

Aplikasi *Local Area Network Messenger* dengan Java

Merliana, Timotius Witono

Jurusan S1 Teknik Informatika

Fakultas Teknologi Informasi, Universitas Kristen Maranatha

Jl. Prof. Drg. Suria Sumantri no. 65 Bandung 40164

email: sllxe@yahoo.com, timotius@itmaranatha.org

Abstract

Local Area Network (LAN) is a local network within a building or an office that is often used to connect two or more computers. One of the objectives of having computer connected to each other within the local network is to have a media communications and make the information distribution process easier. But occasionally when a user wants to communicate a data to others who are in the same local network, user tries to find or meet the person manually. Or when the user needs a file that is located on another computer, user has to move to a computer where the file is located. There are some users connected in a local network that has not been utilizing the benefits of a LAN connection. Because of that, a Local Area Network Messenger application is being developed. This application is equipped with four main features which are instant messaging, voice call, screen sharing and file transfer. Instant message feature enable a user to send text messages to one another. Voice call feature allows the user to conduct voice communication with other users. Through screen sharing feature, a user is able to view the screen activity from other users. In the file transfer feature, a user can send multiple files to multiple users on the network. Those features are implementing the advantages and benefits of local networks, particularly in addressing the problem about time and distance. By implementing this application, the information distribution process becomes faster and easier because users no longer need to switch places every time. And this application can be a good alternative media communications.

Keyword: messenger, Local Area Network, communication

1. Pendahuluan

Jaringan komputer telah menjadi bagian dari kehidupan sehari - hari karena perkembangan yang terjadi di dalam teknologi informasi serta adanya kebutuhan akan pemanfaatan sumber daya bersama melalui teknologi. *Local Area Network (LAN)* merupakan jaringan lokal di dalam sebuah gedung atau kantor yang sering digunakan untuk menghubungkan dua atau lebih komputer. Salah satu tujuan dari komputer yang saling terhubung di dalam jaringan lokal adalah untuk memiliki sarana telekomunikasi dan proses distribusi informasi yang mudah dan cepat. Tetapi masih sering terjadi saat seorang *user* yang terhubung dalam jaringan lokal harus mengkomunikasikan suatu data yang sedang dikerjakannya kepada orang lain yang berada di dalam jaringan lokal yang sama, *user* tersebut tetap mencari atau menemui orang yang bersangkutan secara manual. Atau pun saat *user* memerlukan suatu *file* yang ternyata tidak ada dalam komputernya dan ada pada komputer yang

lain, *user* berpindah tempat kerja ke komputer dimana *file* tersebut tersedia. Masih ada sebagian pengguna komputer yang saling terhubung di dalam *local area network* yang belum memanfaatkan keuntungan dari koneksi LAN.

Pembuatan aplikasi *Local Area Network Messenger* memiliki beberapa tujuan, diantaranya :

1. Mengembangkan aplikasi yang menerapkan manfaat dari *Local Area Network*.
2. Menyediakan alternatif sarana komunikasi yang efektif.
3. Memberi kemudahan dalam proses distribusi informasi dan pertukaran data atau *file* antara *user* yang saling terhubung dalam *local area network*.

2. Dasar Teori

2.1. Java

Teknologi java merupakan sebuah bahasa pemrograman dan *platform*. Dalam bahasa pemrograman java, seluruh *source code* berakhir dengan ekstensi .java. Yang kemudian di-*compile* menjadi .class oleh *javac compiler*. Dua produk utama dalam *platform* Java SE (*Standard Edition*) adalah : *Java Development Kit* (JDK) dan *Java SE Runtime Environment* (JRE). JDK merupakan *superset* dari JRE, dan berisi semua yang ada di dalam JRE, ditambah *compiler* dan *debugger* yang diperlukan untuk mengembangkan applet dan aplikasi. *Java Runtime Environment* (JRE) menyediakan *libraries*, *Java Virtual Machine*, dan komponen lainnya untuk menjalankan applet dan aplikasi yang ditulis dalam bahasa pemrograman java.

