

Mobile Remote Application

TUGAS AKHIR

Oleh :

Ismah Yudhi 3310701085

Danny Rio Setiawan 3310701117

Disusun untuk memenuhi syarat kelulusan Program Diploma III



PROGRAM STUDI TEKNIK INFORMATIKA

POLITEKNIK NEGERI BATAM

BATAM

2011

LEMBAR PENGESAHAN

Batam, 16 Agustus 2011

Pembimbing,

Mir'atul Khusna Mufida S.ST

NIK. 109057

LEMBAR PERNYATAAN

Dengan ini, saya:

NIM : 3310701085

Nama : Ismah Yudhi

adalah mahasiswa Teknik Informatika Politeknik Negeri Batam yang menyatakan bahwa tugas akhir dengan judul:

MOBILE REMOTE APPLICATION

disusun dengan:

1. Tidak melakukan plagiat terhadap naskah karya orang lain
2. Tidak melakukan pemalsuan data
3. Tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa izin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Akhir ini.

Batam, 16 Agustus 2011

Ismah Yudhi
3310701085

LEMBAR PERNYATAAN

Dengan ini, saya:

NIM : 3310701117

Nama : Danny Rio Setiawan

adalah mahasiswa Teknik Informatika Politeknik Negeri Batam yang menyatakan bahwa tugas akhir dengan judul:

MOBILE REMOTE APPLICATION

disusun dengan:

1. Tidak melakukan plagiat terhadap naskah karya orang lain
2. Tidak melakukan pemalsuan data
3. Tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa izin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Akhir ini.

Batam, 16 Agustus 2011

Danny Rio Setiawan
3310701117

KATA PENGANTAR

Segala puji dan syukur kita panjatkan kepada Allah SWT yang telah memberikan Rahmat dan Karunia-Nya, penulis bersyukur kepada kepada Allah SWT atas selesainya Tugas Akhir yang berjudul ”*Mobile Remote Application*”

Dalam kesempatan ini pula penulis mengucapkan terima kasih kepada:

1. Allah SWT atas terselesainya Tugas Akhir ini.
2. Kedua orang tua yang telah memberikan dukungan moril maupun materil
3. Bapak DR.Ir.Priyono Eko Sanyoto, M.Sc, selaku Direktur Politeknik Negeri Batam
4. Bapak Uuf Brajawidagda, ST, MT selaku koordinator Tugas Akhir
5. Ibu Mir’atul Khusna Mufida S.ST selaku dosen Pembimbing Tugas Akhir ini. Terima kasih atas bimbingan dan motivasi yang bapak berikan dalam penyelesaian laporan ini.
6. Keluarga yang telah memberikan do’a, dukungan dan semangat.
7. Bapak/ Ibu dosen program studi Teknik Informatika Politeknik Negeri Batam atas arahan dan bimbingannya.
8. Teman-teman Teknik Informatika angkatan 2007 - 2008 yang memberikan dukungan, semangat dan do’a.

Penulis menyadari bahwa masih banyak kekurangan dalam penyusunan laporan ini. Oleh karena itu, penulis sangat mengharapkan bantuan dari beberapa pihak baik berupa kritik maupun saran yang membangun dari pembaca. Akhir kata penulis mengucapkan terima kasih, semoga penulisan laporan ini dapat bermanfaat bagi pembaca yang ingin mengembangkan sebuah penelitian yang serupa.

Batam, Agustus 2011

Penulis

ABSTRACT

MOBILE REMOTE APPLICATION

The application is widely used to help solve problems or work to be done, is no exception in the presentation process issues that are still using presenter operator, listen to music and play videos on a computer that controls the application is still using a keyboard and mouse so difficult for users to move more widely and spend many times.

Where the current developments in mobile phone technology is not limited to its function as a communication tool only, but the phone can also be used to operate certain tools, for additional bluetooth technology is capable of transferring data. Mobile phones and computers have some relevance in its use both in the operation of the application to data services, so that the relationship between mobile phones and computers can be used as a function of mutual support. operation of today's computers are still using the devices available on the computer itself, for example in an application running on a computer that still uses the keyboard and mouse functions to perform these activities, making it difficult for users to move more widely and spent much time, so the role of the mobile remote application is necessary.

Reducing the problem is required "Mobile Remote Application" that is expected of these applications can help simplify user's authority to operate applications such as Windows Media Player, Winamp and Microsoft Power Point on your PC using a mobile phone remotely using a Bluetooth network.

Keywords: Mobile, bluetooth, Mobile Remote Application.

ABSTRAK

MOBILE REMOTE APPLICATION

Aplikasi sangat banyak digunakan untuk membantu menyelesaikan masalah atau pekerjaan yang akan dilakukan, tidak terkecuali dalam masalah proses presentasi yang masih menggunakan *ASROT* (Asisten Sorot), mendengarkan musik dan memutar *video* pada komputer yang kontrol aplikasinya masih menggunakan *keyboard* dan *mouse* sehingga menyulitkan pengguna bergerak lebih luas dan banyak menghabiskan waktu.

Dimana pada saat ini Perkembangan teknologi pada *handphone* tidak terbatas hanya pada fungsinya sebagai alat komunikasi saja, tetapi *handphone* juga dapat digunakan untuk mengoperasikan alat-alat tertentu, ditambah lagi *handphone* mempunyai teknologi *bluetooth* yang mampu mentransfer data. Hal ini dapat dicontohkan pada *personal computer* (PC), *handphone* dan komputer mempunyai beberapa keterkaitan dalam penggunaannya baik dalam pengoperasian aplikasi hingga pelayanan data, sehingga hubungan antar *handphone* dan komputer dapat dijadikan sebuah fungsi yang saling mendukung. pengoperasian komputer saat ini masih menggunakan perangkat yang tersedia di komputer itu sendiri, contohnya dalam menjalankan aplikasi pada komputer yang masih menggunakan fungsi *keyboard* dan *mouse* untuk melakukan aktifitas tersebut, sehingga menyulitkan pengguna bergerak lebih luas dan banyak menghabiskan waktu, sehingga peran *remote application* sangat dibutuhkan dalam kasus seperti ini.

Mengurangi masalah tersebut maka dibutuhkan "*Mobile Remote Application*" yang diharapkan aplikasi ini dapat membantu memudahkan user dalam mengoperasikan aplikasi seperti *Windows Media Player*, *Winamp* dan *Microsoft Power Point* pada PC dengan menggunakan *handphone* dari jarak jauh menggunakan jaringan *bluetooth*.

Kata Kunci : *Handphone, bluetooth, Mobile Remote Application*

DAFTAR ISI

LEMBAR PENGESAHAN.....	i
LEMBAR PERNYATAAN	ii
KATA PENGANTAR	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI	viii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
Bab I Pendahuluan	1
I.1 Latar Belakang.....	1
I.2 Rumusan Masalah.....	2
I.3 Batasan Masalah	2
I.4 Tujuan	2
I.5 Sistematika Penulisan	2
Bab II Landasan Teori	3
II.1 Pemrograman J2ME.....	3
II.1.1 Connected Limited Device Configuration (CLDC)	4
II.1.2 CDC (Connected Device Configuration)	4
II.1.3 Mobile Information Device Profile (MIDP).....	5
II.1.4 MIDlet	5
II.1.5 NetBeans Mobility Pack	5
II.1.6 J2ME Wireless Toolkit	5
II.1.7 Fitur ponsel yang mendukung J2ME.....	6
II.2 Pemrograman Java	6
II.2.1 Pemrograman Prosedural dan Pemrograman Berorientasi Objek..	6
II.2.2 Sifat-sifat Pemrograman Java	8
II.2.3 Konsep Dasar Dalam Java	8
II.3 NetBeans IDE	18
II.4 Sharing Data Device	22
II.4.1 Inframerah (<i>Infrared</i>)	22

II.4.2	Koneksi Internet Nirkabel (<i>wireless</i>)	22
II.4.3	<i>Bluetooth</i>	23
II.5	Remote	25
II.6	Wireless Presenter.....	26
Bab III	Analisis.....	28
III.2	Fitur Utama Perangkat Lunak	29
III.3	Kebutuhan Fungsional	29
III.4	Batasan Sistem.....	29
III.5	Use Case.....	30
III.5.1	Diagram <i>Use Case</i>	30
III.5.2	Skenario <i>Use Case</i>	30
III.5.3	Use Case Koneksi	31
III.5.4	Use Case Kontrol <i>Mouse</i>	31
III.5.5	Use Case Kontrol Aplikasi	31
III.6	Analisis Kelas	32
III.6.1	Analisis Kelas	32
Bab IV	Perancangan.....	33
IV.1	<i>Sequence Diagram</i>	33
IV.1.1	<i>Sequence Diagram Use Case Koneksi</i>	33
IV.1.2	<i>Sequence Diagram Use Case Kontrol Mouse</i>	34
IV.1.3	<i>Sequence Diagram Use Case Kontrol Aplikasi</i>	34
IV.2	Diagram Kelas	35
IV.3	Rancangan Kelas Rinci	36
IV.3.1	Kelas Koneksi	36
IV.3.2	Kelas Koneksi <i>Server</i>	36
Algoritma	36
IV.3.3	Kelas Navigasi Kontrol	37
Algoritma	37
IV.3.4	Kelas Menu Utama.....	38
Algoritma	39
IV.3.5	Kelas Menu Aplikasi.....	39

Algoritma.....	39
IV.3.6 Kelas <i>Mobile Remote Server</i>	39
Algoritma.....	39
IV.4 Perancangan antar muka	40
IV.4.1 Tampilan Proses Koneksi <i>client</i>	40
IV.4.2 Koneksi <i>Server</i>	41
IV.4.3 Menu Utama.....	42
IV.4.4 Menu Aplikasi.....	43
IV.4.5 Navigasi kontrol.....	44
Bab V Implementasi Dan Pengujian.....	46
V.1 Implementasi Kelas.....	46
V.2 Implementasi Antar Muka.....	46
V.2.1 Implementasi Antar Muka <i>Client</i> dan <i>Server</i>	47
V.3 Pengujian.....	47
V.3.1 Identifikasi Pengujian	48
V.3.2 Skenario Pengujian Aplikasi <i>Mobile Remote Application</i>	48
V.4 Dokumen Rinci Testing	50
V.4.1 Tim Penguji.....	50
V.4.2 Hasil Rinci Pengujian	51
V.5 Daftar Gambar	52
Bab VI Kesimpulan dan Saran	53
VI.1 Kesimpulan.....	53
VI.2 Saran	53
Daftar Pustaka	54

DAFTAR GAMBAR

Gambar II.1. Toolbar pada <i>Netbeans</i>	23
Gambar II.2. Tampilan <i>choose project</i>	24
Gambar II.3. Tampilan name dan location	24
Gambar II... Tampilan <i>default platform selection</i>	25
Gambar II.5. <i>Source code</i> pada j2me	26
Gambar II.6. <i>Source code</i> pada j2me	26
Gambar II.7. Tampilan Emulator.....	27
Gambar II.8. Konfigurasi <i>nokia wireles presenter</i>	32
Gambar II.9. Konfigurasi <i>nokia wireles presenter</i>	33
Gambar II.10. Navigasi kontrol aplikasi	33
Gambar III.1. Deskripsi umum sistem	34
Gambar III.5.1. Use Case sistem Mobile Remote Aplication	36
Gambar III.6.1. Analisis Kelas	39
Gambar IV1.1. Squence diagram use case Koneksi	40
Gambar IV.1.2. Squence diagram use case Kontrol <i>Mouse</i>	41
Gambar IV1.3. Squence diagram use case Kontrol Aplikasi	41
Gambar IV.2. Diagram kelas	42
Gambar IV.4.1. Tampilan koneksi <i>client</i>	47
Gambar IV.4.2. Tampilan koneksi <i>server</i>	48
Gambar IV.4.3. Tampilan Menu Utama	49
Gambar IV.4.4. Tampilan Menu Aplikasi.....	50
Gambar IV.4.5. Tampilan Navigasi Kontrol	51
Gambar V.3.2. Skenario Aplikasi	52

DAFTAR TABEL

Tabel II.1. Deklarasi modifier 1.....	10
Tabel II.1. Deklarasi modifier 2.....	11
Tabel III.1. Fungsi <i>client</i> dan <i>server</i>	30
Tabel V.1. Implementasi kelas	47
Tabel V.4.2.Hasil Rinci Pegujian	52

Bab I Pendahuluan

I.1 Latar Belakang

Perkembangan teknologi pada *handphone* yang beragam mengakibatkan teknologi terus maju dan berkembang, sehingga menjadi salah satu kebutuhan pokok manusia untuk mempermudah pekerjaan dan kegiatan dengan cepat dan efisien. Penggunaan *handphone* tidak terbatas hanya pada fungsinya sebagai alat komunikasi saja, tapi disisi lain *handphone* juga dapat digunakan untuk mengoperasikan alat-alat tertentu seperti mendengarkan lagu, memutar *video*, *browsing* dan bahkan menonton program acara televisi, ditambah lagi *handphone* mempunyai teknologi *bluetooth* yang mampu mentransfer data. Hal ini dapat dicontohkan pada *personal computer* (PC), *handphone* dan komputer mempunyai beberapa keterkaitan dalam penggunaannya baik dalam pengoperasian aplikasi hingga pelayanan data, sehingga hubungan antar *handphone* dan komputer dapat dijadikan sebuah fungsi yang saling mendukung.

Dalam pengoperasian komputer saat ini masih menggunakan perangkat yang tersedia di komputer itu sendiri, contohnya dalam melakukan presentasi, mendengarkan musik dan memutar *video* yang masih menggunakan fungsi *keyboard* dan *mouse* untuk melakukan aktifitas aplikasi, sehingga menyulitkan pengguna bergerak lebih luas dan banyak menghabiskan waktu, sehingga peran *remote* sangat dibutuhkan dalam kasus seperti ini.

