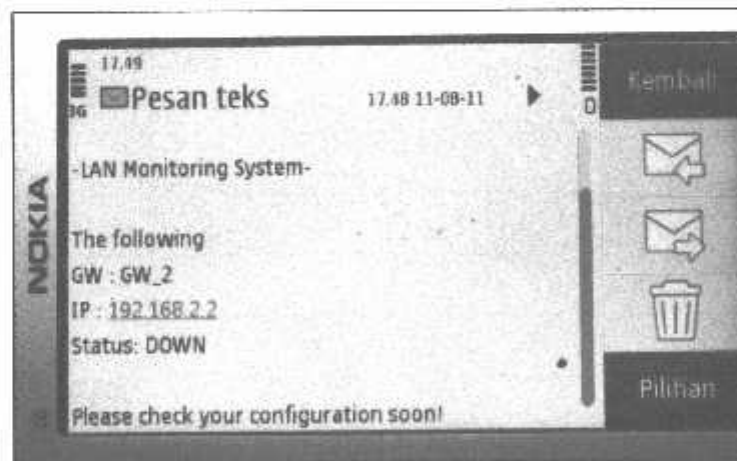
C:\Documents and Settings\Andy\Desktop>C:\xampp\php\php.exe -f C:\xampp\htdocs\lanmon\sysmon\monitorManual.php
new request: IP:192.168.2.2
C:\Documents and Settings\Andy\Desktop>C:\xampp\php\php.exe -f C:\xampp\htdocs\lanmon\sysmon\monitorManual.php
192.168.2.2 is valid IP, found in db, so
C:\Documents and Settings\Andy\Desktop>C:\xampp\php\php.exe -f C:\xampp\htdocs\lanmon\sysmon\monitorManual.php
gw_2 isnt valid IP, not found in db, so
```

Gambar 4.32 Tampilan proses pada command prompt saat menerima request

Hasil dari balasan ping yang di terima di tunjukkan dalam gambar 4.33 dan 4.34



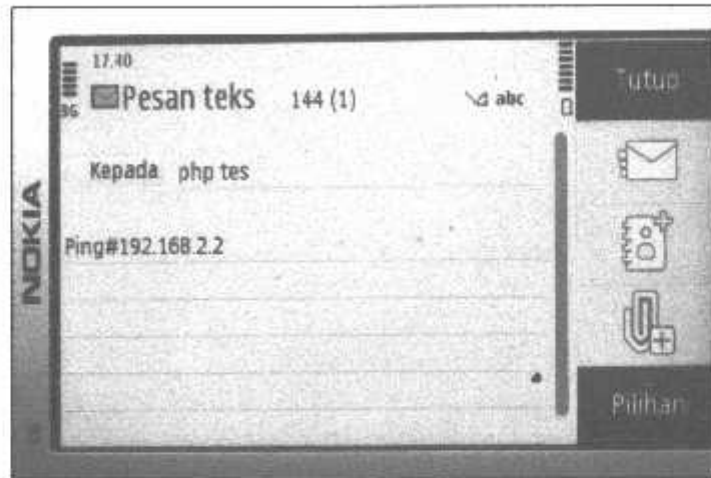
Gambar 4.33 Hasil balasan bila IP dalam keadaan UP



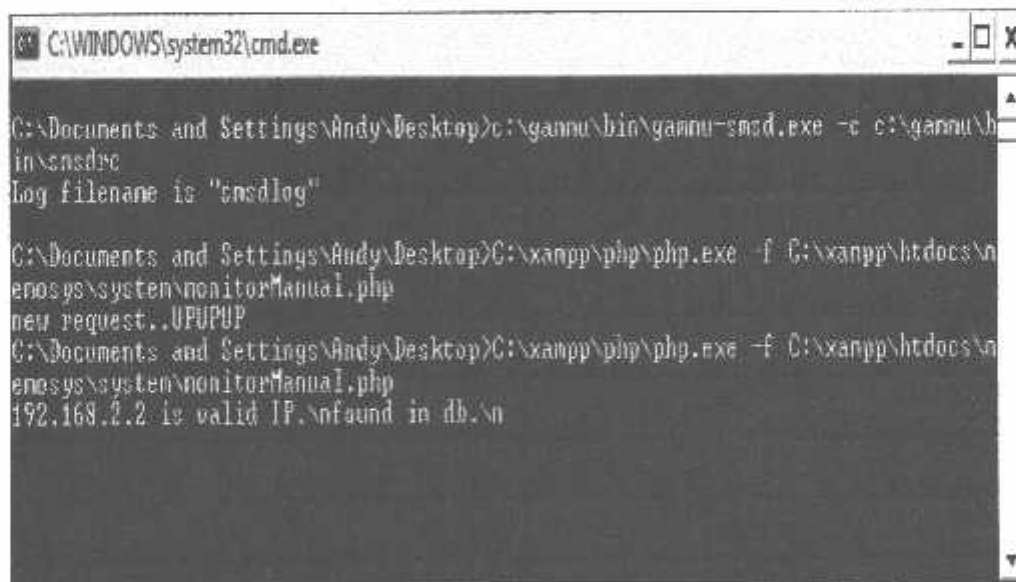
Gambar 4.34 Hasil balasan bila IP dalam keadaan DOWN

Dalam *me-request*, admin juga dapat *me-request* dengan menggunakan format penulisan *ping# nomor IP* yang di tuju. Berikut di tunjukkan format penulisan *request* dengan menggunakan nomor IP.

Ping#192.168.2.2



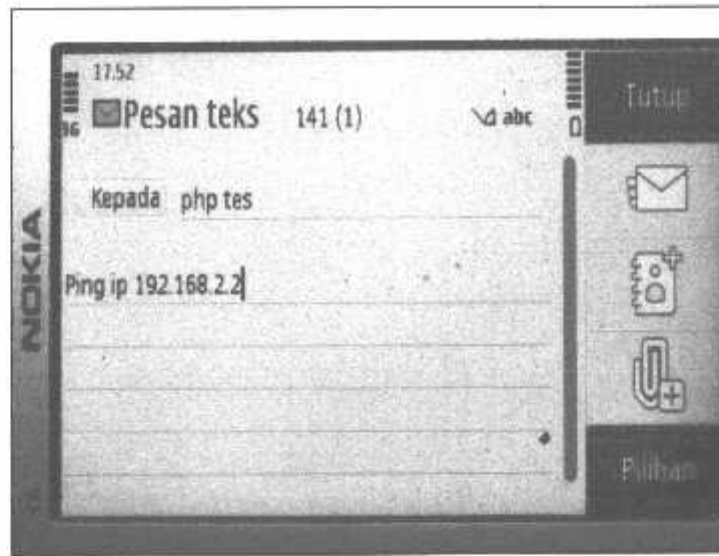
Gambar 4.35 Tampilan format *request* dengan menggunakan nomor IP



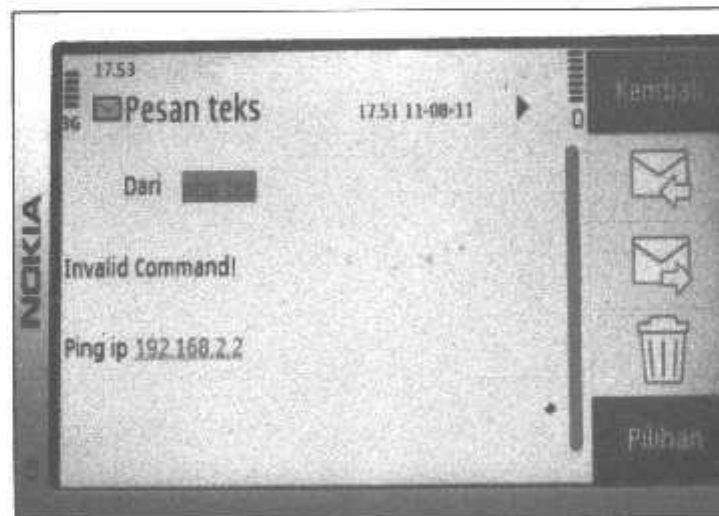
Gambar 4.36 Tampilan proses pada *command prompt* saat menerima *request*

Bila format penulisan tidak sesuai dengan format yang telah di atur dalam sistem monitoring maka, aplikasi untuk akan mengirim pesan *error* ke nomor *handphone*

pengirim SMS. Berikut di tunjukkan dalam contoh format penulisan yang salah dan hasil balasan dari aplikasi, dalam Gambar 4.37 dan 4.38



Gambar 4.37 Tampilan format kesalahan penulisan request



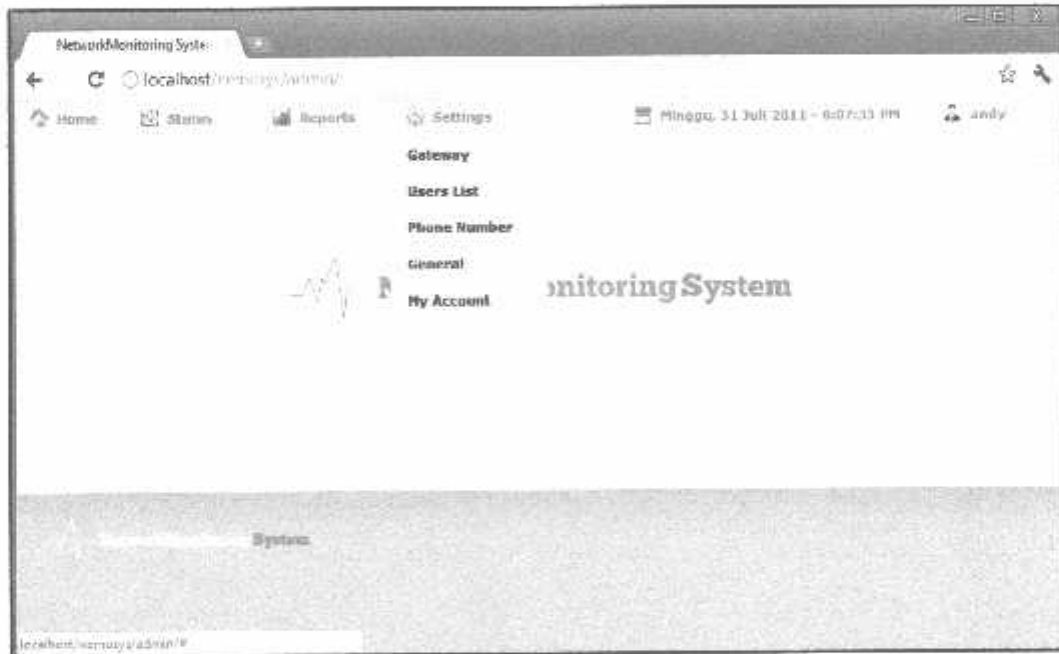
Gambar 4.38 Hasil balasan SMS dari sistem aplikasi

4.2.4 Pengujian pada browser

Berikut ini aplikasi sistem monitoring di ujikan pada beberapa browser yang kompatibel dengan sistem operasi windows diantaranya:

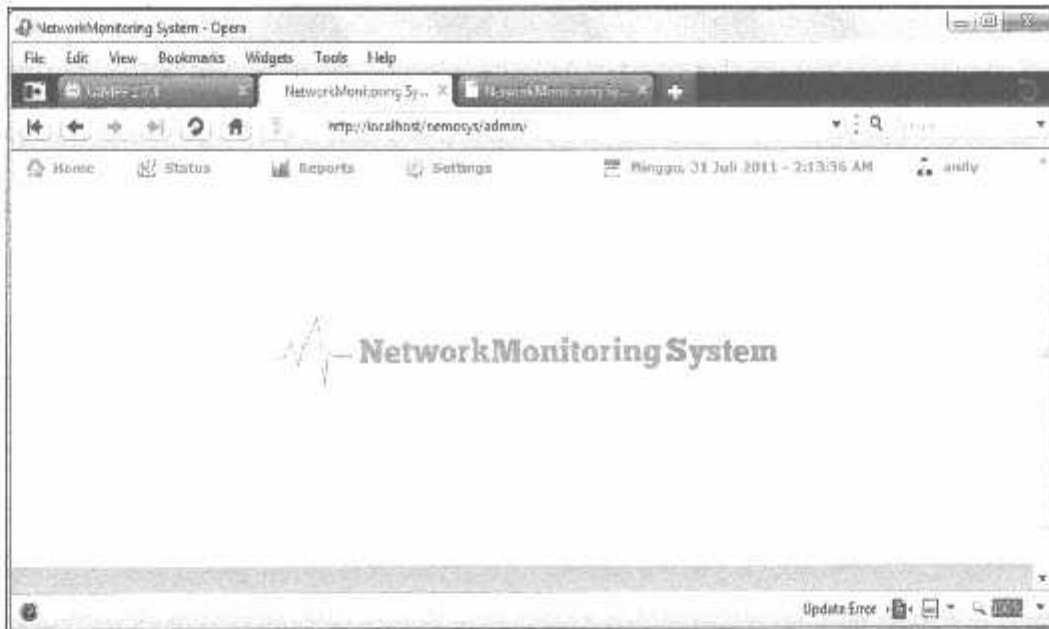
1. Google chrome versi 12
2. Opera versi 10.00 Beta
3. Mozila firefox versi 5.0
4. Internet Explorer versi 8

