

**Perancangan Sistem *Monitoring Jaringan*
Melalui Perangkat Android
Berbasis *Simple Network Management Protocol***

Artikel Ilmiah

**Diajukan kepada
Fakultas Teknologi Informasi
untuk memperoleh Gelar Sarjana Komputer**



**Peneliti :
Lukas Tri Christanto
NIM : 672010240**

**Program Studi Teknik Informatika
Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Salatiga
Januari 2016**

**Perancangan Sistem Monitoring Jaringan Melalui Perangkat Android
Berdasarkan Simple Network Management Protocol**

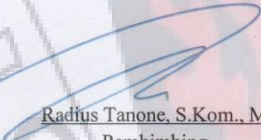
Oleh,

Tri Christanto
NIM : 672010240

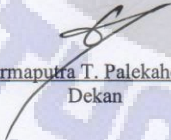
ARTIKEL ILMIAH


Diajukan Kepada Program Studi Teknik Informatika guna memenuhi sebagian dari
persyaratan untuk mencapai gelar Sarjana Komputer

Disetujui oleh,


Radians Tanone, S.Kom., M.Cs.
Pembimbing

Diketahui oleh,


Dr. Dharmaputra T. Palekahelu, M.Pd.
Dekan

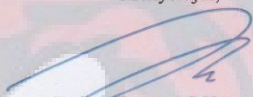

Supriyadi, S.Si., M.Kom.
Ketua Program Studi

**FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN SATYA WACANA
SALATIGA
2016**

Lembar Pengesahan

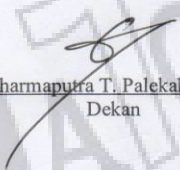
Judul Tugas Akhir : Perancangan Sistem Monitoring Jaringan Melalui Perangkat
Android Berbasis Simple Network Management Protocol.
Nama Mahasiswa : Tri Christanto
NIM : 672010240
Program Studi : Teknik Informatika
Fakultas : Teknologi Informasi


Menyetujui,


Radians Tanone, S.Kom., M.Cs.

Pembimbing

Mengesahkan,


Dr. Dharmaputra T. Palekahelu, M.Pd.
Dekan

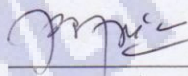
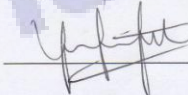

Suprihadi, S.Si., M.Kom.
Ketua Program Studi

1956

Dinyatakan Lulus Ujian tanggal: 3 Februari 2016

Penguji:

1. Wiwin Sulisty, ST., M.Kom.
2. Yos Richard Beeh, ST., M.Cs.



PERNYATAAN PERSETUJUAN AKSES

Saya yang bertanda tangan di bawah ini:

Nama : LUKAS TRI CHRISTANTO
NIM : 672010240 Email : lukasetri.christanto@yahoo.com
Fakultas : TEKNOLOGI INFORMASI Program Studi : TEKNIK INFORMATIKA
Judul tugas akhir : PERANCANGAN SISTEM MONITORING JARINGAN
MELALUI PERANGKAT ANDROID BERBASIS
SIMPLE NETWORK MANAGEMENT PROTOCOL

Dengan ini saya menyerahkan hak *non-eksklusif** kepada Perpustakaan Universitas – Universitas Kristen Satya Wacana untuk menyimpan, mengatur akses serta melakukan pengelolaan terhadap karya saya ini dengan mengacu pada ketentuan akses tugas akhir elektronik sebagai berikut (beri tanda pada kotak yang sesuai):

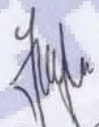
- ☒ a. Saya mengizinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA
- ☐ b. Saya tidak mengizinkan karya tersebut diunggah ke dalam aplikasi Repositori Perpustakaan Universitas, dan/atau portal GARUDA**

* Hak yang tidak terbatas hanya bagi satu pihak saja. Pengajar, peneliti, dan mahasiswa yang menyerahkan hak non-eksklusif kepada Repositori Perpustakaan Universitas saat mengumpulkan hasil karya mereka masih memiliki hak copyright atas karya tersebut.

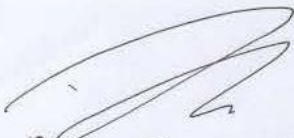
** Hanya akan menampilkan halaman judul dan abstrak. Pilihan ini harus dilampiri dengan penjelasan/ alasan tertulis dari pembimbing TA dan diketahui oleh pimpinan fakultas (dekan/kaprodi).

Demikian pernyataan ini saya buat dengan sebenarnya.

Salatiga, 23 februari 2016


Lukas Tri Christanto
Tanda tangan & nama terang mahasiswa

Mengetahui,


RADIUS TANONE, S.Kom., M.Cs.
Tanda tangan & nama terang pembimbing I

Tanda tangan & nama terang pembimbing II



PERPUSTAKAAN UNIVERSITAS
UNIVERSITAS KRISTEN SATYA WACANA
Jl. Diponegoro 52 – 60 Salatiga 50711
Jawa Tengah, Indonesia
Telp. 0298 – 321212, Fax. 0298 321433
Email: library@adm.uksw.edu ; http://library.uksw.edu

PERNYATAAN TIDAK PLAGIAT

Saya yang bertanda tangan di bawah ini:

Nama : LUKAS TRI CHRISTANTO
NIM : 672010240 Email : lukas.tri.christanto@yahoo.com
Fakultas : TEKNOLOGI INFORMASI Program Studi : TEKNIK INFORMATIKA
Judul tugas akhir : PERANCANGAN SISTEM MONITORING JARINGAN
MELALUI PERANGKAT ANDROID BERBASIS
SIMPLE NETWORK MANAGEMENT PROTOCOL
Pembimbing : 1. Radius Tanong, S.Kom., M.Cs.
2. _____

Dengan ini menyatakan bahwa:

1. Hasil karya yang saya serahkan ini adalah asli dan belum pernah diajukan untuk mendapatkan gelar kesarjanaan baik di Universitas Kristen Satya Wacana maupun di institusi pendidikan lainnya.
2. Hasil karya saya ini bukan saduran/terjemahan melainkan merupakan gagasan, rumusan, dan hasil pelaksanaan penelitian/implementasi saya sendiri, tanpa bantuan pihak lain, kecuali arahan pembimbing akademik dan narasumber penelitian.
3. Hasil karya saya ini merupakan hasil revisi terakhir setelah diujikan yang telah diketahui dan disetujui oleh pembimbing.
4. Dalam karya saya ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali yang digunakan sebagai acuan dalam naskah dengan menyebutkan nama pengarang dan dicantumkan dalam daftar pustaka.

Pernyataan ini saya buat dengan sesungguhnya. Apabila di kemudian hari terbukti ada penyimpangan dan ketidakbenaran dalam pernyataan ini maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya saya ini, serta sanksi lain yang sesuai dengan ketentuan yang berlaku di Universitas Kristen Satya Wacana.