Mengurangi masalah tersebut maka dibutuhkan “*Mobile Remote Application*“ yang diharapkan dapat membantu memudahkan user dalam mengoperasikan aplikasi pada PC dengan menggunakan *handphone* dari jarak jauh menggunakan jaringan *bluetooth*.

I.2 Rumusan Masalah

Bagaimana mengoperasikan aplikasi dan navigasi kontrol pada PC tanpa menggunakan *mouse* dan *keyboard*.

I.3 Batasan Masalah

1. Hanya berjalan pada PC yang menggunakan sistem operasi *Windows*.
2. Hanya dapat diakses dengan spesifikasi *bluetooth* tertentu.
3. Hanya dapat diakses pada aplikasi tertentu pada PC seperti *Microsoft Power Point*, *Music Media Player* dan *Winamp*.
4. Spesifikasi kondisi awal PC dan aplikasi dalam keadaan hidup.
5. Hanya dapat dijalankan melalui *handphone* dengan spesifikasi pendukung *java* dan *bluetooth*.

I.4 Tujuan

Menyediakan media untuk mengendalikan navigasi kontrol PC dan mempermudah kinerja user dalam mengontrol aplikasi seperti *power point*, *windows media player*, *winamp* pada PC menggunakan *handphone* yang memiliki fitur *bluetooth*.

I.5 Sistematika Penulisan

Bab I Pendahuluan

Berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan, dan sistematika penulisan

Bab 2 Tinjauan Pustaka

Berisi tentang teori-teori yang berhubungan dengan penelitian

Bab 3 Analisis

Berisi tentang uraian langkah-langkah penyelesaian masalah

Bab 4 Perancangan

Berisi tentang perancangan dari penyelesaian masalah

Bab 5 Implementasi dan Pengujian

Berisi tentang uraian langkah implementasi dan pengujian/validasi

Bab 6 Kesimpulan dan Saran Berisi tentang kesimpulan dan saran

Bab II Landasan Teori

II.1 Pemrograman J2ME

J2ME atau Java 2 Micro Edition merupakan kategori Java yang dirancang untuk menjalankan program Java pada perangkat seperti PDA atau handphone, yang memiliki karakteristik yang berbeda dengan komputer biasa, misalnya memiliki memori yang terbatas, input yang terbatas, serta layar yang kecil. J2ME sendiri merupakan kombinasi yang terbentuk antara sekumpulan interface Java yang sering disebut dengan Java API (Application Programming Interface) dengan JVM (Java Virtual Machine) yang didesain khusus untuk alat, yaitu JVM dengan ruang terbatas. Arsitektur J2ME mendefinisikan elemen-elemen untuk membentuk lingkungan Java runtime yang lengkap yang terdiri dari:

- Lapisan Konfigurasi:

Configuration sendiri ditentukan perkembangannya oleh JCP (Java Community Process), inilah badan non-profit yang berkecukupan dengan perkembangan teknologi Java. J2ME mempunyai dua konfigurasi yaitu Connected Limited Device Configuration (CLDC) dan Connected Device Configuration (CDC). Pada bagian ini secara detail CLDC diperlukan untuk pengembangan aplikasi wireless dengan MIDP implementasinya CLDC digunakan untuk program Java pada perangkat keras dengan ukuran memori yang terbatas, pada 160 sampai dengan 512 Kilobyte.

- Lapisan Profil:

Profile merupakan kebalikan dari configuration yaitu mengatur hal-hal yang spesifik untuk sebuah device atau tipe market. Misalkan pada profile ini diatur tentang persistent storage dan UI. Saat ini JCP telah mendefinisikan lima buah profile, salah satunya yaitu MIDP yaitu profile yang digunakan pada banyak mobile devices. Akibatnya, fitur-fitur yang kurang penting untuk diimplementasikan dalam handheld device yang bersangkutan dari Java 2 harus dibuang. MIDP dikhususkan untuk digunakan pada handset dengan kemampuan CPU, memori, keyboard dan layer yang terbatas, seperti handphone, pager, PDA dan sebagainya. Aplikasi yang berjalan pada sebuah perangkat yang mendukung

MIDP disebut dengan MIDlets, atau lebih singkatnya MIDlet merupakan aplikasi yang dibuat menggunakan Java 2 Micro Edition dengan profile *Mobile Information Device Profile* (MIDP).

II.1.1 Connected Limited Device Configuration (CLDC)

CLDC atau Connected Limited Device Configuration adalah perangkat dasar dari J2ME, spesifikasi dasar yang berupa library atau API yang diimplementasikan pada J2ME, seperti yang digunakan pada telepon selular, pager, dan PDA. Perangkat tsb dibatasi dengan keterbatasan memory, sumber daya, dan kemampuan memproses. Spesifikasi CLDC pada J2ME adalah spesifikasi minimal pada package, class, dan sebagian fungsi Java Virtual Machine yang dikurangi agar dapat diimplementasikan dengan keterbatasan sumber daya pada alat tersebut, JVM yang digunakan adalah KVM (Kilobyte Virtual Machine).

II.1.2 CDC (Connected Device Configuration)

CDC (Connected Device Configuration) merupakan komunitas proses pada Java yang memiliki standardisasi. CDC terdiri dari virtual machine dan library dasar untuk dipergunakan pada profile industri. Implementasi CDC pada J2ME adalah source code yang menyediakan sambungan dengan macam-macam platform.

II.1.3 Mobile Information Device Profile (MIDP)

MIDP atau Mobile Information Device Profile adalah spesifikasi untuk sebuah profil J2ME. MIDP memiliki lapisan diatas CLDC, API tambahan untuk daur hidup aplikasi, antar muka, jaringan, dan penyimpanan persisten. Pada saat ini terdapat MIDP 1.0 dan MIDP 2.0. Fitur tambahan pada MIDP 2.0 dibandingkan MIDP 1.0 adalah API untuk multimedia. Pada MIDP 2.0 terdapat dukungan memainkan tone, tone sequence, dan file WAV walaupun tanpa adanya Mobile Media API (MMAPI).

II.1.4 MIDlet

MIDlet adalah aplikasi yang ditulis untuk MIDP. Aplikasi MIDlet adalah bagian dari class javax. Microedition.midlet.MIDlet yang didefinisikan pada MIDP. MIDlet berupa sebuah class abstrak yang merupakan subclass dari bentuk dasar aplikasi sehingga antarmuka antara aplikasi J2ME dan aplikasi manajemen pada perangkat bisa terbentuk.

II.1.5 NetBeans Mobility Pack

NetBeans Mobility Pack adalah alat untuk mengembangkan aplikasi yang berjalan pada perangkat bergerak (*mobile*), umumnya telepon seluler, tetapi juga mencakup PDA, dan lain-lain. NetBeans Mobility Pack dapat digunakan untuk menulis, menguji, dan *debugging* aplikasi untuk perangkat bergerak yang menggunakan teknologi berplatform Java Micro Edition (platform Java ME). Paket ini mengintegrasikan dukungan terhadap *Mobile Information Device Profile* (MIDP) 2.0, *Connected Limited Device Configuration* (CLDC) 1.1, dan *Connected Device Configuration* (CDC). Emulator dari pihak ketiga dapat diintegrasikan dengan mudah untuk lingkungan pengujian yang lebih kokoh. *NetBeans Mobility Pack* saat ini tersedia dalam dua klaster yang berbeda, yang satu memuat *CDC* dan yang lainnya *CLDC*.

II.1.6 J2ME Wireless Toolkit

J2ME Wireless Toolkit adalah perkakas yang menyediakan lingkungan emulator, dokumentasi beserta contoh-contoh aplikasi JAVA untuk perangkat kecil. *J2ME WTK* berbasis pada *CLDC* dan *MIDP*. *J2ME WTK* adalah program yang meniru kerja ponsel yang mendukung MIDP atau yang disebut emulator. Oleh karena itu, belum tentu *MIDlet* yang berjalan di emulator juga berjalan pada ponsel yang sebenarnya, karena juga bergantung pada kemampuan dan kapasitas ponsel yang digunakan.

II.1.7 Fitur ponsel yang mendukung J2ME

Fitur *mobile phone* yang mendukung J2ME adalah *mobile phone* yang mendukung konfigurasi dan profil dari J2ME. konfigurasi dan Profil adalah spesifik kemampuan dari paket perangkat yang akan anda jadikan target aplikasi anda, sebagai contoh saya punya HP K608 itu mempunyai spesifikasi CLDC 1.1 dan MIDP 2.0 artinya HP saya bisa menjalankan program dengan paket yang ada diCLDC 1.1 dan MIDP 2.0. MIDP 1.0 seperti nokia 6015 dan nokia yang dengan

slidenya yang saya lupa apa mereknya dan HP lama lainnya. MIDP 2.0 HP baru seperti nokia N70, N73, SE K600, SEW800 dan lain lain. Karena MIDP 2.0 lebih banyak ada baiknya anda pilih MIDP 2.0 dan CLDC 1.1.¹

II.2 Pemrograman Java

“Program adalah sekumpulan perintah-perintah (*instruction*) yang diatur secara sistematis untuk menyelesaikan suatu masalah. Salah satu bahasa pemrogramn komputer adalah bahasa pemrograman Java. Bahasa pemrograman Java adalah bahasa pemrograman berorientasi objek atau *Object Oriented Programming (OOP)*.”

Java bersifat netral, tidak terikat pada suatu platform, dan mengikuti prinsip WORA (*Write Once Run Anywhere*). Beberapa bahasa pemrograman seperti Basic, Fortran, Pascal, COBOL, C, ProLog dan yang lainnya harus bergantung pada suatu platform karena mengharuskan Anda mengkompilasi program tersebut pada setiap sistem operasi yang ditentukan jika ingin menggunakan hasil program tersebut.

II.2.1 Pemrograman Prosedural dan Pemrograman Berorientasi Objek

Program komputer terdiri atas kode dan data. Kode yang dimaksud adalah proses logis program (misalnya fungsi), sedangkan data dapat berupa variabel atau konstanta. Secara konseptual, kode dan data tersebut diatur sedemikian rupa sehingga dapat bekerja sama untuk menghasilkan keluaran sesuai dengan proses yang diinginkan.

Pada model pemrograman tradisional atau prosedural (disebut *process oriented mode*), semua data dan kode digabung menjadi satu bagian dalam program. Untuk program-program sederhana yang hanya membutuhkan beberapa baris kode, penggunaan ini tentu tidak begitu menjadi masalah. Permasalahan baru akan timbul pada saat berkembang ke arah yang lebih besar dan kompleks, yang mungkin membutuhkan ratusan atau bahkan ribuan baris kode. Pada pemrograman berorientasi objek, komponen-komponen penyusun progam secara konseptual akan dipecah menjadi bagian-bagian tersendiri yang disebut “objek”.

¹ tutorial-membuat-program-java-j2me-di-hp.htm

Dalam pemrograman berorientasi objek, setiap objek akan memiliki *data* (sifat berupa variabel atau konstanta) dan *method* (perilaku atau kemampuan melakukan suatu fungsi). Jadi, objek dapat didefinisikan sebagai suatu yang memiliki data dan method.

Contoh:

Manusia adalah suatu objek yang memiliki data-data:

- a. Nama.
- b. Jenis kelamin.
- c. Tinggi badan
- d. Berat badan.
- e. Golongan darah.

Manusia memiliki *method*:

- Cara bicara.
- Cara berjalan.
- Cara marah.
- Sifat.

II.2.2 Sifat-sifat Pemrograman Java

Sifat-sifat pemrograman Java, antara lain: berorientasi objek, terdistribusi, multiplatform, dan multithread.

II.2.2.1 Java Berorientasi Objek

Pemrograman Java merupakan pemrograman yang berorientasi objek. Artinya, penulisan program harus dibuat dalam bentuk objek-objek, kemudian memodelkan sifat masing-masing objek dalam program. Java menentukan dan mengatur interaksi antara objek.

II.2.2.2 Java Bersifat Terdistribusi

Program java sesuai dengan perkembangan teknologi saat ini, yaitu teknologi jaringan lokal seperti jaringan internet. Dalam hal ini, Java dapat digunakan untuk menyebarkan/mendistribusikan data atau informasi dalam workstation *client*, e-mail *server*, dan *webserver* (dalam jaringan komputer lokal dan internet).

II.2.2.3 Java Bersifat *Multiplatform*

Program Java yang sudah diterjemahkan (dikompilasi) dengan benar dan berhasil dapat dijalankan pada platform yang lain atau pada sistem operasi yang lain. Java menggunakan prinsip WORA (**Write Once Run Anywhere**). Sekali menuliskan program maka hasil programnya dapat dijalankan pada sistem operasi yang mana saja. Artinya, sekali menulis program dalam bahasa Java dan mengkompilasinya maka program tersebut dapat dijalankan pada berbagai sistem operasi, seperti DOS, windows, dan Linux.

II.2.2.4 Java Bersifat *Multithread*

Thread merupakan unit dasar penggunaan *Central Processing Unit* (CPU) yang terdiri dari *Thread_ID*, program counter, register set, dan stack. *Multithread* adalah banyaknya pekerjaan yang dikerjakan dalam satuan waktu yang hampir bersamaan. Sebagai contoh, suatu *web server* mempunyai ratusan *client server* -beberapa *client* mengakses *web server* secara bersamaan. Jika *web server* hanya mempunyai satu thread, berarti *client* yang lain harus menunggu. Oleh karena itu diperlukan multithread untuk melayani *client server*.

II.2.3 .Konsep Dasar Dalam Java

Konsep dasar pemrograman berorientasi objek atau *Object Oriented Programming* (OOP) adalah inti dari pemrograman Java. Semua program Java merupakan objek. Dasar pemrograman objek pada pemrograman Java, antara lain:

1. Class

Class adalah pemodelan dari objek yang menjelaskan data (sifat karakteristik data) dan fungsi yang dimiliki suatu objek. Class merupakan tempat yang digunakan untuk menciptakan suatu objek.

