

## APLIKASI REMOTE DEKSTOP BERBASIS MOBILE MELALUI WIRELESS DENGAN TEKNOLOGI ANDROID

Jasman Pardede<sup>1)</sup>, Asep Nana Hermana<sup>2)</sup>

<sup>1,2)</sup> Teknik Informatika, Fakultas Teknik Industri, Institut Teknologi Nasional Bandung  
jasman@itenas.ac.id, asepnana\_h@yahoo.com

### Abstract

Computer development coupled with the higher level of mobility of people who want to access data and applications in a PC or computing device wherever and whenever. Remote Desktop is a term to represent instances where device that is in a particular location is accessed and used from a different location. Some of the Remote Desktop application that has been created is the Windows and Macintosh Remote Desktop. Windows Remote Desktop have functionality to transmit a copy of the screen (screen captured), responsibility keyboard and mouse events, and even voice. Remote Desktop Macintosh congenital further provides additional features such as remote installation. The applications can only be done remotely on each platform by internet device. However, the people want to access data and applications on multiplatform. There for to solve the problem, researcher developed a Remote Desktop Application multiplatform. Programming language that supports multiplatform is Java. To facilitated the application of Remote Desktop that regardless of the location, require mobile-based applications. The mobile-based application that facilitates the researcher used a smart phone-based on Android by wireless technology.

**Keywords**—remote desktop, multiplatform, java, android, wireless

### 1. PENDAHULUAN

Remote desktop adalah sebuah istilah untuk mewakili kejadian di mana sebuah PC atau perangkat komputer yang ada di suatu lokasi tertentu, dapat diakses dan digunakan dari tempat yang berbeda [1]. Penggunaan komputer yang di *remote* tidak hanya terbatas pada tampilan teks tetapi tampilan desktop komputer seperti saat menggunakannya secara langsung. *Remote Desktop* dalam mengendalikan dan menampilkan salinan gambar layar (*screen captured*) komputer yang dikendalikan dipengaruhi oleh kualitas jaringan komunikasi antar kedua komputer dalam interval waktu tertentu. Perangkat lunak *remote desktop* pada komputer yang akan di-*remote* mentransmisikan dan mengendalikan aktivitas *keyboard* atau *mouse* ke komputer yang me-*remote*. Teknologi *remote desktop* bukanlah teknologi baru, teknologi ini sudah cukup lama dikenal dalam dunia komputerisasi. Umumnya teknologi yang digunakan *remote desktop* hanya untuk mengendalikan

komputer dari PC ke PC bukan dari Handphone ke PC.

Beberapa aplikasi *Remote Desktop* yang sudah pernah dibuat diantaranya *Remote Desktop Windows* dan *Macintosh*. *Remote Desktop* bawaan Windows memiliki fungsionalitas mentransmisikan salinan layar (*screen captured*), *event keyboard* dan *mouse*, dan suara [2]. *Remote Desktop Macintosh* disamping fungsionalitas yang dimiliki oleh Windows juga menyediakan fungsionalitas tambahan seperti penginstalan secara *remote* [1]. Tetapi kedua *remote desktop* tersebut hanya dapat melakukan *remote* pada platformnya masing-masing dan jaringan komunikasi menggunakan jaringan internet. Untuk menyelesaikan masalah tersebut, peneliti akan mengembangkan aplikasi *Remote Desktop* yang multiplatform. Bahasa pemrograman yang mendukung multiplatform yaitu bahasa pemrograman Java. Dalam memfasilitasi pengaplikasian *Remote Desktop* yang tanpa memperhatikan lokasi, maka diperlukan aplikasi yang berbasis mobile, yaitu dengan menggunakan *smartphone* yang berbasis **Android** dengan jaringan komunikasinya menggunakan teknologi wireless.