2.2. Java Media Framework (JMF)

Java Media Framework (JMF) adalah sebuah *application programming interface* (API) untuk menangani *time-based* media dalam aplikasi Java. JMF juga mendefinisikan *optional* RTP API untuk memungkinkan transmisi dan penerimaan RTP *stream*.

Model tersebut mempertimbangkan tiga tahap dalam aliran pengolahan data : *input*, pengolahan dan *output*. Tahap *input* dimaksudkan untuk memperoleh data media. Data media dapat diperoleh dari sumber yang berbeda berikut:

1. Dari *capture device* (seperti mikrofon atau kamera)
2. Dari *file* (music.wav)
3. Dari jaringan (menerima RTP *stream*)

Tahap pengolahan mengambil data yang didapat pada tahap input dan menerapkan beberapa proses pengolahan seperti:

1. *Multiplexing / demultiplexing*
2. *Encoding / decoding*
3. *Packetizing / depacketizing*

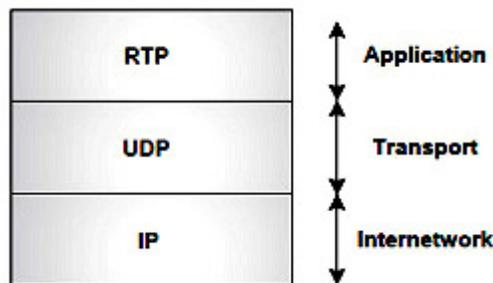
Tahap *output* bertugas untuk mengirimkan data media kepada tujuan. Tujuan yang memungkinkan adalah:

1. Sebuah *presentation device* (*soundcard* dan *speaker*)
2. Sebuah *file*
3. Jaringan (mengirimkan data media sebagai *RTP stream*)

JMF memungkinkan *programmer* untuk mengkonfigurasi skenario pengolahan media yang menggabungkan *input*, *output*, dan pilihan pengolahan yang berbeda. JMF juga menawarkan *high-level* API untuk mengelola data *capture*, presentasi, dan pengolahan *time-based* media.

2.3. RTP

Real-time Transport Protocol (RTP) adalah protokol standar *Internet Engineering Task Force* (IETF STD 64, RFC 3550) yang menyediakan *end-to-end* jasa pengiriman untuk data dengan karakteristik *real-time*, seperti suara dan video. RTP memiliki konsep RTP Session, sebuah RTP Session diidentifikasi oleh transport address dan mencakup hanya satu jenis media. RTP berjalan di atas UDP, sebuah paket RTP terdiri dari *header* dan *data payload*. *Data payload* berisi kode aktual suara atau video, sedangkan *header* berisi informasi yang dibutuhkan untuk memberikan pelayanan yang disediakan oleh protokol. (Perea, 2008)



Gambar-1 RTP (Perea, 2008)

2.4. Port

Address adalah segala yang diperlukan jika setiap komputer tidak melakukan lebih dari satu hal pada saat yang bersamaan. Namun, modern komputer melakukan berbagai hal yang berbeda sekaligus. Email harus dipisahkan dari FTP *requests*, yang juga harus dipisahkan dari *web traffic*. Hal ini dapat dicapai melalui port. Setiap komputer dengan IP Address memiliki ribuan port logis. Setiap port diidentifikasi dengan angka antara 1 hingga 65535. Setiap port dapat dialokasikan ke layanan tertentu. Misalnya HTTP, protokol yang mendasari Web, umumnya menggunakan port 80. (Harold, 2004)

2.5. Multicast

Multicasting lebih luas daripada *unicast*, komunikasi *point-to-multipoint* tetapi lebih sempit dan *targetted* dibandingkan *broadcast communication*. *Multicasting* mengirimkan data dari satu *host* ke banyak *host* yang berbeda, namun tidak ke semua orang. Data hanya pergi ke klien yang telah menyatakan ketertarikannya dengan bergabung dengan kelompok *multicast* tertentu.