Dasar pemrograman berorientasi objek pada Java adalah class. Class dapat membuat atau mendeklarasikan tipe data baru. Class dijalankan sebagai *blueprint*. File sumber java harus diberi nama berdasarkan class didalam program.

Bentuk deklarasi class:

```
Class NamaClass  
  
{  
  
Class Body  
  
}
```

Secara umum, class memiliki struktur anatomi sebagai berikut:

```
(Modifier1) ClassNamaClass (modifier2)  
  
{  
  
Class Body3  
  
}
```

Dalam pemrograman Java, ada beberapa keyword yang sering digunakan sebagai modifier, yaitu kata tambahan yang terletak di depan suatu unsur (kelas/data/method). Modifier sering digunakan untuk mendeklarasikan class dapat dilihat pada tabel 1.1.

Tabel 1.1 Deklarasi Modifier 1

Modifier 1 (Optional)	Keterangan
Public	Class (kelas) public dapat diakses oleh kelas yang lain, baik yang masih berada dalam

	package yang sama maupun berbeda.
Private	Class private tidak dapat diakses oleh kelas lain (baik kelas itu berasal dalam kelas yang sama ataupun dari kelas yang berbeda).
Protected	Kelas yang dilakukannya oleh subkelas turunannya (dari proses pewarisan/ <i>inheritance</i>) dan kelas lain yang terletak dalam package yang sama.
Abstract	Kelas tidak dapat diinstansiasi langsung menjadi objek. Modifier ini hanya dipakai bila kelas tersebut terletak pada hirarki kelas tertinggi sehingga hanya mungkin diturunkan sebagai subkelas, bukan diinstansi menjadi objek.
Final	Kelas tidak dapat diturunkan menjadi subkelas (pewarisan/ <i>inheritance</i>).

Tabel 1.2 Deklarasi Modifier 2

Modifier 2 (Optional)	Keterangan
Extends Superclass	Digunakan dalam prinsip <i>inheritance</i> , yaitu bila kelas tersebut merupakan subkelas dari suatu kelas super (superclass). Superclass adalah suatu kelas yang dapat mewariskan seluruh data dan <i>method</i> yang dimilikinya kepada kelas lainnya. Kelas yang menjadi

	pewaris (ahli waris) disebut subkelas (subclass).
Implements Interfaces	Kelas yang mengimplementasikan satu atau lebih interface. Bila terdapat lebih dari satu interface, digunakan tanda koma (,) diantara interface-interface tersebut.

Aturan pemberian nama class:

1. Dimulai dengan abjad, tanda *underscore* (_), atau tanda dolar (\$).
2. Nama class tidak boleh dimulai dengan bilangan (0,1,2,3,...). setelah karakter pertama, dapat diikuti oleh kombinasi abjad, underscore, dolar, atau bilangan.
3. Tidak boleh menggunakan keyword dalam Java (public, class, boolean, private dan sebagainya).
4. Tidak boleh menggunakan operator (+, -, = dan lain-lain).
5. Java bersifat *case sensitive* (membedakan huruf kecil dan huruf kapital).
6. Nama file source harus sesuai dengan nama class. Misalnya, menuliskan nama mahasiswa dan alamat mahasiswa.

Nama file source: **dataku.java**

Nama class: **dataku**

```
Class dataku{
    Public static void main(String[] args) {
        String nama, alamat;
        nama="Stacia";
```

```

alamat="Jalan Sudirman Medan";

System.out.println("Nama    :"+nama);

System.out.println("Alamat  :"+alamat);

} }

```

2. Object

Object adalah suatu entiti yang memiliki keadaan, *behavior*, dan identitas yang tugasnya dirumuskan dalam suatu lingkup masalah dengan baik. Object dapat disebut sebagai suatu intans (*instance*). Instance adalah proses pembuatan (perwujudan) suatu kelas menjadi objek. Ada dua tahap untuk membuat objek baru dari tipe kelas yang telah didefinisikan:

Tahap 1, mendeklarasikan variabel yang digunakan sebagai referensi ke objek dari kelas yang bersangkutan.

Tahap 2, menginstansiasi (intans) kelas dengan menggunakan kata kunci **new** dan memasukkan instance-nya ke dalam referensi yang baru dideklarasikan.

Class yang menggunakan kata kunci **new** secara dinamis akan mengalokasikan ruang memori untuk menyimpan suatu objek tertentu dan mengembalikan nilai berupa referensi ke objek yang bersangkutan. Referensi objek adalah sebuah alamat memori dari objek yang dialokasikan dengan menggunakan operator **new**. Misalnya, menghitung luas persegi dengan rumus luas= panjang x lebar.

Tahap 1 mendeklarasikan

Nama file: persegi.java

```

Class persegi{

    int lebar;

    int panjang;

```

```
}
```

Tahap 2 menginstansiasi

Nama file: luaspersegi.java

```
Class luaspersegi{  
  
    public static void main (String[] args){  
  
        double luas;  
  
        persegi p=new persegi();  
  
        p.panjang=10;  
  
        p.lebar=3;  
  
        luas= p.panjang*p.lebar;  
  
        System.out.println("Luas Persegi    :"+luas);  
  
    }  
  
}
```

Keterangan:

Persegi adalah suatu kelas.

3. Atribute

Attribute adalah elemen data dari suatu objek. Attribute menyimpan informasi tentang objek. Attribute dapat diartikan sebagai data, variabel, properti, atau sebuah field data. Setiap variabel harus memiliki tipe data dan diberi nama yang unik untuk membedakan yang satu dengan yang lainnya. Misalnya, objek mahasiswa.

Atribut dari seorang mahasiswa berupa:

Nomor mahasiswa

Nama mahasiswa

4. Method

Method adalah subprogram yang dihubungkan pada definisi class tertentu. Setiap subprogram memiliki tugas sendiri. Method dalam pemrograman terstruktur adalah *prosedur* atau *fungsi* yang dimiliki suatu program. Sebuah method menjelaskan behavior dari sebuah objek. Dalam java, method ada 2, yaitu method void dan method nonvoid.

Method void

Method void adalah method yang tidak mengembalikan nilai. Untuk method void tipe data kembalian, harus diisi dengan kata kunci *void*. Cara kerja method void sama seperti prosedur (*procedure*).

Method nonvoid

Method nonvoid adalah method yang mengembalikan nilai. Untuk mengembalikan suatu nilai tertentu maka di dalam sebuah method harus terdapat *statement return* yang diikuti nilai yang akan dikembalikan. Cara kerja method nonvoid sama seperti fungsi (function).

Pendeklarasian/pendefinisian method:

1. Deklarasi method disebut "judul method" (header method).
2. Badan method berada di dalam pasangan kurungan kerawal

cara penulisan:

```
headermethod{
```

```
    badan method
```

```
}
```

3. Badan method membentuk satu blok

Contoh program untuk menghitung perkalian dua nilai ($c=a*b$) menggunakan method void **cetakhasil.java**

```
//class perkalian{  
  
double a;  
  
double b;  
  
void hasil(){  
  
    System.out.println("Hasil perkalian =" +(a*b));  
  
    }  
  
}  
  
class cetak hasil{  
  
public static void main(String[] args){  
  
    perkalian var1, var2;  
  
/*instansi objek var1 dan var2 */  
  
    var1=new perkalian();  
  
    var2=new perkalian();  
  
/*mengisi nilai data a dan nilai data b pada objek var1 */  
  
    var1.a=2;  
  
    var2.b=4;
```

```

/*mengisi nilai data a dan nilai data b pada objek var2 */

    var2.a=2;

    var2.b=5;

/*memanggil method hasil mencetak hasil dari masing-masing objek*/

    var1.hasil();

    var2.hasil();

}

}

```

Hasil dari program:

Hasil perkalian = 8.0

Hasil perkalian 10.0

5. Constructor

Constructor adalah sebuah method yang digunakan dalam kelas untuk membuat dan menginisialisasi sebuah objek baru. Constructor berfungsi untuk menginisialisasi nilai terhadap data yang terdapat pada kelas yang berhubungan. Ketentuan constructor untuk melakukan inisialisasi:

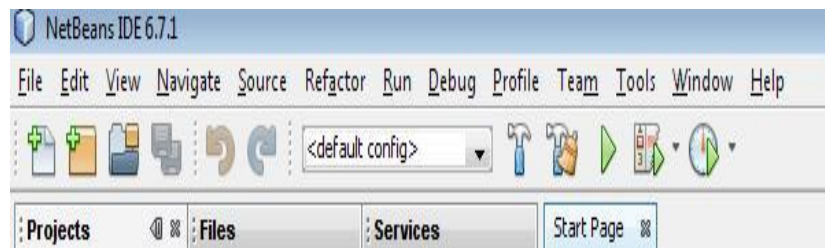
1. Constructor harus bernama sama dengan nama kelas.
2. Constructor tidak memverifikasikan tipe data yang dikirim.
3. Kelas dapat berisi banyak constructor overloading yang memungkinkan objek diinisialisasi.

II.3 NetBeans IDE

NetBeans IDE adalah IDE *open source* yang ditulis sepenuhnya dengan bahasa pemrograman Java menggunakan *platform* NetBeans. NetBeans IDE mendukung pengembangan semua tipe aplikasi Java (J2SE, *Web*, EJB dan aplikasi *mobile*).²

Langkah-langkah untuk menjalankan NetBeans IDE, sebagai berikut:

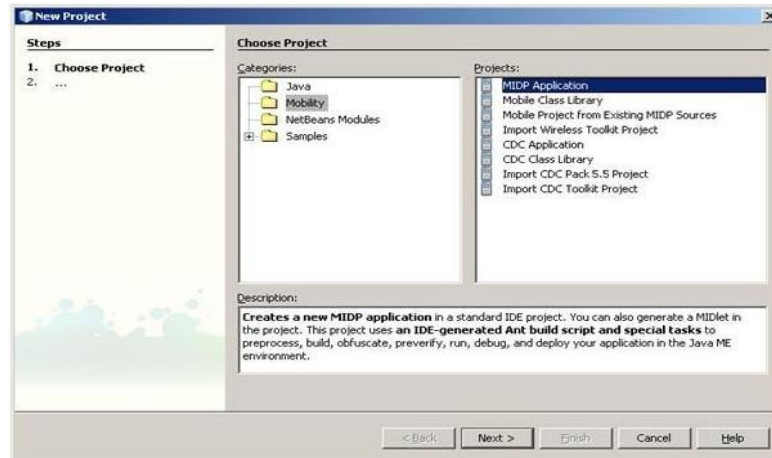
1. Buka aplikasi NetBeans IDE (*Intergrated Development Environment*) 6.7.1
2. Pada IDE, pilihlah *file > New Project* (Ctrl-Shift-N)



Gambar II.1 Toolbar pada NetBeans

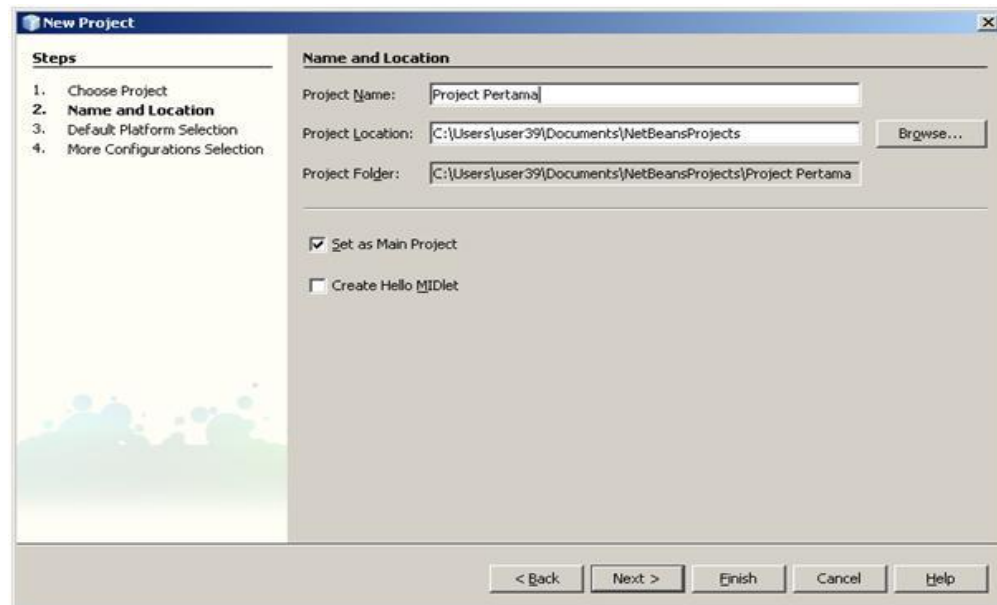
3. Pada *New Project* , pilih kategori *Mobility* dan dilanjutkan dengan meng-klik *next*