Salatiga, 23 Februari 2016


Lukas Tri Christanto
Tanda tangan & nama terang mahasiswa





FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS KRISTEN SATYA WACANA
Jalan Diponegoro 52 – 60
Phone. (0298) 321212 (Hunting)
Fax. (0298) 321433
E-mail: fu@uksw.edu
Salatiga 50711 – INDONESIA



LEMBAR PERSETUJUAN PUBLISH JURNAL

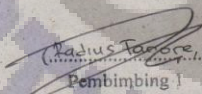
Dengan mempertimbangkan isi dari jurnal mahasiswa :

Nama Mahasiswa : Tri Chrintanto
NIM : 672010240

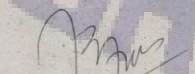
Maka jurnal ini dinyatakan :

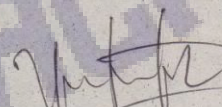
~~LAYAK TERBIT~~ / TIDAK LAYAK TERBIT

Menyetujui,


Pembimbing 1

(.....)
Pembimbing 2


Mengetahui,
Penguji 1


Penguji 2

**Perancangan Sistem *Monitoring Jaringan*
Melalui Perangkat Android
Berbasis *Simple Network Management Protocol***

¹⁾ Lukas Tri Christanto, ²⁾ Radius Tanone

Program Studi Teknik Informatika
Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Jl. Diponegoro 52-60, Salatiga 50711, Indonesia
Email: ¹⁾lukastrichristianto@yahoo.com, ²⁾radius.tanone@staff.uksw.edu

Abstract

Network monitoring activity is needed to help administrator to manage the network. The technology development of smartphone which is based on Android can support the network monitoring activity combined with simple network management protocol. This observation's objective is to built a monitoring application that uses router to display the information about network device. This observation is using waterfall method. The built application will be displaying information such as traffic data, network interface, bandwidth usage, and resource detail. The trial's result shows that the application is capable to display information on smartphone well. This is indicated by the passage of monitoring system on Android smartphone.

Keyword : SNMP, Computer Network, Android

Abstrak

Kegiatan *monitoring* jaringan diperlukan untuk mempermudah *administrator* dalam mengelola jaringan. Perkembangan teknologi *smartphone* berbasis Android dapat mendukung kegiatan *monitoring* jaringan yang dipadukan dengan teknologi *simple network management protocol*. Penelitian ini dirancang aplikasi *monitoring* dengan memanfaatkan *router* untuk menampilkan informasi mengenai perangkat jaringan. Penelitian ini menggunakan metode *waterfall*. Aplikasi yang dihasilkan mampu menampilkan informasi perangkat jaringan, seperti *traffic data*, *network interface*, *bandwidth usage*, dan *resource detail*. Hasil uji coba menunjukkan bahwa aplikasi dapat menampilkan objek-objek yang dimonitor pada *smartphone* dengan baik. Hal ini ditunjukkan dengan berjalannya sistem *monitoring* pada *smartphone* android.

Kata Kunci : SNMP, Jaringan Komputer, Android

¹⁾ Mahasiswa Fakultas Teknologi Informasi Jurusan Teknik Informatika, Universitas Kristen Satya Wacana Salatiga.

²⁾ Staff Pengajar Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana Salatiga.

1. Pendahuluan

Pada saat ini jaringan komputer telah berkembang sangat pesat. Kehadiran jaringan komputer memudahkan manusia dalam melakukan segala aktivitas. Untuk melakukan *monitoring* antara *administrator* dan elemen jaringan diperlukan sebuah komunikasi. Komunikasi tersebut berguna agar seorang *administrator* dapat mengetahui kondisi jaringan. Komunikasi yang dilakukan oleh *administrator* adalah bertanya pada elemen jaringan, kemudian elemen jaringan tersebut memberitahu kepada *administrator* tentang informasi yang ada dalam jaringan tersebut.

Kegiatan *monitoring* jaringan diperlukan protokol aplikasi TCP/IP yaitu SNMP (*Simple Network Management Protocol*). Pada tahap ini *monitoring* jaringan merupakan proses pengumpulan, analisis dan pelaporan data mengenai performa dan komponen-komponen pada sebuah jaringan. Data yang dikumpulkan oleh protokol SNMP berupa statistik kinerja dan informasi yang dapat diberikan pada *administrator* berupa *network interface*, *traffic list data*, *bandwidth usage*, dan *resource detail* melalui *router*.

Pada penelitian ini, android digunakan karena mobilitas jaringan *internet* yang tinggi dan didukung adanya teknologi SNMP, serta pengembangan aplikasi pada perangkat android sebagai perangkat bergerak yang sangat umum digunakan, maka dibangun suatu sistem yang dapat melakukan pemantauan jaringan dengan memadukan kedua teknologi tersebut. Pada penelitian ini digunakan perangkat *smartphone* android untuk melakukan pemantauan jaringan. Pembuatan aplikasi pada perangkat android merupakan hal yang baik, karena dengan memanfaatkan android kegiatan apapun dapat dilakukan termasuk salah satunya adalah *monitoring* jaringan. Penelitian ini juga ditujukan untuk melakukan pemantauan *router* pada jaringan lokal secara *real-time* menggunakan protokol SNMP.

Berdasarkan pada latar belakang masalah tersebut, maka dilakukan penelitian untuk merancang sebuah aplikasi *monitoring* pada perangkat *smartphone* android berbasis *Simple Network Management Protocol*. Sistem ini diharapkan akan mempermudah *administrator* jaringan dalam mengetahui status suatu perangkat yang saling terhubung dalam jaringan melalui perangkat *smartphone* android.

Mengingat luasnya cakupan penelitian yang akan dilakukan maka perlu adanya batasan masalah agar hasil penelitian lebih sesuai dan terarah. Penelitian ini akan memberikan informasi tentang kondisi jaringan yang akan dimonitor seperti *network interface*, *traffic list data*, *bandwidth usage*, dan *resource detail* jaringan. Kegiatan *monitoring* yang dilakukan hanya terbatas pada jaringan lokal saja, atau lebih tepatnya membangun jaringan sendiri kemudian melakukan *monitoring*. Penelitian ini hanya membahas cara kerja dan hasil monitoring yang dilakukan, keamanan dan *troubleshooting* jaringan pada penelitian ini tidak dibahas.

2. Tinjauan Pustaka

Pada penelitian yang pertama yang dilakukan oleh firman arifin, dkk yang berjudul “*Sistem Monitoring Traffic Paket Internet Melalui SMS berbasis Simple Network Management Protocol*”. Pada penelitian tersebut dibuat *software* untuk memonitor *traffic packet internet*. *Data traffic* tersebut ditampilkan melalui SMS (*Short Message Service*) yang berisi informasi paket *internet* yang masuk dan keluar dalam sebuah jaringan[1]. Pada penelitian yang lain yang dilakukan oleh Ary Mazharuddin Shiddiqi dan Andhika Panji Nugraha yang berjudul “*Sistem Monitoring Jaringan Dengan Protokol SNMP Menggunakan Piranti Bergerak*”. Penelitian tersebut membahas tentang pembangunan sistem manajemen jaringan secara *online* berbasis PHP yang memanfaatkan protokol SNMP pada perangkat bergerak android[2].