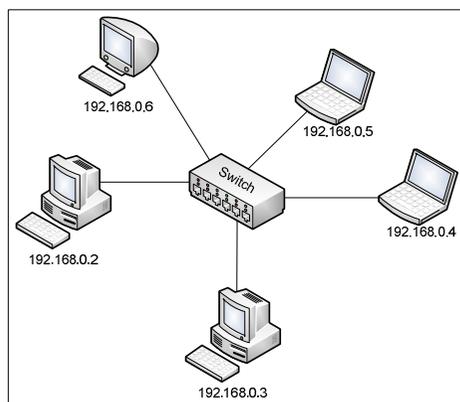
Multicast address adalah alamat bersama sekelompok *host* yang disebut *multicast group*. *Multicast address* adalah IP Address dalam rentang 224.0.0.0 sampai 239.255.255.255. Mereka disebut Class D Addresses untuk membedakan mereka dari Class A, B dan C yang lebih umum. Seperti alamat IP, alamat *multicast* dapat memiliki nama *host*, misalnya alamat *multicast* 224.0.1.1 (alamat dari *Network Time Protocol distributed service*) yang diberi nama *ntp.mcast.net*. Sebuah *multicast group* adalah sekumpulan *host* yang berbagi alamat *multicast*. Setiap data yang dikirim ke alamat *multicast* ini disampaikan kepada semua anggota kelompok. Keanggotaan dalam *multicast group* terbuka, *host* dapat masuk atau meninggalkan kelompok setiap saat. (Harold, 2004)

3. Analisis dan Pemodelan

Dalam teknologi komunikasi, jaringan memiliki peranan yang penting dan merupakan dasar dari proses komunikasi. Tetapi ada beberapa pengguna komputer yang saling terhubung di dalam LAN yang belum menyadari dan memanfaatkan keuntungan dari koneksi jaringan lokal tersebut.

LAN Messenger adalah sebuah aplikasi yang mampu mengimplementasikan beberapa keuntungan dan manfaat dari jaringan lokal. Beberapa kelebihan dari aplikasi ini adalah aplikasi ini berbasis *peer-to-peer* (P2P) sehingga tidak perlu menggunakan *server* ataupun *internet connection*. Aplikasi ini dilengkapi dengan fitur *instant messaging* (IM), *voice call*, *file transfer* dan *screen sharing*. Fitur – fitur tersebut mengimplementasikan keuntungan dan manfaat dari jaringan lokal, khususnya dalam mengatasi permasalahan di sisi waktu dan jarak.

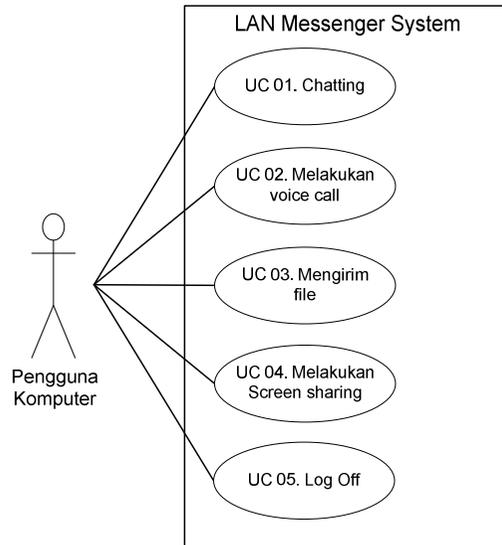
3.1. Topologi



Gambar-2 Topologi

Topologi yang digunakan adalah star topologi. Dimana setiap *node* terhubung langsung ke pusat jaringan. Contoh pada Gambar-2 dimana setiap komputer terhubung ke *switch*.