### 2. METODE PENELITIAN

Metodologi penelitian yang digunakan dalam menyelesaikan penelitian adalah sebagai berikut :

- Studi Pustaka (Literatur), mencari sumber referensi yang berhubungan dengan pengembangan aplikasi *remote desktop* berbasis mobile.
- Studi Sistem, dilakukan dengan cara observasi dan pengambilan data yang berhubungan dengan pengembangan aplikasi *remote desktop* berbasis mobile.
- Pengembangan Sistem, model pengembangan perangkat lunak yang dilakukan adalah menggunakan model waterfall dengan mengikuti tahap-tahap analisis, desain, pengkodean, testing, dan deploy aplikasi.

### 3. LANDASAN TEORI

Pada subbab ini akan membahas tentang pengertian *remote desktop*, *remote desktop* sistem operasi *windows* dan *macintosh*, bahasa pemrograman java, socket, android, dan teknologi wireless.

### 3.1. REMOTE DEKSTOP

Remote desktop adalah sebuah istilah untuk mewakili kejadian dimana sebuah PC atau perangkat computer yang ada di suatu lokasi tertentu, diakses dan digunakan dari tempat yang berbeda lokasi [1]. Teknologi remote desktop sebenarnya bukanlah teknologi yang baru. Kehadiran teknologi ini sudah cukup lama di dunia komputerisasi. Umumnya teknologi yang digunakan untuk remote desktop hanya untuk pengendalian dari PC ke PC.

Dalam komputasi, Remote Desktop merujuk pada sebuah perangkat lunak atau fungsionalitas dari sistem operasi yang memungkinkan untuk menjalankan server secara jarak jauh untuk ditampilkan secara lokal. Cara kerja dari Remote Desktop cukup sederhana yaitu dengan cara mengirimkan *captured screen* komputer yang di-remote ke komputer yang me-remote saat bekerja mengendalikan komputer yang di-remote sehingga komputer yang me-remote dapat melihat hasil eksekusi dari komputer yang di-remote tersebut. Secara umum komputer yang me-remote mengirimkan event-event yang terjadi di komputer yang me-remote dan meng-copy-kan event-event tersebut ke komputer yang akan di-remote. Beberapa fitur yang biasanya ada pada sebuah Remote Desktop adalah menampilkan screen komputer yang di kendalikan, mengirimkan dan mengeksekusi event-event dari keyboard dan mouse, *network printer redirection*, *redirection of local drives*, *copy and paste files*, serta *audio redirection*.

### 3.2. REMOTE DEKSTOP SISTEM OPERASI WINDOWS DAN MACINTOSH

Pada sistem operasi yang berbasis Windows pada umumnya fitur Remote Desktop yang disediakan adalah menampilkan *screen computer* yang dikendalikan, penanganan event-event keyboard dan mouse, *network printer redirect*, *redirect of local drivers*, *copy and paste files*, *audio redirection*, *support for over 256 colors*, *redirection of windows key combinations*, *shared clipboard*, *128 bit encryption*, *port redirection*, *connection bar*. Remote Desktop pada sistem operasi Windows menggunakan protokol RDP (*Remote Desktop Protocol*). Windows menanam protokol tersebut dalam level kernel sehingga Remote Desktop menjadi lebih aman dan lebih cepat dalam pengaksesan [2]. Versi RDP terbaru saat ini adalah RDP versi 7.0 yang sudah ter-include dalam Windows Server 2008 dan Windows 7.

Pada sistem operasi Macintosh terdapat software Apple Remote Desktop 3 yang memiliki beberapa fitur yaitu Remote Spotlight Search, Dashboard Widget, Automator Action, Auto Install, Curtain Mode, Remote Drag and Drop, Power Copy, System Status Indicators, Application Usage Report, User History Reports, Smart Computer List, dan Task Templates [1].

