

SKRIPSI

RANCANG BANGUN APLIKASI *SECURE MESSENGER* MENGUNAKAN *ECLIPSE GALILEO* BERBASIS ANDROID



Disusun Oleh

**ANDIKA SUBRATA
07.12.628**

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2013**

LEMBAR PERSETUJUAN

RANCANG BANGUN APLIKASI SECURE MESSENGER MENGUNAKAN ECLIPSE GALILEO BERBASIS ANDROID

SKRIPSI

*Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Teknik Komputer Dan Informatika Strata Satu (S-1)*

Disusun Oleh :

ANDIKA SUBRATA

NIM : 07.12.628

Diperiksa dan Disetujui

Mengetahui

Program Studi Teknik Elektro S-1



Ilham Ashari, ST, MT
NIP.P.1030100353

Dosen Pembimbing I

Dosen Pembimbing II

Dr. Eng. Aryanto Soetedjo, ST, MT
NIP.P. 1030800417

Michael Ardita, ST, MT
NIP.Y.1031000434

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2013**

RANCANG BANGUN APLIKASI SECURE MESSENGER MENGUNAKAN ECLIPSE GALILEO BERBASIS ANDROID

ANDIKA SUBRATA
NIM : 0712628

Dosen Pembimbing : 1. Dr. Eng. Aryunto Soetedjo, ST, MT
2. Michael Ardita, ST, MT

Jurusan Teknik Elektro S-1, Konsentrasi Teknik Komputer dan Informatika
Fakultas Teknologi Industri, Institut Teknologi Nasional Malang
Jl. Raya Karanglo Km 2 Malang

ABSTRAK

Perkembangan teknologi informasi semakin pesat, tidak terkecuali di Indonesia. Teknologi informasi khususnya chatting saat ini sudah menjadi hal umum bagi tua, muda, pria maupun wanita. Industri dan bisnis pengembangan chatting ini juga sudah menjadi suatu hal yang menjanjikan, terbukti dengan banyaknya perusahaan pengembang chatting di dunia.

Di balik semua itu, tersirat keinginan dari penulis untuk membuat chatting sendiri, tetapi masih saja terbentur dengan masalah ilmu, biaya dan tingkat kesulitan pembuatan chatting yang memang cukup tinggi. Untuk itu penulis berusaha mengembangkan pengamanan chatting sederhana dengan menggunakan suatu metode yang sudah terkenal dan teruji di kalangan komunitas pengembang chatting baik Indonesia maupun luar negeri, yaitu RSA. Penulis berusaha membuat dan menyajikan ilmu dengan bahasa yang sederhana dan mudah dicerna.

Adapun hasil yang telah di peroleh dari pembuatan skripsi ini, yaitu kita dapat membuat aplikasi chatingsendiri, sehingga pada suatu hari dapat melakukan pemembangan lagi.

Kata Kunci: *Handphone, Android, RSA, Aplikasi.*

KATA PENGANTAR

Puji syukur kehadirat Tuhan Yang Maha Esa, yang telah memberikan berkat-Nya, sehingga penulis dapat menyelesaikan laporan Skripsi ini dengan baik dan lancar.

Laporan Skripsi ini merupakan salah satu persyaratan akademik dalam menyelesaikan program Strata 1 Jurusan Teknik Elektro, Konsentrasi Komputer & Informatika, Institut Teknologi Nasional Malang. Adapun judul laporan Skripsi ini adalah:

RANCANG BANGUN APLIKASI *SECURE MESSENGER* MENGUNAKAN *ECLIPSE GALILEO* BERBASIS ANDROID

Selanjutnya pada kesempatan ini penulis juga menyampaikan rasa terimakasih yang sebesar-besarnya kepada pihak-pihak yang telah banyak membantu penulis selama penyusunan tugas akhir, diantaranya :

1. Bapak Ir. Yusuf Ismail Nahkoda, MT selaku Ketua Jurusan Teknik Elektro S-1 ITN Malang.
2. Bapak Dr. Eng. Aryuanto Soetedjo, ST, MT selaku Sekertaris Jurusan Teknik Elektro S-1 ITN Malang dan pengusul serta penyedia ruang Skripsi.
3. Bapak Dr. Eng. Aryuanto Soetedjo, ST, MT selaku Dosen Pembimbing I
4. Bapak Michael Ardita, ST, MT selaku Dosen Pembimbing II
5. Orang Tua, dan adik-adikku serta saudaraku yang telah memberikan dukungan untuk selalu berdoa, berusaha dan nasehat yang telah diberikan sampai saat ini.
6. Seluruh dosen dan pegawai ITN Kampus 2 Malang.
7. Semua teman-teman yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak bisa penulis sebutkan satu persatu.

Penulis berharap agar buku laporan Skripsi ini dapat memberikan banyak manfaat bagi semua pihak yang membutuhkan, khususnya bagi rekan-rekan mahasiswa. Penulis menyadari bahwa dalam penyusunan laporan ini masih banyak kekurangan, oleh karena itu mohon maaf apabila dalam buku ini terdapat hal-hal yang kurang berkenan dihati para pembaca.

Penulis juga mengharap koreksi, kritik serta saran-saran yang bermanfaat demi kesempurnaan buku Laporan Skripsi ini.

Malang, Maret 2013

Penulis

DAFTAR ISI

HALAMAN PERSETUJUAN	i
ABSTRAK	ii
KATA PENGANTAR	iii
DAFTAR ISI	v
DAFTAR GAMBAR	vii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan	2
1.5 Metode Penelitian	2
1.6 Sistematika Penulisan	3
BAB II LANDASAN TEORI	5
2.1 Android Operating System	5
2.1.1. Definsi dan Sejarah Android	5
2.1.2. Arsitektur Android	7
2.1.3. Keunggulan Android	12
2.1.4. <u>Komponen-Komponen Aplikasi Android</u>	12
2.1.5. Beberapa Versi Android	17
2.2 PHP	20
2.2.1. Definisi PHP	20
2.3 MySQL	20
2.3.1. Definisi MySQL	20
2.4 Eclipse	21
2.4.1. Definisi Eclipse	21
2.4.2. Arsitektur Eclipse	22
2.4.3. Versi Peluncuran	22
2.5 Wamp Server	24
2.6 RSA	25
2.7 Android SDK	26
BAB III ANALISA MASALAH DAN PERENCANAAN SISTEM	27
3.1 Perancangan Umum	27
3.2 Analisis Sistem	27

3.2.1 Deskripsi Sistem	27
3.2.2 Messenger Saat ini	28
3.2.3 Otoritas Akses Sistem	28
3.2.4 Analisa Kebutuhan Akses Sistem	28
3.3 Perancangan Aplikasi Secure Messenger	29
3.3.1 Data Flow Diagram	30
3.3.2 Diagram Konteks	30
3.3.3 DFD Level I	31
3.4 Desain Sistem	32
3.4.1 Spesifikasi Sistem	33
3.5 Flowchart	33
BAB IV IMPLEMENTASI DAN PENGUJIAN	34
4.1 Implementasi Sistem	34
4.2.1. Eclipse dan ADT	34
4.2.2. Android SDK	39
4.3 Tampilan Aplikasi	40
4.3.1. Tampilan Awal	40
4.3.2. Tampilan Registrasi	41
4.3.3. Tampilan Login	41
4.3.4. Tampilan Ruang Chat	42
4.3.5. Tampilan Room	44
4.4 Pengujian Hasil	44
4.4.1. Lingkungan Perangkat Keras (<i>Hardware</i>) Mobile	44
4.4.2. Pengujian Aplikasi Secure Messenger di Handphone	45
4.4.2.1. Pengujian Aplikasi	45
BAB V KESIMPULAN DAN SARAN	49
5.1 Kesimpulan	49
5.2 Saran	49
DAFTAR PUSTAKA	50
LAMPIRAN	51