3.2. Use Case



Gambar-3. Use Case

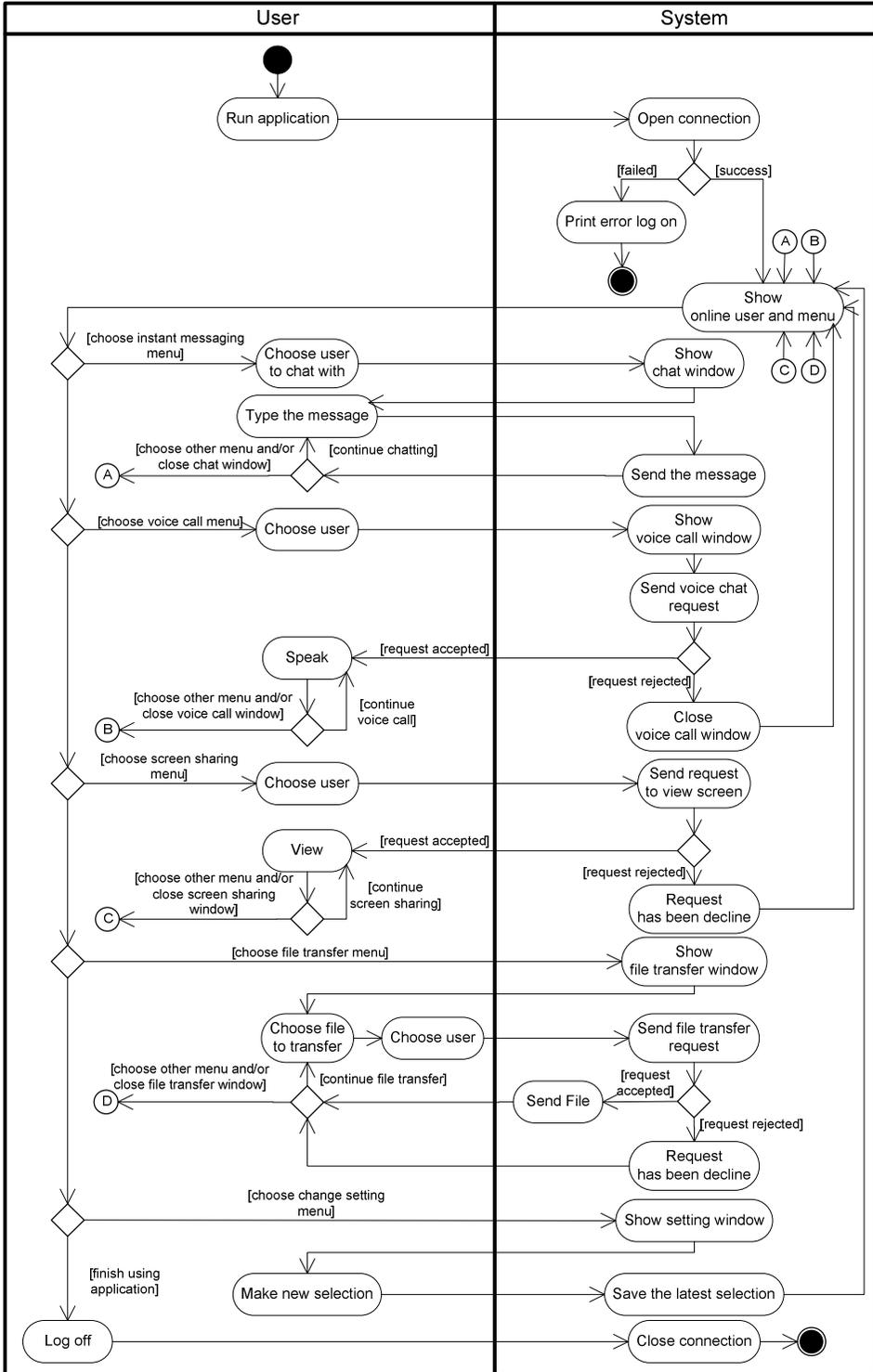
Kode	Nama	Deskripsi
UC 01	<i>Chatting</i>	Dengan <i>use case</i> ini pengguna komputer dapat saling berkirim pesan teks dengan satu atau lebih pengguna komputer lainnya yang juga <i>online</i> .
UC 02	Melakukan <i>voice call</i>	Dengan <i>use case</i> ini pengguna komputer dapat saling berbicara seperti melakukan panggilan telepon dengan pengguna komputer lainnya yang juga <i>online</i> .
UC 03	Mengirim <i>file</i>	Dengan <i>use case</i> ini pengguna komputer dapat mengirimkan <i>multiple file</i> kepada satu atau lebih pengguna komputer lainnya yang juga <i>online</i> secara bersamaan.
UC 04	Melakukan <i>screen sharing</i>	Dengan <i>use case</i> ini pengguna komputer dapat melihat tampilan <i>screen</i> pengguna komputer lainnya yang juga <i>online</i> .
UC 05	<i>Log off</i>	<i>Use case</i> ini digunakan ketika pengguna komputer sudah selesai menggunakan aplikasi.