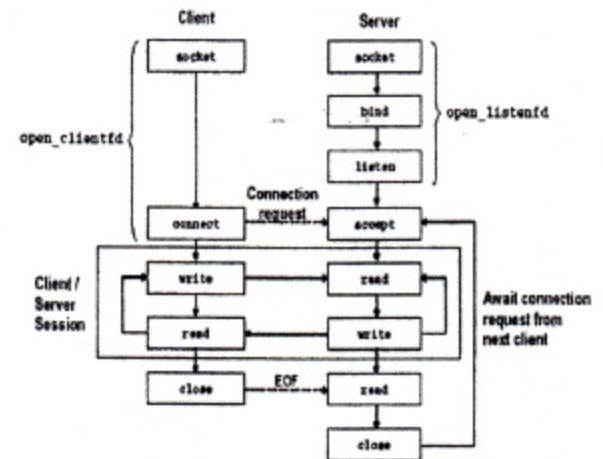
### 3.3. BAHASA PEMROGRAMAN JAVA

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Java berdiri diatas sebuah mesin interpreter yang diberi nama *Java Virtual Machine (JVM)* [3]. JVM inilah yang akan

membaca bytecode dalam file .class dari suatu program sebagai representasi langsung program yang berisi bahasa mesin. Platform Java terdiri dari kumpulan library, JVM, kelas-kelas loader yang dipaket dalam sebuah lingkungan rutin Java dan sebuah kompilator, debugger dan kakas lain yang dipaket dalam Java Development Kit (JDK). Java 2 adalah generasi yang sekarang sedang berkembang dari platform Java. Agar sebuah program Java dapat berjalan dengan baik, maka file dengan ekstensi Java harus dikompilasi menjadi file bytecode. Untuk menjalankan bytecode tersebut dibutuhkan *Java Runtime Environment (JRE)* yang memungkinkan pemakai untuk menjalankan program Java, hanya menjalankan, tidak untuk membuat kode baru lagi. JRE berisi JVM dan library Java yang digunakan [4].

### 3.4. SOCKET

Socket adalah mekanisme komunikasi yang memungkinkan terjadinya pertukaran data antar program atau proses, baik dalam satu mesin maupun antar mesin [5][6]. Socket dibuat dengan menyambungkan dua buah alamat IP melalui port tertentu. Secara umum socket digunakan dalam sistem client/server, dimana sebuah server akan menunggu client pada port tertentu. Saat client me-request server maka server akan berkomunikasi dengan client melalui socket yang dibangun. Salah satu fungsi socket adalah *interface socket* yaitu untuk menghubungkan komputer ke jaringan atau antara client dengan server [6]. System call pada *interface socket* dapat memudahkan suatu aplikasi untuk membuat local socket, dan menghubungkannya ke remote socket. Dengan menghubungkan komputer ke socket, maka komunikasi antar komputer dapat dilakukan. Adapun mekanisme komunikasi socket seperti yang dinyatakan pada Gambar 1.



Gambar 1. Interface Socket

### 3.5. ANDROID [7]

Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan platform terbuka bagi para pengembang buat menciptakan aplikasi

mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Terdapat dua jenis distributor sistem operasi Android, yaitu pertama yang mendapat dukungan penuh dari Google atau Google Mail Services (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai Open Handset Distribution (OHD). Sampai saat ini terdapat berbagai versi – versi android yang sudah ada, yaitu :

- a. Android versi 1.1
- b. Android versi 1.5 (Cupcake)
- c. Android versi 1.6 (Donut)
- d. Android versi 2.0/2.1 (Eclair)
- e. Android versi 2.2 (Froyo: Frozen Yoghurt)
- f. Android versi 2.3 (Gingerbread)
- g. Android versi 3.0 (Honeycomb)
- h. Android versi 4.x (Ice Cream Sandwich)