² <http://javastudy.wordpress.com>



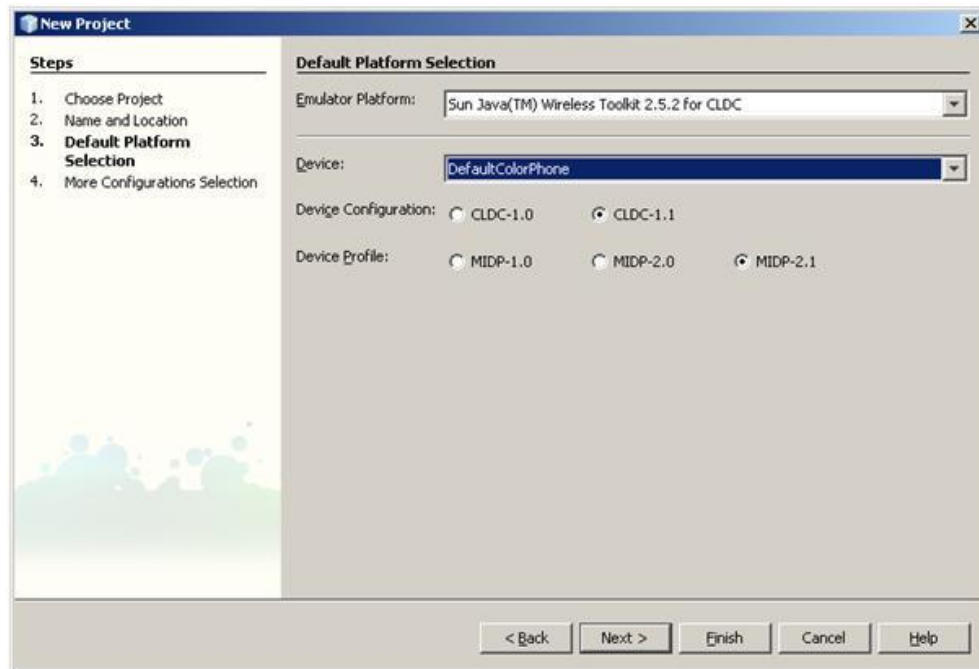
Gambar II.2 Tampilan Choose Project

1. Pilih MIDP Application dan tekan **Next**. Selanjutnya anda akan di bawa ke layer Name and Location seperti di bawah.



Gambar II.3. Tampilan Name and Location

2. Input Project Name yang sesuai dengan nama project anda dan hilangkan centang di Create Hello Midlet. Tekan **Next**



Gambar II.4. Tampilan Default Platform Selection

Ini adalah layar untuk menentukan target perangkat yang akan anda buat aplikasinya. Pada baris **Emulator Platform** adalah pilihan emulator yang anda Install di computer anda, untuk saat ini anda hanya menginstall emulator bawaan dari NetBeans anda. Setelah anda merasa perlu menginstall emulator platform untuk spesifik dari vendor lain seperti Nokia, Sony Ericsson dan Motorola anda dapat mengubahnya disini.

3. Pilih Next dan langsung pilih Finish.

Kita mulai dengan membuat class MIDlet baru untuk project anda. Klik kanan para project anda di project explore dan pilih new-MIDlet, namakan class baru itu dengan nama MIDletPertama. Kode terakhir file MIDlet anda terlihat seperti di bawah ini.

```

/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */

import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;

/**
 * @author Puputs
 */
public class MIDletPertama extends MIDlet {
    public void startApp() {

    }

    public void pauseApp() {

    }

    public void destroyApp(boolean unconditional) {

    }
}

```

Gambar II.5. Source code pada J2ME

Untuk pertama kali kita akan menampilkan “Hello J2ME World” pada layar HP. Ada dua jenis tampilan dalam MIDlet.

- Low Level menggunakan class Canvas
- High Level menggunakan Form

Disini kita akan menggunakan form untuk mempermudah anda mengerti pembentukan tampilan di MIDlet.

```

public void startApp() {
    Form frPertama = new Form("Judul");
    frPertama.append("Hello J2ME World");
    Display.getDisplay(this).setCurrent(frPertama);
}

```

Gambar II.6. Source code pada J2ME

Perhatikan kode di atas, frPertama adalah object Form yang akan kita tampilkan di layer, kita memberi judulForm tersebut dengan “Judul”. Lalu method append adalah method untuk menambahkan Item pada Form, Item yang ingin saya tambahkan adalah String “Hello J2ME World”. Statement Display adalah statement untuk menampilkan object frPertama kedalam layer.

4. Pilih F11 untuk **build** project dan F6 untuk **run project**. Pilih midlet yang ingin dijalankan dan tekan launch di emulator, selanjutnya emulator anda akan menampilkan Form yang anda buat, seperti di bawah.



Gambar II.7 Tampilan Emulator

II.4 Sharing Data Device

II.4.1 Inframerah (*Infrared*)

Inframerah adalah radiasi elektromagnetik dari panjang gelombang lebih panjang dari cahaya tampak, tetapi lebih pendek dari radiasi gelombang radio. Namanya berarti "bawah merah" (dari bahasa Latin *infra*, "bawah"), merah merupakan warna dari cahaya tampak dengan gelombang terpanjang. Radiasi inframerah memiliki jangkauan tiga "order" dan memiliki panjang

gelombang antara 700 nm dan 1 mm. Inframerah ditemukan secara tidak sengaja oleh Sir William Herschell, astronom kerajaan Inggris ketika ia sedang mengadakan penelitian mencari bahan penyaring optik yang akan digunakan untuk mengurangi kecerahan gambar matahari dalam tata surya teleskop.

II.4.2 Koneksi Internet Nirkabel (*wireless*)

Wireless adalah teknologi tanpa kabel, dalam hal ini adalah melakukan hubungan telekomunikasi dengan menggunakan gelombang elektromagnetik sebagai pengganti kabel. Saat ini teknologi *wireless* berkembang dengan pesat, secara kasat mata dapat dilihat dengan semakin banyaknya pemakaian telepon selular, selain itu berkembang pula teknologi *wireless* yang digunakan untuk akses internet.

Wireless LAN menggunakan gelombang *elektromagnetik* (radio dan inframerah) untuk melakukan komunikasi data menyalurkan data dari satu point ke point yang lain tanpa melalui fasilitas fisik. Koneksi ini menggunakan frekuensi tertentu untuk menyalurkan data tersebut, kebanyakan *Wireless LAN* menggunakan frekuensi 2,4 GHz. Frekuensi inilah yang disebut dengan Industrial, Scientific and Medical Band atau sering disebut ISM Band, seperti telah dijelaskan di atas.

II.4.3 Bluetooth

Bluetooth adalah sebuah teknologi komunikasi *wireless* (tanpa kabel) yang beroperasi dalam pita frekuensi 2,4 GHz *unlicensed ISM (Industrial, Scientific and Medical)* dengan menggunakan sebuah *frequency hopping tranceiver* yang mampu menyediakan layanan komunikasi data dan suara secara *real-time* antara *host-host bluetooth* dengan jarak jangkauan layanan yang terbatas. *Bluetooth* sendiri dapat berupa *card* yang bentuk dan fungsinya hampir sama dengan *card* yang digunakan untuk *wireless local area network (WLAN)* dimana menggunakan frekuensi radio standar IEEE 802.11, hanya saja pada *bluetooth* mempunyai jangkauan jarak layanan yang lebih pendek dan kemampuan *transfer* data yang lebih rendah.

Pada dasarnya *bluetooth* diciptakan bukan hanya menggantikan atau menghilangkan penggunaan kabel didalam melakukan pertukaran informasi, tetapi juga mampu menawarkan fitur yang baik

untuk teknologi *mobile wireless* dengan biaya yang relative rendah, konsumsi daya yang rendah, *interoperability* yang menjanjikan, mudah dalam pengoperasian dan mampu menyediakan layanan yang bermacam-macam. Untuk memberi gambaran yang lebih jelas mengenai teknologi *bluetooth* yang relatif baru ini, berikut diuraikan tentang sejarah munculnya *bluetooth* dan perkembangannya, teknologi yang digunakan pada sistem *bluetooth* dan aspek layanan yang mampu disediakan, uraian tentang perbandingan metode modulasi *spread spectrum FHSS* (*Frequency Hopping Spread Spectrum*) yang digunakan oleh *bluetooth* dibandingkan dengan metode *spread spectrum DSSS* (*Direct Sequence Spread Spectrum*) serta interferensi *bluetooth* dengan ponsel.

II.4.3.1 Aplikasi dan Layanan *Bluetooth*

Protokol *bluetooth* menggunakan sebuah kombinasi antara *circuit switching* dan *packet switching*. *Bluetooth* dapat mendukung sebuah kanal data asinkron, tiga kanal suara sinkron simultan atau sebuah kanal dimana secara bersamaan mendukung layanan data asinkron dan suara sinkron. Setiap kanal suara mendukung sebuah kanal suara sinkron 64 kb/s. Kanal asinkron dapat mendukung kecepatan maksimal 723,2 kb/s asimetris, dimana untuk arah sebaliknya dapat mendukung sampai dengan kecepatan 57,6 kb/s. Sedangkan untuk mode simetris dapat mendukung sampai dengan kecepatan 433,9 kb/s.

Sebuah perangkat yang memiliki teknologi *wireless bluetooth* akan mempunyai kemampuan untuk melakukan pertukaran informasi dengan jarak jangkauan sampai dengan 10 meter (*~30 feet*), bahkan untuk daya kelas 1 bisa sampai pada jarak 100 meter. Sistem *bluetooth* menyediakan layanan komunikasi *point to point* maupun komunikasi *point to multipoint*. Produk *bluetooth* dapat berupa *PC card* atau *USB adapter* yang dimasukkan ke dalam perangkat. Perangkat-perangkat yang dapat diintegrasikan dengan teknologi *bluetooth* antara lain : *mobile PC*, *mobile phone*, *PDA* (*Personal Digital Assistant*), *headset*, *kamera digital*, *printer*, *router* dan masih banyak peralatan lainnya. Aplikasi yang dapat disediakan oleh layanan *bluetooth* ini antara lain : *PC to PC file transfer*, *PC to PC file synch* (*notebook to desktop*), *PC to mobile phone*, *PC to PDA*, *wireless headset*, *LAN connection via ethernet access point* dan sebagainya.

II.4.3.2 Diskripsi Umum Sistem *Bluetooth*

Sistem *bluetooth* terdiri dari sebuah *radio transceiver*, *baseband link Management dan Control*, *Baseband (processor core, SRAM, UART, PCM USB Interface)*, *flash* dan *voice code*. sebuah *link manager*. *Baseband link controller* menghubungkan perangkat keras radio ke *baseband processing* dan *layer* protokol fisik. *Link manager* melakukan aktivitas-aktivitas protokol tingkat tinggi seperti melakukan *link setup*, autentikasi dan konfigurasi.

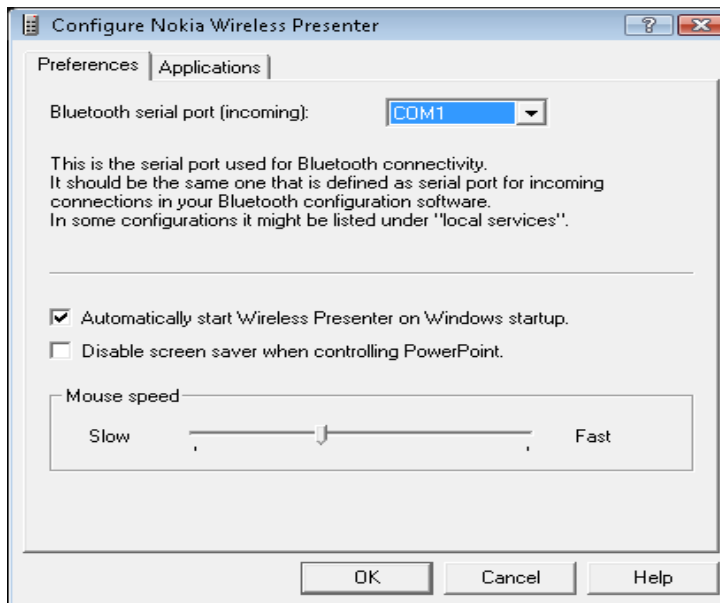
II.5 Remote

Pengendali jarak jauh (*remote control*) adalah sebuah alat elektronik yang digunakan untuk mengoperasikan sebuah mesin dari jarak jauh.

Istilah *remote control* juga sering disingkat menjadi "remote" saja. *Remote* juga sering kali mengacu pada istilah "controller", "donker", "doofer", "zapper" "click-buzz", "box", "flipper", "zippity", "clicker", or "changer". Pada umumnya, pengendali jarak jauh digunakan untuk memberikan perintah dari kejauhan kepada televisi atau barang-barang elektronik lainnya seperti sistem stereo dan pemutar DVD. *Remote control* untuk perangkat-perangkat ini biasanya berupa benda kecil nirkabel yang dipegang dalam tangan dengan sederetan tombol untuk menyesuaikan berbagai setting, seperti misalnya saluran televisi, nomor trek, dan volume suara. Malah, pada kebanyakan peranti modern dengan kontrol seperti ini, remote controlnya memiliki segala kontrol fungsi sementara perangkat yang dikendalikan itu sendiri hanya mempunyai sedikit kontrol utama yang mendasar. Kebanyakan remote berkomunikasi dengan perangkatnya masing-masing melalui sinyal-sinyal infra merah dan beberapa saja melalui sinyal radio. Remote control biasanya menggunakan baterai AAA yang kecil atau AA sebagai catu dayanya.

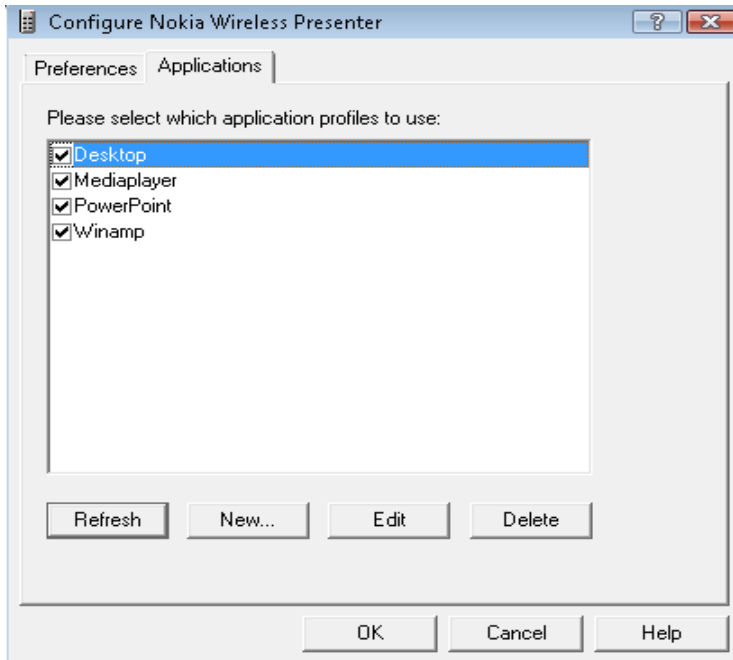
II.6 Wireless Presenter

adalah sebuah layanan aplikasi yang terdapat pada *handphone* yang mendukung keterhubungan antar koneksi dari satu media ke media lain dengan menggunakan *wireless (nircable)*. Hal ini yang biasanya terdapat pada jenis *Handphone Nokia* yang di sebut dengan *Nokia wireless Presenter* seperti gambar di bawah ini.



