PENERAPAN ALGORITMA SAPA PADA JARINGAN *AD-HOC* UNTUK MENGIDENTIFIKASI KOMPUTER BERBASIS MAC *ADDRESS*

Chris Tri Widyarto, Hero Wintolo, Yuliani Indrianingsih
Jurusan Teknik Informatika
Sekolah Tinggi Teknologi Adisutjipto Yogyakarta
informatika@stta.ac.id

Abstract

Wi-Fi waves are commonly used as an intermediary in the communication relationships in the wireless network. However, the use of Wi-Fi waves can be used in addition to getting the relationship in the network, the user is also able to detect and identify other devices by using Wi-Fi waves. Ad-hoc network is a network that can be built independently by the computer with the Microsoft Windows operating system, to enable ad-hoc device has to have an identity that is SSID, the SSID is then activate the device can be detected by other devices. Activation is done ad-hoc by the netsh command is a shell command in the Windows 7 operating system. An ID must be unique for each device as there is no dual identity, to maximize the identification process. Utilizing MAC address as an identity is used, allowing the device to be uniquely identified. In addition to the MAC address is added to the device profile in the identity of the device for easier identification. Sapa *application users can identify the device is detected, and recognize it as a co-owned devices. Scheme which is the concept of the algorithm* Sapa. Sapa *application allows device activation or notebook PC with the Windows 7 operating system to be detected and detected by other devices with the Windows 7 operating system and applications are also active* Sapa. *Communication is established indirect communication that occurs in a single direction. Detection range can be done up to 77 meters distance.*
Keyword : ***ad-hoc, Wi-Fi, MAC Address, Netsh, Windows***

1. Pendahuluan

Perkembangan zaman telah membawa manusia pada era *gadget* dengan berbagai fungsi dan kemampuan yang ditanamkan. Algoritma dalam dunia pemrograman komputer adalah suatu proses-proses atau tahapan-tahapan untuk menyelesaikan masalah hingga didapat hasil dari proses. Saat pertama kali komputer diciptakan, manusia tidak pernah membayangkan bahwa komputer dapat menjadi sebuah *gadget* jinjing yang dapat dibawa kemanapun. Bahkan saat ini komputer jinjing atau umum disebut laptop atau *notebook*, seakan menjadi *gadget* wajib bagi mahasiswa selayaknya telepon genggam. Khususnya kemampuan untuk mengakses *internet*, *notebook* saat ini telah berkembang sangat pesat.

Jaringan *internet* dapat secara langsung diakses melalui jaringan *mobile broadband* seperti *General Packet Radio Service* (GPRS), *Enhanced Data rates for GSM Evolution* (EDGE), *Universal Mobile Telecommunications System* (UMTS/3G), *High Speed Downlink Packet* Access (HSDPA), *High Speed Uplink Packet Access* (HSUPA), *Worldwide Interoperability for Microwave* Access (WiMAX), *Long Term* Evolution (LTE) atau melalui jaringan *Wireless Fidelity* (*Wi-Fi*). Saat ini fitur jaringan *Wi-Fi* telah menjadi keharusan tersendiri bagi produsen *notebook* untuk

menanamkannya di setiap produknya. Sehingga keberadaan fitur *Wi-Fi* dalam sebuah *notebook* telah menjadi keharusan. Untuk mengakses jaringan internet selain menggunakan jaringan *Wi-Fi*, sebuah *notebook* harus menambahkan perangkat modem untuk dapat mengakses internet melalui jaringan EDGE, UMTS dan HSDPA. Keberadaan fitur *Wi-Fi* mendukung mobilitas komputer jinjing dalam mengakses terhadap sumber daya internet lebih efisien dibanding dengan *Personal Computer* (PC). Hubungan dengan internet telah menjadi kebutuhan para pengguna *gadget*. Tidak hanya untuk terhubung ke dalam jaringan internet, koneksi *Wi-Fi* juga dibutuhkan untuk saling bertukar data antar perangkat.

Pertukaran data antar perangkat komputer membutuhkan protokol. Ada dua jenis protokol yang umum digunakan, yaitu *Transmission Control Protocol / Internet Protocol* (TCP/IP) dan *User Datagram Protocol* (UDP). TCP/IP adalah *connection-oriented protocol* yang berarti protokol ini membutuhkan adanya negoisasi koneksi yang disebut dengan *three-way-handshake* antara perangkat asal dan perangkat tujuan paket yang akan dikirim. Sedangkan UDP bersifat *connectionless* yang berarti protokol ini akan mengirimkan paket tanpa harus adanya *three-way-handshake*. Dengan sifatnya yang *connectionless* maka dengan protokol ini dimungkinkan terjadinya komunikasi antar perangkat tanpa harus membentuk koneksi penuh.

Pemakaian protokol UDP dalam mekanisme transmisi data dapat mendukung mekanisme "Sapa" pada sebuah aplikasi yang diinstalkan pada setiap perangkat dengan fitur *Wi-Fi*. Karena untuk terciptanya mekanisme "Sapa" dibutuhkan pengiriman dan penerimaan data tanpa perlu menciptakan *dedicated connection*. Penggunaan protokol UDP membutuhkan *bandwidth* yang relatif lebih kecil, sehingga tidak akan terlalu menghabiskan *resource* baterai perangkat khususnya. Dalam implementasinya dapat menggunakan koneksi *ad-hoc* sebagai contoh yang paling mudah untuk dipakai. Koneksi ini terdapat pada sistem operasi Windows besutan raksasa teknologi Microsoft. Koneksi *ad-hoc* ini memungkinkan perangkat *notebook* dapat saling bertukar data dan berbagi koneksi internet melalui jaringan *Wi-Fi*.

2. Landasan Teori

A. *Handshake*

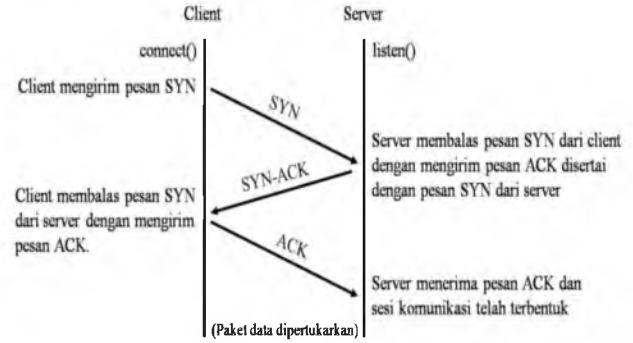
Tata cara manusia untuk memulai berkomunikasi dengan orang yang baru dikenal dimulai dengan cara melakukan pengenalan nama diikuti dengan jabat tangan (handshake). Seperti halnya manusia, perangkat-perangkat yang akan berkomunikasi dalam sebuah jaringan membutuhkan perkenalan yang disebut dengan *handshake* untuk menentukan protokol yang digunakan, atau algoritma serta kunci enkripsi yang dipakai agar komunikasi yang terjalin dapat berjalan lancar dan dapat dimengerti oleh perangkat-perangkat yang berkomunikasi.

Istilah *handshake* yang umum dalam dunia jaringan komputer adalah *3-way-handshake*, saat dua perangkat komputer hendak saling bertukar data dengan membangun sebuah *dedicated connection* maka terjadi proses *3-way-handshake* sebagai permulaan komunikasi. Proses *3-way-handshake* ini terjadi pada protokol dengan sifat *connection oriented* yaitu TCP/IP. Langkah-langkah yang terjadi pada proses *3-way-handshake* adalah sebagai berikut :

1. Perangkat yang meminta berkomunikasi mengirimkan paket TCP SYN yang dalam istilah lengkap *synchronize*.

2. Perangkat lain sebagai "lawan bicara" mengirimkan paket TCP SYN-ACK kepada lawan bicaranya.
3. Proses terakhir adalah perangkat yang meminta berkomunikasi setelah menerima paket SYN-ACK, mengirimkan pesan ACK yang dalam istilah lengkap adalah *acknowledge*.

Berikut bagan komunikasi dengan menggunakan *3-way-handshake* dapat dilihat pada Gambar 1.



Gambar 1 Bagan Komunikasi *3-Way-Handshake*

B. Perbedaan TCP dan UDP

Protokol yang digunakan dalam berkomunikasi dalam jaringan komputer ada dua jenis, yaitu TCP dan UDP. Protokol TCP adalah sebuah protokol komunikasi yang membentuk *connection-oriented protocol* yang memungkinkan pengiriman paket-paket data sampai di tujuan tanpa hilang satupun. UDP atau *User Datagram Protocol* dipakai menggantikan TCP ketika dalam pengiriman data tidak diperlukan pengiriman yang *reliable*. Protokol UDP digunakan pada *streaming media* seperti *audio, video* dan lainnya. UDP relatif lebih cepat dibanding TCP dikarenakan tidak ada *error checking, error correction*, atau *acknowledgement*. Selain itu besarnya paket data pada protokol UDP relatif lebih kecil dibandingkan dengan pada protokol TCP Pada pengaplikasiannya kedua protokol ini sangat penting dalam transfer data. Sebagai perbandingan kinerja antara TCP dengan UDP bisa dilihat pada Tabel 1.

Tabel 1 Perbandingan TCP dan UDP

Karakteristik / Deskripsi	**UDP**	**TCP**
Deskripsi Umum	"Pembungkus paket" yang memiliki kecepatan tinggi namun rendah dalam fungsi.	Protokol dengan fitur lengkap yang memungkinkan aplikasi untuk mengirim data dengan terpercaya tanpa khawatir tentang masalah pada *network layer*.
***Data Interface* Ke Aplikasi**	*Message base-based* dikirim dalam paket dengan ciri tersendiri oleh aplikasi.	*Stream-based*; Data dikirim oleh aplikasi tanpa struktur khusus.

Karakteristik / Deskripsi	**UDP**	**TCP**
Pengiriman Ulang	Tidak dilakukan. Aplikasi harus mendeteksi kehilangan data dan mengirim kembali jika dibutuhkan.	Pengiriman dari seluruh data telah diatur, dan kehilangan data akan dikirim kembali secara otomatis.
Fitur yang disediakan untuk mengatur *flow of Data*	Tidak ada.	*Flow control* menggunakan *sliding windows*, pengaturan ukuran jendela *heuristics*, algoritma penghindar kemampatan.
Overhead	Sangat kecil.	Kecil, tapi lebih besar dari UDP.
Kecepatan pengiriman	Sangat tinggi.	Tinggi tapi tidak setinggi UDP.
Kesesuaian kuantitas data	Kuantitas data kecil hingga sedang.	Kuantitas data kecil hingga sangat besar.

C. *Connectionless Communication*

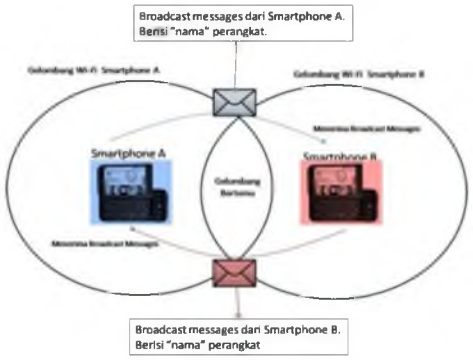
Pada umumnya ketika dua perangkat akan berkomunikasi, maka sebuah *dedicated communication* harus dibangun terlebih dahulu. Dengan memanfaatkan prinsip *connectionless communication* maka sebuah komunikasi satu arah yang sangat singkat akan dapat terjadi dengan lebih cepat dan hanya menggunakan *resource* data yang cukup kecil.

D. Koneksi *Ad-Hoc*

Komunikasi perangkat komputer dalam lingkungan WLAN biasa terjadi dengan bantuan perangkat tambahan berupa *access point*. Komunikasi ini pun bisa tetap diwujudkan dengan tanpa perangkat *access point*, yaitu dengan mengaktifkan koneksi *ad-hoc*. Koneksi *ad-hoc* adalah sebuah mode koneksi yang disediakan pada komputer dengan sistem operasi *Microsoft Windows* untuk menyediakan sebuah koneksi sementara bagi komputer agar bisa berkomunikasi seraca *point-to-point*. Sedangkan pada sistem operasi *Android* yang lazim digunakan pada *smartphone* dan perangkat *tablet* disebut sebagai *tethering and portable hotspot*. Komunikasi *ad-hoc* ini menjadikan perangkat sebagai *access point* untuk perangkat lain agar bisa berkomunikasi.