### 3.6. TEKNOLOGI WIRELESS JAVA

Secara konsep, teknologi wireless dapat dibagi atas dua kategori, pertama untuk lokal dan yang kedua untuk area yang sangat luas. Perangkat yang termasuk kedalam kategori pertama misalnya remote control untuk membuka atau mengunci mobil maupun garasi, telepon cordless 900Mhz, peralatan mainan dengan radio control, atau jaringan wireless [5]. Peralatan wireless jenis pertama ini hanya bekerja untuk daerah dengan jangkauan yang tidak terlalu jauh. Sedangkan peralatan jenis aplikasi yang kedua diantaranya adalah pager, telepon genggam, PDA, dan jenis lainnya. Jangkauan dari perangkat tersebut jauh lebih besar dari aplikasi jenis pertama. Jangkauan yang besar tersebut dilayani oleh *cell-tower* [5]. Dengan adanya *cell-tower* maka peralatan komunikasi bergerak seperti telepon genggam dapat menerima layanan dari sebuah *wireless carrier* atau perusahaan yang mengoperasikan *cell-tower* tersebut.

Pada awal perkembangan aplikasi komunikasi bergerak masing-masing vendor menghasilkan platform aplikasi dan sistem operasi sendiri-sendiri. Sebagai contoh, telepon genggam Nokia dan Siemens masing-masing memiliki platform aplikasi yang berbeda. Perbedaan ini menyebabkan platform aplikasi maupun sistem operasi yang dikembangkan pada telepon genggam Nokia tidak dapat berjalan pada telepon genggam Siemens [5]. Agar aplikasi yang hendak dikembangkan pada perangkat telepon genggam dapat berjalan pada berbagai platform maka diperlukan bahasa pemrograman yang memiliki kebebasan atau *platform independece*. Java adalah teknologi "write once run anywhere", sehingga portabilitas Java merupakan suatu kekuatan yang dimiliki Java [3]. Java dijalankan pada sistem operasi apapun tanpa perlu kompilasi ulang program Java yang dibuat. Untuk pengembangan aplikasi yang bergerak, seperti telepon genggam, Sun Microsystem mengenalkan J2ME yang merupakan salah satu bagian teknologi Java yang digunakan untuk aplikasi Java yang berjalan pada perangkat *mobile device* dan teknologi aplikasi wireless [3]. Pada teknologi Android, pengaturan koneksi wireless terdapat pada class *WifiManager*, pada package

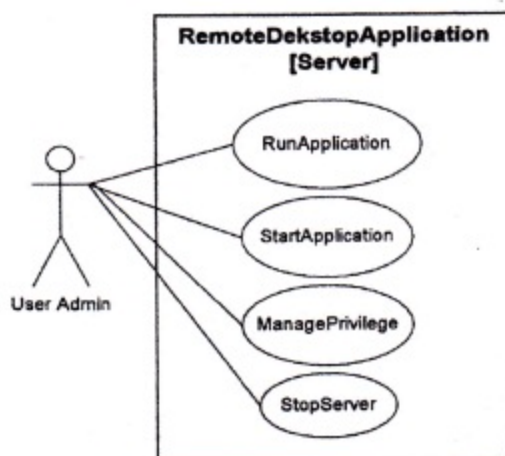
*android.net.wifi*. Class *WifiManager* menyediakan API utama dalam mengatur segala aspek komunikasi Wi-Fi. Untuk memperbolehkan aplikasi melaksanakan segala operasi ke koneksi jaringan pada level abstrak menggunakan class *ConnectivityManager* [7].

## 4. HASIL PENELITIAN

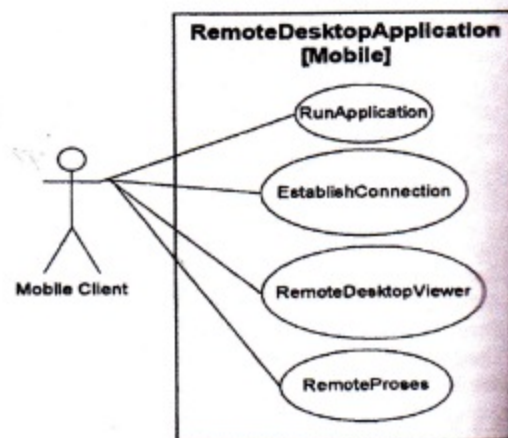
Pada subbab berikut ini akan membahas tentang analisis kebutuhan sistem yang dilakukan, perancangan sistem, implementasi sistem dan pengujian terhadap aplikasi yang telah dikembangkan.