DAFTAR GAMBAR

Gambar 2.1	Arsitektur Android	7
Gambar 2.2	Siklus Activity	15
Gambar 2.3	Siklus Services	15
Gambar 2.4	Jendela kerja Eclipse	24
Gambar 2.6	Halaman Utama Wamp	25
Gambar 2.7	SDK Manager.....	26
Gambar 3.1	Desain Umum Secure Messenger.....	29
Gambar 3.2	Diagram Konteks Secure Messenger.....	30
Gambar 3.3	DFD Level 1 Secure Messenger.....	32
Gambar 3.4	Diagram Block Aplikasi.....	32
Gambar 3.5	Flowchart.....	33
Gambar 4.1	Eclipse.exe.....	35
Gambar 4.2	Workspace Eclipse	35
Gambar 4.3	Tampilan Awal Eclipse	36
Gambar 4.4	Tampilan kerja Eclipse.....	36
Gambar 4.5	Tampilan Instal ADT.....	37
Gambar 4.6	Tampilan pop-up Plug-in ADT	37
Gambar 4.7	Tampilan pop-up Repository Plug-in Eclipse	38
Gambar 4.8	Tampilan Instalasi ADT Plug-in Eclipse.....	38
Gambar 4.9	Tampilan Proses Instalasi ADT Plug-in Eclipse	39
Gambar 4.10	Android SDK Manager	39
Gambar 4.11	Tampilan Lokasi SDK Android	40
Gambar 4.12	Tampilan Awal Aplikasi	40
Gambar 4.13	Menu Registrasi	41
Gambar 4.14	Tampilan Sourcecode Register.....	41
Gambar 4.15	Menu Login	42
Gambar 4.16	Tampilan Sourcecode Login	42
Gambar 4.17	Tampilan Ruang Chat.....	43
Gambar 4.18	Tampilan Sourcecode Chat.....	43
Gambar 4.19	Tampilan Room.....	44
Gambar 4.20	Tampilan Menu Awal Aplikasi	45
Gambar 4.21	Tampilan Registrasi	46
Gambar 4.22	Tampilan Login	46

Gambar 4.23 Tampilan Chat	47
Gambar 4.24 Tampilan RSA.....	48
Gambar 4.25 Tampilan Hasil RSA	49

BAB I

PENDAHULUAN

1.1 Latar Belakang

Android merupakan system operasi berbasis Linux untuk telepon seluler. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia.

Pada saat perilisan perdana Android, 5 November 2007, Android bersama Open Handset Alliance menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, Google merilis kode-kode Android di bawah lisensi apache, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler.

Messenger mengacu kepada segala bentuk komunikasi menggunakan Internet, tetapi secara spesifik mengacu kepada percakapan berbasis teks antara dua orang pengguna Internet. Percakapan di Internet dapat menggunakan perangkat lunak seperti pengirim pesan instan, Internet Relay Chat, dan lain-lain.

RSA di bidang kriptografi adalah sebuah algoritma pada enkripsi public key. RSA merupakan algoritma pertama yang cocok untuk digital signature seperti halnya ekripsi, dan salah satu yang paling maju dalam bidang kriptografi public key. RSA masih digunakan secara luas dalam protokol electronic commerce, dan dipercaya dalam mengamankan dengan menggunakan kunci yang cukup panjang.

1.2. Rumusan Masalah

Berdasarkan uraian dari latar belakang di atas maka terdapat beberapa rumusan masalah sebagai berikut :

1. Bagaimana penjelasan tersebut dapat digunakan sebagai media messenger ?
2. Bagaimana merancang dan membuat enkripsi ?

1.3. Tujuan

Berdasarkan rumusan masalah yang telah dikemukakan, tujuan utama yang diharapkan adalah:

- 1) Untuk mempermudah dalam melakukan komunikasi melalui messenger.
- 2) Membantu melindungi pesan yang bersifat privasi didalam messenger dalam bentuk enkripsi .

1.4. Batasan Masalah

Berkaitan dengan indentifikasi masalah diatas, maka diberikan beberapa batasan masalah yaitu:

1. Bagaimana membuat aplikasi chatting pada Android.
2. Bagaimana membuat pengaman pada aplikasi tersebut.
3. Bagaimana cara mengabungkan antara aplikasi chatting dan pengaman tersebut.
4. Bagaimana cara kerja aplikasi tersebut pada handphone Android

1.5. Metodologi Penelitian

Adapun metode penelitian yang digunakan adalah sebagai berikut:

1. Studi literatur

Pengumpulan data yang dilakukan dengan mencari bahan-bahan kepustakaan dan referensi dari berbagai sumber sebagai landasan teori yang ada hubungannya dengan permasalahan yang dijadikan objek penelitian.

2. Analisa Kebutuhan Sistem

Data dan informasi yang telah diperoleh akan dianalisa agar didapatkan kerangka global yang bertujuan untuk mendefinisikan kebutuhan sistem di mana nantinya akan digunakan sebagai acuan perancangan sistem.

3. Perancangan dan pembuatan

Berdasarkan data dan informasi yang telah diperoleh serta analisa kebutuhan untuk membangun sistem ini, akan dibuat rancangan kerangka global yang menggambarkan mekanisme dari sistem yang akan dibuat dan diimplementasikan kedalam sistem.

4. Pengujian dan Evaluasi

Pada tahap ini, sistem yang telah selesai dibuat akan diuji coba, yaitu pengujian berdasarkan fungsionalitas program, dan akan dilakukan koreksi dan penyempurnaan program jika diperlukan.

1.6. Sistematika Penulisan

Untuk mempermudah dan memahami pembahasan penulisan skripsi ini, maka sistematika penulisan disusun sebagai berikut :

- Bab I : Pendahuluan**
Berisi Latar Belakang, Rumusan Masalah, Tujuan Penelitian, Pembatasan Permasalahan, Metode Penelitian dan Sistematika Penulisan.
- Bab II : Tinjauan Pustaka**
Berisi tentang landasan teori mengenai permasalahan yang berhubungan dengan penelitian yang dilakukan.
- Bab III : Perancangan dan Analisa Sistem**
Dalam bab ini berisi mengenai analisa kebutuhan sistem baik software maupun hardware yang diperlukan untuk membuat kerangka global yang menggambarkan mekanisme dari sistem yang akan dibuat.
- Bab IV : Pembuatan dan Pengujian Sistem**
Berisi tentang pembuatan dari perancangan sistem yang telah dibuat serta pengujian terhadap sistem tersebut.
-

Bab V : Penutup

Merupakan bab terakhir yang memuat intisari dari hasil pembahasan yang berisikan kesimpulan dan saran yang dapat digunakan sebagai pertimbangan untuk pengembangan penulisan selanjutnya

BAB II

TINJAUAN PUSTAKA

2.1 Android Operating System

2.1.1 Definisi dan Sejarah Android

Android adalah sistem operasi yang berbasis Linux untuk telepon seluler seperti telepon pintar dan komputer tablet. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia.

Pada saat perilisannya perdana Android, 5 November 2007, Android bersama Open Handset Alliance menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, Google merilis kode-kode Android di bawah lisensi Apache, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler.

Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau Google Mobile Services (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung Google atau dikenal sebagai Open Handset Distribution (OHD).

Pada Juli 2005, Google bekerjasama dengan Android Inc., perusahaan yang berada di Palo Alto, California Amerika Serikat. Para pendiri Android Inc. bekerja pada Google, di antaranya Andy Rubin, Rich Miner, Nick Sears, dan Chris White. Saat itu banyak yang menganggap fungsi Android Inc. hanyalah sebagai perangkat lunak pada telepon seluler. Sejak saat itu muncul rumor bahwa Google hendak memasuki pasar telepon seluler. Di perusahaan Google, tim yang dipimpin Rubin bertugas mengembangkan program perangkat seluler yang didukung oleh kernel Linux. Hal ini menunjukkan indikasi bahwa Google sedang bersiap menghadapi persaingan dalam pasar telepon seluler.

Sekitar September 2007 sebuah studi melaporkan bahwa Google mengajukan hak paten aplikasi telepon seluler (akhirnya Google mengenalkan Nexus One, salah satu jenis telepon pintar GSM yang menggunakan Android pada sistem operasinya. Telepon seluler ini diproduksi oleh HTC Corporation dan tersedia di pasaran pada 5 Januari 2010).

Pada 9 Desember 2008, diumumkan anggota baru yang bergabung dalam program kerja Android ARM Holdings, Atheros Communications, diproduksi oleh Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, dan Vodafone Group Plc. Seiring pembentukan Open Handset Alliance, OHA mengumumkan produk perdana mereka, Android, perangkat bergerak (*mobile*) yang merupakan modifikasi kernel Linux 2.6. Sejak Android dirilis telah dilakukan berbagai pembaruan berupa perbaikan bug dan penambahan fitur baru.