E. Jaringan *Wi-Fi*

Pada masa sekarang, konektifitas antara perangkat sangat dimanjakan dengan kehadiran teknologi nirkabel. Pengguna *gadget* seperti *smartphone* atau *notebook* dapat mendeteksi keberadaan *access point* di sekitarnya yang berada dalam jangkauan. Memanfaatkan bukti ini dengan memodifikasi protokol dan proses yang digunakan, memungkinkan bagi sebuah perangkat untuk bisa mendeteksi keberadaan perangkat lain yang juga menyalakan *Wi-Fi* hingga membentuk sebuah komunikasi satu arah tanpa mengharuskan sebuah perangkat itu menjadi *access point* sementara dan tanpa menjalin koneksi yang tetap. Sebagai gambaran dapat dilihat pada Gambar 2.



Gambar 2 Mekanisme Sapa antar perangkat

F. Netsh

Netsh atau *Network shell* adalah *script utility* yang memungkinkan pengguna komputer dengan sistem operasi *Microsoft Windows* untuk dapat menampilkan atau memodifikasi pengaturan jaringan yang sedang berjalan. Perintah ini ditanamkan mulai pada sistem operasi *Windows* 2000 hingga sistem operasi terbaru dari *Microsoft* yaitu *Windows* 8. Beberapa perintah dalam Netsh memerlukan *privilege* seorang admin dalam pengeksekusiannya.

3. Perancangan Dan Cara Uji Coba

A. Kebutuhan Perangkat Keras

Perangkat keras yang digunakan dalam perancangan dan uji coba adalah sebagai berikut :

1. *Personal Computer* (PC) : Prosesor Intel Core 2 Duo E7300 2,66 GHz, RAM 2 Gb DDR2, VGA Nvidia GeForce 8600GTS 256 Mb, Gigabyte *motherboard socket* LGA775, D-Link *Wi-Fi adapter* DWA 525, *onboard* LAN adapter, Lite-On DVD R/W SATA, HDD 160 GB SATA.
2. *Notebook* : Asus A43E prosesor Intel Core i3 2,1 GHz, RAM 2Gb DDR3, Intel HD Graphics, *Wi-Fi adapter*, LAN *adapter*, DVD R/W, HDD 500 GB.
3. *Smartphone* : Samsung Galaxy Young.

Spesifikasi minimum untuk *notebook* dan komputer PC adalah sebagai berikut :

1. *Processor* : Intel Dual Core 1,8 GHz (atau prosesor *dual core* lainnya).
2. RAM : DDR2 1 Gb.
3. VGA : Nvidia 256 Mb atau lainnya dengan spesifikasi yang sama.
4. WLAN *adapter* : 802.11a/b/g/n yang mendukung fitur *hosted network* (pembahasan terdapat pada Bab 4 halaman 27).

Spesifikasi diatas adalah spesifikasi minimun untuk dapat menjalankan program pada sistem operasi Windows 7 dengan lancar.

B. Kebutuhan Perangkat Lunak

Perangkat lunak yang digunakan dalam perancangan sistem adalah sebagai berikut :

1. Windows 7 32-bit sebagai sistem operasi yang berjalan pada *notebook* dan *personal computer*.
2. Delphi 7 sebagai *environment* dalam pembuatan program.
3. *Command Prompt*, penggunaan *shell programming* dibutuhkan untuk bisa mengaktifkan mode *ad-hoc*.

C. Analisa Awal

Pada umumnya perangkat-perangkat yang dapat dideteksi keberadaannya melalui gelombang *Wi-Fi* tanpa dilakukannya sambungan adalah *access point* dan perangkat bukan *access point* yang sedang menjadi *access point*. Jika hal ini bisa diterapkan pada perangkat bukan *access point* tanpa harus menjadi *access point*. Maka bisa menjadi acuan untuk mendeteksi *gadget* lainnya menggunakan gelombang *Wi-Fi* seperti radar. Namun tentu perangkat yang dideteksi harus berada dalam radius gelombang *Wi-Fi gadget* yang dipakai untuk mendeteksi.

Kepentingan untuk dapat mendeteksi adalah untuk menemukan rekan dalam sebuah lingkup yang cukup besar namun masih dalam jarak gelombang *Wi-Fi* perangkat, misalnya ruang pameran atau *hotspot area*. Semakin umumnya kepemilikan perangkat *notebook* oleh seorang mahasiswa atau pelajar, bisa menjadi alat untuk mendeteksi sesama *notebook* dengan memanfaatkan gelombang *Wi-Fi* yang terpasang pada setiap *notebook*. Aplikasi yang memiliki fungsi untuk mengaktifkan mode *ad-hoc* secara instan dengan menggunakan data *Media Access Control Address* (MAC *address*) perangkat sebagai *Service Set Identifier* (SSID).

D. Konfigurasi Jaringan

Pengujian dilakukan menggunakan koneksi tanpa kabel atau *wireless* dengan gelombang *Wi-Fi*. Rancangan sistem uji coba menggunakan gelombang *Wi-Fi* dapat dilihat pada Gambar 3.

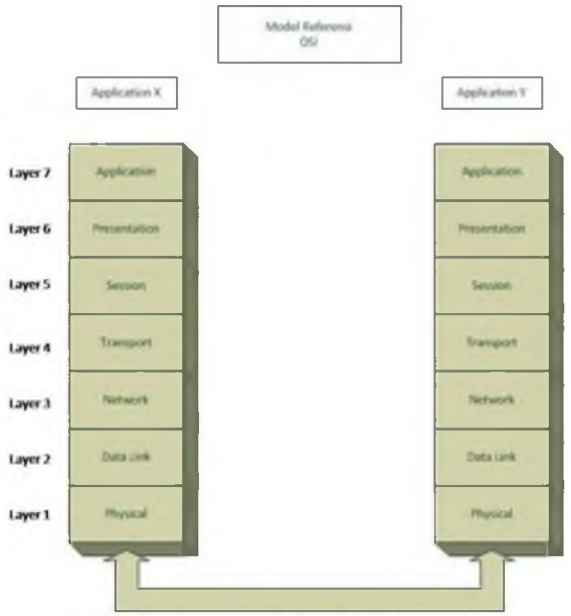
Gambar 3 Rancangan sistem uji coba

Sistem yang dibangun ini dimaksudkan untuk bisa mendeteksi perangkat dengan menggunakan gelombang *Wi-Fi*. Perangkat yang mendeteksi dan dideteksi menggunakan *software* hasil perancangan dengan perangkat *Wi-Fi* menyala. Mekanisme pendeteksian adalah dengan menerima SSID yang berisi identitas perangkat pengirim. Identitas perangkat pengirim merupakan hasil enkripsi dari MAC *Address*. Data berisi hasil enkripsi dari MAC *address* ini adalah identitas yang akan diidentifikasi oleh perangkat pendeteksi paket. Sehingga dapat dikenali asal perangkat dari paket yang diterima.

E. Komunikasi yang Dirancang

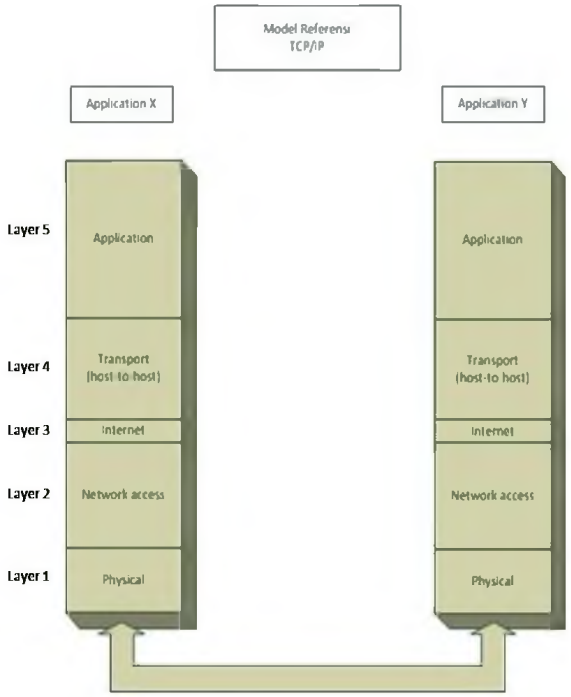
Komunikasi yang dirancang dalam tugas akhir ini merupakan komunikasi yang terjadi pada tingkat aplikasi. Pada tingkat ini terjadi proses menerima pesan dan identifikasi pesan yang diterima.

Dunia jaringan komputer mengenali dua protokol standar yang populer dalam komunikasi data. Standar protokol yang pertama adalah *Open System interconnection* (OSI) dan standar protokol yang kedua adalah TCP/IP. Sebagai gambaran komunikasi antara dua buah komputer dengan model referensi OSI dapat dilihat pada Gambar 4



Gambar 4 Model referensi OSI

Pada model referensi OSI ini lapisan aplikasi merupakan lapisan ke tujuh. Sedangkan pada model referensi TCP/IP, lapisan aplikasi terletak pada lapisan ke lima. Perbedaan jumlah ini adalah perbedaan dari konsep proses dalam komunikasi data yang diusung oleh kedua model referensi. Komunikasi antara dua buah komputer dengan model referensi TCP/IP dapat dilihat pada Gambar 5



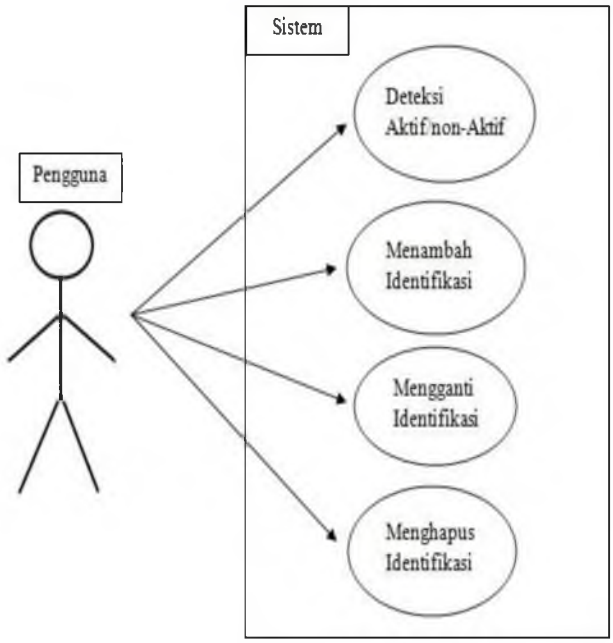
Gambar 5 Model referensi TCP/IP

Standar protokol umum disebut dengan model referensi. Perbedaan antara model referensi OSI dan TCP/IP adalah pada *layer* atau lapisan yang mewakili proses yang terjadi dalam komunikasi data tiap-tiap model referensi. Selain itu perbedaan juga terdapat pada bahasa yang digunakan dalam komunikasi data. Sehingga membutuhkan

perangkat jaringan yang disebut *router* agar bisa menghubungkan dua jaringan yang memiliki model referensi yang berbeda, agar komunikasi dapat berjalan sesuai dengan harapan. Model referensi OSI memiliki tujuh lapisan atau *layer*. Sedangkan pada model referensi TCP/IP terdapat 5 *layer*.

Urutan berjalannya komunikasi pada komunikasi pengiriman data, berlangsung dari lapisan pertama (terbawah) sampai dengan lapisan terakhir pada tiap-tiap model referensi. Lapisan aplikasi sebagai lapisan yang menjadi fokus utama tugas akhir ini berada pada lapisan terakhir, baik pada OSI maupun pada TCP/IP. Rancangan aplikasi dibuat dengan menggunakan bahasa pemrograman Delphi 7. Pada setiap aplikasi dibuat mekanisme atau prosedur untuk mengirimkan sinyal "Sapa" dalam jaringan komputer yang menggunakan model referensi OSI atau TCP/IP.