Pengujian yang pertama di lakukan pada *browser* google chrome versi 12 seperti di tunjukkan dalam Gambar 4.39



Gambar 4.39 Tampilan halaman admin pada *browser google chrome*

Pengujian kedua di lakukan pada browser opera versi 10 seperti di tunjukkan dalam Gambar 4.40



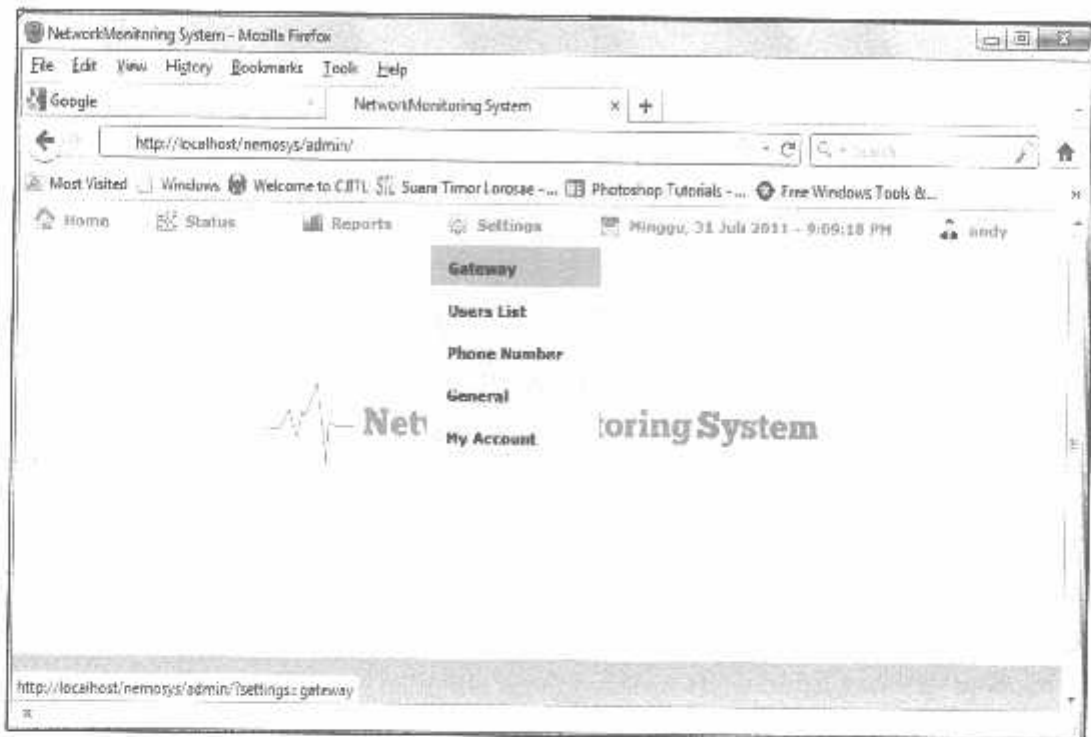
Gambar 4.40 Tampilan halaman admin pada *browser opera*

Pengujian ketiga di lakukan pada browser internet explorer versi 8 seperti di tunjukkan dalam Gambar 4.41



Gambar 4.41 Tampilan halaman admin pada *browser internet explorer*

Pengujian terakhir dilakukan pada *browser* mozilla firefox versi 5 seperti di tunjukkan dalam Gambar 4.42



Gambar 4.42 Tampilan halaman admin pada *browser mozilla firefox*

4.2.5 pengujian dengan USB *modem*

Dalam pengembangan sistem monitoring ini juga digunakan sebuah USB *modem* GSM untuk menggantikan HP sebagai *gateway* aplikasi. Dari pengujian pada sistem monitoring secara otomatis hasil yang ditampilkan sama dengan hasil yang dilakukan dengan menggunakan HP sebagai *gateway*, namun pada percobaan secara manual dimana admin harus mengirimkan pesan *request* untuk hasil yang ditampilkan mendapatkan *error* karena USB *modem* membutuhkan *memory* yang lebih besar pada PC aplikasi untuk memproses pesan SMS yang masuk ke PC aplikasi. Hasil dari percobaan di tunjukkan dalam Gambar 4.43

BAB V

PENUTUP

5.1 Kesimpulan

Dari semua hasil pengujian yang telah dilakukan, maka dapat diambil kesimpulan bahwa:

1. Pada saat aplikasi melakukan ping ke tiap IP *gateway/server*, aplikasi tidak melakukan ping secara serentak ke semua IP, melainkan aplikasi melakukan ping secara bergantian sesuai dengan urutan IP yang telah terdaftar dalam sistem aplikasi tersebut dan tenggang waktu untuk ping antara satu IP dengan IP berikutnya dapat di atur dalam sistem aplikasi.
2. Sistem aplikasi monitoring ini telah di uji dan dapat berjalan dengan baik pada sistem operasi windows.
3. Aplikasi monitoring dapat berjalan pada hampir semua browser yang telah kompatibel dengan sistem operasi windows, diantaranya yang telah diuji adalah opera, mozilla firefox, internet explorer dan google chrome.