### 4.1. ANALISIS KEBUTUHAN SISTEM

Berdasarkan permasalahan yang terjadi, peneliti menemukan *requirement* pada bagian server seperti yang dinyatakan pada Gambar 2. dan bagian client seperti yang dinyatakan pada Gambar 3.



Gambar 2. Use case diagram Remote Dekstop pada bagian Server



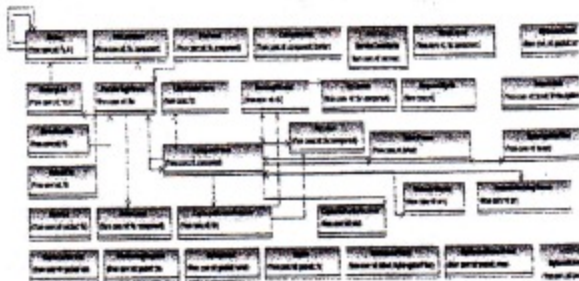
Gambar 3. Use case diagram Remote Dekstop pada bagian Client

## 4.2. PERANCANGAN SISTEM

Berdasarkan kebutuhan fungsionalitas yang harus dipenuhi dalam memfasilitasi kebutuhan perangkat lunak seperti yang dinyatakan pada use case diagram pada Gambar 2, maka peneliti mendapatkan class diagram dari aplikasi remote desktop pada bagian server seperti yang dinyatakan pada Gambar 4, sedangkan pada bagian client diperoleh class diagram seperti yang dinyatakan pada Gambar 5.



Gambar 4. Class diagram Remote Dekstop pada bagian Server



Gambar 5. Class diagram Remote Dekstop pada bagian Client

## 4.3. IMPLEMENTASI SISTEM

Untuk mengimplementasikan perancangan sistem yang dinyatakan pada Gambar 2, membutuhkan bahasa pemrograman java JDK1.6.7 atau versi yang lebih tinggi. Selain itu juga menggunakan software pendukung lainnya, seperti database MySQL Server 5.0, NetBeans 6.8 IDE, eclipse Indigo dan Android versi 2.2 (Froyo: Frozen Yoghurt).

## 4.4. TEKNIK PENGUJIAN

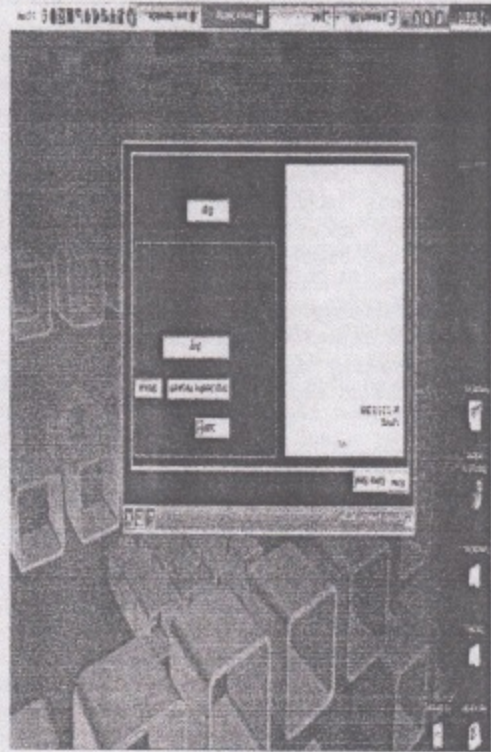
Teknik pengujian yang digunakan adalah teknik pengujian black box testing. Pengujian ini memungkinkan pemrogram untuk memperoleh sekumpulan kondisi masukan (input) yang akan secara penuh menjalankan semua kebutuhan fungsional untuk sebuah program. Dalam hal ini diambil salah satu butir uji, yaitu Remote Desktop Viewer, seperti yang ditunjukkan pada Tabel 1.