F. Diagram *Use Case*



Gambar 6 Diagram Algoritma Sapa

Sebagai gambaran jalannya program, maka alur program dapat digambarkan melalui bantuan diagram *Use Case*. Diagram program Sapa dapat dilihat pada Gambar 6. Program Sapa ini melibatkan *user* dan sistem dalam program. *Login* untuk menggunakan program tidak diperlukan.

G. Algoritma Sapa

Algoritma Sapa dalam Tugas Akhir ini adalah algoritma dalam penyaringan data yang diterima oleh program menjadi sebuah informasi yang bisa digunakan oleh pengguna program dalam membaca hasil deteksi program. penyaringan yang dilakukan adalah penyaringan dari data yang telah dikumpulkan di dalam sebuah file **.txt* yang menampung hasil pendeteksian perangkat. Namun data hasil pendeteksian perangkat ini

masih bersifat data mentah. Hasil deteksi program dibagi menjadi dua jenis hasil deteksi, yaitu :

1. Hasil deteksi baru
   Hasil deteksi baru adalah hasil deteksi perangkat yang belum dikenali program sebagai perangkat yang telah masuk dalam daftar identifikasi.
2. Hasil deteksi yang teridentifikasi
   Hasil deteksi yang teridentifikasi adalah hasil deteksi perangkat yang telah dikenali dan program mengganti identitas perangkat tersebut dengan identitas yang diberikan oleh pengguna program.

Berikut adalah tahapan berjalannya algoritma Sapa :

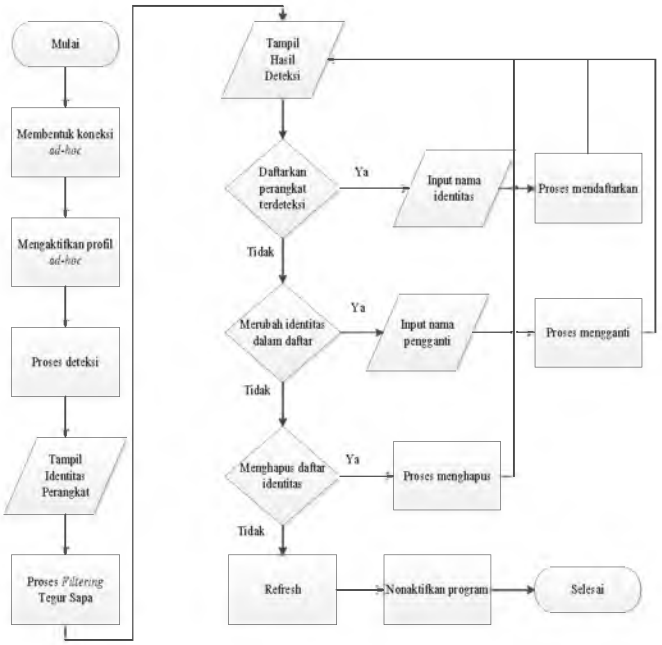
1. Membaca file *list.txt* yang berisi hasil pendeteksian perangkat.
2. Memilah data yang berisi SSID perangkat, dengan data biasa.
3. Penyaringan dilakukan dengan menemukan kata = SSID, karena pada kalimat dengan kata SSID merupakan baris yang berisi data SSID yang dicari.
4. Setelah menemukan kata = SSID, selanjutnya baris tersebut dimasukan ke dalam variabel untuk kemudian menjadi nama SSID hasil deteksi.
5. Hasil deteksi ditampilkan untuk selanjutnya bisa didaftarkan secara manual oleh pengguna.
6. Apabila SSID yang dideteksi telah terdaftar, maka tampilan identitas yang tampil di program merupakan identitas baru yang telah didaftarkan oleh pengguna.

H. Cara Uji Coba

Setelah selesai pengembangan dan telah menjadi sebuah program utuh. Maka uji coba algoritma Sapa dilakukan dengan tahapan-tahapan berikut ini :

1. Pengguna mengaktifkan program untuk memulai proses pembentukan dan pengaktifan koneksi *ad-hoc*, serta mengaktifkan pendeteksian.
2. Setelah program aktif, pengguna dapat melihat identitas dari perangkat.
3. Membandingkan identitas perangkat pada program, dengan identitas perangkat tersebut pada hasil pendeteksian perangkat lain.
4. Program akan langsung melakukan penyaringan file *list.txt* yang dibuat dalam proses deteksi. Proses penyaringan untuk mendapatkan informasi SSID dari perangkat yang terdeteksi.
5. Membandingkan daftar deteksi dengan perangkat yang aktif.
6. Mendaftarkan identitas perangkat yang terdeteksi dengan identitas yang diinginkan.
7. Melihat daftar identitas yang telah didaftarkan.
8. Mengganti informasi SSID yang ada pada daftar identitas.
9. Memperbaharui tampilan perangkat yang terdeteksi dengan menekan tombol "*Refresh*".
10. Menonaktifkan program, sehingga tidak terdeteksi lagi oleh perangkat lain.

Diagram alur uji coba dapat dilihat pada Gambar 7.



Gambar 7 Diagram alur cara uji coba

4. Pengujian Dan Analisa

A. Penjelasan Aplikasi

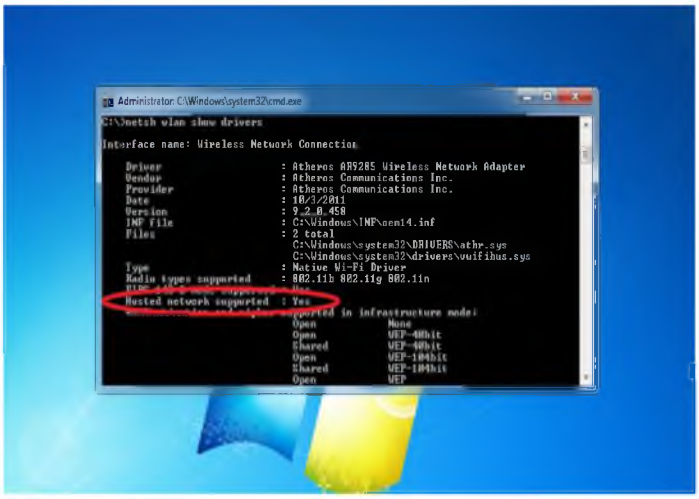
Aplikasi Sapa adalah aplikasi yang menerapkan Algoritma Sapa dalam mendeteksi serta menyaring data dari jaringan *Wi-Fi* yang terdeteksi oleh perangkat pengguna aplikasi. Pendeteksian dilakukan dengan menangkap *broadcast* SSID perangkat lain, dan dilakukan identifikasi perangkat berdasarkan daftar identifikasi. Perangkat *notebook* dengan sistem operasi Windows 7 yang mengaktifkan aplikasi Sapa, maka perangkat secara otomatis akan mengaktifkan *mode ad-hoc* pada sistem operasi dengan *mode* ini maka perangkat dapat melakukan deteksi dan dapat terdeteksi oleh perangkat lain.

a. Sebelum Aplikasi Berjalan

Pembaharuan *driver Wi-Fi adapter* pada perangkat sebelum menggunakan aplikasi ini perlu dilakukan, karena tidak semua *driver* bisa mendukung proses Netsh yang dijalankan oleh program. Bahkan ada perangkat yang sama sekali tidak bisa menjalankan proses Netsh, ini bisa di cek melalui jendela *command prompt* dengan mengetikan perintah "netsh wlan show drivers", lalu dilihat pada keterangan "*Hosted Network*" yang dapat dilihat pada Gambar 8. Jika keterangannya "No" maka pembaharuan *driver* bisa menjadi solusi. Pembaharuan *driver* perangkat, diperlukan karena sebuah perangkat harus bisa mendukung berjalannya "Microsoft *Virtual WiFi Miniport Adapter*", hal ini penting untuk bisa membuat sebuah jaringan *Ad-hoc* langsung melalui perintah dalam Netsh.

Aplikasi Sapa memerlukan jendela *command prompt* dengan *privilege administrator* agar semua perintah Netsh dapat dijalankan dengan lancar, untuk itu perubahan pada *registry* sistem operasi yang digunakan yaitu Windows harus dilakukan. Perubahan dilakukan dengan menambah *value* baru pada lokasi "HKEY_CURRENT_USER \ Software \ Microsoft \ Windows NT \ CurrentVersion \ AppCompactFlags \ Layers". *Value* yang ditambahkan adalah *path* menuju aplikasi

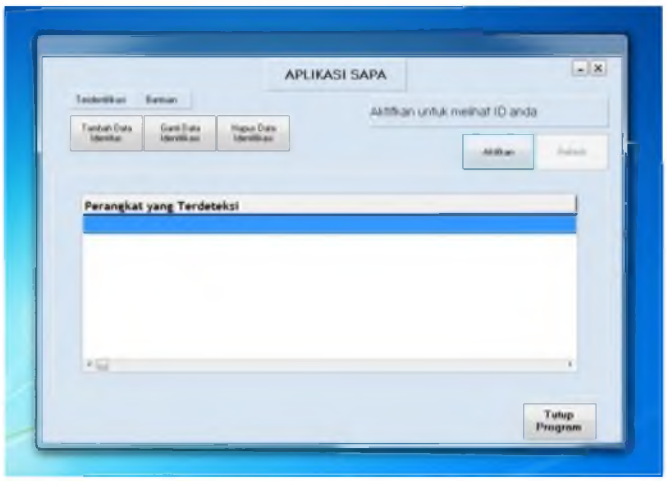
cmd.exe yang berada di lokasi C:\Windows\System32\cmd.exe dengan *value data* RUNASADMIN. Penambahan ini memungkinkan untuk setiap perintah dalam bentuk *shell command* dijalankan dengan otoritas seorang *administrator*.



Gambar 8 Keterangan *driver* perangkat

b. Saat Aplikasi Berjalan

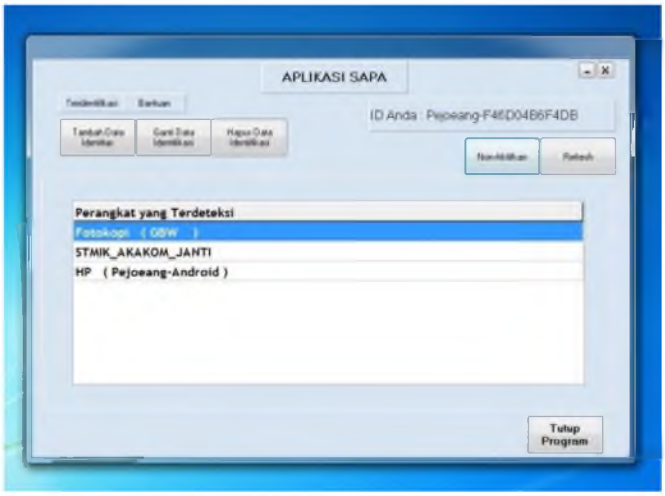
Aplikasi Sapa membutuhkan sebuah informasi dari perangkat pengguna sebagai identitas yang bersifat unik (tidak sama dengan perangkat lain). Sifat unik dibutuhkan sehingga dapat meminimalkan kemungkinan terdapatnya nama deteksi ganda. Aplikasi akan memakai informasi nama profil perangkat yang telah diinputkan oleh pengguna ditambah dengan data MAC *address* perangkat. MAC *address* terdiri dari 12 digit tersusun dengan format MM:MM:MM:SS:SS:SS, 6 digit pertama adalah ID pabrik asal perangkat, 6 digit selanjutnya adalah ID dari perangkat yang diberikan oleh pabrik. Karena inilah maka informasi MAC *address* bersifat unik bagi setiap perangkat.