4. Batas jumlah *subnet* dalam LAN yang di monitoring hanya 5 *subnet* dan jumlah IP yang telah diuji oleh sistem aplikasi adalah 3 IP.
5. Dalam pengujian sistem monitoring pada browser IE, proses mengeksekusi program menjadi sedikit lebih lambat, ini di sebabkan karena tidak semua *script* bahasa pemrograman dapat dibaca dan dieksekusi dengan baik pada browser IE.

5.2 Saran

1. Sistem aplikasi ini dapat dikembangkan lagi untuk memonitor jaringan yang skalanya lebih besar dengan salah satu contoh menambah jumlah IP yang dimonitor. Dan juga bisa menambah fitur – fitur lebih yang tidak memonitor sebatas *error* koneksi tapi juga bisa memonitor *bandwith*
2. sistem aplikasi ini juga dapat dikembangkan lagi agar bisa diakses baik itu lewat SMS maupun secara *online* melalui *internet*.
3. Sebaiknya menggunakan HP sebagai interface *gateway* dari pada modem, karena modem membutuhkan memori yang besar untuk memproses baik itu pengiriman SMS maupun untuk menerima SMS, sehingga membuat kinerja dari sistem akan menurun.

DAFTAR PUSTAKA

Anhar, 2010. *Panduan Menguasai PHP & MySQL secara otodidak* , Penerbit MediaKita, Jakarta.

ANDI, 2009, *Menguasai XHTML, CSS, PHP, MySQL melalui dreamweaver*, Penerbit ANDI,yogyakarta.

Behrouz A.Forouzan, 2007. *Data Communications and Networking fourth edition*, McGraw-Hill Education, USA.

C. Widyo Hermawan, 2009, *PHP programing* penerbit ANDI, Yogyakarta.

Douglas E. Comer, 2009. *Computer Networks and Internet fifth edition*, Penerbit Pearson Education International, USA.

Iwan Sofana, 2009. *Cisco CCNA & Jaringan Komputer*, Penerbit INFORMATIKA, Bandung.

Jogiyanto H. M. 1998. *Accounting Information System*, penerbit INFORMATIKA, Bandung.

Niall Mansfield, 2002. *PRACTICAL TCP/IP mendesain,menggunakan, dan Troubleshooting jaringan TCP/IP*, Penerbit ANDI, Yogyakarta.

Wahyu Gunawan, 2009, *Kebut sehari jadi master PHP*, penerbit Genius, Jakarta.

Wendel Odom, 2004. *Computer Networking first-step*, Penerbit ANDI, Yogyakarta.

Gammu, <http://wammu.eu/gammu/>