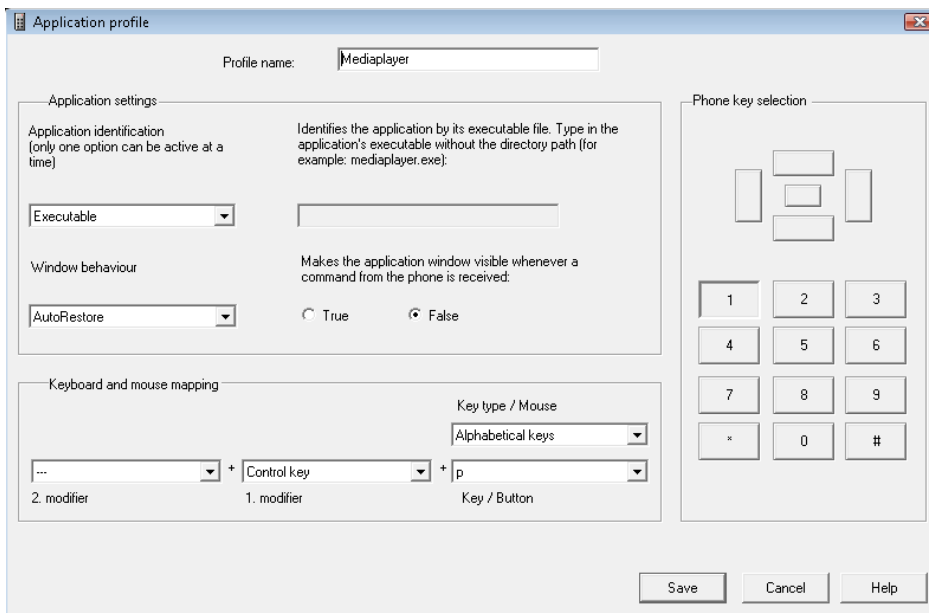
Gambar II.8 Konfigurasi Nokia Wireless Presenter

Menurut gambar yang ada di atas kita dapat mengkonfigurasi *port* yang akan kita pilih sesuai dengan port yang ada di *handphone*, setelah itu dapat memilih aplikasi yang ingin kita jalankan seperti gambar di bawah ini
mengkonfigurasi control panel seperti gambar di bawah ini



Gambar II.9 Konfigurasi Nokia Wireless Presenter

Setelah memilih aplikasi yang ingin kita *remote* kemudian menentukan fungsi *buton* pada layar navigasi control seperti gambar di bawah ini.

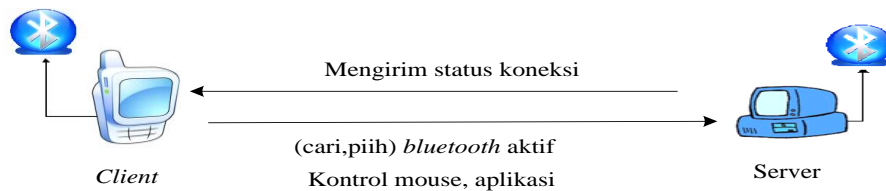


Gambar II.10. Navigasi kontrol aplikasi

Bab III Analisis

III.1 Deskripsi Umum Sistem *Mobile Remote Application*

Deskripsi umum Sistem *Mobile Remote Application* ini menggambarkan interaksi antara pengguna dan sistem. Untuk lebih lanjutnya tentang deskripsi umum sistem *Mobile Remote Application* dapat dilihat pada Gambar III.1 Deskripsi umum sistem.



Gambar III.1 Deskripsi umum sistem

Keterangan :

Sebelum user mengontrol aplikasi yang ada pada pc, user akan mengaktifkan *Bluetooth* pada *handphone* dan *PC* terlebih dahulu, setelah itu user mengaktifkan aplikasi *Mobile Remote kontrol client* dan *server*, kemudian user akan mencari dan memilih jaringan transfer data (*Bluetooth*) yang aktif pada *client*, jika sistem telah mengidentifikasi jaringan *bluetooth* maka *server* akan menampilkan status koneksi apakah sudah terhubung atau belum. Setelah terhubung user mengaktifkan navigasi kontrol *mouse* seperti *up, right, left, down* pada *client* yang nantinya dapat memilih aplikasi yang ingin digunakan seperti *Microsoft Power Point, Windows Media Player* dan *Winamp*.

No	<i>Client</i>	<i>Server</i>
1.	Cari <i>bluetooth</i>	Tampil koneksi
2.	Pilih <i>bluetooth</i>	<i>Bluetooth</i> terkoneksi
3.	Kontrol <i>Mouse</i>	Jalankan aplikasi
4.	Kontrol Aplikasi	Navigasi dapat diatur melalui <i>client</i>

Tabel III.1 fungsi *client* dan *server*.

III.2 Fitur Utama Perangkat Lunak

III.3 Kebutuhan Fungsional

Kebutuhan Fungsional pada aplikasi yaitu :

F-01 Sistem ini mampu melakukan koneksi aplikasi *client* ke *server* menggunakan jaringan *bluetooth*.

F-02 Sistem ini mampu menjalankan kontrol *mouse*

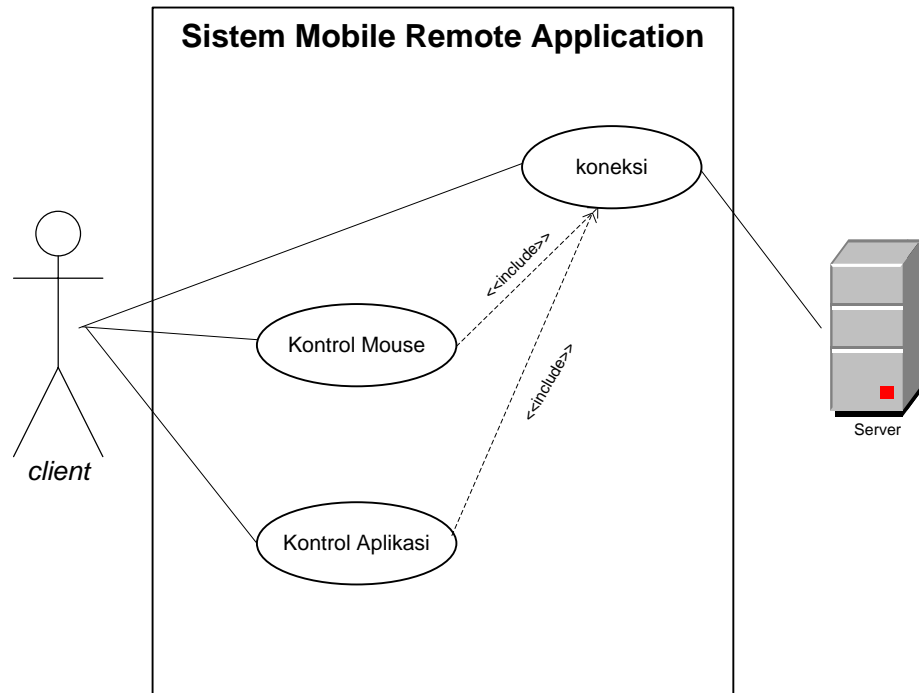
F-03 Sistem ini mampu menjalankan kontrol aplikasi.

III.4 Batasan Sistem

- Sistem tidak dapat berjalan selain pada *handphone* dengan fitur MIDP 2.0 dan CLDC 1.1

III.5 Use Case

III.5.1 Diagram Use Case



Gambar III.5.1 Diagram Use Case Sistem mobile remote application

Keterangan : Pada diagram use case ini menjelaskan interaksi antar pengguna dengan aplikasi. Interaksi pengguna setiap *use case* dapat dilihat pada skenario *use case*.

III.5.2 Skenario Use Case

Skenario *use case mobile remote application* adalah, sebelum menggunakan sistem *mobile remote application* untuk mengontrol *mouse* dan aplikasi. *Client* dan *server* sudah saling terhubung.

III.5.3 Use Case Koneksi

Kondisi awal : Koneksi ke *server*

Kondisi akhir : Telah terkoneksi ke *server*

Skenario : User mengaktifkan *Bluetooth* pada *client* dan *server*. Kemudian *client* mencari koneksi *bluetooth* yang aktif pada *server* setelah ketemu user akan memilih *bluetooth* yang aktif untuk melakukan koneksi ke *server* jika terkoneksi *server* akan menampilkan status koneksi dan menu aplikasi pada *client*.

III.5.4 Use Case Kontrol *Mouse*

Kondisi awal : *Mouse* belum aktif

Kondisi akhir : *Mouse* telah aktif dan dapat digunakan

Skenario : Sebelum menggerakkan cursor seperti *up, right, left, down*. *Client* dan *server* sudah harus terhubung setelah terhubung user akan menggerakkan cursor yang nantinya *client* akan mengirim perintah ke *server* kemudian *server* akan melakukan perintah sesuai kebutuhan *client*.

III.5.5 Use Case Kontrol Aplikasi

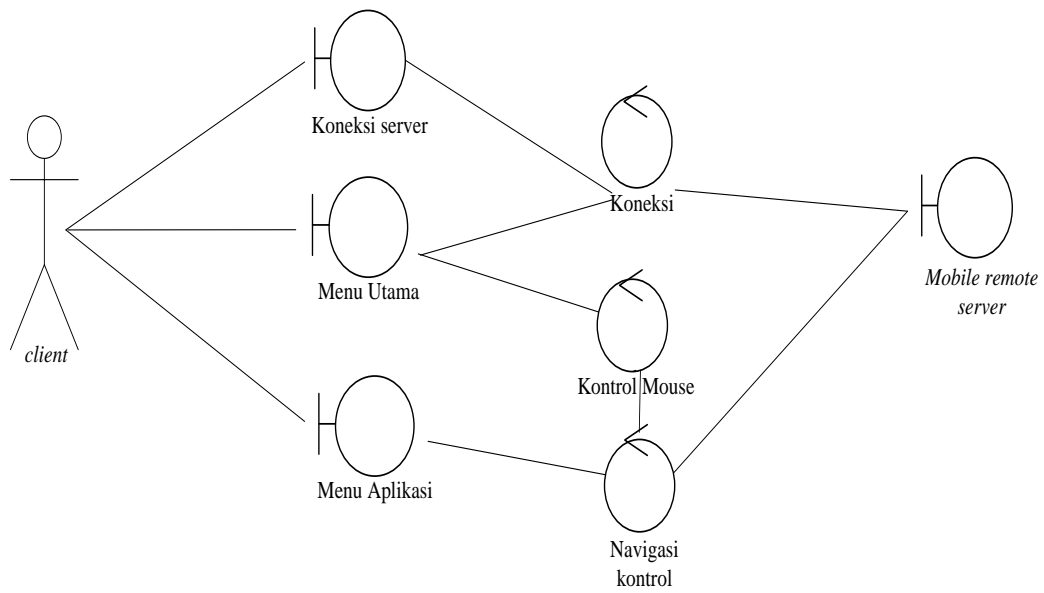
Kondisi Awal : User melakukan kontrol aplikasi

Kondisi Akhir : User telah melakukan kontrol aplikasi

Skenario : Setelah *client server* telah terhubung. Pada *client* akan tampil pilihan menu aplikasi yang ingin digunakan user untuk mengontrol aplikasi seperti *power point, Winamp* dan *windows media player*. Setelah user telah memilih aplikasi yang ingin dikontrol. *client* akan menampilkan pilihan seperti *next, play, pause, stop, volume up/down, previous, mute*, yang nantinya perintah tersebut dikirim *client* ke *server* kemudian *server* akan mengoperasikan sesuai perintah dari *client*.

III.6 Analisis Kelas

III.6.1 Analisis Kelas



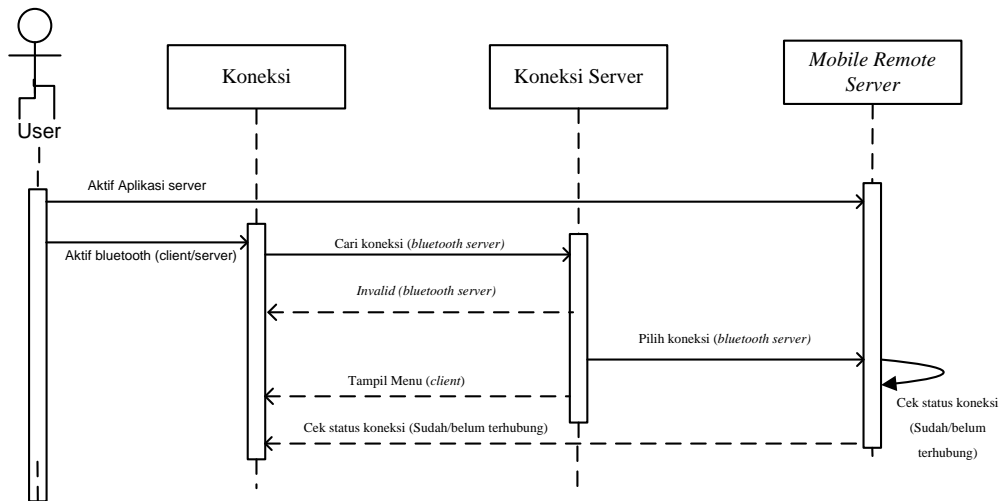
Gambar III.6.1 Analisis Kelas

Bab IV Perancangan

Sequence Diagram digunakan untuk menjelaskan lebih rinci jalannya aplikasi setiap use case. diagram ini merupakan gambaran dari skenario use case pada bab sebelumnya yang merupakan respon dari sebuah *event* untuk menghasilkan *output* tertentu.