Gambar 9 Aplikasi Sapa sebelum aktif

Saat aplikasi dijalankan, pengguna diharuskan untuk mengaktifkan aplikasi dengan menekan tombol aktif. Karena untuk bisa mendapatkan MAC *address*

perangkat sebagai identitas, kondisi perangkat harus aktif. Pada Gambar 10 dapat dilihat identitas yang muncul dari aplikasi setelah aplikasi dijalankan.



Gambar 10 Aplikasi Sapa setelah aktif

Setelah aktif, aplikasi langsung mengambil informasi deteksi perangkat WLAN dari *file* "list.txt". Informasi dari *file* langsung disaring hingga didapatkan data SSID yang terdeteksi. Data SSID yang didapat dibandingkan dengan daftar deteksi pada file "daftar.txt". Perangkat yang terdeteksi, dan teridentifikasi sebagai perangkat yang terdaftar, maka akan dirubah nama deteksi dari perangkat tersebut. Perubahan dilakukan dengan mengganti identitas asli perangkat dengan identitas yang terdapat dalam daftar. Identitas asli dari perangkat tetap ditampilkan dalam tanda kurung. Pendeteksian perangkat beserta identitasnya dilakukan tanpa harus terbangun sebuah jaringan yang *dedicated*, ini memungkinkan sebuah perangkat untuk secara mandiri tanpa terikat pada jaringan manapun untuk dapat melakukan deteksi.

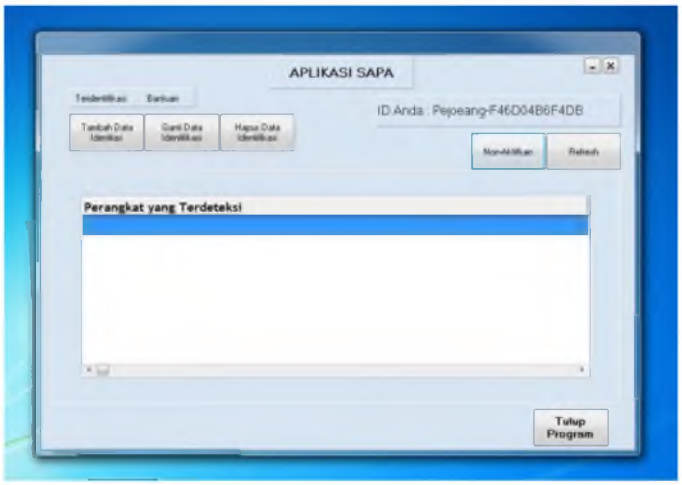
Semua perangkat dengan aplikasi Sapa yang aktif, berlaku sebagai *server* dalam jaringan *ad-hoc*. Aplikasi dapat mendeteksi semua SSID yang berada dalam area jangkauan gelombang *Wi-Fi*. Kelebihan yang didapat adalah aplikasi dapat mendeteksi dan mengidentifikasi perangkat berdasarkan daftar identifikasi yang telah dibuat oleh pengguna. Sehingga pengguna dapat dengan cepat mendeteksi keberadaan perangkat rekan yang telah dikenali.

B. Pengujian

Aplikasi Sapa memerlukan pengujian untuk mendapatkan data tentang *platform* apa saja yang dapat menjalankan aplikasi. Berjalannya fungsi penyaringan serta jarak maksimal yang bisa dicapai dalam mendeteksi dan terdeteksi bagi *platform* yang dapat menjalankan aplikasi Sapa.

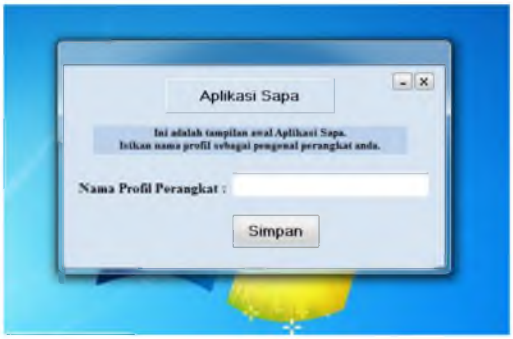
a. Microsoft dan Microsoft

1. Microsoft Windows 7



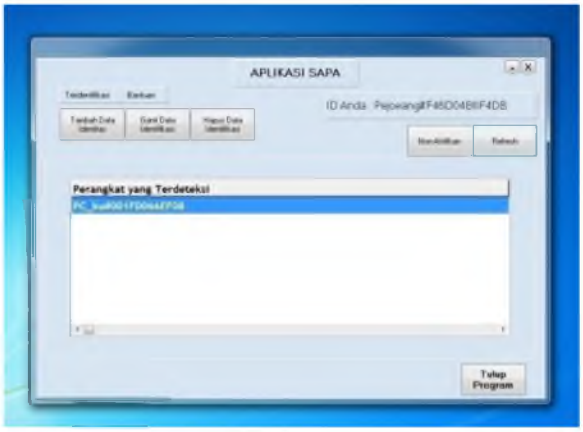
Gambar 11 Aplikasi Sapa pada sistem operasi Windows 7

Pada sistem operasi Windows 7, aplikasi Sapa berjalan seperti seharusnya. Karena pada sistem operasi ini juga aplikasi ini dibangun, sehingga proses dan komponen pendukung yang dipakai merupakan proses dan komponen dalam sistem operasi Windows 7. Pengujian menggunakan 2 buah perangkat dengan sistem operasi Windows 7 32-bit, perangkat-perangkat tersebut adalah *Notebook* dan PC yang spesifikasinya telah dijabarkan pada bagian perancangan. Fungsi deteksi pada aplikasi Sapa berjalan, dan fungsi penyaringan data untuk mengidentifikasi perangkat yang terdeteksi pun berjalan sebagaimana mestinya.

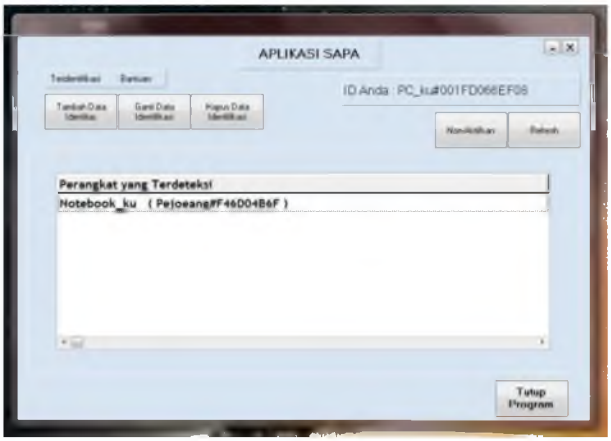


Gambar 12 Jendela pengisian nama profil perangkat

Saat pertama kali aplikasi dijalankan maka, pengguna harus mengisikan nama profil perangkat seperti ditampilkan pada Gambar 12. Proses ini hanya berjalan sekali, untuk selanjutnya jendela pengisian nama profil ini tidak akan muncul lagi (selama program belum di *uninstall*). Pada Gambar 11 menunjukan tampilan aplikasi saat belum mendeteksi keberadaan perangkat. Tampilan perangkat setelah mendeteksi keberadaan perangkat, namun perangkat tersebut belum berada dalam daftar identifikasi bisa dilihat pada Gambar 13 dan tampilan aplikasi pada perangkat PC bisa dilihat pada Gambar 14.

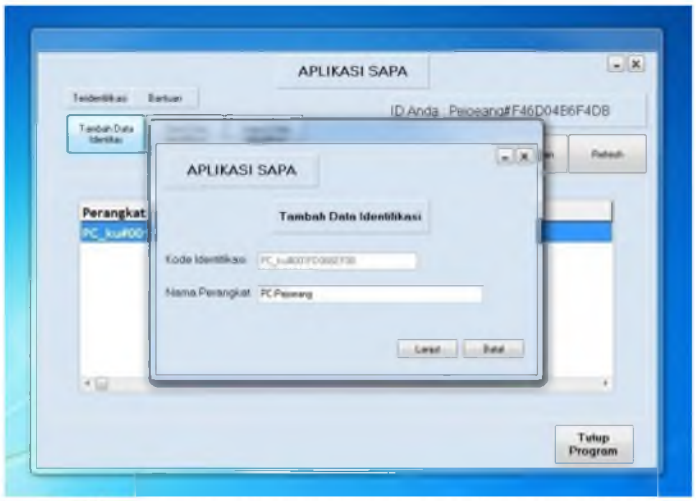


Gambar 13 Tampilan deteksi perangkat yang belum teridentifikasi

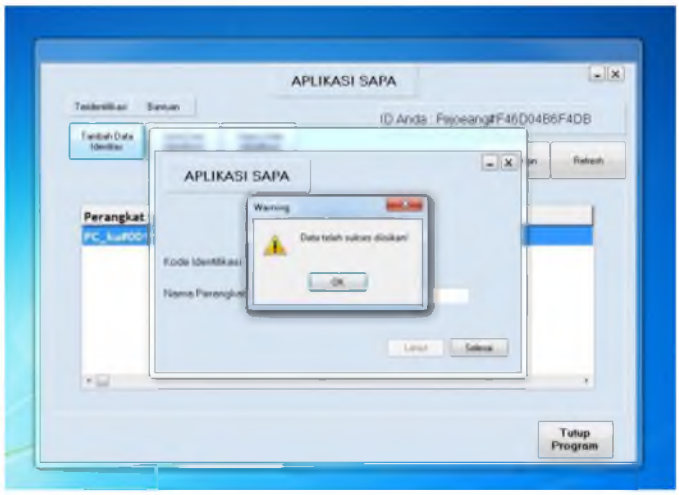


Gambar 14 Tampilan pada layar PC

Perangkat yang terdeteksi adalah perangkat dengan kode identitas "PC_ku#001FD066EF08" perangkat tersebut adalah perangkat PC yang juga menggunakan aplikasi Sapa. Namun belum masuk ke dalam daftar identifikasi. Untuk memasukan perangkat ke dalam daftar identifikasi, pengguna dapat harus mendaftarkannya melalui menekan tombol "Tambah Data Identitas" yang proses nya dapat dilihat pada Gambar 14 dan Gambar 16. Pada gambar 17 layar deteksi perangkat *wireless* milik dari sistem operasi dimunculkan sebagai pembanding, juga untuk mempercepat proses pendeteksian pada aplikasi. Jika layar deteksi tersebut tidak dimunculkan maka pendeteksian oleh aplikasi bisa berlangsung lebih lama (lebih dari 70 detik).

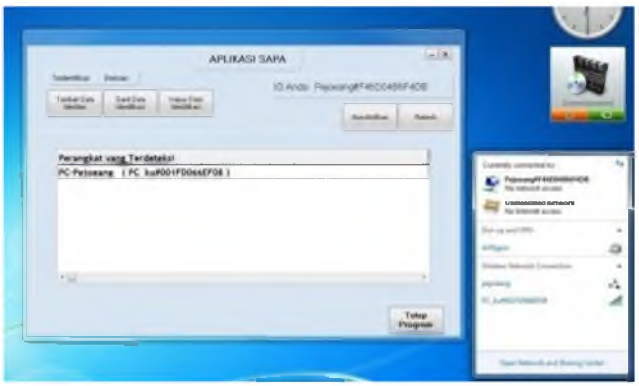


Gambar 15 Proses pengisian nama perangkat untuk identifikasi



Gambar 16 Data perangkat telah sukses dimasukan

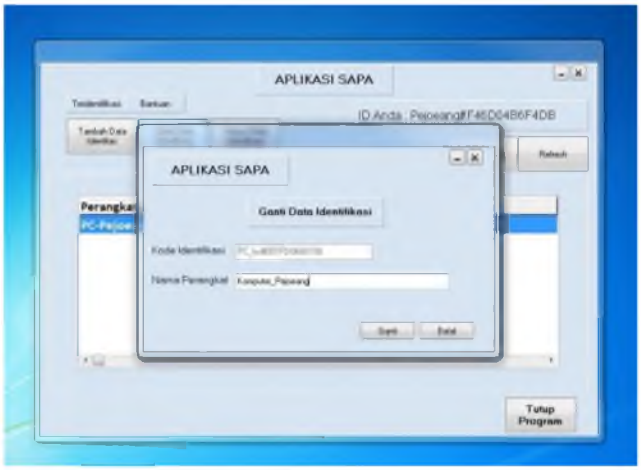
Setelah masuk ke daftar identifikasi, maka aplikasi dapat mengenali perangkat tersebut dalam setiap identifikasi, kecuali jika data perangkat telah dihapus dari daftar identifikasi. Tampilan dari aplikasi dapat dilihat pada Gambar 17.