Penelitian yang akan dilakukan ini memiliki perbedaan dengan penelitian sebelumnya. Penelitian pertama hasil ditampilkan berupa sebuah SMS (*Short Message Service*) berisi tentang informasi jaringan yang telah dimonitor. Hasil yang ditampilkan berupa trafik data dan utilitas layanan *internet* yang sudah diatur dalam periode waktu tertentu. Cara untuk mendapatkan informasi menggunakan dua metode yaitu *program sms* mengirim data SNMP ke *administrator* setiap periode waktu tertentu, dan yang kedua dengan cara *program sms* mengirim data SNMP ke *user* atau pengguna ponsel jika *user* melakukan *request* kepada *sms-gateway*. Sedangkan pada penelitian yang akan dilakukan hasil yang dimonitor akan ditampilkan dalam bentuk aplikasi pada perangkat android. Informasi yang ditampilkan berupa *network interface*, *traffic list data*, *bandwidth usage*, dan *resource detail* yang informasinya diambil melalui *router*.

Pada penelitian yang kedua *router* yang digunakan adalah menggunakan cisco, hasil yang dimonitor ditampilkan melalui *internet browser* pada *smartphone* android. Hasil pada perancangan ini adalah sebuah sistem yang menampilkan informasi mengenai keadaan *router* dalam jaringan. Objek-objek yang dipantau berupa nomor *serial router*, suhu, dan memori yang tersedia. Hasil tersebut ditampilkan secara *real-time* pada perangkat android melalui *internet-browser*. Sedangkan pada penelitian yang akan dilakukan adalah menggunakan *router* mikrotik. Hasil yang ditampilkan juga sama dengan penelitian kedua, tetapi pada penelitian ini informasi mengenai keadaan *router* ditampilkan lebih banyak, dan diakses melalui sebuah aplikasi.

SNMP (*Simple Network Management Protocol*) adalah protokol yang bersifat *connectionless* dikarenakan opsionalnya yang menggunakan UDP (*User Datagram Protocol*) . Konsekuensinya adalah tidak ada jaminan lalu lintas manajemen diterima oleh entitas lain dengan sempurna. Model ini memiliki kelebihan yaitu *overhead* proses dapat dikurangi dan diperoleh kesederhanaan[2]. MIB (*Management Information Base*) berfungsi sebagai struktur *database* variabel elemen jaringan yang dikelola. MIB memiliki struktur bersifat hierarki yang diatur sedemikian

rupa sehingga informasi nilai setiap variabel dengan mudah diketahui maksudnya[2]. SNMP adalah sebuah protokol *under-valued*. SNMP digunakan untuk mengatur atau mengontrol elemen-elemen jaringan seperti *host*, *routers*, *switch*, *firewall*, dan lainnya. SNMP berjalan pada UDP sehingga tidak memberikan beban berat pada jaringan atau pada peralatan yang menggunakannya[3]. Secara konseptual, variable SNMP diatur dalam *tree structure*. Setiap peralatan yang berbeda, memerlukan set variable yang berbeda untuk mendeskripsikannya dan mengontrolnya. Kumpulan variable yang berkaitan biasanya spesifik terhadap peralatan dan didefinisikan dalam modul MIB yang membentuk *subtree* dari *tree*. Seluruh *abstract tree* tersebut disebut *Management Information Base*[3].

Jaringan komputer adalah sekelompok komputer yang saling berkomunikasi menggunakan protokol komunikasi melalui media komunikasi sehingga dapat berbagi data, informasi, *program* aplikasi dan perangkat keras lainnya, serta memungkinkan untuk saling berkomunikasi secara elektronik[4]. Jaringan komputer dapat dikelompokkan berdasarkan luas yang dijangkau. Secara umum jaringan komputer dapat dibedakan menjadi beberapa jenis yaitu *local area network*, *metropolitan area network*, dan *wide area network*. *Local area network* adalah jaringan yang dibatasi oleh area yang relatif kecil dan umumnya berada pada lingkungan kecil seperti kantor pada sebuah gedung atau tiap-tiap ruangan pada universitas atau instansi lainnya[4]. Konsep awal dari jaringan komputer adalah komunikasi data dari satu komputer ke komputer yang lain. Untuk pertama kalinya komunikasi data antar komputer tersebut hanyalah bersifat *point-to-point* jadi hanya ada dua komputer terhubung. Setelah lama berkembang konsep tersebut dikembangkan sehingga pertukaran data yang saat itu bersifat pertukaran antar dua komputer berubah menjadi jaringan komputer[5].

Pertukaran informasi yang diatur dengan baik, informasi mengenai kondisi suatu jaringan dapat diambil dan kemudian digunakan untuk dianalisis. Informasi ini sangat berguna bagi para manajer jaringan untuk melakukan pengaturan kinerja jaringan, melakukan perbaikan jika ada masalah, atau bahkan dapat digunakan untuk merencanakan perkembangan jaringannya[6].

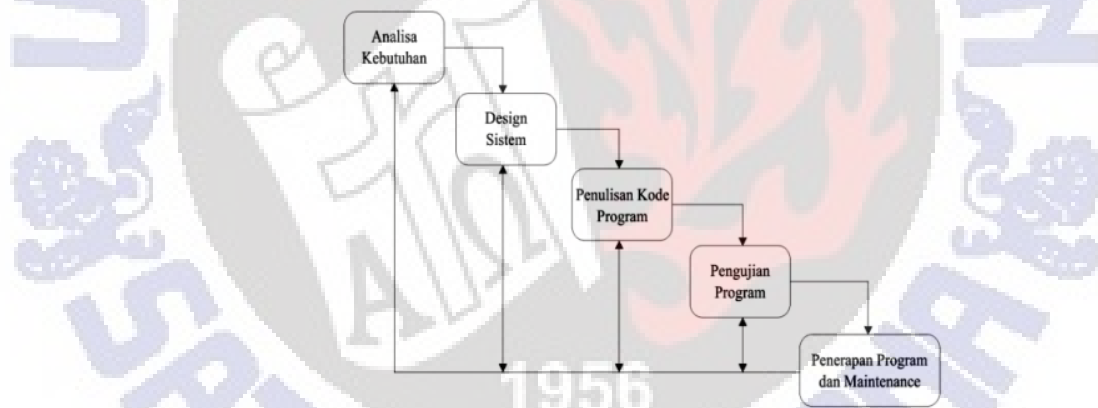
Manajemen information base adalah sekumpulan objek pada jaringan yang dapat dikelola dengan menggunakan protokol SNMP. Pada sebuah MIB juga didapat sebuah *Object Identifier* atau OID yang merupakan sebuah entitas manajemen yang merujuk pada objek tertentu pada suatu jaringan[7].

Android menyediakan *platform* terbuka bagi para pengembang untuk membuat aplikasi mereka sendiri. Pada awalnya dikembangkan oleh *Android Inc*, sebuah perusahaan pendatang baru yang membuat perangkat lunak untuk ponsel yang kemudian dibeli oleh *Google Inc*. Pada pengembangannya, dibentuklah *Open*

Handset Alliance (OHA), konsorsium dari 34 perusahaan perangkat keras, perangkat lunak, dan telekomunikasi termasuk *Google, HTC, Intel, Motorola, Qualcomm, T-Mobile*, dan *Nvidia*[8]. Perangkat android secara garis besar dibagi menjadi beberapa arsitektur. Arsitektur yang pertama berupa *application* dan *widgets* yaitu layer dimana berhubungan dengan aplikasi saja. Kedua *application frameworks* yaitu para pembuat aplikasi melakukan pengembangan aplikasi yang akan dijalankan di sistem operasi Android. Ketiga *libraries* yaitu layer dimana fitur-fitur android berada, biasanya pembuat aplikasi mengakses *libraries* untuk menjalankan aplikasi. Keempat *Android run-time layer*, dimana aplikasi android dapat dijalankan dan prosesnya memakai implementasi Linux. Terakhir adalah *Linux Kernel* yaitu layer dimana inti dari sistem operasi Android berada[9].