Topologi jaringan <http://www.uhb.ac.id/>



FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata 1 Jurusan Teknik Elektro Konsentrasi Teknik Komputer, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : ANDY DA CRUZ
Nim : 09.12.916
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer S-1
Judul : **PENGEMBANGAN SISTEM MONITORING LAN BERBASIS SMS GATEWAY**

No.	Penguji	Tanggal	Uraian	Paraf
1	Penguji I	9 – Agustus 2011	1. DFD di sesuaikan dengan Program 2. BAB IV di perbaiki untuk Gambar	
2	Penguji II	9 – Agustus 2011	1. Perbaiki kalimat pada halaman 50 2. Perbaiki DFD 3. Perbaiki kualitas gambar	

Disetujui :

Dosen Penguji I

Irmalia S. Faradisa, ST, MT
NIP.P.1030000365

Dosen Penguji II

Ahmad Faisol, ST
NIP.P.1030000431

Mengetahui :

Dosen Pembimbing I

SotyoHady, ST
NIP.Y.103970030

Dosen Pembimbing II

Sonny Prasetio, ST, MT
NIP.P.1031000433

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER**



PROPOSAL JUDUL

**PENGEMBANGAN SISTEM MONITORING LAN
GATEWAY BERBASIS SMS**

Disusun oleh :

Nama : Andy Da Cruz

Nim : 0912916

Diperiksa dan disetujui Oleh :
Ketua Jurusan Teknik Elektro

Ir. Yusuf Ismail Nakhoda, MT
NIP.Y.1018800189



PERMOHONAN PERSETUJUAN SKRIPSI

Yang betanda tangan dibawah ini :

Nama : ANDY DA CRUZ
 NIM : 0912916
 Semester : VII
 Fakultas : Teknologi Industri
 Jurusan : Teknik Elektro S-1
 Konsentrasi : TEKNIK ELEKTRONIKA
TEKNIK ENERGI LISTRIK
TEKNIK KOMPUTER DAN INFORMATIKA
TEKNIK KOMPUTER
TEKNIK TELEKOMUNIKASI
 Alamat : JLN. PERUBAHAN R. 2 RW. 8

Dengan ini kami mengajukan permohonan untuk mendapatkan persetujuan untuk membuat *SKRIPSI Tingkat Sarjana*. Untuk melengkapi permohonan tersebut, bersama kami lampirkan persyaratan-persyaratan yang harus dipenuhi.

Adapun persyaratan-persyaratan pengambilan *SKRIPSI* adalah sebagai berikut :

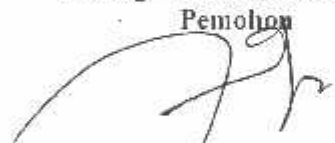
1. Telah melaksanakan semua praktikum sesuai dengan konsentrasinya (.....)
2. Telah lulus dan menyerahkan Laporan Praktek Kerja (.....)
3. Telah lulus seluruh mata kuliah keahlian (MKB) sesuai konsentrasinya (.....)