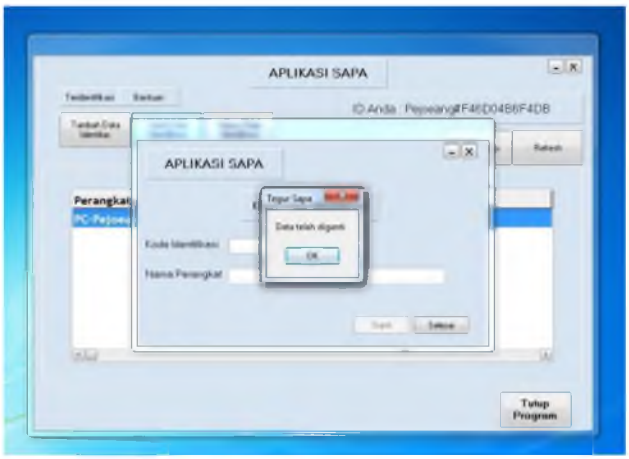
Gambar 17 Tampilan aplikasi jika perangkat telah teridentifikasi

Pengguna dapat melakukan penggantian data identifikasi perangkat dengan menekan tombol dengan *caption* "Ganti Data Identifikasi". Penggantian ini adalah penggantian nama

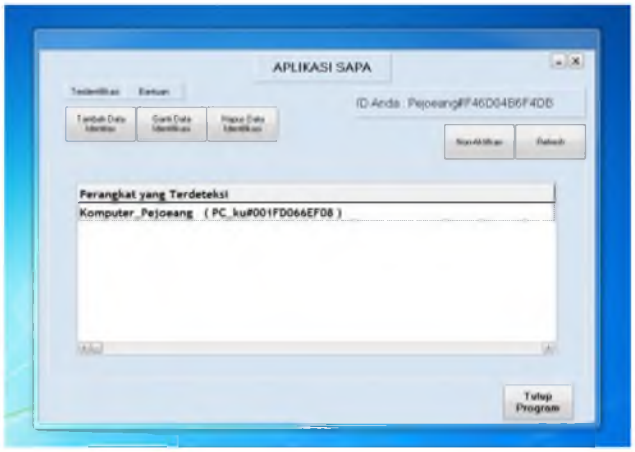
identifikasi dari perangkat. Tampilan rangkaian proses penggantian data identifikasi dapat dilihat pada Gambar 18, Gambar 19, dan Gambar 20.



Gambar 18 Tampilan penggantian data identifikasi perangkat



Gambar 19 Penggantian data identifikasi sukses

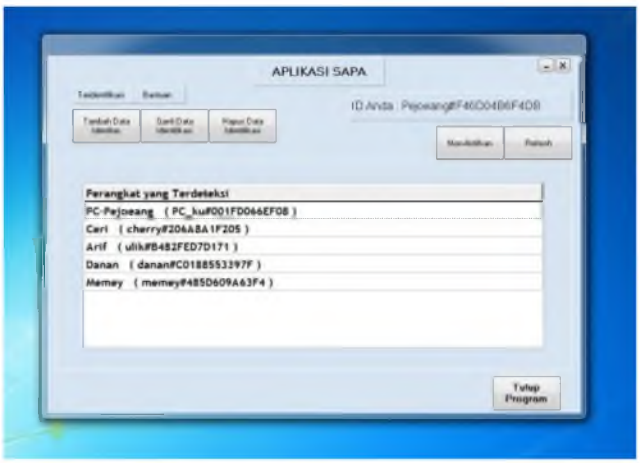


Gambar 20 Identifikasi perangkat setelah data identifikasi diganti

Pengguna dapat menghilangkan data perangkat dari daftar identifikasi untuk membuatnya kembali tidak teridentifikasi. Penghapusan dilakukan dengan menghapus data perangkat dari file "*daftar.txt*". Fungsi ini dapat diakses dengan menekan tombol dengan *caption* "Hapus Data Identifikasi" yang ditunjukan oleh Gambar 21.

Gambar 21 Tampilan proses penghapusan data perangkat

Perangkat dengan Aplikasi Sapa dapat mendeteksi perangkat lain dengan Aplikasi Sapa juga, lebih dari satu perangkat. Tampilan uji coba Aplikasi Sapa saat mendeteksi sebanyak 5 perangkat dapat dilihat pada Gambar 22.



Gambar 22 Aplikasi Sapa saat mendeteksi lebih dari satu perangkat

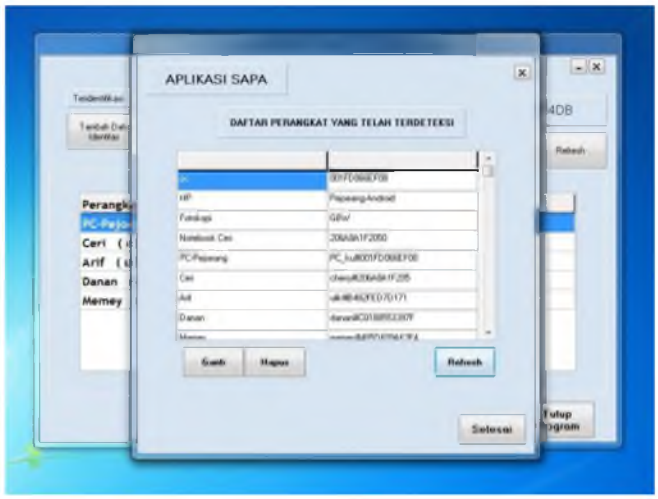
Berikut adalah daftar perangkat yang masuk ke dalam daftar identifikasi:

Tabel 2 Daftar perangkat yang terdeteksi

No.	Nama Perangkat	Identitas	Sistem Operasi
1.	PC Intel Core 2 Duo	PC_ku#001FD066EF08	Windows 7
2.	Asus X42JY	memey#485D609A63F4	Windows 7
3.	Acer Travelmate 4740	cherry#206A8A1F2050	Windows 7

No.	Nama Perangkat	Identitas	Sistem Operasi
4.	Toshiba L645	ulik#B482FED7D171	Windows 7
5.	Asus A43E	danan#C0188553397F	Windows 7

Daftar identifikasi perangkat yang berada pada *file* "*daftar.txt*" dapat dilihat secara *visual* oleh pengguna pada tabel daftar identifikasi yang dapat dilihat pada Gambar 23, dengan menekan pilihan "Teridentifikasi" yang terletak pada pojok kiri atas aplikasi. Pada tabel ini pengguna dapat melihat seluruh perangkat yang telah teridentifikasi dan juga melakukan penggantian atau penghapusan data identifikasi.



Gambar 23 Daftar perangkat yang telah teridentifikasi

Pada sistem operasi Microsoft Windows 7 ini seluruh fungsi dari aplikasi Sapa berjalan sebagaimana mestinya. Data MAC *address* sebagai identitas perangkat pada aplikasi, dapat diambil dan ditampilkan serta menjadi identitas deteksi pada perangkat lain.

1. Pengujian Jarak Tanpa Objek Penghalang (*Line of Sight*)

Setelah dilakukan pengujian fungsi aplikasi, maka selanjutnya dilakukan pengujian jarak. Pengujian dilakukan dengan mengecek pendeteksian pada aplikasi Sapa yang dijalankan pada perangkat *notebook* dengan sistem operasi yang dipakai adalah Windows 7. Setelah itu, perangkat *notebook* digeser menjauh, untuk menguji deteksi perangkat hingga jarak 80 meter. Penghitungan jarak dibantu dengan aplikasi *Smart Distance* pada sistem operasi Android pada *smartphone*. Aplikasi ini dapat mengukur jarak dengan akurat berdasarkan patokan tinggi suatu benda. Aplikasi yang dipakai adalah Smart Distance, dengan tampilan yang dapat dilihat pada Gambar 24.

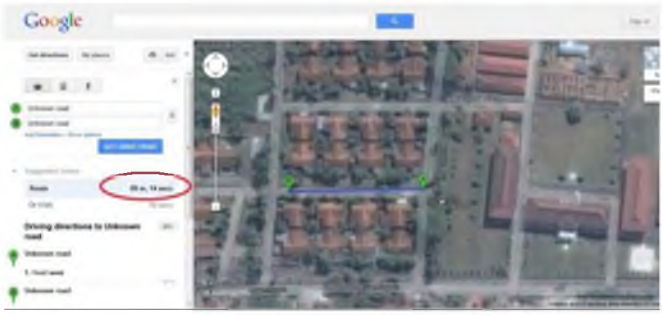
Gambar 24 Tampilan aplikasi *Smart Distance*

Jarak didapat dengan mengambil selisih jarak antara *notebook* dan *notebook* terhadap pintu yang terlihat pada Gambar 23. Berikut adalah tabel data yang didapat dari hasil pengujian.

Tabel 2 Hasil pengujian jarak deteksi (tanpa terhalang)

Pengujian *Notebook* dengan *Notebook*	
Jarak	Status Deteksi
1 Meter	Terdeteksi
2 Meter	Terdeteksi
3 Meter	Terdeteksi
4 Meter	Terdeteksi
5 Meter	Terdeteksi
10 Meter	Terdeteksi
15 Meter	Terdeteksi
20 Meter	Terdeteksi
25 Meter	Terdeteksi
30 Meter	Terdeteksi
35 Meter	Terdeteksi
40 Meter	Terdeteksi
45 Meter	Terdeteksi
50 Meter	Terdeteksi
55 Meter	Terdeteksi
60 Meter	Terdeteksi
65 Meter	Terdeteksi
70 Meter	Terdeteksi
75 Meter	Terdeteksi
78 Meter	Tidak Terdeteksi
80 Meter	Tidak Terdeteksi

Hasil pengujian secara *line of sight* (tanpa objek penghalang) yang ditunjukan pada Tabel 2 menunjukan bahwa gelombang *Wi-Fi* sudah tidak dapat terdeteksi mulai jarak 78 Meter. Pengujian menggunakan *notebook* ASUS A43E dan ASUS X42JY dengan lokasi pengujian yang dapat dilihat pada Gambar 25.



Gambar 25 Peta lokasi pengujian Aplikasi Sapa

2. Pengujian Jarak Dengan Objek Penghalang

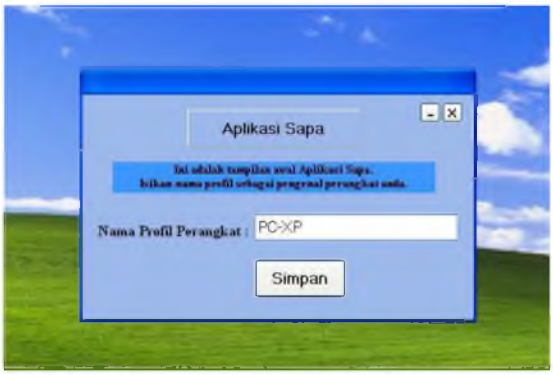
Pengujian dilakukan di kampus Sekolah Tinggi Teknologi Adisutjipto (STTA) Yogyakarta, perangkat 1 yaitu *notebook* ASUS X42JY dengan Aplikasi Sapa aktif diletakan di gedung Wiweko lantai 2 dalam ruangan laboran. Sedangkan perangkat 2 yaitu *notebook* ASUS A43E dengan Aplikasi Sapa aktif bergerak keluar gedung hingga perangkat 2 tidak dapat mendeteksi perangkat 1. Gambaran lokasi pengujian dapat dilihat pada Gambar 26.