3. Metode Alur Penelitian

Metode yang digunakan dalam penelitian ini adalah metode *waterfall*. Metode *waterfall* merupakan salah satu metode yang sistematis dan konvensional yang diawali dengan tahap analisis kebutuhan, desain sistem, pengkodean, pengujian program, dan *maintenance*.



Gambar 1 Metode Waterfall [9]

Metode *waterfall* juga dikenal sebagai *linier sequential* atau *classic life cycle*. Pada penelitian ini metode *waterfall* melingkupi aktivitas-aktivitas seperti:

1. Analisis

Pada tahap ini penulis merencanakan konsep dasar tentang perancangan sistem yang akan dibangun.

2. Desain Sistem

Pada tahap desain, penulis akan melakukan perancangan tampilan aplikasi dan merancang jaringan yang akan dibangun.

3. Pengkodean

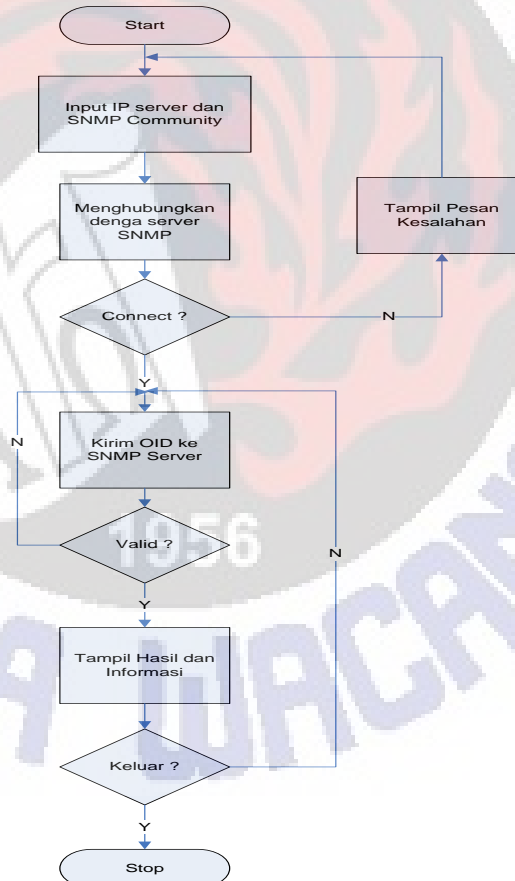
Setelah melakukan desain, tahap selanjutnya adalah menulis kode program. Pada tahap ini penulis menggunakan bahasa pemrograman *java* untuk merancang aplikasi.

4. Pengujian Sistem

Tahap selanjutnya adalah melakukan uji sistem. Pengujian dilakukan dengan menghubungkan perangkat Android dengan jaringan komputer yang akan dimonitor. Selanjutnya adalah melakukan analisa dari hasil yang sudah dimonitor.

5. Penerapan program dan *maintenance*

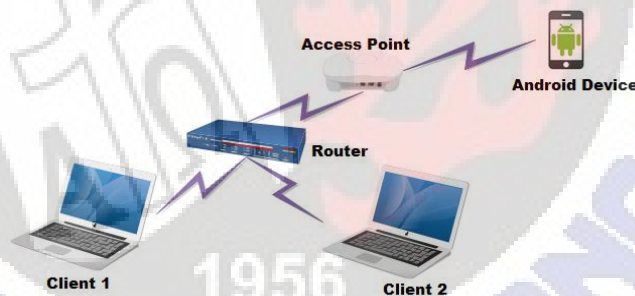
Tahap ini adalah tahap dimana aplikasi atau sistem yang sudah berjalan dikembangkan lagi dan menarik sebuah kesimpulan apakah aplikasi atau sistem ini sudah bekerja dengan baik atau sebaliknya.



Gambar 2 Flowchart kinerja aplikasi

Pada penelitian ini juga dijelaskan *flowchart* kinerja aplikasi. *Flowchart* digunakan untuk memperlihatkan urutan atau hubungan antar proses beserta instruksinya.

1. Pertama *user* diminta untuk menginputkan alamat *IP* dari *server* tujuan dan *SNMP community* dari *server*
2. Kemudian sistem membuka koneksi dengan *SNMP server* tujuan
3. Jika koneksi berhasil maka, sistem akan melanjutkan ke proses selanjutnya. Sedangkan jika koneksi tidak berhasil, maka sistem akan menampilkan pesan kesalahan, dan kemudian *user* diharuskan untuk melakukan pengisian pada proses ke satu.
4. Pada proses selanjutnya sistem mengirimkan *Obejct Identifier* atau *OID* sesuai dengan menu yang dipilih oleh *user*. Setelah itu kemudian sistem mengirimkan *OID* ke *SNMP server* tujuan.
5. Sistem memeriksa hasil balasan dari *server* tujuan, apakah hasil yang diberikan valid atau tidak. Jika hasil tidak valid maka proses berulang pada langkah empat.
6. Setelah sistem menerima hasil valid dari *server* maka sistem akan menampilkan hasil dan informasi berdasarkan *OID* yang sudah dikirimkan pada langkah ke empat.
7. Setelah itu jika *user* hendak melanjutkan proses, maka proses akan berulang ke langkah empat, jika tidak maka proses akan berakhir dan selesai.



Gambar 3 Arsitektur Perancangan Sistem

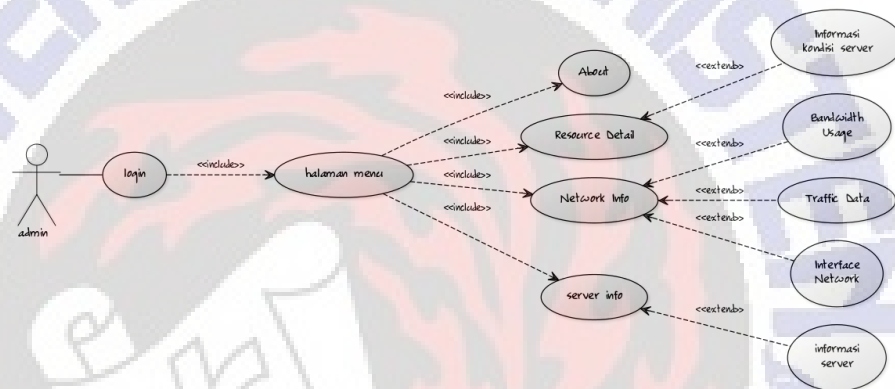
Pada Gambar 3 adalah gambaran dari arsitektur perancangan sistem. Perancangan ini berada pada jaringan lokal yang mempunyai terdiri dari *router*, *access point*, 2 *client server* dan perangkat android. Konfigurasi alamat *IP* akan dijelaskan pada Tabel 1.