Telepon pertama yang memakai sistem operasi Android adalah HTC Dream, yang dirilis pada 22 Oktober 2008. Pada penghujung tahun 2009 diperkirakan di dunia ini paling sedikit terdapat 18 jenis telepon seluler yang menggunakan Android.

Para penggemar open source kemudian membangun komunitas yang membangun dan berbagi Android berbasis firmware dengan sejumlah penyesuaian dan fitur-fitur tambahan, seperti FLAC lossless audio dan kemampuan untuk menyimpan download aplikasi pada microSD card. Mereka sering memperbaharui paket-paket firmware dan menggabungkan elemen-elemen fungsi Android yang belum resmi diluncurkan dalam suatu carrier-sanction firmware.

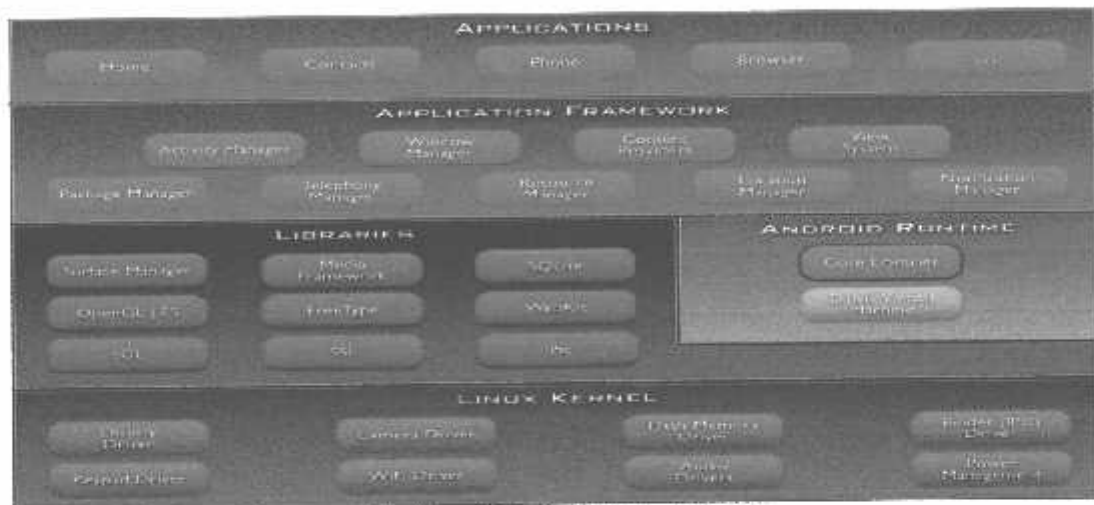
Android dipuji sebagai "platform mobile pertama yang lengkap, terbuka dan bebas". Android merupakan generasi baru platform mobile yang memberikan pengembangan untuk melakukan pengembangan sesuai dengan keinginan. Sistem operasi yang mendasari android dilisensikan dibawah GNU, *General Public Lisensi versi 2 (GPLv2)*, yang sering dikenal dengan istilah "copyleft" lisensi.

Salah satu kunci dari android adalah *Dalvik Virtual Machine (DVM)*. Android berjalan di dalam *Dalvik Virtual Machine (DVM)* bukan di *Java Virtual Machine (JVM)*. *Dalvik Virtual Machine (DVM)* adalah "register bases" sementara *Java Virtual Machine (JVM)* adalah "stack based". DVM menggunakan karel linux untuk

menangani fungsionalitas tingkat rendah termasuk keamanan, threading, dan proses serta manajemen memori.

2.1.2 Arsitektur Android

Google mengibaratkan Android sebagai sebuah tumpukan software. Setiap lapisan dari tumpukan ini menghimpun beberapa program yang mendukung fungsi-fungsi spesifik dari sistem operasi. Berikut ini susunan dari lapisan – lapisan tersebut jika di lihat dari lapisan dasar hingga lapisan teratas:



Gambar 2.1 Arsitektur Android

1. Linux Kernel

Tumpukan paling bawah pada arsitektur Android ini adalah kernel. Google menggunakan kernel Linux versi 2.6 untuk membangun sistem Android, yang mencakup memory management, security setting, power management, dan beberapa driver hardware. Kernel berperan sebagai abstraction layer antara hardware dan keseluruhan software. Sebagai contoh, HTC G1 dilengkapi dengan kamera. Kernel Android terdapat driver kamera yang memungkinkan pengguna mengirimkan perintah kepada hardware kamera.

2. *Android Runtime*

Lapisan setelah Kernel Linux adalah Android Runtime. Android Runtime ini berisi *Core Libraries* dan *Dalvik Virtual Machine*. *Core Libraries* mencakup serangkaian inti library Java, artinya Android menyertakan satu set library-library dasar yang menyediakan sebagian besar fungsi-fungsi yang ada pada library-library dasar bahasa pemrograman Java.

Dalvik adalah *Java Virtual Machine* yang memberi kekuatan pada sistem Android. *Dalvik VM* ini di optimalkan untuk telepon seluler. Setiap aplikasi yang berjalan pada Android berjalan pada processnya sendiri, dengan instance dari *Dalvik Virtual Machine*. *Dalvik* telah dibuat sehingga sebuah piranti yang memakainya dapat menjalankan multi *Virtual Machine* dengan efisien.

Dalvik VM dapat mengeksekusi file dengan format *Dalvik Executable (.dex)* yang telah dioptimasi untuk menggunakan minimal memory footprint. *Virtual Machine* ini register-based, dan menjalankan class-class yang dicompile menggunakan compiler Java yang kemudian ditransformasi menjadi format *.dex* menggunakan "dx" tool yang telah disertakan. *Dalvik Virtual Machine (VM)* menggunakan kernel Linux untuk menjalankan fungsi-fungsi seperti threading dan low-level memory management.

3. *Libraries*

Bertempat di level yang sama dengan *Android Runtime* adalah *Libraries*. Android menyertakan satu set library-library dalam bahasa C/C++ yang digunakan oleh berbagai komponen yang ada pada sistem Android. Kemampuan ini dapat diakses oleh programmer melewati *Android application framework*. Sebagai contoh Android mendukung pemutaran format audio, video, dan gambar. Berikut ini beberapa core library tersebut :

- o *System C library*
diturunkan dari implementasi standard C system library (libc) milik BSD, dioptimasi untuk piranti embedded berbasis Linux

- *Media Libraries*
berdasarkan PacketVideo's OpenCORE; library-library ini mendukung playback dan recording dari berbagai format audio and video populer, meliputi MPEG4, H.264, MP3, AAC, AMR, JPG, and PNG
 - *Surface Manager*
mengatur akses pada display dan lapisan composites 2D and 3D graphic dari berbagai aplikasi
 - *LibWebCore*
web browser engine modern yang mensupport Android browser maupun embeddable web view
 - *SGL*
the underlying 2D graphics engine
 - *3D libraries*
implementasi berdasarkan OpenGL ES 1.0 APIs; library ini menggunakan hardware 3D acceleration dan highly optimized 3D software rasterizer
 - *FreeType*
bitmap dan vector font rendering
 - *SQLite*
relational database engine yang powerful dan ringan tersedia untuk semua aplikasi
-

4. Application Framework

Lapisan selanjutnya adalah application framework, yang mencakup program untuk mengatur fungsi-fungsi dasar smartphone. Application Framework merupakan serangkaian tool dasar seperti alokasi resource smartphone, aplikasi telepon, pergantian antar - proses atau program, dan pelacakan lokasi fisik telepon. Para pengembang aplikasi memiliki aplikasi penuh kepada tool-tool dasar tersebut, dan memanfaatkannya untuk menciptakan aplikasi yang lebih kompleks.

Programmer mendapatkan akses penuh untuk memanfaatkan API-API (Android Protocol Interface) yang juga digunakan core applications. Arsitektur aplikasi didesain untuk menyederhanakan pemakaian kembali komponen-komponen, setiap aplikasi dapat menunjukkan kemampuannya dan aplikasi lain dapat memakai kemampuan tersebut. Mekanisme yang sama memungkinkan pengguna mengganti komponen-komponen yang dikehendaki. Di dalam semua aplikasi terdapat servis dan sistem yang meliputi :

- Satu set Views yang dapat digunakan untuk membangun aplikasi meliputi lists, grids, text boxes, buttons, dan embeddable web browser

 - Content Providers yang memungkinkan aplikasi untuk mengakses data dari aplikasi lain (misalnya Contacts), atau untuk membagi data yang dimilikinya.