Gambar 26 Peta lokasi pengujian Aplikasi Sapa dengan objek penghalang

Objek penghalang dalam pengujian ini adalah tembok dan lantai yang menghalangi pandangan langsung antara perangkat 1 dengan perangkat 2. Pada Gambar 26 titik dengan abjad "B" sebagai posisi perangkat 2 adalah titik ketika perangkat 2 **tidak dapat mendeteksi** perangkat 1(titik dengan abjad "C") dengan jarak 36,1 meter. Sedangkan point dengan abjad "C" adalah titik ketika perangkat 2 berada di lantai 1 gedung atau hampir tepat dibawah perangkat 1. Pada titik ini, perangkat 2 juga **tidak dapat mendeteksi** perangkat 1, titik ini berjarak kurang dari 10 meter.

2. Microsoft Windows XP

Sistem operasi Windows XP merupakan sistem operasi yang diluncurkan sebelum Windows 7. Sistem operasi ini merupakan salah satu sistem operasi yang paling sukses di jajaran sistem operasi yang pernah dirilis oleh Microsoft. Pada sistem operasi ini, Aplikasi Sapa dapat dijalankan, pada awal menjalankan aplikasi jendela untuk memasukan nama profil perangkat muncul dan data sukses dimasukan seperti dapat dilihat pada Gambar 27.



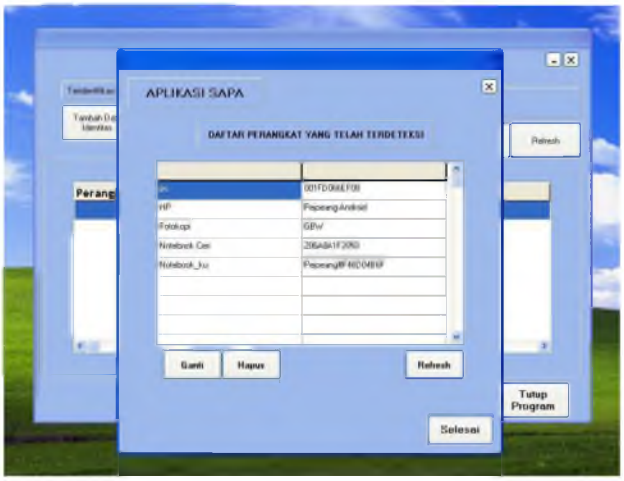
Gambar 27 Jendela pengisian nama profil perangkat pada Sistem Operasi Windows XP

Namun fungsi pembangunan *ad-hoc* dan pendeteksian tidak berjalan. Seperti ditampilkan pada Gambar 28, perangkat *wireless* dapat secara normal mendeteksi keberadaan perangkat *Wi-Fi* namun fungsi pendeteksian pada aplikasi tidak berjalan.

Gambar 28 Aplikasi Sapa pada Sistem Operasi Windows XP

Penyebabnya adalah perintah netsh pada sistem operasi Windows XP tidak berjalan seperti pada sistem operasi Windows 7. Pada sistem operasi Windows XP, perintah netsh untuk membangun sebuah koneksi *ad-hoc* tidak dapat dijalankan. Selain itu perintah netsh lainnya untuk mendeteksi perangkat *Wi-Fi* tidak berjalan. Tidak berjalannya kedua perintah netsh tersebut, membuat proses utama dari aplikasi Sapa tidak dapat berjalan. Namun untuk proses penyaringan data masih berjalan, seperti ditampilkan pada Gambar 30.

Gambar 29 Pengujian aplikasi Sapa antara sistem operasi Windwos 7 dengan sistem operasi Windows XP



Gambar 30 Daftar perangkat teridentifikasi pada sistem operasi Windows XP

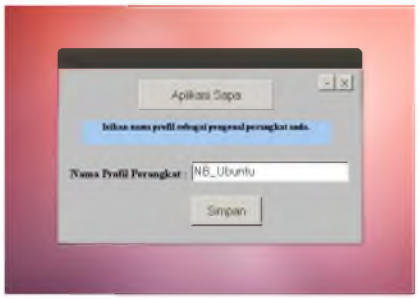
Proses penyaringan data ini masih berjalan karena, proses ini adalah proses pencarian pada *file* "*daftar.txt*" dengan algoritma perulangan yang berjalan pada aplikasi. Jika proses perulangan ini tidak dapat berjalan, maka kemungkinan keseluruhan program tidak dapat berjalan. Pengisian identitas perangkat yang terdeteksi pada *file* "*daftar.txt*" didapat saat aplikasi berjalan pada sistem operasi Windows 7 yang juga terdapat pada perangkat PC yang dipakai.

Perangkat *notebook* dengan sistem operasi Windows 7 dan aplikasi Sapa aktif, tidak dapat mendeteksi perangkat PC juga sebaliknya. Karena tidak berjalannya dua fungsi utama pada aplikasi Sapa. Selain dua fungsi yang telah disebutkan, identitas perangkat yang merupakan MAC *address* perangkat juga tidak dapat ditampilkan, karena aplikasi tidak mendapatkan informasi tersebut.

b. Microsoft Windows 7 dan Ubuntu 12.04

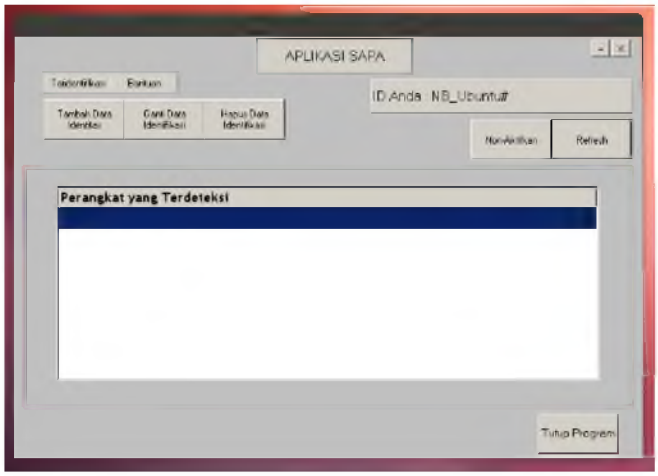
Gambar 31 Aplikasi Sapa pada sistem operasi Ubuntu

Sistem operasi Ubuntu adalah sistem operasi berbasis *open source* yang dikembangkan dengan *kernel* UNIX. Sistem operasi dengan *kernel* UNIX dikenal juga dengan sistem operasi Linux. Pada sistem operasi ini, aplikasi Sapa harus menggunakan bantuan *software* Wine untuk dapat dijalankan. Karena pada sistem operasi Ubuntu, *file* *.*exe* tidak dapat dijalankan begitu saja sebagai *file* yang *executable*. Tampilan berjalannya aplikasi Sapa pada sistem operasi Ubuntu dapat dilihat pada Gambar 31.



Gambar 32 Jendela pengisian nama profil Aplikasi Sapa pada sistem operasi Ubuntu

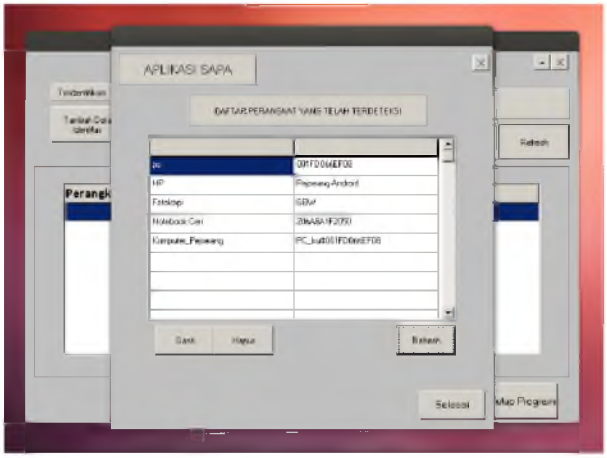
Pertama kali mengaktifkan aplikasi, jendela untuk mengisikan nama profil perangkat muncul, dapat dilihat pada Gambar 32. Proses ini masih dapat berjalan dan data dapat diisikan dengan sukses.



Gambar 33 Aplikasi Sapa dalam kondisi aktif

Pada Gambar 33 terlihat bahwa aplikasi telah berada pada kondisi aktif, namun data identitas perangkat yang merupakan MAC *address* tidak dapat diambil. Begitu juga dengan fungsi pembuatan koneksi *ad-hoc* dan fungsi deteksi perangkat tidak dapat dijalankan, kedua fungsi tersebut menggunakan perintah netsh. Seperti telah dijelaskan bahwa perintah netsh adalah perintah yang terdapat pada sistem operasi Windows. Meskipun telah menggunakan aplikasi bantuan Wine, namun perintah netsh tetap tidak dapat dijalankan. Karena perintah tersebut tidak dikenali, juga perintah *shell command* lain yang dijalankan oleh aplikasi.

Namun seperti halnya pada sistem operasi Windows XP. Pada sistem operasi Ubuntu 12.04 ini, fungsi penyaringan data tetap dapat berjalan seperti terlihat pada Gambar 34.



Gambar 34 Daftar perangkat teridentifikasi yang masih dapat dilihat

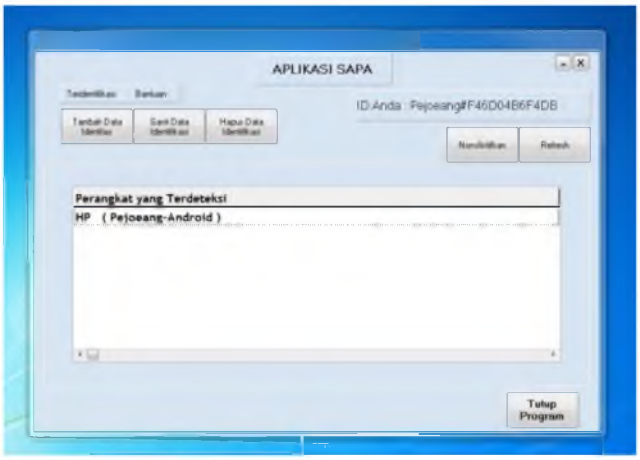
File "daftar.txt" masih dikenali sebagai *textfile* dalam sistem operasi ini. Selain itu proses pembacaan *file* pada aplikasi masih dapat dijalankan sehingga memungkinkan penyaringan *file* berlangsung.

c. *Notebook* dan *Smartphone* Android

Pengujian kali ini bermaksud untuk melihat apakah perangkat *notebook* dapat mendeteksi juga mengidentifikasi *mode tethering* pada *smartphone* Android. *Smartphone* yang digunakan adalah Samsung Galaxy Young GT-S5360 yang menyalakan *mode tethering and portable hotspot*. Sehingga memungkinkan bagi perangkat lain untuk dapat melihat keberadaan perangkat ini juga terhubung ke dalam jaringan *hotspot* yang telah dibangun.

Gambar 35 Pengujian antara *notebook* dengan *smartphone*

Smartphone Android dapat terdeteksi oleh aplikasi Sapa pada perangkat *notebook*. Begitu juga *smartphone* dapat mendeteksi *notebook* namun tidak menggunakan aplikasi Sapa. Karena untuk *smartphone*, aplikasi yang dibuat adalah aplikasi *mobile*. Sedangkan aplikasi Sapa yang diuji adalah aplikasi *desktop*. *Smartphone* dapat dideteksi karena setelah mengaktifkan *mode tethering*, perangkat memiliki SSID yang memang untuk dideteksi sehingga perangkat lain dapat terhubung dan memanfaatkan koneksi yang telah dibangun. Pada Gambar 36 adalah tampilan saat aplikasi mendeteksi perangkat *smartphone*.