4. Telah menempuh mata kuliah ≥ 134 sks dengan IPK ≥ 2 dan tidak ada nilai E (.....)
5. Telah mengikuti secara aktif kegiatan seminar skripsi yang diadakan Jurusan (.....)
6. Memenuhi persyaratan administrasi (.....)

Demikian permohonan ini untuk mendapatkan penyelesaian lebih lanjut dan atas perhatiannya kami ucapkan terima kasih.

Telah diteliti kebenaran data tersebut diatas
 Recording Teknik Elektro


Malang, NOV 2010



 (.....)

Pemohon

 (ANDY DA CRUZ)

Disetujui
 Ketua Jurusan Teknik Elektro

Mengetahui
 Dosen Wali


 Ir. Yusuf Ismail Nakhoda, MT
 NIP. Y. 1018800189


 (Ir. YUSUF ISMAIL N. MT.)

Catatan :


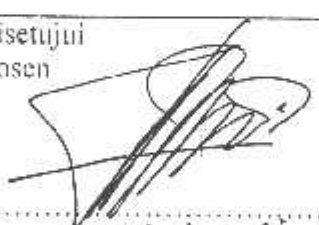
Bagi mahasiswa yang telah memenuhi persyaratan mengambil SKRIPSI agar membuat proposal dan mendapat persetujuan dari Ketua Jurusan/Sekretaris Jurusan T. Elektro S-1

1. IPK 3,935 / 4 = 3,72
2. 126
3. - MK -> Base data, CRB, CBT Embedded RPL



LEMBAR PENGAJUAN JUDUL SKRIPSI JURUSAN TEKNIK ELEKTRO S-1

Konsentrasi : Teknik Energi Listrik / Teknik Elektronika / Teknik Komputer & Informatika / Teknik ~~Komputer~~ / Teknik Telekomunikasi*)

1.	Nama Mahasiswa: <u>ANDY DA CRUZ</u>	Nim: <u>0912916</u>
2.	Waktu Pengajuan	Tanggal: <u>15</u>
		Bulan: <u>12</u>
		Tahun: <u>2010</u>
Spesifikasi Judul (berilah tanda silang)**)		
3.	a. Sistem Tenaga Elektrik	e. Elektronika & Komponen
	b. Energi & Konversi Energi	<input checked="" type="checkbox"/> f. Elektronika Digital & Komputer
	c. Tegangan Tinggi & Pengukuran	g. Elektronika Komunikasi
	d. Sistem Kendali Industri	h. lainnya <u>SISTEM INFORMASI</u>
4.	Konsultasikan judul sesuai materi bidang ilmu kepada Dosen*)	Ketua Jurusan
	<u>Sotyhadi, ST</u>	 Ir. Yusni Ismail Nakhoda, MT NIP. Y. 10/8800189
5.	Judul yang diajukan mahasiswa:	<u>PENGEMBANGAN SISTEM MONITORING LAN GATEWAY BERBASIS SMS</u>
6.	Perubahan judul yang disetujui Dosen sesuai materi bidang ilmu	<u>PENGEMBANGAN SISTEM MONITORING LAN BERBASIS SMS GATEWAY</u>
Catatan:		
7.	Persetujuan Judul skripsi yang dikonsultasikan kepada Dosen materi bidang ilmu	Disetujui Dosen 201...  <u>Sotyhadi, ST</u>