 - Resource Manager, menyediakan akses ke non-code resources misalnya localized strings, graphics, dan layout files

 - Notification Manager yang memungkinkan semua aplikasi untuk menampilkan custom alerts pada the status bar
-

- o Activity Manager yang manage life cycle of dari aplikasi dan menyediakan common navigation backstack

5. Application

Di lapisan teratas bercokol aplikasi itu sendiri. Di lapisan inilah anda menemukan fungsi-fungsi dasar smartphone seperti menelepon dan mengirim pesan singkat, menjalankan web browser, mengakses daftar kontak, dan lain-lain. Bagi rata-rata pengguna, lapisan inilah yang paling sering mereka akses. Mereka mengakses fungsi-fungsi dasar tersebut melalui user interface.

Google menjabarkan aplikasi Android ke dalam empat blok bangunan dasar (tidak semua aplikasi mempunyai keempatnya) yaitu :

- o *Activities*
yaitu ketika sebuah aplikasi memunculkan screen di layar. Sebagai contoh, sebuah aplikasi GPS mempunyai screen peta dasar, screen rencana perjalanan, dan screen rute di atasnya. Ketiga penampakan screen ini disebut activities.
 - o *Intents*
yaitu mekanisme perpindahan dari suatu activity ke activity lainnya. Sebagai contoh ketika merencanakan perjalanan pada aplikasi GPS, intent akan menginterpretasi input dan mengaktifkan screen rute di atas screen peta.
 - o *Services*
yaitu serupa dengan service di PC dan server, program yang berjalan di belakang layar, tanpa interferensi dari pengguna.
-

- o *Content Provider*
yaitu mekanisme yang memungkinkan sebuah aplikasi berbagi informasi dengan aplikasi lainnya.

2.1.3 Keunggulan Android

Beberapa keunggulan Android adalah :

- o *Keterbukaan*
Android menyediakan akses ke fungsi dasar perangkat mobile menggunakan standar panggilan ke API.
- o *Penghancuran perbatasan*
Anda dapat menggabungkan informasi dari Internet ke dalam telepon, seperti informasi kontak, atau data pada lokasi geografis untuk mendapatkan kesempatan baru
- o *Kesamaan aplikasi*
untuk Android ada perbedaan antara telepon utama aplikasi dan perangkat lunak lain. anda bahkan dapat mengubah program untuk memutar nomor, atau screen saver.
- o *Cepat dan mudah perkembangan*
dalam SDK memiliki semua yang anda butuhkan untuk membuat dan menjalankan aplikasi Android, termasuk simulator ini instrumen, dan alat debugging maju.

2.1.4 Komponen – Komponen Aplikasi Android

Fitur penting android adalah bahwa satu aplikasi dapat menggunakan elemen dari aplikasi lain (untuk aplikasi yang memungkinkan). Sebagai contoh, sebuah aplikasi memerlukan fitur scroller dan aplikasi lain telah mengembangkan fitur scroller yang baik dan memungkinkan aplikasi lain menggunakannya. Maka pengembang tidak perlu lagi mengembangkan hal serupa untuk aplikasinya, cukup menggunakan *scroller* yang telah ada. Agar fitur tersebut dapat bekerja, sistem harus dapat menjalankan

aplikasi ketika setiap bagian aplikasi itu dibutuhkan, dan pemanggilan objek java untuk bagian itu. Oleh karenanya android berbeda dari sistem-sistem lain, Android tidak memiliki satu tampilan utama program seperti fungsi main() pada aplikasi lain. Sebaliknya, aplikasi memiliki komponen penting yang memungkinkan sistem untuk memanggil dan menjalankan ketika dibutuhkan. Komponen tersebut yaitu:

1. *Activity*

Activity merupakan bagian yang paling penting dalam sebuah aplikasi, karena Activity menyajikan tampilan visual program yang sedang digunakan oleh pengguna. Setiap Activity dideklarasikan dalam sebuah kelas yang bertugas untuk menampilkan antarmuka pengguna yang terdiri dari Views dan respon terhadap Event. Setiap aplikasi memiliki sebuah activity atau lebih.

Perpindahan antara activity dengan activity lainnya diatur melalui sistem, dengan memanfaatkan activity stack. Keadaan suatu activity ditentukan oleh posisinya dalam tumpukan activity, LIFO (Last In First Out) dari semua aplikasi yang sedang berjalan. Bila suatu activity baru dimulai, activity yang sebelumnya digunakan maka akan dipindahkan ketumpukan paling atas. Jika pengguna ingin menggunakan activity sebelumnya, cukup menekan tombol Back, atau menutup activity yang sedang digunakan, maka activity yang berada diatas akan aktif kembali. Memory Manager android menggunakan tumpukan ini untuk menentukan prioritas aplikasi berdasarkan activity, memutuskan untuk mengakhiri suatu aplikasi dan mengambil sumber daya dari aplikasi tersebut.

Ketika activity diambil dan disimpan dalam tumpukan activity terdapat 4 kemungkinan kondisi transisi yang akan terjadi, yaitu:

(a) *Active*

Setiap activity yang berada ditumpukan paling atas, maka dia akan terlihat, terfokus, dan menerima masukan dari pengguna. Android akan berusaha untuk membuat activity aplikasi ini untuk tetap hidup dengan segala cara, bahkan akan menghentikan activity yang berada dibawah tumpukkannya jika diperlukan. Ketika activity sedang aktif, maka yang lainnya akan dihentikan sementara.

(b) *Paused*

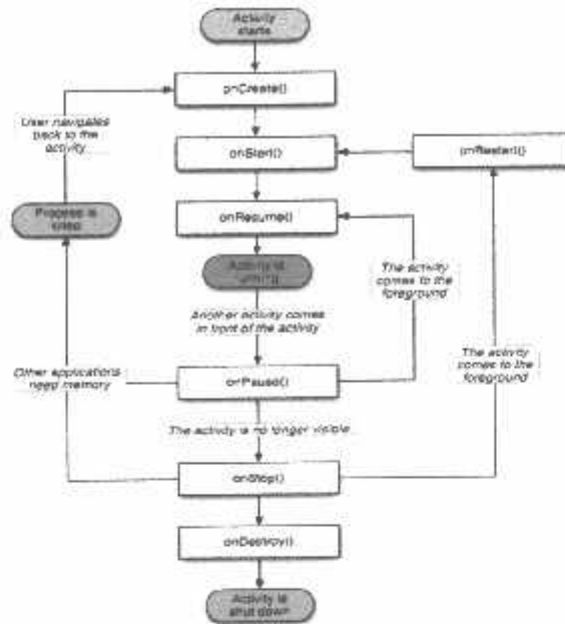
Dalam beberapa kasus activity akan terlihat tapi tidak terfokus pada kondisi inilah disebut paused. Keadaan ini terjadi jika activity transparan dan tidak fullscreen pada layar. Ketika activity dalam keadaan paused, dia terlihat active namun tidak dapat menerima masukan dari pengguna. Dalam kasus ekstrim, android akan menghentikan activity dalam keadaan paused ini, untuk menunjang sumber daya bagi activity yang sedang aktif.

(c) *Stopped*

Ketika sebuah activity tidak terlihat, maka itulah yang disebut stopped. Activity akan tetap berada dalam memori dengan semua keadaan dan informasi yang ada. Namun akan menjadi kandidat utama untuk dieksekusi oleh sistem ketika membutuhkan sumberdaya lebih. Oleh karenanya ketika suatu activity dalam kondisi stopped maka perlu disimpan data dan kondisi antarmuka saat itu. Karena ketika activity telah keluar atau ditutup, maka dia akan menjadi inactive.

(d) *Inactive*

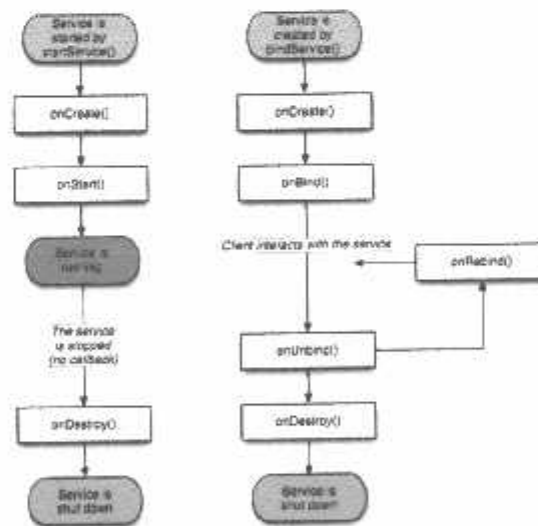
Kondisi ketika activity telah dihentikan dan sebelum dijalankan. Inactive activity telah ditiadakan dari tumpukan activity sehingga perlu restart ulang agar dapat tampil dan digunakan kembali.