Gambar 36 Aplikasi Sapa mendeteksi *smartphone*

d. *Notebook* dan *Access Point*

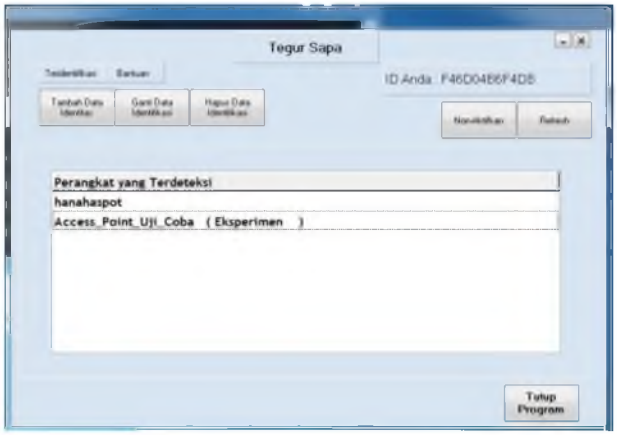
Pengujian ini dimaksudkan untuk melihat kompabilitas aplikasi dengan *mode* gelombang *Wi-Fi* yang dipakai oleh *access point*. Gelombang standar *Wi-Fi* ada empat macam yang dikenali dan telah mendapat sertifikat dari *Institute of Electrical and Electronics Engineers* (IEEE), yaitu 802.11a, 802.11b, 802.11g, dan 802.11n, untuk mendapatkan informasi gelombang yang didukung oleh sebuah *access point* bisa dengan cara melihat pada bagian deteksi *wireless network* dari sistem operasi Windows, lalu sorot dengan *cursor* SSID yang akan dicek, tampilan informasi SSID tersebut muncul beserta gelombang yang dipakai. Pada Gambar 37 adalah tampilan halaman pengaturan *access point* D-Link DWL-2000AP+.

Gambar 37 *Setting mode* gelombang yang akan dipakai pada *Access Point*

Pembagian jenis gelombang ini menurut frekuensi dan modulasi data yang dipakai. Setiap *mode* memiliki karakteristik dan kecepatan transfer data yang berbeda. Khusus untuk standar gelombang 802.11a, standar ini tidak bisa dipakai sembarangan, karena banyak Negara yang tidak melegalkan penggunaan frekuensi yang dipakai oleh standar gelombang ini. *Access point* yang dipakai adalah D-Link DWL-2000AP+ bentuk fisik dapat dilihat pada Gambar 38, dengan SSID "Eksperimen" dan pengujian dilakukan pada jarak +/- 2,5 meter.

Gambar 38 Bentuk fisik *Access Point* D-Link DWL-2000AP+

1. *Mode* Gelombang 802.11g

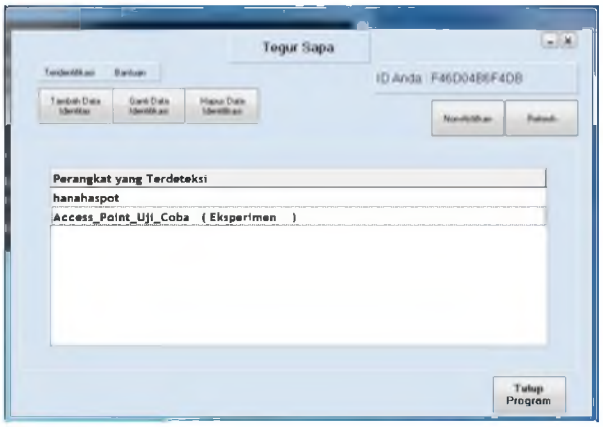


Gambar 39 Pendeteksian *access point* dengan *mode* gelombang 802.11g

Mode gelombang 802.11g merupakan *mode* gelombang yang umum digunakan oleh perangkat saat ini. *Mode* ini menggunakan modulasi *Orthogonal Frequency Division Multiplexing* (OFDM) dan frekuensi 2,400 – 2,495 GHz dengan kecepatan transfer data maksimum hingga 54 Mbps. Pada *mode* ini perangkat dapat mendeteksi keberadaan *access point* seperti dapat dilihat pada Gambar 39, karena merupakan *mode* gelombang yang dapat dideteksi oleh perangkat *Wi-Fi* pada *notebook*.

2. *Mode* Gelombang 802.11 b/g/n (*mix*)

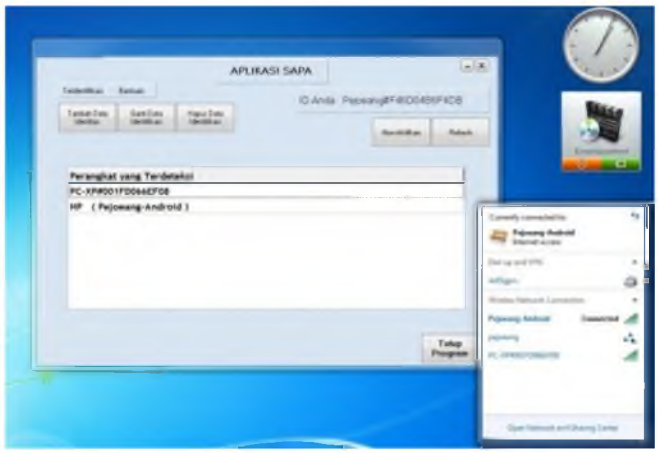
Mode gelombang ini adalah *mode* gelombang campuran yang diaktifkan oleh *access point* agar perangkat dengan *mode* gelombang yang spesifik hanya 802.11b, 802.11g, atau 802.11n saja, dapat mendeteksi dan menggunakan koneksi yang disediakan oleh *access point*. Pendeteksian berjalan lancar dan dapat dideteksi oleh aplikasi Sapa pada *notebook* seperti terlihat pada Gambar 40.



Gambar 40 Pendeteksian *access point* dengan *mode* gelombang 802.11b/g/n (*mix*)

e. Aplikasi Sapa Dalam Jaringan WLAN

Aplikasi Sapa adalah aplikasi yang memanfaatkan koneksi *ad-hoc* untuk dapat dideteksi oleh perangkat lain. Pengujian aplikasi saat perangkat dalam keadaan terkoneksi dalam sebuah jaringan WLAN dilakukan untuk melihat pengaruh terhadap pendeteksian aplikasi. Pengujian dilakukan dengan menghubungkan perangkat *notebook* dengan sebuah *access point* sementara yaitu sebuah perangkat *smarphone* Samsung Galaxy Young dengan *mode tethering and portable hotspot*.



Gambar 41 Aplikasi Sapa saat perangkat telah terkoneksi dalam jaringan WLAN

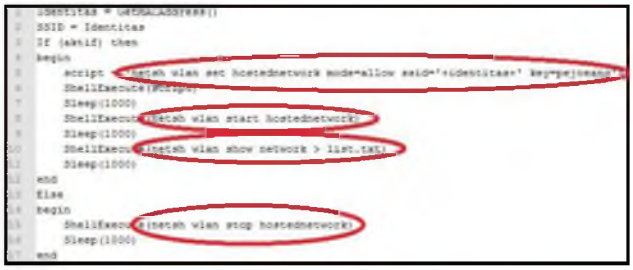
Pada gambar 41 terlihat bahwa *notebook* telah terkoneksi dalam *hotspot* yang disediakan oleh *smartphone* dengan SSID Pejoeang-Android. Namun aplikasi masih tetap dapat mendeteksi perangkat lain dan juga perangkat *smartphone* tersebut dengan normal.

C. Analisa

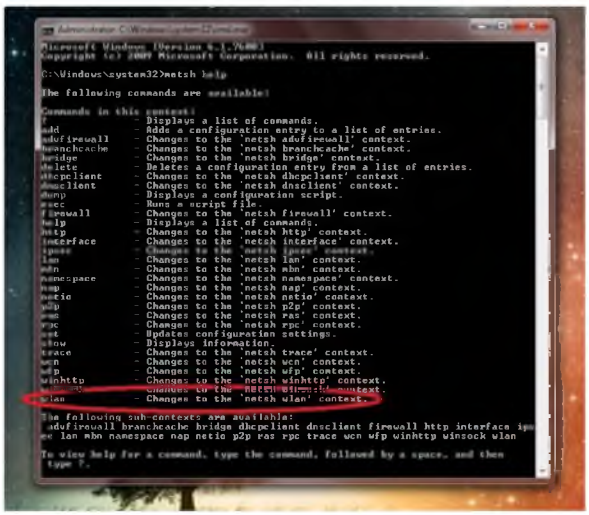
a. Kompabilitas Program

Aplikasi Sapa dapat berjalan dengan sempurna pada sistem operasi Windows 7, tetapi tidak berfungsi pada sistem operasi Windows XP dan Ubuntu 12.04. Karena aplikasi Sapa memanfaatkan perintah netsh untuk proses pembentukan koneksi *ad-hoc* dan juga pada proses pendeteksian perangkat. Pada sistem operasi Windows XP, perintah netsh dapat ditemukan. Tetapi fungsi netsh untuk mengaktifkan *ad-hoc* dan melakukan pendeteksian perangkat tidak ditemukan seperti pada sistem operasi

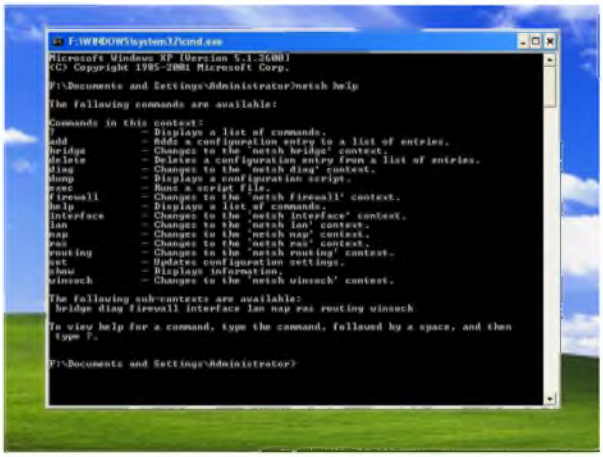
Windows 7. Selain itu, perintah netsh pada sistem operasi Windows XP untuk melihat *driver* perangkat *wireless* perangkat juga tidak dikenali.



Gambar 42 Perintah netsh yang dipakai oleh Aplikasi Sapa



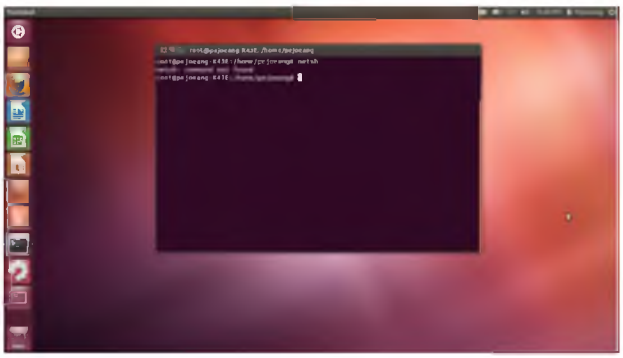
Gambar 43 Perintah netsh pada Windwos 7



Gambar 44 Perintah netsh pada Windows XP

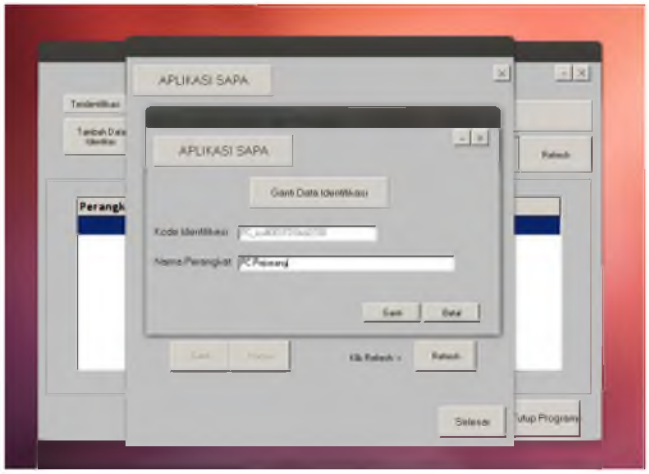
Pada Gambar 42 memperlihatkan baris program yang menggunakan perintah netsh dan dijalankan sebagai *shell command*. Keempat perintah tersebut tidak terdapat pada sistem operasi Windows XP, meskipun sistem operasi tersebut memiliki fungsi netsh. Seperti yang ditampilkan oleh Gambar 43, merupakan perintah netsh pada Windows 7. Terdapat perintah "wlan" untuk jaringan *wireless* yang tidak terdapat pada sistem operasi Windows XP. Perintah netsh pada Windows XP dapat dilihat pada Gambar 44.