IV.1 *Sequence Diagram*

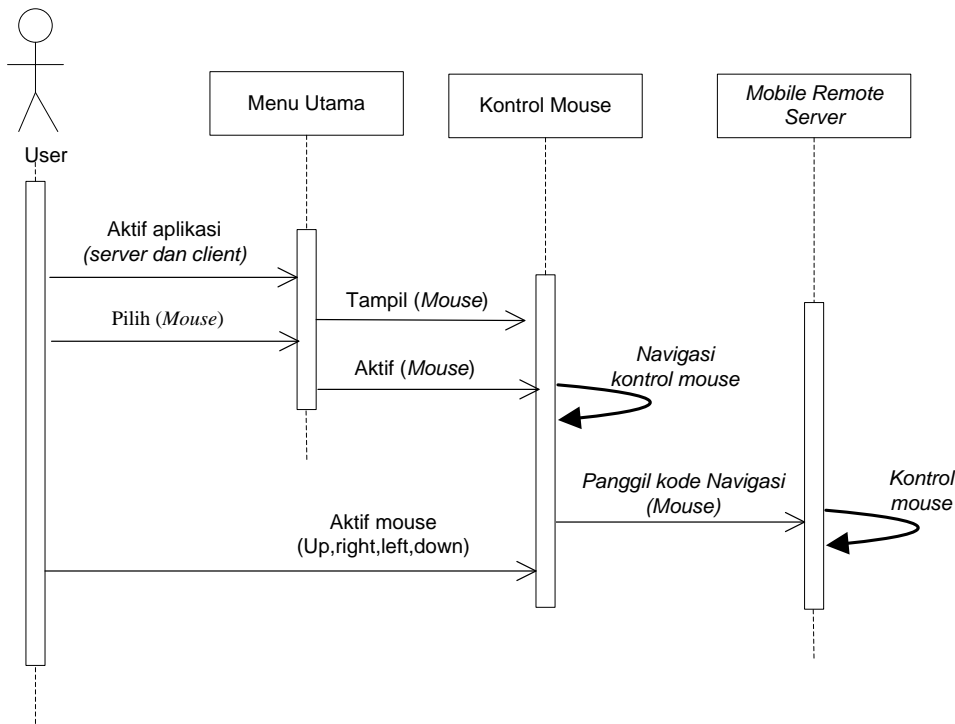
IV.1.1 *Sequence Diagram Use Case Koneksi*



Gambar V.1.1 *Sequence Diagram Use Case Koneksi*

Keterangan : user mengaktifkan *bluetooth server* dan *client*, kemudian user mencari *bluetooth* yang aktif setelah ketemu, user akan memilih *bluetooth* untuk melakukan koneksi *server*. Kemudian pada aplikasi *server* akan mencari *bluetooth* benar atau salah jika benar sistem akan mengirimkan status koneksi.

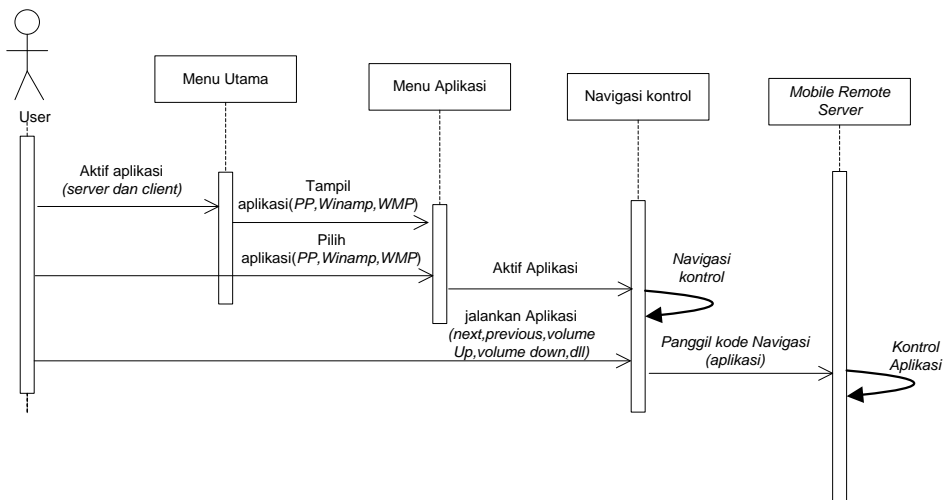
IV.1.2 Sequence Diagram Use Case Kontrol Mouse



Gambar IV.1.2 Sequence Diagram Use Case Kontrol Mouse

Keterangan : Setelah aplikasi client dan server telah terhubung, kemudian pada layar menu utama user memilih control mouse pada client untuk mengaktifkan kursor mouse seperti up ,right, left, down.

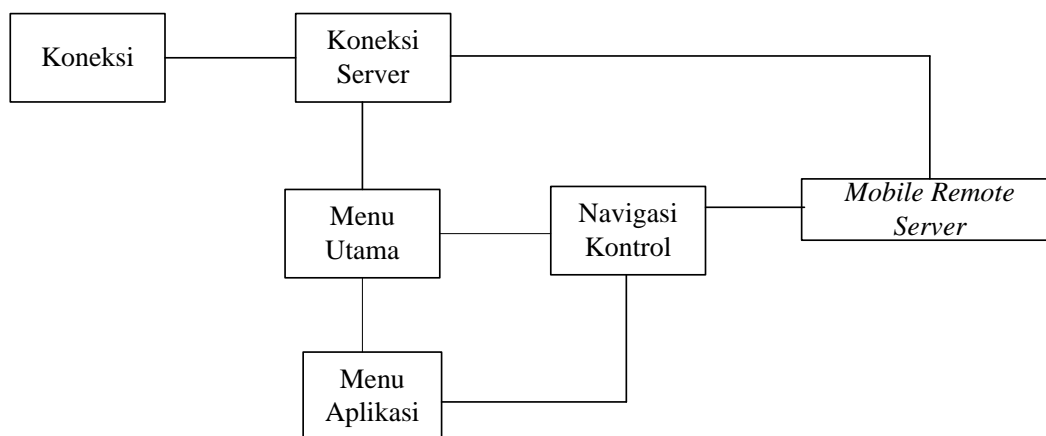
IV.1.3 Sequence Diagram Use Case Kontrol Aplikasi



Gambar IV.1.3 Sequence Diagram Use Case Kontrol Aplikasi

Keterangan : Setelah aplikasi *client* dan *server* sudah terhubung kemudian pada layar kontrol aplikasi user memilih aplikasi pada menu aplikasi di *client* seperti (*Power point, Winamp, Windows Media Player*) untuk melakukan kontrol aplikasi yang nantinya akan diproses di navigasi kontrol seperti : *next, previous, pause, stop, volume up/down*. kemudian user dapat menggunakan navigasi kontrol untuk mengontrol aplikasi yang dipilih.

IV.2 Diagram Kelas



Gambar IV.2 Diagram Kelas

IV.3 Rancangan Kelas Rinci

IV.3.1 Kelas Koneksi

Koneksi
Nama_bluetooth()
+Cari <i>bluetooth</i> ()
+Status koneksi()

Deskripsi :

- Kelas ini berfungsi untuk mencari *bluetooth* aktif pada *server*.

Algoritma

Nama Kelas : Koneksi.class

Nama Operasi : Koneksi

Algoritma : *(Algo-001)*

```
Koneksi {new String: DEVICE}
Read Koneksi :
if kon == Koneksi
then Device Ketemu
else
then Error Sending Command
```

IV.3.2 Kelas Koneksi Server

KoneksiServer
Nama_bluetooth()
+Pilih <i>Bluetooth</i> ()
+ <i>invalid bluetooth</i> ()
+Tampil Menu
+Status koneksi()

Deskripsi :

- Kelas ini berfungsi untuk menghubungkan koneksi antara client dan *server*.

Algoritma

Nama Kelas : *KoneksiServer.class*

Nama Operasi : *KoneksiServer*

Algoritma : *(Algo-002)*

Koneksi {new String: DEVICE} Read Koneksi : if konn == Koneksi then Welcome else then Tidak terhubung dengan <i>server</i>

IV.3.3 Kelas Navigasi Kontrol

NavigasiKontrol
Nama_ <i>bluetooth</i>
+search() +Show <i>bluetooth</i> () +Aktif Aplikasi() +Jalan aplikasi()

Deskripsi :

- Kelas ini berfungsi untuk proses jalan aplikasi yang berjalan

Algoritma

Nama Kelas : *NavigasiKontrol.class*

Nama Operasi : *NavigasiKontrol*

Algoritma : *(Algo-003)*

Read App=(aplikasi, <i>mouse</i>) String ketemu; If ketemu Then Aplikasi If Ketemu Then <i>Mouse</i> Else Then Tidak Terhubung dengan <i>server</i>

IV.3.4 Kelas Menu Utama

MenuUtama
-
+ <i>show</i> MenuUtama()

Deskripsi :

- Kelas ini berfungsi menampilkan menu utama

Algoritma

Nama Kelas : MenuUtama.class

Nama Operasi : MenuUtama

Algoritma : (Algo-004)

Read Menu = (Aplikasi , <i>Mouse</i> ,) Show.Aplikasi; Show. <i>Mouse</i> ;

IV.3.5 Kelas Menu Aplikasi

Menu Aplikasi
- <i>Windows media player</i> - <i>Winamp</i> - <i>Power point</i>
+Pilih aplikasi() + <i>Back</i> ()

Deskripsi :

- Kelas ini berfungsi menampilkan pilihan aplikasi

Algoritma

Nama Kelas : Menu Aplikasi.class

Nama Operasi : Aplikasi

Algoritma : (Algo-005)

```
Read Aplikasi == (Windows Media Player, Winamp, PowerPoint)
If aplikasi Windows Media Player
Then show.Windows Media Player
If aplikasi Winamp
Then show.winamp
If aplikasi PowerPoint
Then show.PowerPoint
```

IV.3.6 Kelas Mobile Remote Server

<i>Server</i>
-
+Koneksi() +Exit()

Deskripsi :

- Kelas ini berfungsi untuk menerima koneksi dari *client* dan perintah dari *client*

Algoritma

Nama Kelas : *Mobile Remote Server.class*

Nama Operasi : *Server*

Algoritma : (Algo-006)

```
Read Koneksi == Koneksi.server.
If Koneksi
Then Server AND Client Aktif
Else
Gagal Terhubung
```

IV.4 Perancangan antar muka

IV.4.1 Tampilan Proses Koneksi *client*

IV.4.1.1 Proses koneksi *client*



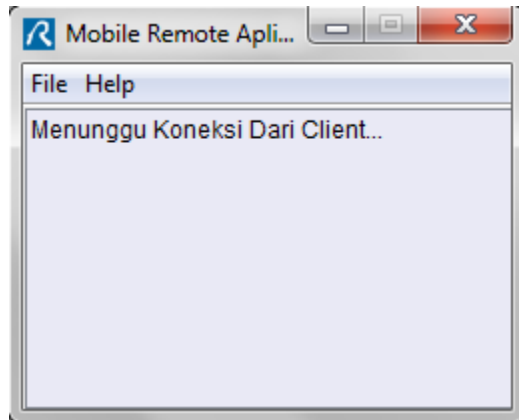
Gambar IV.4.1 Tampilan Koneksi *client*

IV.4.1.2 Deskripsi

Id_Objek	Jenis	Nama	Keterangan
<i>Proses bar</i>	<i>Proses bar</i>	<i>Searching for server</i>	Jalannya pencarian <i>bluetooth</i> aktif
<i>Button</i>	<i>Button</i>	<i>Cancel</i>	Jika di klik, akan membatalkan pencarian dan keluar dari aplikasi

IV.4.2 Koneksi Server

IV.4.2.1 Tampilan *Mobile Remote Application*



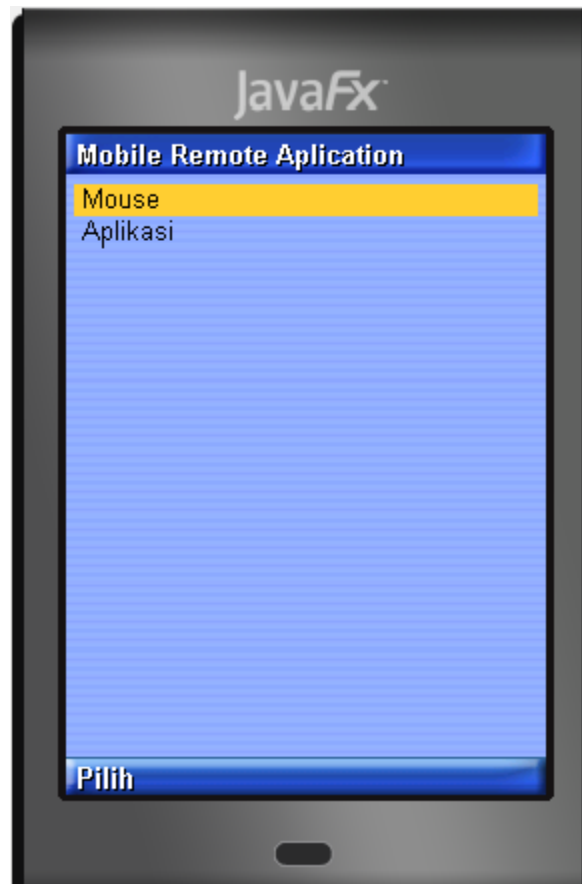
Gambar IV.4.2 Tampilan *Mobile Remote Application*