Tabel 1 Pengujian RemoteDesktopViewer

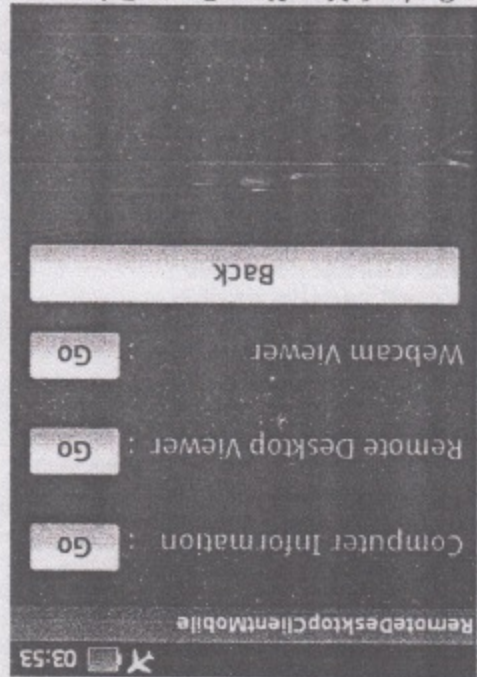
Identifikasi	ARDM -03										
Nama Butir Uji	RemoteDesktopViewer										
Tujuan	Media dalam memfasilitasi pengguna untuk menampilkan tampilan remote desktop pada smartphone client										
Deskripsi	Komunikasi dengan teknologi wireless sudah terhubung dengan baik, pengguna sudah berada pada menu utama client, kemudian pengguna akan memilih menu tombol Go dari Remote Desktop Viewer										
Kondisi Awal	Pengguna sudah terkoneksi dengan komputer server yang akan di remote dan sudah berada pada menu utama client										
Pengujian	<p style="text-align: center;"><b>Skenario Uji</b></p> <p>1. Pilih Tombol Go Remote Desktop Viewer</p> <p style="text-align: center;"><b>Kriteria Evaluasi Hasil</b></p> <p>Tunggu beberapa detik (tergantung dari spesifikasi komputer server) akan ditampilkan desktop server yang akan di remote</p> <p style="text-align: center;"><b>Kasus dan Hasil Uji (Data Normal)</b></p> <table border="1"> <thead> <tr> <th>Masukkan</th> <th>Harapan</th> <th>Pengamatan</th> <th>Kesimpulan</th> </tr> </thead> <tbody> <tr> <td>Action Event click Button Go</td> <td>Menampilkan remote desktop server yang akan di remote oleh pengguna ke smartphone pengguna, tanpa harus me-lock komputer server yang di remote</td> <td>Menampilkan remote desktop server yang akan di remote oleh pengguna ke smartphone pengguna, tanpa harus me-lock komputer server yang di remote</td> <td>[X] Terima [ ] Tolak</td> </tr> </tbody> </table>			Masukkan	Harapan	Pengamatan	Kesimpulan	Action Event click Button Go	Menampilkan remote desktop server yang akan di remote oleh pengguna ke smartphone pengguna, tanpa harus me-lock komputer server yang di remote	Menampilkan remote desktop server yang akan di remote oleh pengguna ke smartphone pengguna, tanpa harus me-lock komputer server yang di remote	[X] Terima [ ] Tolak
Masukkan	Harapan	Pengamatan	Kesimpulan								
Action Event click Button Go	Menampilkan remote desktop server yang akan di remote oleh pengguna ke smartphone pengguna, tanpa harus me-lock komputer server yang di remote	Menampilkan remote desktop server yang akan di remote oleh pengguna ke smartphone pengguna, tanpa harus me-lock komputer server yang di remote	[X] Terima [ ] Tolak								