Tabel 1 Konfigurasi alamat *IP* pada masing-masing perangkat

-
- | | |
|---|--|
| - | <i>Client 1</i> 192.168.220.22 /24 |
| - | <i>Client 2</i> 192.168.230.22/24 |
| - | <i>Ip Mikrotik</i> 192.168.88.1/24 |
| - | <i>Access point</i> 192.168.88.22/24 |
| - | <i>Android device</i> mendapat <i>IP DHCP</i> dari <i>Access point</i> |
-

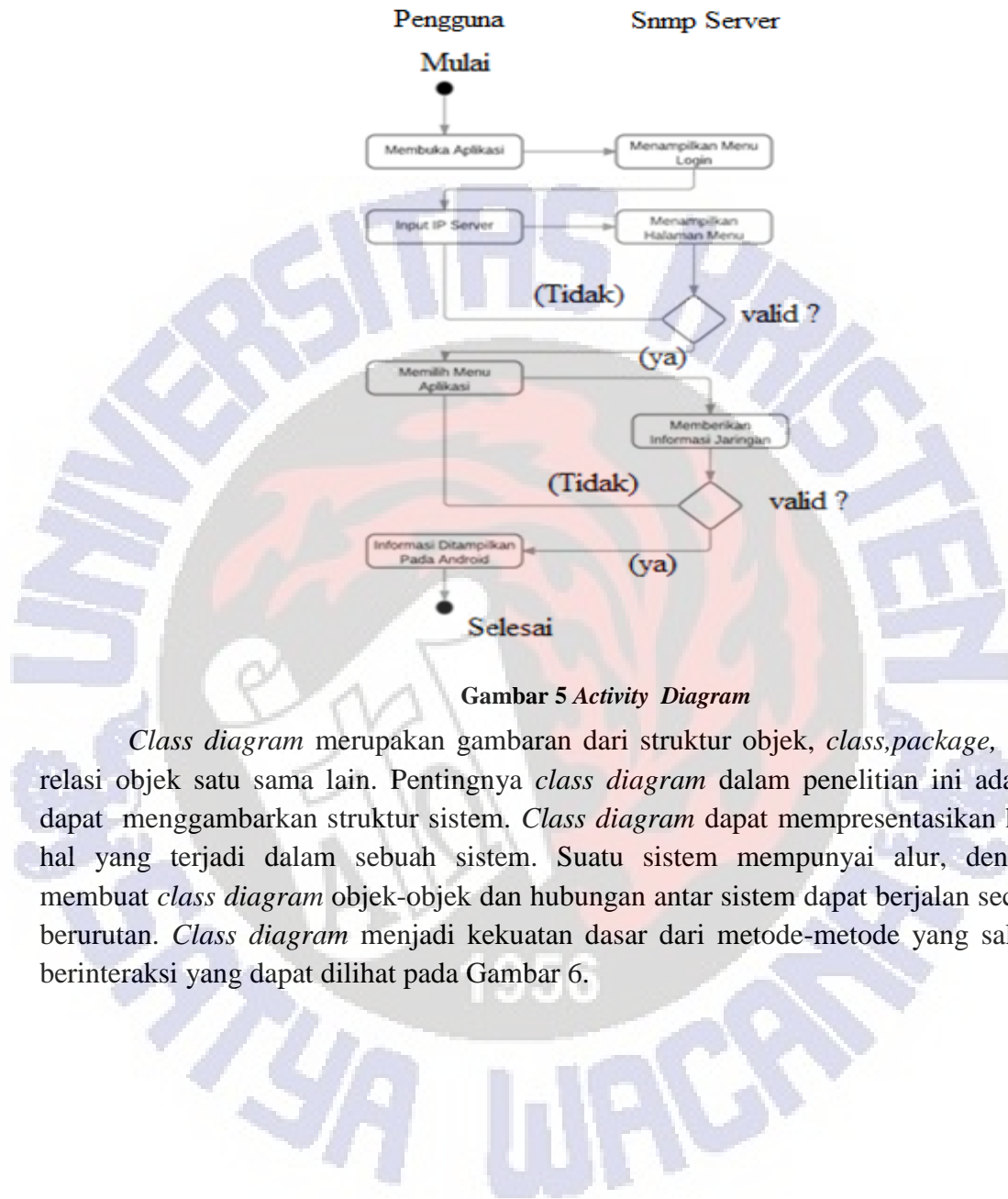
Pada penelitian ini, dilakukan perancangan sistem perangkat lunak untuk menggambarkan prosedur dan proses kerja dari sistem aplikasi. Sistem dibangun menggunakan UML (*Unified Modeling Language*) yang terdiri dari *use case diagram*, *activity diagram* dan *class diagram*.

Use case diagram, pengguna aplikasi android dapat memilih menu yang disediakan pada aplikasi. Didalam aplikasi menu terdapat *server info*, *network info*, *resource*, dan *about*. Jika salah satu dipilih maka akan menampilkan informasi tentang perangkat jaringan yang dimonitor. *Use case diagram* dapat dilihat pada Gambar 4.



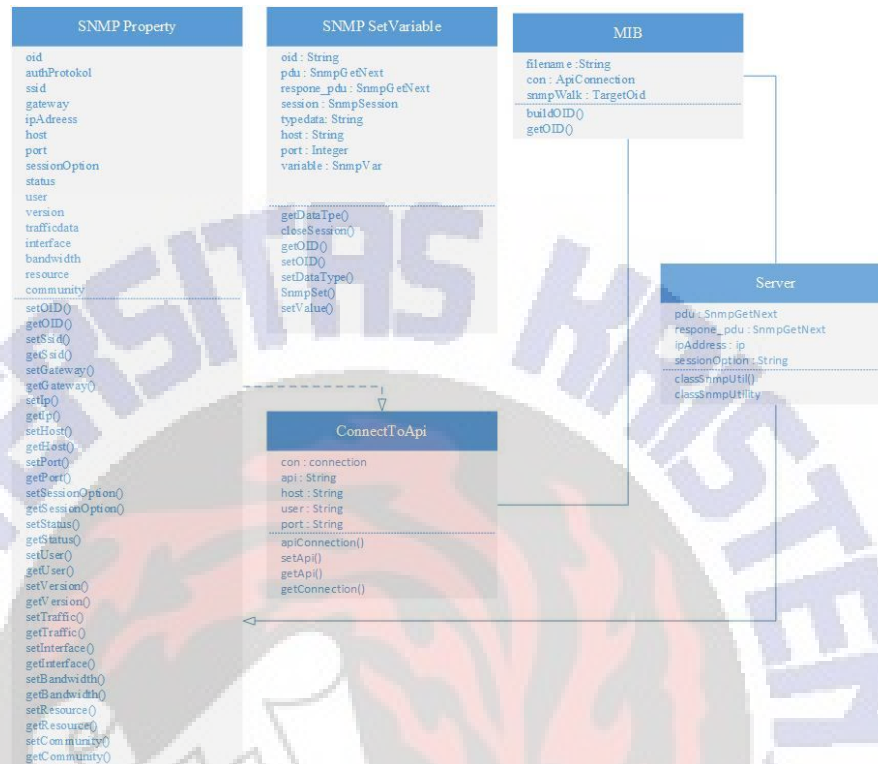
Gambar 4 Use Case Diagram

Activity diagram, pengguna memulai proses dengan membuka aplikasi yang terdapat pada perangkat android. Pengguna membuka dan SNMP *server* atau sistem akan menampilkan tampilan menu *login*. Setelah menu *login* tampil, pengguna memasukkan IP *server* yang akan dimonitor. Ketika pengguna sudah memasukkan IP *server*, sistem akan menampilkan halaman menu aplikasi, apabila terjadi kesalahan halaman menu aplikasi tidak akan muncul, dan sistem memerintahkan pengguna untuk memasukkan IP *server* dengan benar. Setelah masuk pada halaman menu, pengguna dapat memilih menu yang tersedia pada aplikasi android. Setelah pengguna memilih salah satu menu yang tersedia, maka sistem akan menampilkan informasi jaringan sesuai dengan yang dipilih pengguna. Jika informasi yang ditampilkan ada kesalahan, sistem akan mengembalikan tampilan ke halaman menu aplikasi. Apabila pengguna sudah melihat informasi yang ada, pengguna dapat menutup aplikasi dan sistem akan berhenti. *Activity diagram* dapat dilihat pada Gambar 5.