Perhatian:

1. Formulir pengajuan ini harap dikembalikan kepada jurusan paling lambat satu minggu setelah disetujui kelompok dosen keahlian dengan dilampirkan proposal skripsi beserta persyaratan skripsi sesuai form S-1
2. Keterangan: *) Coret yang tidak perlu
**) dilingkari a, b, c, atau g sesuai bidang keahlian

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai dengan permohonan dari Mahasiswa/i :

Nama : Andy da Cruz
Nim : 0912916
Semester : VII
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer

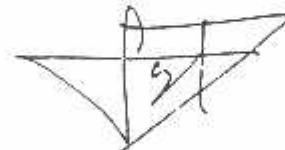
Dengan ini menyatakan Bersedia /~~Tidak Bersedia~~*) Membimbing Skripsi dari Mahasiswa tersebut, dengan Judul :

**PENGEMBANGAN SISTEM MONITORING LAN
BERBASIS SMS GATEWAY**

Demikian surat pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Malang, 22 - Desember - 2010

Kami yang membuat pernyataan,



SONNY PRASETIO, ST, MT
NIP. P.1031000433

Catatan :

Setelah disetujui agar formulir ini diserahkan Mahasiswa/i yang bersangkutan kepada Jurusan untuk diproses lebih lanjut.

*) Coret yang tidak perlu

FORM. S -3b

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai dengan permohonan dari Mahasiswa/i :

Nama : Andy da Cruz

Nim : 0912916

Semester : VII

Jurusan : Teknik Elektro S-1

Konsentrasi : Teknik Komputer

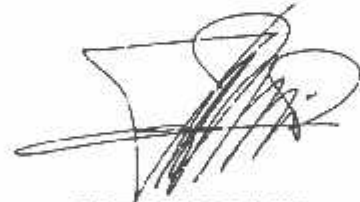
Dengan ini menyatakan Bersedia / Tidak Bersedia *) Membimbing Skripsi dari Mahasiswa tersebut, dengan Judul :

**PENGEMBANGAN SISTEM MONITORING LAN
BERBASIS SMS GATEWAY**

Demikian surat pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Malang, 22 - Desember -- 2010

Kami yang membuat pernyataan,



SOTYOHADI, ST
NIP. Y.103970030

Catatan :

Setelah disetujui agar formulir ini diserahkan Mahasiswa/i yang bersangkutan kepada Jurusan untuk diproses lebih lanjut.





*) Coret yang tidak perlu

FORM. S -3b



BERITA ACARA SEMINAR PROPOSAL SKRIPSI JURUSAN TEKNIK ELEKTRO S-1

Konsentrasi : Teknik Energi Listrik/Teknik Elektronika/Teknik Komputer& Informatika*)

1.	Nama Mahasiswa: <u>ANDY DA CRUZ</u>		Nim. <u>0912916</u>	