Gambar 2.1.1 Siklus Activity

2. Service

Suatu service tidak memiliki tampilan antarmuka, melainkan berjalan di background untuk waktu yang tidak terbatas. Komponen service diproses tidak terlihat, memperbarui sumber data dan menampilkan notifikasi. Service digunakan untuk melakukan pengolahan data yang perlu terus diproses, bahkan ketika Activity tidak aktif atau tidak tampak.



Gambar 2.1.2 Siklus Service

3. *Intent*

Android menggunakan class *Intent* untuk melakukan aksi, berubah dari satu layar ke layar yang lain. Dua hal utama yang perlu diperhatikan pada struktur class *Intent* adalah aksi yang dilakukan dan data yang dibawa.

Sebuah class *intent* tidak dapat melakukan sendiri, namun membutuhkan sebuah *intent receiver* dan *intent filter*. *Intent receiver* akan menerima semua *activity* yang diberikan kemudian oleh *intent filter* akan dipilih aksi mana yang akan dijalankan.

4. *Broadcast Receiver*

Broadcast Receivers merupakan komponen yang sebenarnya tidak melakukan apa-apa kecuali menerima dan bereaksi menyampaikan pemberitahuan. Sebagian besar *Broadcast* berasal dari sistem misalnya, Baterai sudah hampir habis, informasi zona waktu telah berubah, atau pengguna telah merubah bahasa default pada perangkat. Sama halnya dengan *service*, *Broadcast Receivers* tidak menampilkan antarmuka pengguna. Namun, *Broadcast Receivers* dapat menggunakan *Notification Manager* untuk memberitahukan sesuatu kepada pengguna.

5. *Content Provider*

Content Providers digunakan untuk mengelola dan berbagi database. Data dapat disimpan dalam file sistem, dalam database *SQLite*, atau dengan cara lain yang pada prinsipnya sama. Dengan adanya *Content Provider* memungkinkan antar aplikasi untuk saling berbagi data. Komponen ini sangat berguna ketika sebuah aplikasi membutuhkan data dari aplikasi lain, sehingga mudah dalam penerapannya.

2.1.5 Beberapa Versi Android

Android versi 1.1

Pada 9 Maret 2009, Google merilis Android versi 1.1. Android versi ini dilengkapi dengan pembaruan estetis pada aplikasi, jam alarm, voice search (pencarian suara), pengiriman pesan dengan Gmail, dan pemberitahuan email

Android versi 1.5 (Cupcake)

Pada pertengahan Mei 2009, Google kembali merilis telepon seluler dengan menggunakan Android dan SDK (Software Development Kit) dengan versi 1.5 (Cupcake). Terdapat beberapa pembaruan termasuk juga penambahan beberapa fitur dalam seluler versi ini yakni kemampuan merekam dan menonton video dengan modus kamera, mengunggah video ke Youtube dan gambar ke Picasa langsung dari telepon, dukungan Bluetooth A2DP, kemampuan terhubung secara otomatis ke headset Bluetooth, animasi layar, dan keyboard pada layar yang dapat disesuaikan dengan sistem.

Android versi 1.6 (Donut)

Donut (versi 1.6) dirilis pada September dengan menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus; kamera, camcorder dan galeri yang diintegrasikan; CDMA / EVDO, 802.1x, VPN, Gestures, dan Text-to-speech engine; kemampuan dial kontak; teknologi text to change speech (tidak tersedia pada semua ponsel; pengadaan resolusi VWGA.

Android versi 2.0/2.1 (Eclair)

ada 3 Desember 2009 kembali diluncurkan ponsel Android dengan versi 2.0/2.1 (Eclair), perubahan yang dilakukan adalah pengoptimalan hardware, peningkatan Google Maps 3.1.2, perubahan UI dengan browser baru dan dukungan HTML5, daftar

kontak yang baru, dukungan flash untuk kamera 3,2 MP, digital Zoom, dan Bluetooth 2.1.

Untuk bergerak cepat dalam persaingan perangkat generasi berikut, Google melakukan investasi dengan mengadakan kompetisi aplikasi mobile terbaik (killer apps – aplikasi unggulan). Kompetisi ini berhadiah \$25,000 bagi setiap pengembang aplikasi terpilih. Kompetisi diadakan selama dua tahap yang tiap tahapnya dipilih 50 aplikasi terbaik.

Dengan semakin berkembangnya dan semakin bertambahnya jumlah handset Android, semakin banyak pihak ketiga yang berminat untuk menyalurkan aplikasi mereka kepada sistem operasi Android. Aplikasi terkenal yang diubah ke dalam sistem operasi Android adalah Shazam, Backgrounds, dan WeatherBug. Sistem operasi Android dalam situs Internet juga dianggap penting untuk menciptakan aplikasi Android asli, contohnya oleh MySpace dan Facebook.

Android versi 2.2 (Froyo: Frozen Yoghurt)

Pada 20 Mei 2010, Android versi 2.2 (Froyo) diluncurkan. Perubahan-perubahan umumnya terhadap versi-versi sebelumnya antara lain dukungan Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, integrasi V8 JavaScript engine yang dipakai Google Chrome yang mempercepat kemampuan rendering pada browser, pemasangan aplikasi dalam SD Card, kemampuan WiFi Hotspot portabel, dan kemampuan auto update dalam aplikasi Android Market.

Android versi 2.3 (Gingerbread)

Pada 6 Desember 2010, Android versi 2.3 (Gingerbread) diluncurkan. Perubahan-perubahan umum yang didapat dari Android versi ini antara lain peningkatan kemampuan permainan (gaming), peningkatan fungsi copy paste, layar antar muka (User Interface) didesain ulang, dukungan format video VP8 dan WebM, efek audio baru (reverb, equalization, headphone virtualization, dan bass boost), dukungan kemampuan Near Field Communication (NFC), dan dukungan jumlah kamera yang lebih dari satu.

Android versi 3.0 (Honeycomb)

Android 3.0 adalah versi yg secara khusus dioptimalkan untuk gadget dengan layar lebar, khususnya, dan yaitu tablet. Dalam versi ini diperkenalkan desain UI yang baru, virtual dan holografis tapi juga elegan dengan model interaksi memfokuskan pada konten.

Dalam versi ini Google membuat multitasking, notifications, Home screen customization, widgets, dan yang lainnya juga menjadi lebih interaktif, vibrant, dan memberikan pengalaman 3D tapi tetap familiar dan juga lebih baik dari sebelumnya

Perubahan-perubahan yang dilakukan di versi ini antara lain pada System Bar (di bagian global status dan notifikasi), Action Bar (di bagian application control), customizable Home screens, recent Apps (untuk multitasking yang lebih mudah), keyboard (di desain ulang agar mengetik bisa lebih cepat dan akurat), text selection (copy dan paste yang lebih baik lagi), opsi connectivity yang baru, browser, camera dan gallery, contact, dan email. Semuanya itu ditujukan agar lebih maksimal di dalam tablet. dan memang versi ini sampai saat versi 4.0 resmi diumumkan hanya beredar dan dipakai di tablet.

Android versi 4.0 (Ice Cream Sandwich)

Ice Cream Sandwich didesain untuk baik itu telepon ataupun tablet. Android ICS menawarkan banyak peningkatan dari apa yg sudah ada di Gingerbread dan Honeycomb dengan pada saat yang sama memberikan inovasi-inovasi baru.

Beberapa peningkatan itu antara lain kemampuan copy paste yang lebih baik, data logging dan warnings, dan kemampuan utk mengambilscreenshot dengan menekan power dan volume bersamaan. Selain itu keyboardnya dan kamus juga mendapat perbaikan. *Inovasi-inovasi baru di ICS* antara lain penggunaan fony "Roboto."