Berdasarkan hasil pengujian yang dilakukan pengguna terhadap butir uji Remote Desktop Viewer dengan mengikuti skenario yang dinyatakan pada Tabel 1 dengan menu utama client seperti Gambar 6, diperoleh hasil pengujian seperti yang ditunjukkan pada Gambar 7. Dengan melakukan proses pengujian yang sama, maka untuk mengatur koneksi antara client dengan server dilakukan dengan menggunakan Atur Koneksi Remote Dekstop seperti pada Gambar 8. Untuk Computer Information dari komputer server yang di remote seperti yang ditunjukkan pada Gambar 9. Sedangkan menu utama server yang meresponi segala permintaan client seperti yang ditunjukkan pada Gambar 10.

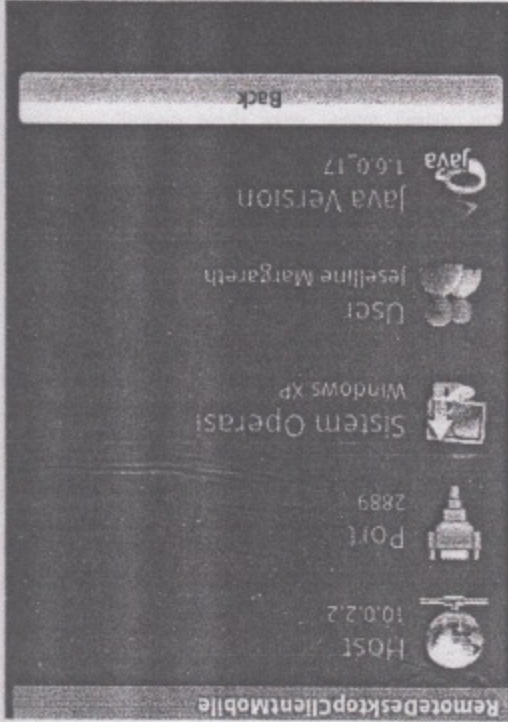
Gambar 7. Tampilan Remote Desktop Viewer Client