IV.4.2.2 Deskripsi

Id_Objek	Jenis	Nama	Keterangan
<i>Proses bar</i>	<i>Proses bar</i>	<i>Searching for server</i>	Jalannya menunggu koneksi <i>client</i>
<i>Button 1</i>	<i>Button</i>	<i>File</i>	Jika ditekan, akan ada menu <i>exit</i> untuk keluar aplikasi
<i>Button 2</i>	<i>Button</i>	<i>Help</i>	Jika ditekan, akan muncul layar <i>about</i>

IV.4.3 Menu Utama

IV.4.3.1 Tampilan Menu Utama



Gambar IV.4.3 Tampilan Menu Utama

IV.4.3.2 Deskripsi

Id_Objek	Jenis	Nama	Keterangan
<i>Button</i>	Button	<i>Mouse</i>	Jika ditekan, akan mengaktifkan navigasi kontrol <i>mouse</i>
<i>Button</i>	<i>Button</i>	<i>Aplikasi</i>	Jika ditekan, akan masuk ke layar Menu Aplikasi

IV.4.4 Menu Aplikasi

IV.4.4.1 Tampilan Menu Aplikasi



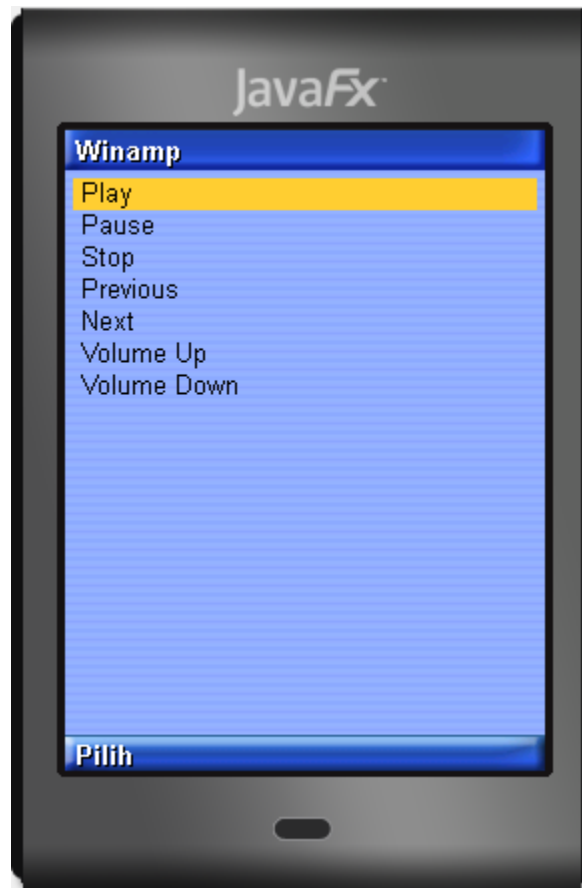
Gamabar IV.4.4Tampilan Menu Aplikasi

IV.4.4.2 Deskripsi

Id_Objek	Jenis	Nama	Keterangan
<i>Button</i>	<i>Button</i>	<i>Winamp</i>	Jika ditekan, akan ke layar navigasi kontrol <i>winamp</i>
<i>Button</i>	<i>Button</i>	<i>Windows Media Player</i>	Jika ditekan, akan ke layar navigasi kontrol <i>Windows Media layer</i>
<i>Button</i>	<i>Button</i>	<i>Power Point</i>	Jika ditekan, akan ke layar navigasi kontrol <i>winamp</i>

IV.4.5 Navigasi kontrol

IV.4.5.1 Tampilan Navigasi kontrol



Gambar IV.4.5 Tampilan Navigasi Kontrol

IV.4.5.2 Deskripsi

Id_Objek	Jenis	Nama	Keterangan
<i>Button</i>	<i>Button</i>	<i>Play</i>	Jika ditekan, maka akan menjalankan aplikasi
<i>Button</i>	<i>Button</i>	<i>Pause</i>	Jika ditekan, maka akan menghentikan sejenak jalannya aplikasi
<i>Button</i>	<i>Button</i>	<i>Stop</i>	Jika ditekan, maka akan memberhentikan jalannya aplikasi

Id_Objek	Jenis	Nama	Keterangan
<i>Button</i>	<i>Button</i>	<i>Previous</i>	Jika ditekan, maka akan mengulang jalannya aplikasi.
<i>Button</i>	<i>Button</i>	<i>Next</i>	Jika ditekan, maka akan melanjutkan jalannya aplikasi.
<i>Button</i>	<i>Button</i>	<i>Volume Up</i>	Jika ditekan, maka akan menambah <i>volume</i> / suara pada aplikasi
<i>Button</i>	<i>Button</i>	<i>Volume Down</i>	Jika ditekan, maka akan menurunkan volume / suara pada aplikasi.

Bab V Implementasi Dan Pengujian

Setelah dilakukan perancangan maka tahap selanjutnya adalah implementasi dan pengujian pada perangkat lunak. Implementasi akan menghasilkan aplikasi yang dapat dijalankan di lingkungan operasional. Untuk mengetahui apakah aplikasi tersebut dapat melakukan fungsi sesuai deskripsi perancangan maka perlu dilakukan pengujian. Pada perangkat keras akan dilakukan pengukuran dan pengujian untuk mengetahui kinerja sistem.

V.1 Implementasi Kelas

No	Nama Kelas	Nama File Fisik	Nama File Executable
1	Koneksi	Koneksi.java	<i>MobileRemoteClient.jar</i>
2	MenuUtama	MenuUtama.java	
3	MenuAplikasi	MenuAplikasi.java	
4	KoneksiServer	Koneksi.java	
5	NavigasiKontrol	NavigasiKontrol.java	
6	<i>MobileRemoteServer</i>	<i>MobileRemoteServer.java</i>	<i>MobileRemoteServer.class</i>

Tabel V.1 Implementasi kelas

Keterangan : pada tabel di atas merupakan tabel implementasi kelas. untuk mobile remote client memiliki 5 kelas yaitu kelas koneksi, KoneksiServer, NavigasiKontrol, MenuUtama dan MenuAplikasi. Pada implementasi ini kelas-kelas yang digunakan sesuai dengan kelas perancangan yaitu memiliki 5 kelas. Pada sisi Server memiliki 1 kelas yaitu *MobileRemoteServer*. Adapun berupa tambahan kelas yaitu : *Bluetooth*, *mainCanvas*, *SplashScreen*.

V.2 Implementasi Antar Muka

V.2.1 Implementasi Antar Muka *Client* dan *Server*

No	Antar Muka	Nama File Fisik	Nama File Executable
1	Koneksi	Koneksi.java	Koneksi.class
2	MenuAplikasi	MenuAplikasi.java	MenuAplikasi.class
3	MenuUtama	MenuUtama.java	MenuUtama.class
4	<i>MobileRemote Server</i>	<i>MobileRemoteServer.java</i>	<i>MobileRemoteServer.class</i>

Tabel V.2 Implementasi antar muka *server client* dan *server*

V.3 Pengujian

V.3.1 Identifikasi Pengujian

V.3.1.1 Pada tahap pengujian ini digunakan metode pengujian.

1. Pengujian dengan menggunakan *handphone* dan *laptop(Bluetooth)*

Dalam pengujian ini, dibutuhkan *laptop server* yang menjadi pusat seluruh data.

1.1. Adapun spesifikasi *laptop* yang dibutuhkan adalah:

- Mempunyai fasilitas *Bluetooth*.
- Emulator *Netbeans*.
- Menjalankan semua aplikasi *server* seperti : *Power Point*, *Winamp* dan *Windows Media Player*.

1.2. Adapun hal-hal yang dibutuhkan adalah :

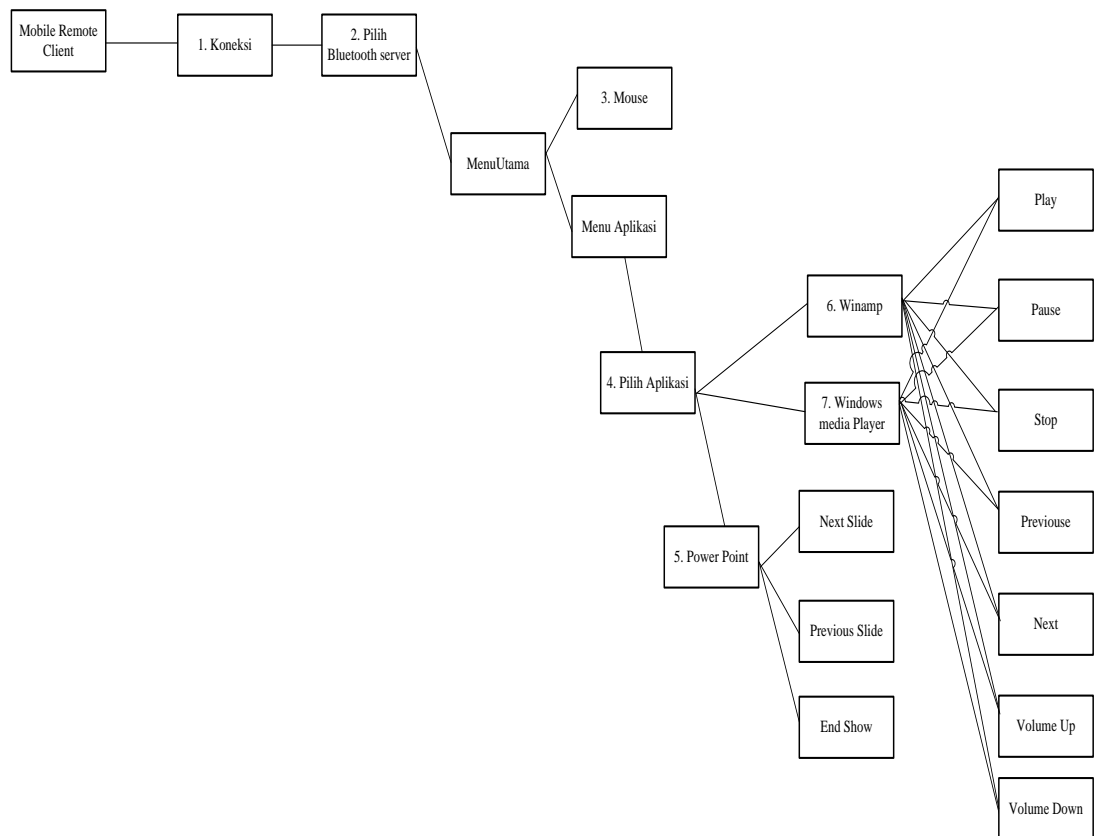
- Fitur *CLDC* sebagai konfigurasi untuk menjalankan program *java*.
- Fitur *MIDP* sebagai penyedia *library java* berupa jaringan, *interface* dan *timer*.
- *Bluetooth* sebagai media untuk menghubungkan antara *client* dan *server*.

1.3. Pengujian pada *handphone* nokia N72

Tahap-tahap menjalankan *Mobile Remote Application* pada nokia N72 adalah :

- Mengaktifkan *Bluetooth* pada *handphone*.
- *Install MobileRemoteClient.jar* ke *handphone*.
- *Mobile Remote client* dapat dijalankan.

V.3.2 Skenario Pengujian Aplikasi *Mobile Remote Application*



Gambar V.3.2 Skenario pengujian

V.4 Dokumen Rinci Testing

V.4.1 Tim Penguji

Mir'atul Khusna Mufida S.ST

V.4.2 Hasil Rinci Pengujian

Hasil rinci pengujian pada aplikasi *Mobile Remote Application* di sajikan di tabel di bawah berikut ini.