Di Android 4.0 Ice Cream Sandwich System Bar dan Action Bar, adanya Android 4.0 Ice Cream Sandwich voice control yang memungkinkan kita mendikte teks yang ingin kita ketik. Selain itu Face Unlock merupakan salah satu hal yang menonjol di Android versi baru ini. Juga ada NFC based app yang disebut Android Bump, yang memungkinkan kita untuk bertukar informasi/data hanya dengan menyentuh gadget kita.

2.2. PHP

2.2.1 Definisi PHP

PHP adalah bahasa pemrograman *script* yang paling banyak dipakai saat ini atau dalam kata lain bisa diartikan sebuah bahasa pemrograman web yang bekerja di sisi server (*server side scripting*) yang dapat melakukan konektivitas pada database yang di mana hal itu tidak dapat dilakukan hanya dengan menggunakan sintaks-sintaks *HTML* biasa. *PHP* banyak dipakai untuk memrogram situs web dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain.

2.3. MySQL

2.3.1 Definisi MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL. MySQL adalah Relational Database Management System (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat closed source atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (Structured Query Language).

SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem database (DBMS) dapat diketahui dari cara kerja optimizer-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya. Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan database server lainnya dalam query data. Hal ini terbukti untuk query yang dilakukan oleh single user, kecepatan query MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan Interbase.

2.4 Eclipse

2.4.1 Definisi Eclipse

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (*platform-independent*). Berikut ini adalah sifat dari Eclipse:

- **Multi-platform:** Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
- **Mult-language:** Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.
- **Multi-role:** Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari Eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in*.

Eclipse awalnya dikembangkan oleh IBM untuk menggantikan perangkat lunak IBM Visual Age for Java 4.0. Produk ini diluncurkan oleh IBM pada tanggal 5 November 2001, yang menginvestasikan sebanyak US\$ 40 juta untuk pengembangannya. Semenjak itu konsorsium Eclipse Foundation mengambil alih untuk pengembangan Eclipse lebih lanjut dan pengaturan organisasinya.

2.4.2 Arsitektur Eclipse

Sejak versi 3.0, Eclipse pada dasarnya merupakan sebuah *kernel*, yang mengangkat *plug-in*. Apa yang dapat digunakan di dalam Eclipse sebenarnya adalah fungsi dari *plug-in* yang sudah diinstal. Ini merupakan basis dari Eclipse yang dinamakan *Rich Client Platform* (RCP). Berikut ini adalah komponen yang membentuk RCP:

- *Core platform*
- OSGi
- SWT (*Standard Widget Toolkit*)
- JFace
- *Eclipse Workbench*

Secara standar Eclipse selalu dilengkapi dengan JDT (*Java Development Tools*), *plug-in* yang membuat Eclipse kompatibel untuk mengembangkan program Java, dan PDE (*Plug-in Development Environment*) untuk mengembangkan *plug-in* baru. Eclipse beserta *plug-in*-nya diimplementasikan dalam bahasa pemrograman Java.

Konsep Eclipse adalah IDE yang terbuka (*open*), mudah diperluas (*extensible*) untuk apa saja, dan tidak untuk sesuatu yang spesifik. Jadi, Eclipse tidak saja untuk mengembangkan program Java, akan tetapi dapat digunakan untuk berbagai macam keperluan, cukup dengan menginstal *plug-in* yang dibutuhkan. Apabila ingin mengembangkan program C/C++ terdapat *plug-in* CDT (*C/C++ Development Tools*). Selain itu, pengembangan secara visual bukan hal yang tidak mungkin oleh Eclipse, *plug-in* UML2 tersedia untuk membuat diagram UML. Dengan menggunakan PDE setiap orang bisa membuat *plug-in* sesuai dengan keinginannya. Salah satu situs yang menawarkan *plug-in* secara gratis seperti *Eclipse downloads by project*

2.4.3 Versi Peluncuran

Sejak tahun 2006, Eclipse Foundation mengkoordinasikan peluncuran Eclipse secara rutin dan simultan yang dikenal dengan nama *Simultaneous Release*. Setiap versi peluncuran terdiri dari Eclipse Platform dan juga sejumlah proyek yang terlibat dalam proyek Eclipse. Tujuan dari sistem ini adalah untuk menyediakan distribusi

Eclipse dengan fitur-fitur dan versi yang terstandarisasi. Hal ini juga dimaksudkan untuk mempermudah deployment dan maintenance untuk sistem enterprise, serta untuk kenyamanan. Peluncuran simultan dijadwalkan pada bulan Juni setiap tahunnya.

Kode Peluncuran	Tanggal Peluncuran	Platform	Nama Proyek
Eclipse 3.0	28 Juni 2004	3.0	
Eclipse 3.1	28 Juni 2005	3.1	
Callisto	30 Juni 2006	3.2	Callisto projects ^{id}
Europa	29 Juni 2007	3.3	Europe projects ^{id}
Ganymede	25 Juni 2008	3.4	Ganymede projects ^{id}
Galileo	24 Juni 2009	3.5	Galileo projects ^{id}
Helios	23 Juni 2010	3.6	

Hadirnya Eclipse di tengah-tengah persaingan IDE, terutama Java IDE, menimbulkan banyak kontroversi. Salah satunya adalah penggunaan *user interface* Eclipse. Sun Microsystems sebagai perusahaan yang membuat Java, selama ini mengeluarkan dua *library* untuk pengembangan *Graphical User Interface* (GUI), yaitu *Abstract Windowing Toolkit* (AWT) dan *Swing*. Akan tetapi dalam pembuatan Eclipse, dua *library* ini ditinggalkan karena berbagai alasan, yang salah satunya adalah kedua *library* ini tidak memberikan *look and feel* yang baik. Sehingga untuk hal ini dikembangkanlah *GUI-library* yang baru, SWT. Selain itu, Sun yang mempunyai IDE sendiri untuk pengembangan Java bernama NetBeans, menjadi gerah atas kehadiran Eclipse yang sampai saat ini komunitasnya cukup besar. Hal ini terlihat dengan menolaknya Sun untuk bekerja sama dalam Eclipse Foundation, walau telah diundang. Nama *Eclipse* seolah-olah memberi pesan eksplisit: cahaya *Sun* (matahari) yang pancarannya dihalangi oleh *Eclipse* (gerhana).



Gambar 2.4. *Jendela kerja Eclipse*

2.5 WAMPP

Wamp adalah sebuah aplikasi yang dapat menjadikan komputer kita menjadi sebuah server. Kegunaan wamp server ini untuk membuat jaringan local sendiri dalam artian kita dapat membuat website secara offline untuk masa coba-coba di komputer sendiri. Jadi fungsi dari wamp server itu sendiri merupakan server website kita untuk cara memakainya. Mengapa harus menjadi server ? Karena dalam hal ini komputer yang akan kita pakai harus memberikan pelayanan untuk pengaksesan web, untuk itu komputer kita harus menjadi server.

Biasanya para perancang web atau web master jika akan merencanakan (planning), kemudian membangun (buliding)dilakukan di komputer local atau bisa juga di jaringan local, tidak langsung di host --> internet. Oleh karena itu perlu di komputer kita di jadikan server sehingga kita seolah olah sedang meng update di hostnya (tempat penyimpanan file2 yang diperlukan website) internet.

Dengan di tempatkannya file file pendukung website di komputer kita kita tidak perlu buka masuk internet .. sehingga hal ini mengurangi presentasi waktu dan biaya,

2.7 Android SDK

Dalam membuat suatu aplikasi Android tidak hanya eclipse yang digunakan untuk membuatnya, tetapi dibutuhkan Android SDK (*Software Development Kit*) yang berisi tool maupun library yang digunakan dalam aplikasi Android. Di dalam Android SDK terdapat banyak library yang terdiri atas berbagai macam versi Android mulai dari yang pertama sampai dengan yang terbaru. Dalam SDK terdapat AVD (*Android Virtual Device*) Manager yang berfungsi sebagai emulator Android. Aplikasi yang dibuat dapat digunakan di AVD sebagai pengganti perangkat Android yang asli.



Gambar 2.6 SDK Manager

BAB III PERANCANGAN SISTEM

3. Perancangan sistem

3.1 Perancangan Umum

Perancangan digunakan untuk memberikan gambaran secara umum terhadap aplikasi, hal ini berguna untuk meninjau perancangan aplikasi yang akan dikembangkan, sehingga kebutuhan akan aplikasi desain aplikasi multimedia untuk enkripsi messenger tersebut dapat diketahui sebelumnya, kemudian hasil perancangan akan menjadi dasar untuk melakukan perancangan atau desain aplikasi sesuai kebutuhan sistem.