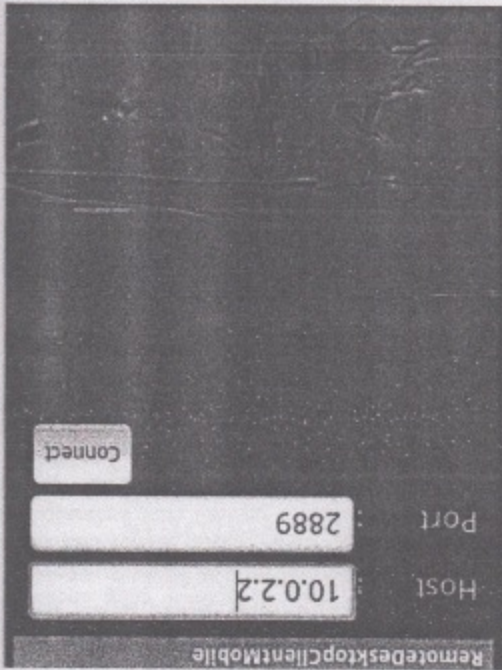
Gambar 6. Menu Utama Remote Desktop

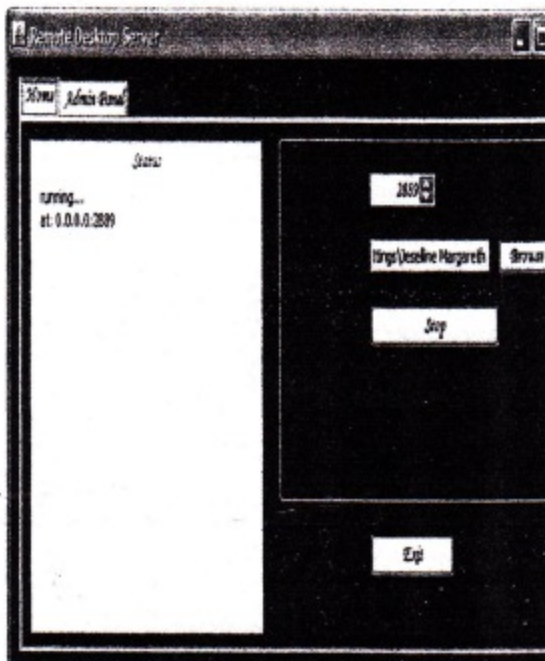


Gambar 9. Menu Computer Information Remote Desktop



Gambar 8. Menu Atur Koneksi Remote Desktop





Gambar 10. Menu Utama Server Remote Dekstop

## 5. KESIMPULAN

Berdasarkan penelitian yang dilakukan, beberapa kesimpulan yang dapat diambil sebagai berikut :

1. Aplikasi Remote Dekstop yang telah dikembangkan dapat diakses melalui berbagai *device (multiplatform)* dan sudah mengimplementasikan fitur minimal aplikasi remote desktop.
2. Berdasarkan cara pengaksesannya, aplikasi yang telah dikembangkan mempunyai perbedaan dengan remote desktop yang dimiliki oleh Remote Desktop versi Windows yaitu komputer yang di-remote tidak akan di lock (*log off*) saat aplikasi di-remote.
3. Waktu respon aplikasi yang dikembangkan masih lebih lambat dibandingkan dengan remote desktop bawaan Windows dan Macintosh.

## 6. SARAN

Berdasarkan hasil penelitian yang sudah dilakukan, aplikasi yang berjalan masih lebih lambat dibandingkan dengan remote desktop bawaan Windows atau Macintosh. Keterlambatan yang terjadi sangat dipengaruhi oleh kualitas jaringan dalam proses pengiriman data. Untuk meningkatkan kecepatan pengiriman data, untuk penelitian selanjutnya disarankan perlu memperhatikan konsep *Multithread*, dimana layar yang hendak di-remote dapat dibagi atas beberapa wilayah, setiap wilayah ditangani oleh satu *Thread*, sehingga server hanya mengirimkan *screen captured* yang mengalami perubahan pada wilayah tertentu.

## 7. UCAPAN TERIMA KASIH

Penelitian ini telah dikerjakan dengan baik berkat bantuan dari Hibah Penelitian Dosen Itenas 2011. Untuk itu penulis mengucapkan terima kasih kepada LPPM Itenas yang telah memberi dukungan financial terhadap penelitian ini.

## DAFTAR PUSTAKA

- [1] Apple. Apple - *Remote Desktop 3 - New Features* [on line], Available : <http://www.apple.com/remotedesktop/newfeatures.html> [27 Februari 2011]
- [2] Microsoft. *The Features of the Remote Desktop Client in Windows XP* [on line], Available: <http://support.microsoft.com/kb/300698> [20 Maret 2011]
- [3] Garrido., J., M., (2003), *Object-Oriented Programming : From Problem Solving to Java*, Charles River Media, Inc.Hingham, Massachusetts
- [4] Poo, D., Kiong, D., Ashok, S., (2008), *Object-Oriented Programming and Java*, Second Edition, Springer-Verlag, London.
- [5] Dov Bulka, *Java™ Performance and Scalability Volume 1: Server-Side Programming Techniques*, Addison Wesley, 2001
- [6] Robert Orfali and Dan Harkey, *Client/Server Programming with Java and CORBA*, 2nd Edition, John Wiley & Sons, Inc., 1998
- [7] Wikipedia. *Android (sistem operasi)* [on line], Available: [http://id.wikipedia.org/wiki/Android\\_\(sistem\\_operasi\)](http://id.wikipedia.org/wiki/Android_(sistem_operasi)) [18 April 2011]