2.	Keterangan	Tanggal	Waktu	Tempat
	Pelaksanaan	<u>31-01-11</u>	<u>09.00</u>	Ruang:
Spesifikasi Judul (berilah tanda silang)**)				
3.	a. Sistem Tenaga Elektrik		e. Elektronika & Komponen	
	b. Energi & Konversi Energi		f. Elektronika Digital & Komputer	
	c. Tegangan Tinggi & Pengukuran		g. Elektronika Komunikasi	
	d. Sistem Kendali Industri		h. lainnya <u>SISTEM INFORMASI</u>	
4.	Judul Proposal yang diseminarkan Mahasiswa	<u>PENGEMBANGAN SISTEM MONITORING LAN BERBASIS SMS GATEWAY</u>		
5.	Perubahan Judul yang diusulkan oleh Kelompok Dosen Keahlian			
6.	Catatan:			
	Catatan:			
Persetujuan Judul Skripsi				
7.	Disetujui, Dosen Keahlian I		Disetujui, Dosen Keahlian II	
				
	Mengetahui, Ketua Jurusan		Disetujui, Calon Dosen Pembimbing ybs	
<u>Ir Yusuf Ismail Nakhoda, MT</u> NIP. Y. 1018800189		Pembimbing I	Pembimbing II	
		 <u>Sotyo Hadi, ST</u>	 <u>SANJAY KARBETIO, SHT</u>	

Perhatian

1. Keterangan: *) Coret yang tidak perlu

**) dilingkari a, b, c, atau g sesuai bidang keahlian



FORMULIR BIMBINGAN SKRIPSI

Nama : Andy Da Cruz
NIM : 09.12.916
Masa Bimbingan : 31 January 2011 s/d 31 Juli 2011 *BY*
Judul : Pengembangan Sistem Monitoring LAN berbasis SMS gateway

NO	Tanggal	Uraian	Paraf Pembimbing
1	20/06/11	Konsultasi BAB I dan BAB II	<i>fadi</i>
2	5/07/11	Konsultasi BAB III	<i>fadi</i>
3	13/7/11	Test program	<i>fadi</i>
4	20/7/11	Konsultasi BAB I, II & III revisi	<i>fadi</i>
5	22/7/11	Konsultasi makalah seminar hasil	<i>fadi</i>
6	5/08/11	Konsultasi BAB IV	<i>fadi</i>
7	5/08/11	Konsultasi BAB V	<i>fadi</i>
8			
9			
10			

Malang, 05-08-2011
Dosen Pembimbing I


Sotyo Dady, ST, MT
NIP. Y 103970030

Form S-4b



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA

NIM

Perbaikan meliputi

Andy da Cruz

0512016

- DFD disesuaikan di program

- Bab Impact diperbaiki utl gambar

Malang,

(_____)



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA

Andy Da Cruz

NIM

0912916

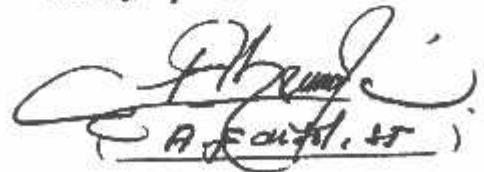
Perbaikan meliputi

Perbaiki kalimat pada hal 50

Perbaiki DFD

Perbaiki layout gambar

Malang, 9-08-2011


A. Fauzi

Monitor auto