Dalam merancang aplikasi proyek ini terlebih dahulu dilakukan pembuatan desain antar muka aplikasi dan menentukan fasilitas yang dimasukkan kedalam aplikasi. Desain antarmuka dibuat semenarik mungkin, hal ini berguna untuk memikat daya tarik pada pengguna. Setelah desain antarmuka selesai dirancang, langkah selanjutnya adalah menentukan fasilitas – fasilitas yang ada di dalam aplikasi enkripsi messenger, sehingga pengguna dapat mengoperasikan aplikasi yang dibuat.

Aplikasi eclipse untuk mempelajari cara membuat enkripsi messenger dan bahan-bahan yang diperlukan, desain tampilannya dibuat serinci mungkin untuk mempermudah yang sedang mempelajarinya. Dalam aplikasi ini juga disertai gambar dan video tutorial pada beberapa penjelasannya agar lebih mudah dipahami dan dimengerti.

3.2 Analisis Sistem

3.2.1 Deskripsi Sistem

Sistem *secure messenger* ini memiliki fungsi untuk mempermudah para pengguna dalam menjamin hal privasi. Semua *messenger* yang masuk akan dikelola kemudian dienkripsi dengan RSA (*Rivest Shamir Adleman*) dan akan dikeluarkan menjadi messenger biasa pada pengguna *messenger* yang lain.

3.2.2 Messenger Saat Ini

Pada umumnya *messenger* saat ini yang sering di gunakan oleh kebanyakan orang masih belum menggunakan tingkat keamanan yang memadai. Untuk melindungi pengguna dari kebocoran informasi yang penting, maka di gunakanlah tingkat keamanan yang harus bisa menjamin pengguna.

3.2.3 Otoritas Akses Sistem

Desain sistem informasi merupakan serangkaian aliran data yang terbetuk dari beberapa aktivitas atau proses yang terkomputerisasi, sehubungan dengan otorisasi akses data dalam sistem ini akan dikelompokkan sebagai berikut :

- Admin
Administrator yang memiliki hak penuh untuk mengakses keseluruhan system yang masuk. Kemudian mengelompokkan menjadi beberapa *group*. dan mengirimkan ke *user* berupa *messenger*.
- User
User hanya bisa menggunakan *software* ini dan menambah teman atau menghapus teman.

3.2.4 Analisa Kebutuhan Perangkat Lunak

Beberapa spesifikasi kebutuhan yang dibutuhkan dalam rancangan perangkat lunak ini antara lain:

1. Dapat menghandle semua permintaan dari client, seperti, kirim pesan, mendapatkan pesan, sign in dan sign out.
 2. Dapat menampilkan daftar user dan juga pesan yang sedang diperbincangkan di antara client.
 3. Dapat menampilkan user yang sedang online maupun yang sedang offline.
 4. Dapat menggunakan sebuah list view untuk menampilkan pesan.
 5. Dapat mengurus semua tampilan atau pencitraan dari aplikasi android.
 6. Dapat mengamankan pesan yang di terima dari user 1 dan user 2.
-

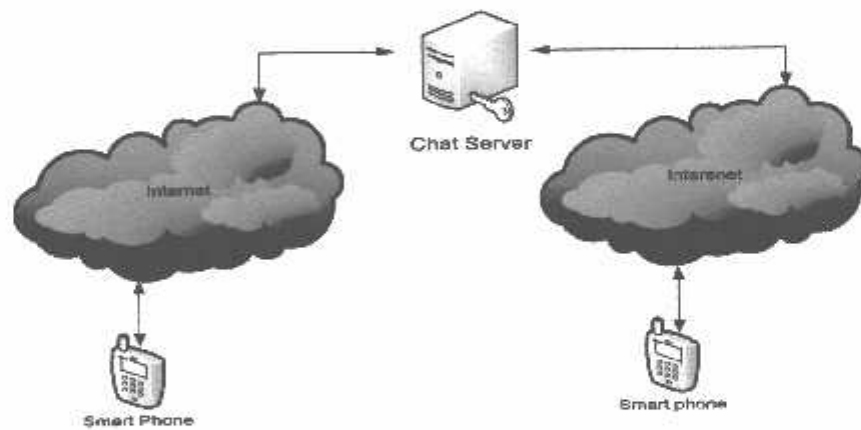
3.3 Perancangan Aplikasi Secure Messenger

Perancangan aplikasi Secure Messenger untuk mempermudah kita dalam berkomunikasi secara nyaman, karena kita tidak perlu khawatir akan privasi yang kita buat melalui messenger berbasis android ini.

Untuk proses pembuatan enkripsi messenger ini, dimulai dari pembuatan program messenger dulu. Setelah program selesai di buat maka akan terbentuklah messenger yang di buat untuk chatting, tapi masih belum menjamin tingkat keamanan tersebut, maka oleh itu dibuatlah enkripsi untuk melindungi privasi user.

Langkah-langkah yang dilakukan untuk memulai program sebagai berikut :

1. Membuat sourcecode pada masing-masing langkah pada program Eclipse.
2. Menganalisa dan merancang sistem dengan menggunakan hasil yang di dapat dari emulator Android.
3. Membuat sistem berdasarkan program dan perancangan yang dilakukan.
4. Uji coba program.
5. Evaluasi hasil uji coba.



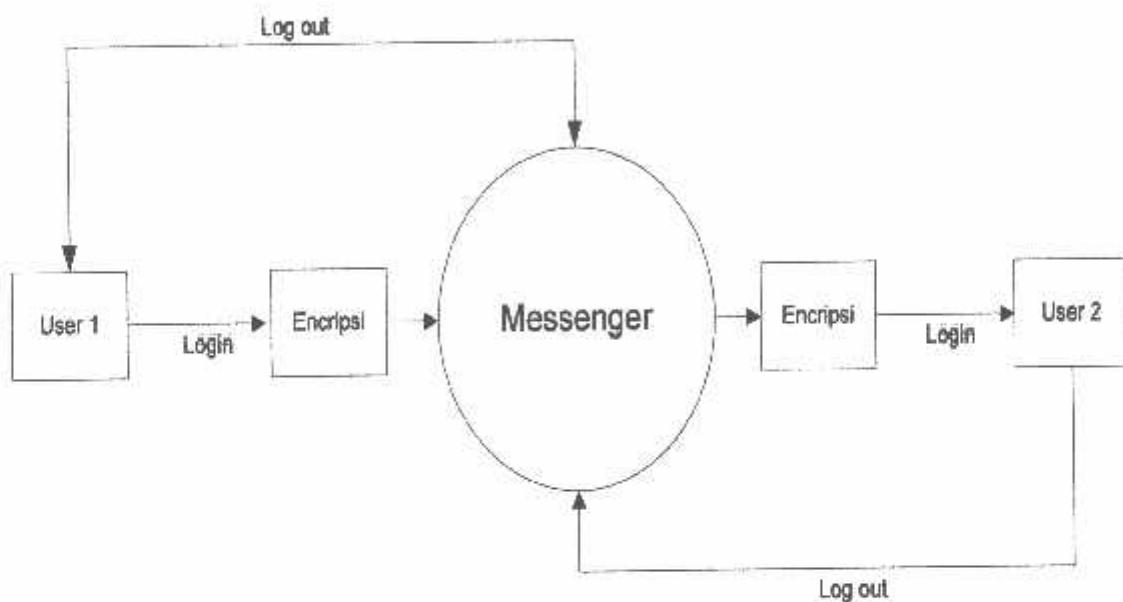
Gambar 3.1 Desain Sistem *Secure Messenger*.

User 1 mengirim *messenger* kepada *User 2* yang sama-sama memakai program Android *Secure Messenger*. Kemudian pesan diterima oleh server yang diubah menjadi enkripsi, pesan tersebut diteruskan kepada *User 2* sudah dalam keadaan normal lagi.

3.3.1 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) merupakan alat yang digunakan untuk menggambarkan keseluruhan sistem informasi yang akan dibuat atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir ataupun lingkungan fisik dimana data tersebut akan disimpan. DFD yang pertama kali dibuat adalah DFD level 0 atau *Context Diagram*.