Gambar 5 Activity Diagram

Class diagram merupakan gambaran dari struktur objek, *class*, *package*, dan relasi objek satu sama lain. Pentingnya *class diagram* dalam penelitian ini adalah dapat menggambarkan struktur sistem. *Class diagram* dapat mempresentasikan hal-hal yang terjadi dalam sebuah sistem. Suatu sistem mempunyai alur, dengan membuat *class diagram* objek-objek dan hubungan antar sistem dapat berjalan secara berurutan. *Class diagram* menjadi kekuatan dasar dari metode-metode yang saling berinteraksi yang dapat dilihat pada Gambar 6.



Gambar 6 Class Diagram

Konfigurasi untuk menampilkan *traffic list data* pada jaringan dapat dilihat pada kode program 1.

Kode Program 1 *traffic list data*

```

community = "public";
targetOid = ".1.3.6.1.2.1.4.20.1.1
sb.append("TRAFFIC LIST:\n");
ret = snmpWalk(ip, community, targetOid);
targetOid = ".1.3.6.1.2.1.2.2.1.10
retIN = snmpWalk(ip, community, targetOid);
targetOid = ".1.3.6.1.2.1.2.2.1.16
retOut = snmpWalk(ip, community, targetOid);
  
```

OID adalah rumus untuk menampilkan data yang sesuai dengan perintah SNMP. OID adalah sebuah objek khusus yang ditujukan untuk menampilkan informasi perangkat jaringan. Hasil dari target OID pada kode program 1 dapat dilihat pada gambar 9. *Traffic list data* merupakan sebuah ukuran *transfer data* yang terjadi dalam lalu lintas jaringan. Jumlah *traffic* yang terjadi dalam jaringan yang dibangun adalah 3 buah *traffic*, dimana dibagi dalam dua buat paket data, paket masuk dan keluar. Manfaat yang dapat diambil dengan melihat *traffic* data ini adalah mengidentifikasi beban *traffic* yang keluar dan masuk sehingga performa *server* atau *router* dapat dikontrol dengan baik.

Kode Program 2 *Bandwidth Usage*

```
ip = IP;
community = "public";
targetOid = ".1.3.6.1.2.1.2.2.1.2";
targetOid = ".1.3.6.1.2.1.4.20.1.1";
sb.append("BANDWIDTH  USAGE LIST:\n");
ret = snmpWalk(ip, community, targetOid);
targetOid = ".1.3.6.1.2.1.2.2.1.10";
retIN = snmpWalk(ip, community, targetOid);
targetOid = ".1.3.6.1.2.1.2.2.1.16";
retOut = snmpWalk(ip, community, targetOid);
targetOid = ".1.3.6.1.2.1.2.2.1.5";
retSpeed = snmpWalk(ip, community, targetOid);
```

Bandwidth usage pada perancangan ini mengacu pada data yang lewat dalam jaringan melalui perangkat-perangkat yang saling terhubung. *Bandwidth* yang ditampilkan berdasar perhitungan *upload* dan *download* data yang ada pada jaringan. Pada Kode program 2 menunjukkan asumsi atau rumus yang digunakan untuk menampilkan besaran *bandwidth* yang tersedia pada jaringan. Acuan informasi yang ditampilkan merujuk pada rumus OID yang dipanggil sesuai dengan perangkat.

Besar kecilnya *bandwidth* berpengaruh terhadap aplikasi yang dijalankan. Beberapa faktor dapat mempengaruhi naik turunnya *bandwidth* pada jaringan ini, misalnya sering atau tidaknya melakukan akses data.. Hasil pada perancangan ini menunjukkan aktivitas *bandwidth* dengan frekuensi rendah dan tinggi. Kedua frekuensi tersebut menunjukkan aktivitas pengiriman data berjalan dengan baik sesuai dengan informasi yang dapat dikirim melalui koneksi jaringan.

Server information berisi tentang informasi *server* atau *router* yang menghubungkan berbagai komputer pada sebuah jaringan yang sedang dimonitoring. Pada perintah ini hasil yang didapatkan berupa alamat IP *server* atau *router* yang dipakai, nama *router*, lokasi *router* berada, *uptime data* yang digunakan, serta *operating system router* yang digunakan.

Kode Program 3 Server Information

```
@Override

protected String doInBackground(String... url1) {

    int count;

    try {

        Log.d("SNMP.Log", url1[0] + "");

        SnmpUtility util = new
        SnmpUtility(SnmpUtility.VERSION_1,IP);

        List<VariableBinding> vbs = util.walk(new
        OID("1.3.6.1.2.1.1"),COMM);

        StringBuffer sb = new StringBuffer("Host Info :\n");

        for (VariableBinding vb : vbs) {

            System.out.println(vb.getVariable().toString());

            strHasil.add(vb.getVariable().toString());

        }return null;

    }
```

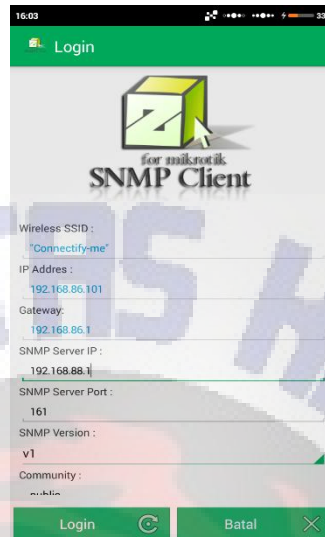
SNMP Versi 1 memiliki beberapa protokol data unit salah satu nya adalah *GetNextRequest*. Fungsi dari protokol data unit *GetNextRequest* dapat digunakan secara interatif untuk mendapatkan sekuen dari informasi manajemen yang dimonitor. Untuk menjalankan fungsi *GetNextRequest* dapat dijelaskan pada Kode Program 4.

Kode Program 4 Perintah untuk memanggil Protokol data Unit *GetNextRequest*

```
PDU requestPDU = new PDU();
requestPDU.add(new VariableBinding(oid));
requestPDU.setType(PDU.GETNEXT);
```

4. Hasil dan Pembahasan

Konfigurasi dan pengkodean telah dilakukan, maka proses selanjutnya adalah tahap pengujian. Pada tahap pengujian dilakukan uji konektivitas dari sistem yang sudah dirancang. Pengujian ini langsung memakai perangkat android yang sudah terhubung pada jaringan lokal yang akan dimonitor. Proses pengujian dilakukan dari awal bagaimana perangkat android dapat berkomunikasi dengan *router* sebagai *base* informasi. Pengujian sistem akan dijelaskan dalam bentuk gambar.