```
<?php
session_start();
require_once('../settings/db.php');

//$mode = $_POST['mode'];
$cur = $_POST['param'];

if ( $cur >= 0 ) {

    // get count of gw
    $sql_get_gw = "select * from `nms_gateway`";
    $rs = mysql_query($sql_get_gw);
    $num_gw = mysql_num_rows( $rs );

    if ($cur == $num_gw) {
        echo 'stop';
    } else {

        $sql_next = "select * from `nms_gateway` order by `id_gateway` asc limit ".$cur.", 1";
        $res = mysql_query($sql_next);

        if ( mysql_num_rows( $res ) == 1 ) {

            $gw = mysql_fetch_array($res);
            $alias = $gw['alias'];
            $ip = $gw['ip_gateway'];
            $id = $gw['id_gateway'];

            // new monitoring status
            $new_status = "insert into `nms_status` values(, 0, 0,
            '$id',
            'waiting',
            NOW()
            );";
            $rs = mysql_query($new_status);
            $id_status = mysql_insert_id();

            // ping
            $pingCount = 5;
            $countParam = '-n';
            $ping = 'ping '.$countParam.' '.$pingCount.' '.$ip;

            $ret = exec($ping, $output, $retVal);
```

```

$status = $pingCount;
// cast ping result
for ($i = 0; $i < count($output); $i++) {

    if ( preg_match('/unreachable/i', $output[$i]) > 0 ||
        preg_match('/Request timed out./i', $output[$i]) > 0 ) {
        // save ping result
        $save_result = "insert into `nms_status_detail` values('$id_status', '$output[$i]')";
        mysql_query($save_result);

        $status--;

    } else if (preg_match('/Ping request could not find host/i', $output[$i]) > 0) {

// save ping result
        $save_result = "insert into `nms_status_detail` values('$id_status', '$output[$i]')";
        mysql_query($save_result);

        $status = 0;
        break;
    } else if ( preg_match('/Reply from/i', $output[$i]) > 0 ) {
// save ping result
        $save_result = "insert into `nms_status_detail` values('$id_status', '$output[$i]')";
        mysql_query($save_result);

    } else {
// other reply goes here
    }
}

// DOWN or UP
if ($status == 0) {
    $statusText = 'DOWN';
} else {
    $statusText = 'UP';
}
$sendSMS = 0;

// send SMS Notification
if ( $statusText == 'DOWN' ) {

    // if this gateway never get down before
    $sqlGetDown = "select * from `nms_status` where id_gateway = '$id' and status =
'DOWN'";
    $rsGetDown = mysql_query($sqlGetDown);

```

```

if ( mysql_num_rows($rsGetDown) == 0 ) {
    // notification
    $content = "LAN Monitoring System\n\n";
    $content .= "Following gateway is DOWN!\n";
    $content .= "GW : ".$Salias."\n";
    $content .= "IP : ".$Sip."\n\n";
    $content .= "Please check your configuration soon!";
    $sendSMS = 1;

} else {
    //echo 'cek1'; exit;
    // if prev is UP, then sms
    $sql_cek = "select * from `nms_status` where `id_gateway` = '$id' order by
'id_status' desc limit 1,1";
    $rsCekLastStatus = mysql_query($sql_cek);
    $n = mysql_num_rows($rsCekLastStatus);

    if ($n == 1) {
        //echo 'cek2'; exit;
        $statusGW = mysql_fetch_array($rsCekLastStatus);
        $gw_status = $statusGW['status'];

        if ( $gw_status == 'UP' ) {

            // notification
            $content = "LAN Monitoring System\n\n";
            $content .= "Following gateway is DOWN!\n";
            $content .= "GW : ".$Salias."\n";
            $content .= "IP : ".$Sip."\n\n";
            $content .= "Please check your configuration soon!";
            $sendSMS = 1;

        } else {

            $sqlGetMaxDowntime = "select * from `nms_settings`";
            $rsGetMaxDownTime = mysql_query($sqlGetMaxDowntime);

            $maxDowntime = mysql_fetch_array($rsGetMaxDownTime);
            $maxDowntime = $maxDowntime['maxDowntime'];

            // if down == maxDowntime, then send notif
            $sqlGetGatewayDowntime = "select * from `nms_status` where id_gateway
= '$id' and status <> 'waiting' order by id_status asc";
            $rsDowntime = mysql_query($sqlGetGatewayDowntime);

            if ( mysql_num_rows($rsDowntime) > 0 ) {

```

```

//echo 'eek3'; exit;
$downCnt = 0;
while ($downtime = mysql_fetch_array($rsDowntime)) {
    if ($downtime['status'] == 'UP') { $sendSMS = 2; $downCnt = 0; }
    else
        $downCnt++;
        //$sendSMS = 2;
    }
//echo $downCnt;
if (($downCnt+1) % $maxDowntime == 0) {
    $sendSMS = 1;
    $content = "LAN Monitoring System\n\n";
    $content .= "GW: ".$alias."\n";
    $content .= "IP: ".$ip."\n\n";
    $content .= "Is DOWN for about ".$maxDowntime." times!";
    $content .= 'Please check your configuration soon!';
    //break;
    }
} else {
    //echo 'eek'; exit;
}
}
}
}

if ($sendSMS == 1) {
    $getPhone = "select * from `nms_phonenumber` where active = 1";
    $rsPhone = mysql_query($getPhone);

    if (mysql_num_rows($rsPhone) > 0) {
        while ( $phone = mysql_fetch_array($rsPhone) ) {
            $number = $phone['number'];

            // try sending notification
            $sql_sendSMS =
            "INSERT INTO `nemosys`.`outbox` (`DestinationNumber`
('TextDecoded')".
            " VALUES ('$number', '$content')";

            mysql_query($sql_sendSMS);
        }
    } else {

```