Pendeteksian aplikasi terhadap perangkat *smartphone* bergantung kepada kemampuan dari *smartphone* tersebut untuk dapat menciptakan sebuah *mode* pembagian koneksi jaringan melalui gelombang *Wi-Fi*. Pada pengujian ini digunakan *smartphone* dengan sistem operasi Android yang memiliki fitur berbagi koneksi jaringan internet dengan istilah "*Tethering and Portable Hotspot*". Perangkat *smartphone* dengan sistem operasi selain Android mungkin memiliki istilah lain.

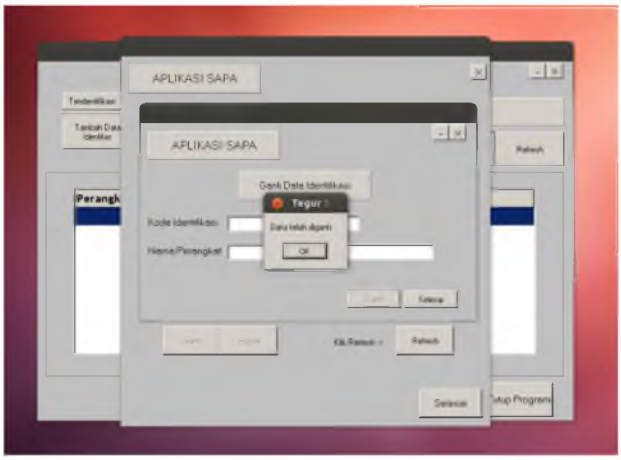


Gambar 45 Perintah netsh yang tidak dikenali di sistem operasi Ubuntu

Pada sistem operasi Ubuntu 12.04, aplikasi Sapa tidak dapat berjalan, seperti dapat dilihat pada Gambar 45, perintah netsh tidak terdapat pada sistem operasi ini. Namun fungsi-fungsi lain seperti menampilkan data perangkat yang telah teridentifikasi, mengganti data identifikasi perangkat, dan penghapusan data identifikasi perangkat masih dapat dijalankan. Pada Gambar 46 dan Gambar 47, proses penggantian data identifikasi masih dapat dilakukan dan sukses. Fungsi untuk menambahkan perangkat ke dalam daftar identifikasi tidak dapat berfungsi, karena tidak ada perangkat yang dapat dideteksi oleh aplikasi.



Gambar 46 Proses penggantian data identifikasi yang masih dapat berjalan



Gambar 47 Proses penggantian data identifikasi sukses

Tabel perbandingan kompabilitas Aplikasi Sapa terhadap perbedaan sistem operasi yang dipakai dapat dilihat pada Tabel 3. Perangkat yang bertugas mendeteksi dan dideteksi adalah perangkat *notebook* ASUS A43E dengan sistem operasi Windows 7.

Tabel 3 Perbandingan berjalannya fungsi pada Aplikasi Sapa berdasarkan sistem operasi yang dipakai

No.	Sistem Operasi	Perangkat	Mendeteksi Perangkat Lain	Terdeteksi Perangkat Lain	Fungsi Lainnya Berjalan?
1.	Windows 7	PC dan Notebook	Ya	Ya	Ya
2.	Windows XP	PC	Tidak	Tidak	Ya
3.	Ubuntu 12.04	Notebook	Tidak	Tidak	Ya

Berdasarkan gelombang *Wi-Fi* yang dipakai, maka aplikasi Sapa bergantung pada kompabilitas dari perangkat itu sendiri. Khususnya untuk standar gelombang 802.11a. Karena gelombang ini memiliki frekuensi yang berbeda dengan standar 802.11b dan 802.11g. Gelombang 802.11a berjalan pada frekuensi 5,745 dan 5,805 GHz. Selain itu, standar gelombang ini bersifat tidak legal digunakan pada sebagian besar Negara di dunia. Khusus untuk di Indonesia, perangkat *notebook* dan *access point* umum menggunakan standar gelombang 802.11b, 802.11g, dan 802.11n. Jadi jika gelombang yang dipakai oleh *access point* sesuai dengan gelombang yang dimiliki oleh *adapter Wi-Fi* pada perangkat, maka aplikasi Sapa dapat mendeteksi dan memanfaatkan gelombang tersebut. Terhubungnya perangkat pada sebuah jaringan WLAN tidak mempengaruhi aplikasi dalam mendeteksi dan dideteksi. Karena seperti telah ditunjukan pada Gambar 36, bahwa aplikasi ini menggunakan fungsi netsh untuk mengaktifkan koneksi *ad-hoc*. Koneksi *ad-hoc* ini membuat komputer secara mandiri dapat dideteksi. Saat mendeteksi perangkat lain, aplikasi ini mengandalkan fungsi netsh dalam pendeteksiannya. Pendeteksian perangkat *wireless*

oleh fungsi netsh mengandalkan kemampuan dari *wireless adapter* komputer itu sendiri untuk dapat mendeteksi perangkat lain.

Aplikasi Sapa berjalan pada lapisan *application* pada standar jaringan OSI maupun TCP/IP. Sehingga aplikasi ini sangat bergantung pada kompabilitas *hardware* perangkat dan sistem operasi yang dipakai. Pada perangkat yang mendukung aplikasi Sapa dari segi *hardware* dan sistem operasi, diharuskan untuk mengaktifkan perangkat *wireless* perangkat tersebut, untuk menghindari kegagalan sistem yang bisa membuat komputer mengalami *bluescreen* meskipun tidak selalu terjadi.

b. Jarak Jangkauan

Jangkauan pendeteksian aplikasi, sesuai dengan jangkauan dari perangkat yang dipakai. Pada jarak 75 meter aplikasi masih dapat melakukan deteksi, namun pada jarak 78 meter gelombang *Wi-Fi* sudah tidak terdeteksi. Pengujian jarak jangkauan ini menggunakan jarak perangkat secara *line of sight*, artinya perangkat tidak terhalangi objek seperti tembok. Sehingga masih dapat terlihat secara langsung. Karena bagi gelombang *Wi-Fi*, keberadaan penghalang seperti tembok, sangat memperngaruhi jangkauan rambat gelombang. Pada pengujian dengan objek penghalang, terdapat dua kejadian saat gelombang *Wi-Fi* sudah tidak terdeteksi, yaitu:

1. Pada saat perangkat 2 berada di lantai 1 gedung (ilustrasi dapat dilihat pada Gambar 4.18). Kejadian ini dimungkinkan, karena gelombang *Wi-Fi* tidak dapat menembus tembok beton yang menjadi penopang gedung tersebut.
2. Pada jarak 36,1 meter dari titik perangkat 1. Pada jarak ini gelombang *Wi-Fi* dari perangkat 1 sudah tidak terdeteksi oleh perangkat 2. Karena adanya objek penghalang seperti pintu, tembok (bukan beton), dan jendela. Telah mengurangi kekuatan jangkauan dari gelombang, hingga hanya bisa menjangkau kurang dari 36 meter.

Standar gelombang yang dipakai berpengaruh terhadap jarak jangkauan yang dimiliki oleh perangkat. Perangkat yang memiliki standar gelombang 802.11n memiliki jarak jangkauan yang lebih jauh dibanding dengan standar gelombang 802.11g. Menurut sumber, standar gelombang 802.11n memiliki jangkauan hingga 250 meter (*outdoor*), sedangkan pada standar gelombang 802.11g memiliki jangkauan hingga 140 meter (*outdoor*).

Setelah dilakukan pengujian, jarak yang dapat dijangkau adalah hingga 77 meter (lihat Tabel 1). Perbedaan ini (dengan sumber) dimungkinkan karena pemancaran gelombang *Wi-Fi* sangat berpengaruh dengan kekuatan pancar dari perangkat itu sendiri. Pada perangkat *notebook* yang tidak memiliki antena luar khusus, tentu akan kurang kekuatan pancarnya dibandingkan dengan perangkat *access point* yang memiliki antena luar khusus. Juga sumber tenaga listrik yang dipakai oleh perangkat. Pada perangkat *notebook* sumber tenaga listrik adalah baterai, sedangkan perangkat *access point* memakai tenaga listrik langsung dari *wall socket*.

5. Kesimpulan

Setelah dilakukan pengujian terhadap aplikasi Sapa, maka dapat diambil kesimpulan sebagai berikut:

1. Aplikasi Sapa memungkinkan terjadinya komunikasi satu arah dengan banyak perangkat.
2. Algoritma Sapa yang digunakan pada aplikasi ini dapat dijalankan pada komputer yang *wireless card adapter*-nya aktif. Aplikasi dapat bekerja pada TCP/IP dan UDP dengan menggunakan mekanisme *broadcast* terhadap SSID yang berbasis MAC *address*.
3. Algoritma Sapa pada aplikasi ini dapat bekerja pada jaringan *ad-hoc* dengan perangkat *notebook* atau PC (dengan tambahan *wireless card adapter*) yang memiliki sistem operasi Microsoft Windows 7 dan *driver* perangkat *wireless* yang telah diperbaharui.
4. Setelah dilakukan pengujian, jarak maksimal untuk dapat mendeteksi perangkat menggunakan aplikasi Sapa adalah 77 meter. Kekuatan pancar gelombang *Wi-Fi* dari setiap perangkat sangat mempengaruhi jarak jangkauan pendeteksian.

6. Saran

Saran yang dapat diambil untuk pengembangan dari tugas akhir ini kedepannya antara lain:

1. Algoritma Sapa dapat dikembangkan untuk dapat mendeteksi perangkat secara *connectionless* tanpa menggunakan fitur *ad-hoc*. Aplikasi dapat dikembangkan untuk ditanam pada perangkat *smartphone*.
2. Memperbaiki algoritma penyaringan data dan identifikasi perangkat untuk mengurangi kemungkinan terdapatnya *bug* pada saat penyimpanan, perubahan, dan penghapusan data dari *file*.
3. Pembaharuan *driver* perangkat *Wi-Fi* diperlukan untuk dapat mendukung berjalannya fungsi netsh dan Microsoft *Virtual Wi-Fi Miniport Adapter* yang dijalankan dan dibutuhkan oleh aplikasi Sapa.

7. Daftar Pustaka

Febrian Jack., Januari 2004., *Kamus Komputer dan Teknologi Informasi*, Penerbit Informatika Bandung, Bandung, Indonesia.

Kuntoro Priyambodo Tri dan Heriadi Dodi., 2005., *Jaringan Wi-Fi*, Penerbit Andi Offset, Yogyakarta, Indonesia.

Purbo Onno W dan Tanuhandaru Protus., Desember 2007., *Jaringan Wireless di Dunia Berkembang*, Penerbit Andi Offset, Yogyakarta, Indonesia.

Stallings William., 2001., *Komunikasi Data dan Komputer Dasar-Dasar Komunikasi Data*, Penerbit Salemba Teknika, Jakarta, Indonesia.

Tanenbaum Andrew S., 1997., *Jaringan Komputer*, Penerbit Prentice-Hall, New Jersey.

http://en.wikibooks.org/wiki/Communication_Networks/TCP_and_UDP_Protocols (diakses pada 6 November 2012)

http://www.microsoft.com/resources/documentation/windows/xp/all/proddocs/en-us/netsh.mspx?mfr=true (diakses pada Januari 2013)

http://linkyrt.blogspot.com/2012/04/kecepatan-perbedaan-dan-tipe-80211-abgn.html (diakses pada 26 Februari 2013)