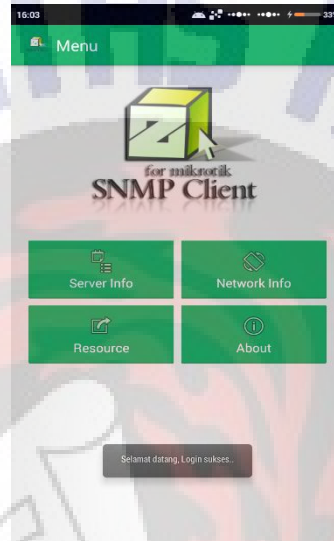
Gambar 7 Halaman Login

Pada tampilan aplikasi *login* ada beberapa bagian-bagian yang digunakan untuk menampilkan informasi yang diambil yang dapat dilihat pada Gambar 7. Tampilan menu login dijelaskan melalui tabel 2.

Tabel 2 Alokasi Menu Login

Informasi	Keterangan
<i>Wireless SSID</i>	Nama <i>Access Point</i> yang terhubung
<i>IP Address</i>	Alamat ip yang diterima oleh <i>access point</i>
<i>Gateway</i>	Alamat IP dari <i>access point</i>
<i>SNMP Server IP</i>	Alamat IP dari <i>mikrotik router</i>
<i>SNMP Server Port</i>	<i>Port</i> dari <i>services SNMP</i>
<i>SNMP Version</i>	Versi <i>SNMP</i> pada <i>router</i>
<i>SNMP Community</i>	<i>Username</i> dari <i>SNMP</i>

Pada tombol *login* dibuat *file class activity*, dimana *file class activity* tersebut digunakan untuk mendefinisikan komponen-komponen yang digunakan untuk membuat halaman *login*. Untuk mengakses tombol *login* pada tampilan aplikasi tersebut, sudah diberi perintah sesuai dengan IP *server* terhubung pada jaringan yang akan dimonitor.

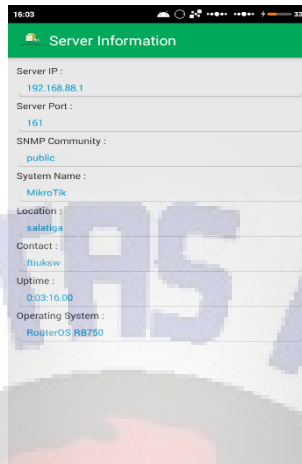


Gambar 8 Halaman Menu

Halaman menu memperlihatkan beberapa pilihan diantaranya menu *Server Info*, *Network Info*, *Resource Detail*, dan *About* yang dapat dilihat pada Gambar 8. Pada tabel 3 menjelaskan fungsi dari masing-masing pilihan yang ada pada tampilan menu aplikasi.

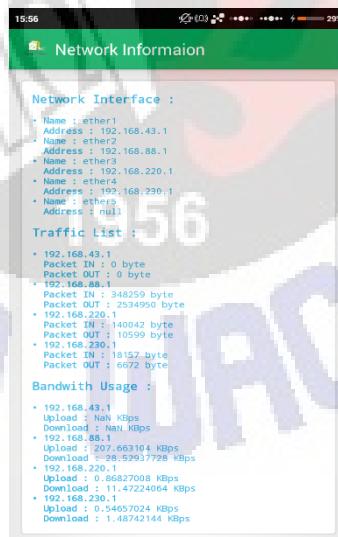
Tabel 3 Alokasi Menu Aplikasi

Informasi	Keterangan
<i>Server Info</i>	Informasi mengenai server yang tersambung
<i>Network Info</i>	Informasi mengenai koneksi dan <i>bandwith</i> pada jaringan
<i>Resource Detail</i>	Penggunaan sumber daya pada <i>router</i>
<i>About</i>	Informasi tentang aplikasi



Gambar 9 Halaman Menu *Server Information*

Pada halaman *server information* dijelaskan mengenai Server IP yaitu alamat IP yang ditujukan untuk diambil informasinya, pada penelitian ini diambil IP 192.168.88.1 dimana IP tersebut dideklarasikan pada perangkat *router* yang terhubung pada jaringan. *Port* yang digunakan adalah port 161 dengan *public* sebagai deklarasi SNMP. *Sistem name* yang muncul adalah mikrotik, karena pada jaringan yang dimonitoring memakai *router* mikrotik dengan menggunakan sistem operasi RouterOs RB750. Lokasi dan *Contact* hanya penjelasan letak *router* berada yang dapat dilihat pada Gambar 9.



Gambar 10 Tampilan Menu *Network Information*

Halaman Menu *Network Information* menjelaskan tentang informasi jaringan yang sedang diambil informasinya. Pada halaman *network information* terdapat tiga informasi yang dapat ditampilkan yang dapat dilihat pada Gambar 10. *Network interface* berisi informasi tentang *port* dalam *router* yang terhubung pada jaringan.

Ether-1 Gateway biasanya digunakan untuk akses ke jaringan publik, *Ether-2-master-local* digunakan untuk menghubungkan *router* dengan laptop sebagai pengganti *access point*. *Ether-3-slave-local* dan *Ether-4-slave-local* digunakan untuk menghubungkan antara *router* dengan perangkat komputer, sedangkan *Ether 5*, tidak digunakan.

Traffic data menjelaskan tentang mekanisme pertukaran data dalam suatu jaringan. Pada hasil yang sudah ditampilkan dijelaskan bahwa *traffic* yang terjadi adalah pada *ethernet-2*, *ethernet-3*, dan *ethernet-4*.

Bandwidth usage menjelaskan informasi tentang berapa jumlah *bandwidth* yang digunakan dalam jaringan tersebut. Informasi dari *bandwidth* yang diambil adalah saat melakukan *upload* dan *download* dalam beberapa kurun waktu sesuai dengan jumlah perangkat yang saling terhubung dalam jaringan tersebut. *Bandwidth* adalah suatu nilai konsumsi transfer data yang dihitung dalam bit/detik atau yang biasanya di sebut dengan *bit per second (bps)*, antara *server* dan *client* dalam waktu tertentu.



Gambar 11 Tampilan Menu Resource Information

Pada *menu resource information* menjelaskan tentang informasi perangkat *router* yang digunakan. Informasi yang diambil berisi tentang IP yang digunakan pada *router* tersebut, *CPU router*, *Platform*, *Versi Router*, *Memori* yang tersedia pada *router*, *uptime router* dalam menangani sebuah jaringan dan masih banyak lagi informasi yang dapat dilihat pada Gambar 11.

Proses yang terjadi pada saat aplikasi meminta dan menerima informasi dari *server* melalui protokol *SNMP* adalah proses *GetNextRequest*. Proses ini memanfaatkan *PDU (Protocol Data Unit)* yang ada pada *SNMP Versi 1*. *PDU GetNextRequest* dapat dilihat pada Gambar 12.