No	User	Use Case	Fungsi yang di Uji	Kondisi Awal	Input	Hasil yang di Harapkan	Ouput	H	
1	User	Koneksi	Koneksi <i>bluetooth</i>	Layar koneksi	pilih <i>bluetooth</i> = POLTEK	Sistem mampu mengidentifikasi jaringan <i>bluetooth</i> yang digunakan	Muncul Menu Utama		
2	User	Kontrol <i>mouse</i>	Mengontrol <i>mouse</i>	Layar menu utama	Pilih <i>mouse</i>	Sistem dapat mengaktifkan navigasi kontrol <i>mouse</i>	<i>Kursor mouse</i> pada PC dapat di jalankan		
3	User	Kontrol Aplikasi	Pilih aplikasi (<i>Power Point</i> ,	Layar menu	Pilih aplikasi (<i>Power Point</i> ,	Sistem dapat mengaktifkan navigasi	Muncul navigasi		

			<i>Winamp dan Windows Media Player)</i>	aplikasi	<i>Winamp dan Windows Media Player)</i>	Kontrol aplikasi yang dipilih user	aplikasi		
4	User	-	<i>Power Point</i>	Layar menu aplikasi	<i>Next slide, Previous slide dan End show</i>	Sistem navigasi kontrol dapat melanjutkan slide, kembali slide sebelumnya dan mengakhiri slide.	Slide		
	User	-	<i>Winamp dan Windows Media Player</i>		<i>Play</i>	Sistem dapat memulai		<i>Winamp</i>	<i>WMP</i>
					<i>Pause</i>	Sistem dapat memberhentikan tanpa harus berhenti total dan mengulanginya dari awal			
					<i>Stop</i>	Sistem dapat memberhentikan			
					<i>Previous</i>	Sistem dapat mengembalikan sebelumnya			

					<i>Next</i>	Sistem dapat melanjutkan file berikutnya			
					<i>Volume Up</i>	Sistem dapat menambahkan suara/ <i>volume</i>			
					<i>Volume Down</i>	Sistem dapat mengurangi suara/ <i>volume</i>			

Tabel V.4.2 Hasil rinci pengujian

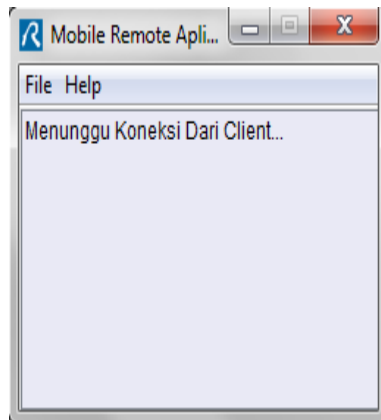
Keterangan :

- **H** : Pengujian dengan menggunakan 1 laptop dan *handphone*

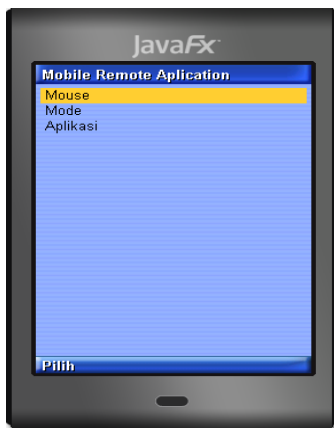
V.5 Daftar Gambar



Gambar V.5.1 Search koneksi



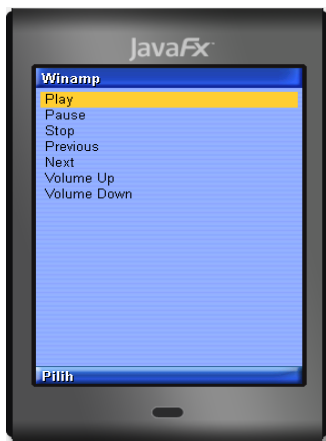
Gambar V.5.2 Mobile Remote Application



Gambar V.5.3 Menu Utama



Gambar V.5.4 Menu Aplikasi



Gambar V.5.5 Navigasi Kontrol

Bab VI Kesimpulan dan Saran

VI.1 Kesimpulan

1. Sistem sudah dapat menjalankan dan mengontrol navigasi kontrol pada aplikasi *server* dengan menggunakan *handphone* yang memiliki fitur *Bluetooth*, *java*, MIDP 2.0 dan CLDC 1.1
2. Aplikasi tidak dapat dijalankan pada *handphone* dengan Sistem Operasi *Android*.

VI.2 Saran

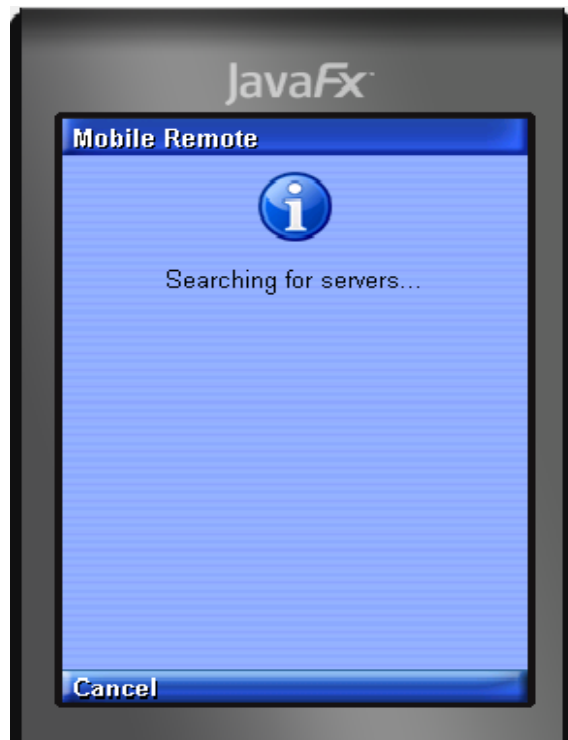
1. Diharapkan aplikasi *server* ini nantinya dapat berjalan pada sistem operasi lainnya (*linux*, *mac*,).
2. Diharapkan aplikasi *client* ini nantinya dapat berjalan di *handphone* seperti (*Android*, *Blackberry*, *iOS*).

Daftar Pustaka

1. Kadir, Abdul. Bahasa Pemrogram Java™ 2. Yogyakarta: Andi Yogyakarta, 2004
2. Raharjo,Budi., Heryanto,imam & Haryono,Arif.Tuntunan Pemrograman Java untuk handphone.Bandung:Informatika,2007.
3. <http://netbeans.org/kb/docs/java/gui-functionality.html/>,_diakses pada tanggal 3 maret 2010.
4. Nugroho, Adi. 2009. *Rekayasa Perangkat Lunak menggunakan UML dan Java*. Yogyakarta: Andi.
5. <http://ilmukomputer.org/2006/09/28/tips-tips-j2me/>, diakses pada tanggal 3 maret 2011.
6. http://id.wikipedia.org/wiki/Basis_data/, diakses pada tanggal 4 maret 2011.
7. <http://id.wikipedia.org/wiki/sharingdatadevice/>, diakses pada tanggal 9 maret 2011.
8. <http://id.wikipedia.org/wiki/kumpulan/modulmobile/Bloototh/Bluetooth/>,_diakses pada tanggal 9 maret 2011.
9. <http://id.wikipedia.org/wiki/remote/>, di akses pada tanggal 9 maret 2011.

Manual Program

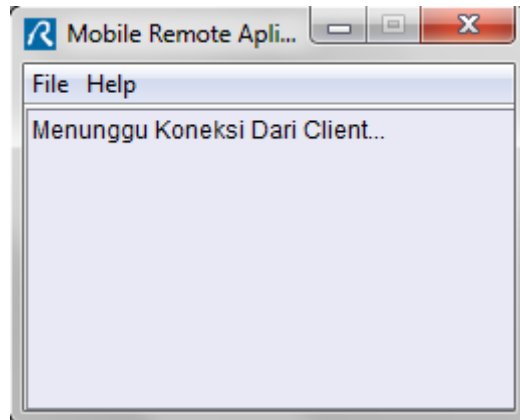
1. Instal dan aktifkan terlebih dahulu aplikasi pada *handphone* (*Client*) dan PC (*Server*) anda.
2. Aktifkan *bluetooth* pada *handphone* dan PC anda
3. Layar proses koneksi *client*



Ket :

1. Pada saat anda menjalankan aplikasi pada *client* ini maka anda harus mencari koneksi *bluetooth* aktif pada *server* terlebih dahulu.

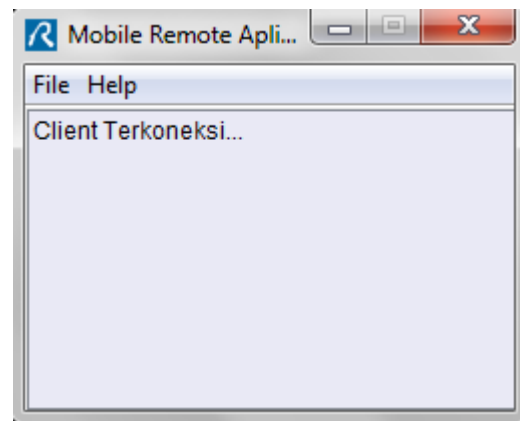
4. Layar koneksi *server*



Ket:

1. Pada *server* akan tampil layar koneksi *server* ini berarti koneksi *bluetooth* antara aplikasi client dengan *server* belum terkoneksi.

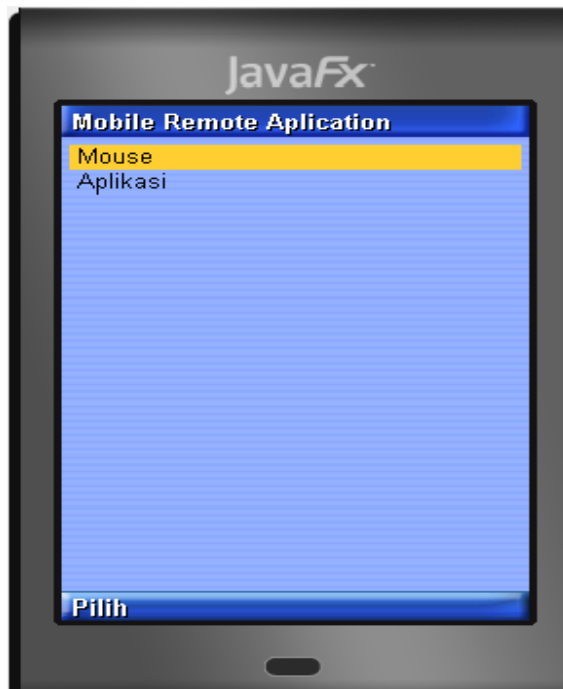
5. Layar Client Terkoneksi



Ket:

1. Jika tampil teks "Client Terkoneksi" berarti koneksi *bluetooth* antara aplikasi client dengan *server* sudah terkoneksi.
2. Setelah terkoneksi kemudian pada *client* muncul layar menu utama.

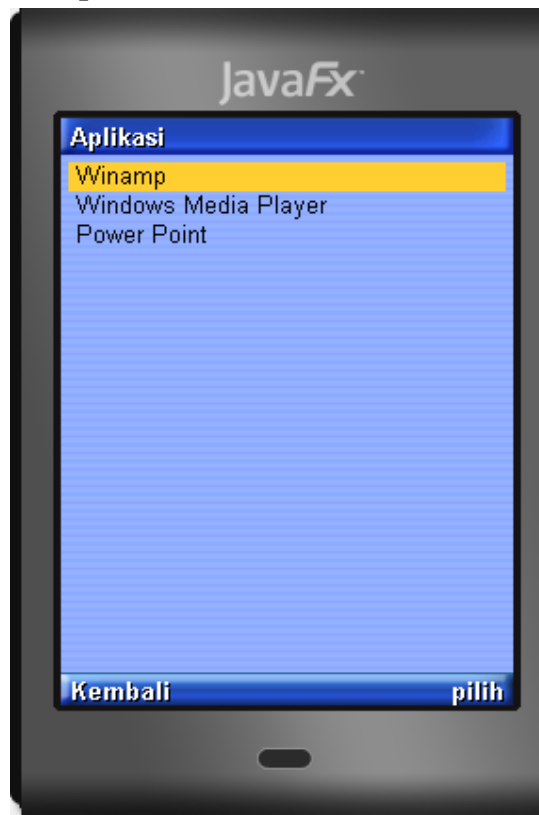
6. Layar Menu Utama



Ket:

1. Jika anda memilih *mouse* maka anda akan mengaktifkan navigasi kontrol mouse PC (*cursor*) menggunakan handphone seperti (*left, right, up, down* dan *ok*)
2. Jika anda memilih aplikasi maka akan muncul layar menu aplikasi, yang nantinya anda dapat memilih aplikasi yang ingin anda gunakan

7. Layar Menu Aplikasi



Ket :

Layar Input Nilai berguna untuk user memilih aplikasi yang ingin di gunakan.

1. Pilih aplikasi "Winamp"
2. Kemudian akan tampil layar navigasi kontrol aplikasi tersebut.

8. Layar Navigasi Kontrol Winamp

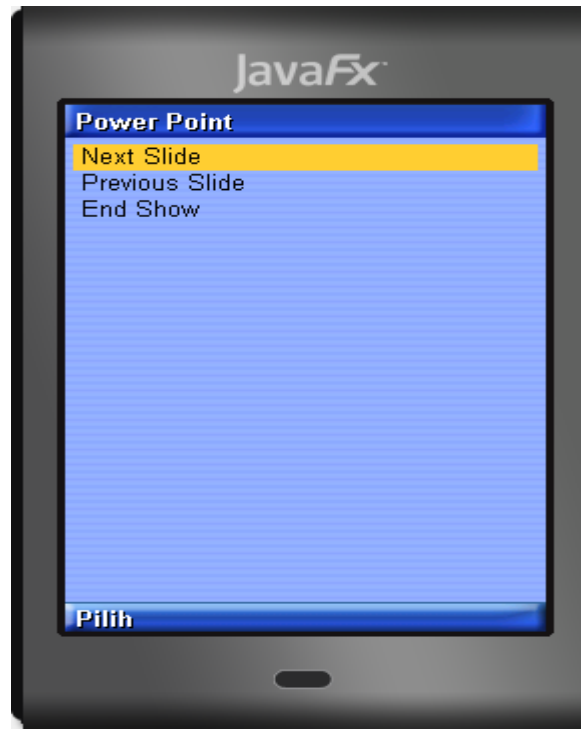


Ket:

Layar navigasi kontrol aplikasi ini berguna untuk mengontrol jalannya aplikasi.

1. Jika tekan *Play* : maka akan menjalankan *track* / lagu .
2. Jika tekan *Pause* : maka akan memberentikan *track* tanpa harus mengulanginya dari awal
3. Jika tekan *Stop* : maka akan memberhentikan lagu yang sedang di putar.
4. Jika tekan *Previous* : maka akan kembali mengulang ke *track* sebelumnya.
5. Jika tekan *Next* : maka melanjutkan ke *track* berikutnya.
6. Jika tekan *Volume Up* : maka akan menambah besarnya volume/suara.
7. Jika tekan *Volume Down* : maka akan menurunkan volume/suara.

9. Layar Navigasi Kontrol Power Point



Ket :

Layar Navigasi kontrol *Power Point* ini berfungsi untuk mengontrol jalannya aplikasi *Microsoft Power Point*.

1. Jika tekan *Next Slide* : maka akan melanjutkan *slide* berikutnya.
2. Jika tekan *Previous Slide* : maka akan mengulang kembali ke *slide* sebelumnya.
3. Jika tekan *Ed Show* : maka akan mengakhiri *slide* presentasi.

“SELAMAT MENCOBA”