Tabel-1 Use Case

3.3. Activity Diagram

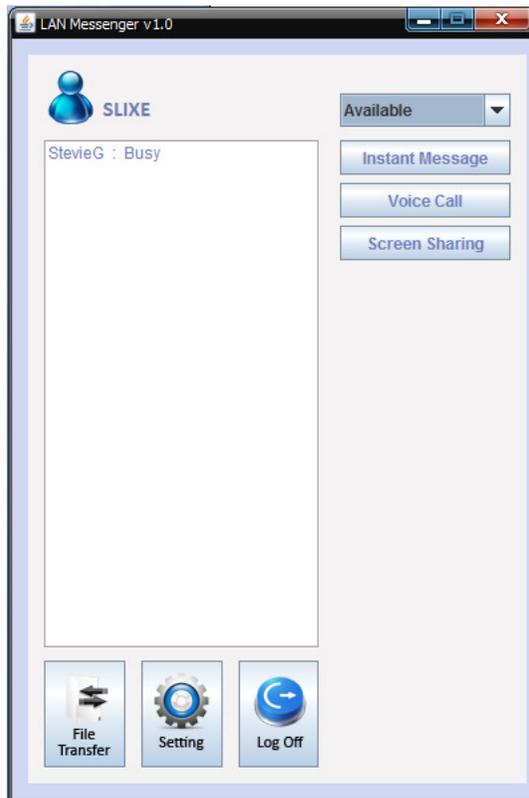
Berikut akan dijelaskan secara umum mengenai *activity diagram* sistem secara keseluruhan:

1. User melakukan aktivitas *Run Application* dan dilanjutkan dengan sistem melakukan aktivitas *Open Connection*
2. Saat aktivitas *Open Connection* selesai, sistem berlanjut pada *decision node*
 - a. Jika *success*, maka sistem akan melakukan aktivitas *Show Online User and Menu* (lanjut ke langkah 3)
 - b. Jika *failed*, maka sistem akan melakukan aktivitas *Print Error Log On* dan berlanjut pada *final node* dari aliran *activity diagram*
3. Saat aktivitas *Show Online User and Menu* oleh sistem telah selesai, user berlanjut pada *decision node* untuk memilih
 - a. *Instant Messaging*
 - b. *Voice Call*
 - c. *Screen Sharing*
 - d. *File Transfer*
 - e. *Change Setting*
 - f. *Log Off*