Id.	Time	Source	Destination	Protocol	Length	Info
5135	53.	192.168.88.22	192.168.88.1	SNMP	86	get-next-request 1.3.6.1.2.1.4.20.1.1
5136	53.	192.168.88.22	192.168.88.1	SNMP	96	get-next-request 1.3.6.1.2.1.4.20.1.1.192.168.43.1
5137	53.	192.168.88.22	192.168.88.1	SNMP	96	get-next-request 1.3.6.1.2.1.4.20.1.1.192.168.88.1
5138	53.	192.168.88.22	192.168.88.1	SNMP	97	get-next-request 1.3.6.1.2.1.4.20.1.1.192.168.220.
5139	53.	192.168.88.22	192.168.88.1	SNMP	97	get-next-request 1.3.6.1.2.1.4.20.1.1.192.168.230.
5140	53.	192.168.88.22	192.168.88.1	SNMP	86	get-next-request 1.3.6.1.2.1.2.2.1.10
5141	53.	192.168.88.22	192.168.88.1	SNMP	88	get-next-request 1.3.6.1.2.1.2.2.1.10.1
5142	53.	192.168.88.22	192.168.88.1	SNMP	90	get-next-request 1.3.6.1.2.1.2.2.1.10.2
5143	53.	192.168.88.22	192.168.88.1	SNMP	88	get-next-request 1.3.6.1.2.1.2.2.1.10.3
5144	53.	192.168.88.22	192.168.88.1	SNMP	88	get-next-request 1.3.6.1.2.1.2.2.1.10.4
5145	53.	192.168.88.22	192.168.88.1	SNMP	88	get-next-request 1.3.6.1.2.1.2.2.1.10.5
5146	53.	192.168.88.22	192.168.88.1	SNMP	86	get-next-request 1.3.6.1.2.1.2.2.1.16
5147	53.	192.168.88.22	192.168.88.1	SNMP	88	get-next-request 1.3.6.1.2.1.2.2.1.16.1
5148	53.	192.168.88.22	192.168.88.1	SNMP	90	get-next-request 1.3.6.1.2.1.2.2.1.16.2
5149	53.	192.168.88.22	192.168.88.1	SNMP	88	get-next-request 1.3.6.1.2.1.2.2.1.16.3
5150	53.	192.168.88.22	192.168.88.1	SNMP	88	get-next-request 1.3.6.1.2.1.2.2.1.16.4
5151	53.	192.168.88.22	192.168.88.1	SNMP	88	get-next-request 1.3.6.1.2.1.2.2.1.16.5

Gambar 12 Proses yang terjadi pada SNMP

PDU merupakan unit data yang terdiri atas sebuah *header* dan beberapa data yang ditempelkan. Penggunaan SNMP dengan memanfaatkan variable melalui PDU *GetNextRequest* dapat memberikan informasi lebih cepat dibanding dengan PDU yang lain pada SNMP Versi 1. Protokol SNMP memiliki variabel-variabel yang didefinisikan dalam MIB sehingga dalam memberikan informasi dapat berupa teks, bilangan, dan waktu. Struktur variabel tersebut bersifat hirarki dan memiliki aturan yang diinstruksikan kedalam OID sehingga informasi setiap variabel dapat dikelola atau ditetapkan dengan mudah.

Tabel 4 Pengujian Sistem

Pengujian	Aksi	Hasil Aksi	Status Pengujian
Tampilan antar muka	<i>Login</i> , masuk ke halaman menu, pilih menu.	Jika berhasil <i>login</i> , maka aplikasi akan mengarahkan ke halaman menu. Pada halaman menu terdapat pilihan untuk menampilkan informasi, seperti <i>server info</i> , <i>network info</i> , <i>resource detail</i> dan <i>about</i> .	<i>Valid</i>

Kesesuaian protokol SNMP memberi informasi kepada aplikasi	Klik menu server info, klik menu <i>network info</i> , klik menu <i>resource</i> , dan klik menu <i>about</i> .	Ketika memilih menu <i>server info</i> , aplikasi menampilkan informasi <i>server</i> . <i>Network info</i> akan memberikan informasi mengenai <i>interface</i> jaringan, <i>traffic data</i> , dan <i>bandwidth usage</i> . Pada menu <i>resource</i> akan menampilkan kondisi <i>server</i> , dan informasi mengenai keadaan <i>server</i> . <i>About</i> menampilkan informasi aplikasi.	Valid
PDU (Protocol data Unit) <i>GetNextRequest</i>		Jika protokol sudah memberikan informasi jaringan dengan baik kepada pengguna aplikasi	Valid

5. Kesimpulan

Penerapan protokol data unit *GetNextRequest* pada SNMP untuk melakukan *monitoring* jaringan menggunakan perangkat android dapat berjalan dengan baik. Informasi mengenai status perangkat pada jaringan yang dimonitor ditampilkan dengan akurat. Aplikasi ini dapat membantu *administrator* jaringan untuk memperoleh informasi saat mengelola sebuah jaringan, karena penyampaian informasi dapat diperoleh dengan cepat. Saran untuk pengembangan aplikasi kedepan adalah: dapat melakukan kontrol jaringan tidak terbatas pada *monitoring* saja; jaringan yang dimonitor menggunakan jaringan publik; dan pengembangan aplikasi menggunakan protokol data unit lain, yang ada pada SNMP.

6. Daftar Pustaka

- [1] Arifin, Firman, dkk. 2004. "Sistem Monitoring Traffic Packet Internet Melalui SMS Berbasis Simple Network Management Protocol (SNMP)". Makalah Teknologi dan Informasi (IES) ,2004.
- [2] Shididiqi, Mazharuddin Ary & Nugraha, Andika Panji, 2011. "Sistem Monitoring Jaringan Dengan Protokol SNMP Menggunakan Prinati Bergerak". Jurnal Teknologi Informasi (ITS), 2007.
- [3] Mansfield, Nial, 2002. "Practical TCP/IP Designing Using and Troubleshooting TCP/IP Networks On Linux and Windows". Yogyakarta : ANDI.
- [4] Herlambang, Moch Linto & Catur, Aziz, 2008. "Panduan Lengkap Menguasai Router Masa Depan Menggunakan Mikrotik Router OS". Yogyakarta : ANDI.
- [5] Setyawan, Eddy & Kristianto, Lewi S, 2002. "Pembangunan Jaringan Komputer Nirkabel dengan FreeBSD Sebagai Gateway". Vol:III, Nomor: 2, 2002.
- [6] Zaini, Muhammad Rizky, 2013. "Implementasi Protokol SNMP Untuk Jaringan Di Kantor Gubernur Sumatera Barat". Makalah Teknologi dan Informasi Ilmiah (TEKNOIF). Vol:I, Nomor:I, April 2013.
- [7] Hidalgo, Fernando & Gamess, Eric, 2014. "Integrating Android Devices into Network Management Systems Based on SNMP". Vol: V, Nomor: 5, 2014.
- [8] Ichwan, Muhammad & Hakiky Fifin, 2011. " Pengukuran Kinerja Goodreads Application Programing Interface Pada Aplikasi Mobile Android". Vol :II, Nomor : 2, 2011.
- [9] Murtiwiayati & Lauren, Glenn. 2013. "Rancang Bangun Aplikasi Pembelajaran Budaya Indonesia Untuk Anak Sekolah Dasar Berbasis Android". Vol: XII , Nomor: 2, 2013.