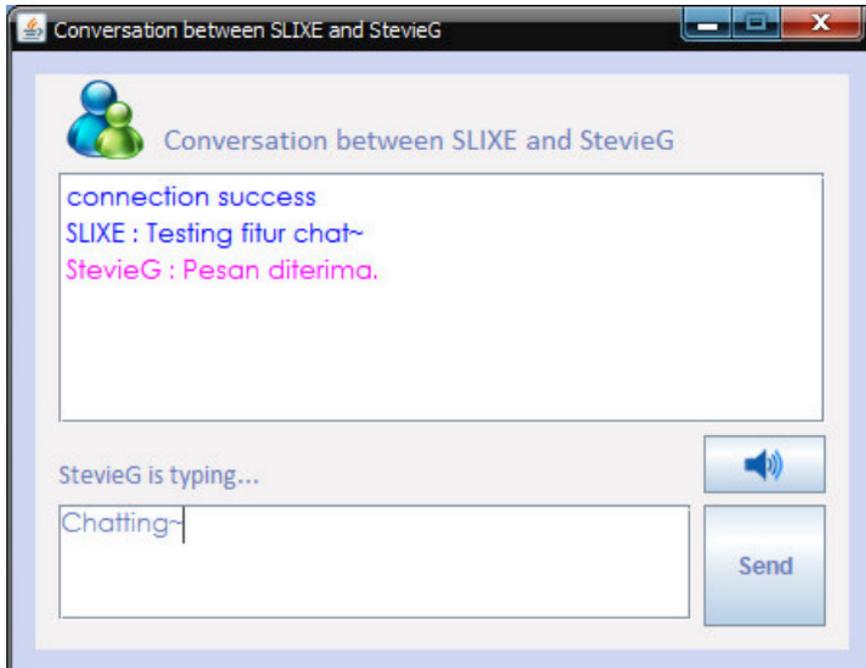
Gambar-4 Activity Diagram sistem secara keseluruhan

4. Perancangan dan Implementasi



Gambar-5 Tampilan Utama

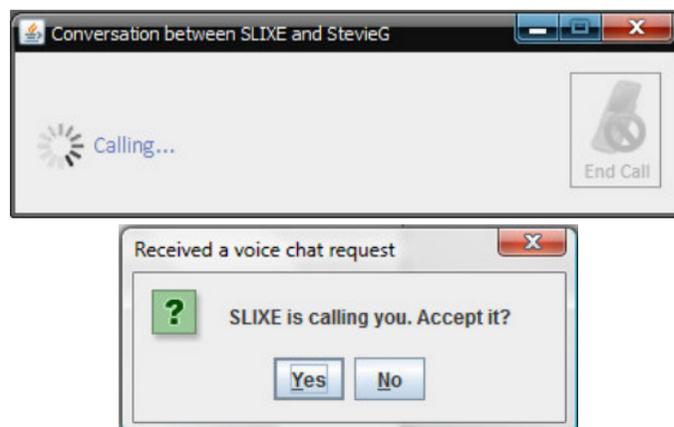
Gambar-5 adalah tampilan *main form* saat *user* pertama kali menjalankan aplikasi. Pada *main form* ini, *user* dapat melihat *list* dari *online user* (pengguna komputer lain dalam jaringan lokal yang juga sedang menggunakan aplikasi ini) dan pilihan menu yang terdapat pada aplikasi ini.



Gambar-6 Tampilan *Instant Message*

Gambar-6 adalah *form conversation* yang ditampilkan ketika user memilih menu *instant message*. User dapat mengetikkan pesan yang akan dikirim pada *text field*, kemudian menekan tombol *send* atau menekan tombol *Enter* pada *keyboard* untuk mengirim pesan. Saat user lain sedang mengetik, akan ditampilkan notifikasi “*user is typing...*”.

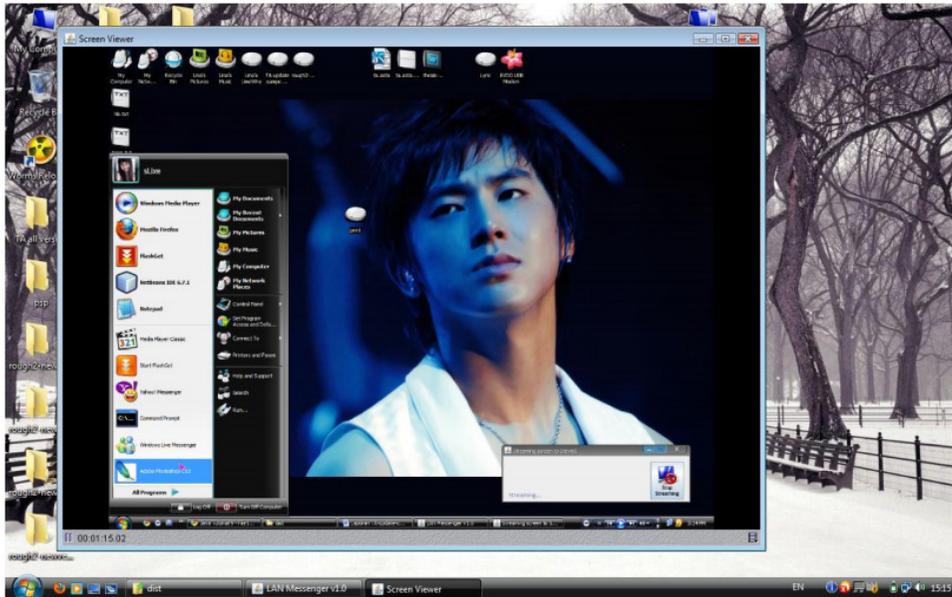
Tombol  berguna untuk mengirimkan nada kejut.



Gambar-7 Tampilan *Voice Call*

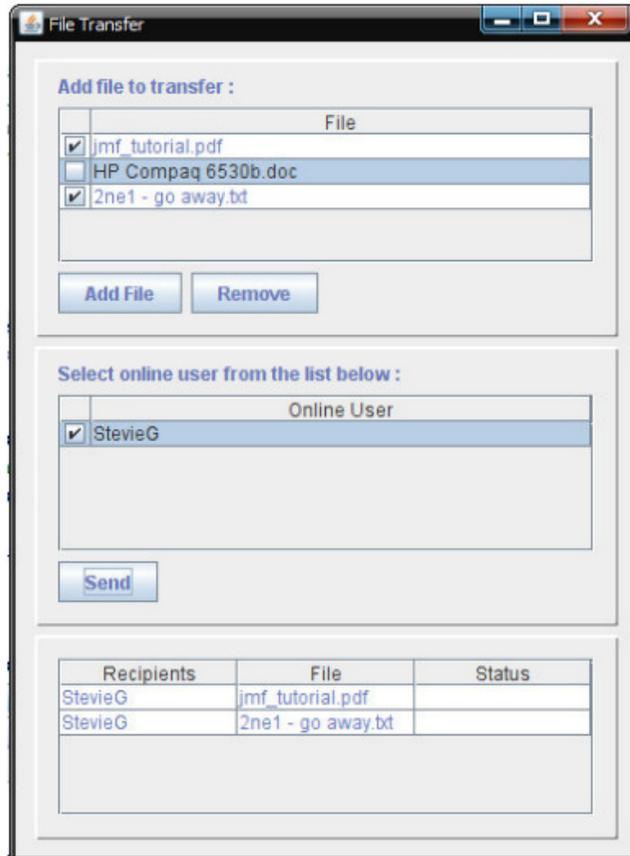
Gambar-7 sebelah kiri adalah *form* yang ditampilkan saat user memilih menu *voice call*. Form ini mengirimkan *request* untuk melakukan koneksi VoIP kepada user

yang dituju. Gambar-7 sebelah kanan adalah *form* yang muncul saat user menerima *request* koneksi VoIP dari *user* lain. *User* dapat memutuskan untuk menerima atau menolak *request* tersebut.



Gambar-8 Tampilan *Screen Sharing*

Gambar-8 adalah tampilan *screen sharing* yang akan muncul ketika *request* untuk melakukan koneksi *screen sharing* diterima dan proses *streaming* mulai berlangsung.



Gambar-9 Tampilan File Transfer

Gambar-9 adalah tampilan ketika user berhasil melakukan proses *file transfer*. Pada tabel paling bawah dari Gambar-9, akan ditampilkan nama user yang menerima file dan nama file yang dikirimkan.

5. Pengujian

Dari 20 koresponden, 8 orang laki – laki dan 12 orang perempuan, dengan rentang usia antara 21 – 24 tahun. Berdasarkan analisa data kuesioner yang berisi *feedback* dari *user* yang telah mencoba aplikasi ini, dapat disimpulkan bahwa:

1. Aplikasi ini mudah digunakan dan *user* tidak mengalami kesulitan dalam memahami menu – menu yang terdapat di dalam aplikasi.
2. Tampilan *user interface* cukup baik, tetapi belum begitu menarik.
3. Fitur yang dimiliki oleh aplikasi sudah lengkap.
4. *Error checking* dan *error prevention* cukup baik.
5. *User* menyetujui bahwa aplikasi ini sudah mengimplementasikan manfaat dari *local area network* dan mampu mengatasi permasalahan di sisi waktu dan jarak dalam hal distribusi informasi.

6. *User* setuju bahwa aplikasi ini dapat dikategorikan sebagai alternatif sarana komunikasi yang efektif.
7. *User* juga berpendapat bahwa fitur yang menjadi keunggulan aplikasi ini dibandingkan dengan aplikasi LAN *Messenger* lainnya adalah fitur *screen sharing*.

6. Kesimpulan dan Saran

Kesimpulan yang didapat setelah proses penelitian, perancangan dan implementasi aplikasi *local area network messenger* adalah sebagai berikut :

1. Aplikasi yang dikembangkan sudah menerapkan manfaat dari *local area network*.
2. Aplikasi dapat mengatasi permasalahan di sisi waktu dan jarak dalam hal distribusi informasi.
3. Aplikasi ini merupakan alternatif sarana komunikasi yang efektif.
4. *Java Media Framework* (JMF) adalah pilihan yang baik untuk menangani *time-based* media dalam aplikasi Java.

Saran yang didapat untuk mengembangkan aplikasi ini menjadi lebih baik adalah sebagai berikut :

1. Tampilan *user interface* dibuat lebih menarik.
2. *List online user* dilengkapi dengan foto atau status.
3. Fitur *instant messaging* dilengkapi dengan *emoticon*.
4. Dikembangkan fitur *chat conference* atau *webcam*.

Daftar Pustaka

- [Har04] Harold, Elliotte Rusty. (2004). *Java Network Programming* (3rd ed.). California: O'Reilly Media, Inc.
- [Par96] Parker, T. (1996). *Teach Yourself TCP/IP in 14 Days* (2nd ed.). Indianapolis: Sams Publishing.
- [Per08] Perea, Rogelio Martínez. (2008). *Internet Multimedia Communications Using SIP : A Modern Approach Including Java® Practice*. Massachusetts: Morgan Kaufmann Publishers.
- [Rei02] Reilly, D & Reilly, M. (2002). *Java™ Network Programming and Distributed Computing*. Boston: Pearson Education, Inc.
- [Sch04] Schmuller, J. (2004). *Teach Yourself UML in 24 Hours* (3rd ed.). Indianapolis: Sams Publishing.
- [Sun10a] Sun Microsystem. (2010). *About the Java Technology*. Dipetik Mei 11, 2010, dari A Sun Developer Network Site. [Java.sun.com : Electronic references. : http://java.sun.com/docs/books/tutorial/getStarted/intro/definition.html](http://java.sun.com/docs/books/tutorial/getStarted/intro/definition.html)

- [Sun10b] Sun Microsystem. (2010). *Java SE 6 Documentation*. Dipetik Mei 11, 2010, dari A Sun Developer Network Site. Java.sun.com : Electronic references. : <http://java.sun.com/javase/6/docs/>
- [Uni10] University of South Florida. (2010). *An Educator's Guide to School Networks*. Dipetik Mei 11, 2010, dari Florida Center For Instructional Technology. Fcit.usf.edu : Electronic references. : <http://fcit.usf.edu/network/>