3.3.2 Diagram Konteks



Gambar 3.2 Diagram konteks *Secure Messenger*

Berdasarkan gambar 3.2 proses "*Secure Messenger* pada android menggunakan Eclipse" mempunyai data masukan berupa :

1. **Login User :**

Login berfungsi untuk mengatur para pengguna yang mempunyai *Username* dan *Password* agar bisa masuk kedalam sebuah messenger. Parameter yang akan digunakan adalah :

- a. *Username*

Berisi masukan *username* yang sudah didaftarkan oleh pengguna aplikasi ini untuk bisa melangkah ke tahap selanjutnya.

b. *Password*

Berisi masukan *password* yang sudah didaftarkan oleh pengguna dan akan disimpan untuk masuk ke program lagi.

2. *Messenger*

Messenger berisi tentang pesan yang ada didalam aplikasi ini. Pesan-pesan tersebut akan bisa digunakan oleh para pengguna program tersebut dengan tingkat keamanan yang sudah memakai RSA (*Rivest Shamir Adleman*).

3. *Log out*

Log out berfungsi untuk mengeluarkan pengguna aplikasi program tersebut. Supaya keamanan dari pengguna bisa terjamin disaat *handphone* sedang dipakai oleh orang lain. Dan untuk memulai kembali menjalankan aplikasi harus *Login* terlebih dahulu.

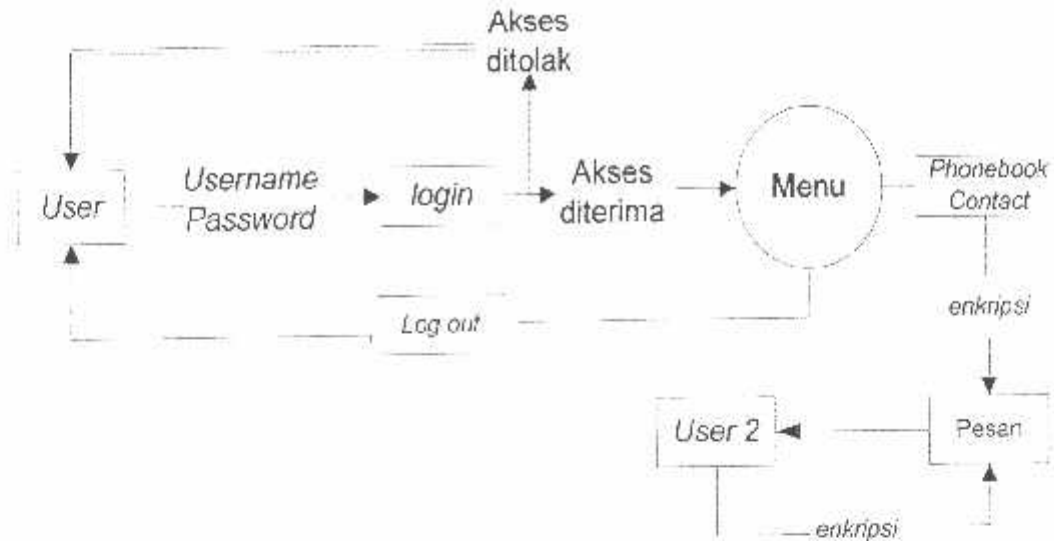
Berdasarkan gambar 3.2 proses "*Secure Messenger* pada android menggunakan Eclipse" para pengguna dapat menikmati komunikasi yang bisa di akses secara lancar. Aplikasi ini juga bisa mendekatkan orang, meskipun letaknya berjauhan.

Aplikasi *Secure Messenger* juga sangat berguna buat kantor-kantor, karena aplikasi ini bisa melindungi privasi pengguna supaya tidak terjadi kebocoran informasi ke masyarakat ataupun ke *rival* kerja pengguna. Tetapi para pengguna aplikasi ini harus menggunakan perangkat seluler yang harus memakai program android. Karena aplikasi ini di kususkan bagi pengguna android, minimal android versi 2.2 (*frozen yogurt*).

Para pengguna aplikasi ini dapat masuk kedalam dengan cara mendaftarkan dulu melalui *server* supaya bisa terdaftar di sana. Selesai mendaftar dan melengkapi syarat-syarat yang di tentukan baru pengguna aplikasi ini bisa melakukan *messenger*.

3.3.3 DFD Level 1

DFD level 1 menggambarkan proses – proses yang lebih detail dari aplikasi *Secure Messenger* pada android menggunakan Eclipse berbasis android.

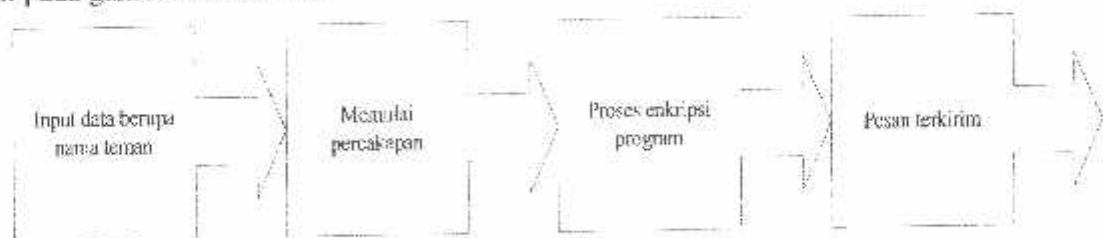


Gambar 3.3 DFD Level 1 *Secure Messenger*

Gambar diatas merupakan DFD level 1 dari aplikasi *Secure Messenger* pada android menggunakan Eclipse berbasis android. Mencakup menu.

3.4 Desain Sistem

Sebelum membangun sebuah sistem, perlu merancang desain sistem terlebih dahulu, untuk dapat mempermudah pembuatan program, dan memberikan gambaran umum tentang program tersebut. Secara garis besar, alur program dapat dilihat pada gambar berikut ini :



Gambar 3.4 Diagram Blok Aplikasi

Keterangan :

1. Inputan teman berdasarkan sesama pengguna aplikasi
2. Proses *enkripsi* merupakan proses keamanan di dalam *messenger*.

Prinsip kerja :

1. Internet berguna untuk melakukan *chatting*.
2. Data yang dikirim akan diproses menjadi enkripsi.
3. Smartphone Android sebagai pengirim dan penerima *chatting*.

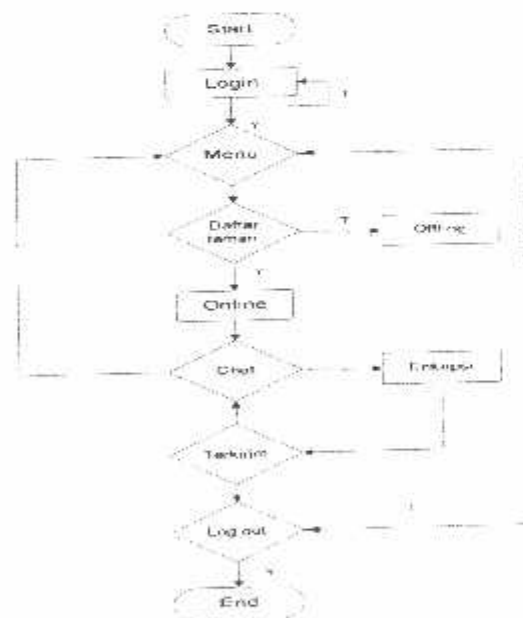
3.4.1 Spesifikasi Sistem

Sesuai dengan tujuan Aplikasi Scure Messenger, maka spesifikasi sistem adalah sebagai berikut :

1. User hanya dapat berkomunikasi dan mencari informasi nama teman yang sudah ada di database.
2. Sistem diimplementasikan dengan bahasa pemrograman PHP dan MySQL sebagai basis data.
3. Sistem operasi yang digunakan adalah sistem operasi Android.
4. Aplikasi RSA ini, diharapkan dapat memberikan keamanan kepada *user* untuk kenyamanan berkomunikasi.
5. Sistem ini menggunakan *gadget* Android sehingga memudahkan mobilitas *user* dalam kegiatan sehari-hari.

3.5 Flowchart

Alur program secara lengkap dapat digambarkan dengan flowchart berikut ini :



Gambar 3.2. Flowchart Program

BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Tahap implementasi pengembangan perangkat lunak merupakan proses perubahan analisa sistem menjadi sistem yang dapat dijalankan. Tahap ini merupakan lanjutan dari proses perancangan, yaitu proses pemrograman perangkat lunak sesuai dengan analisa dan desain sistem.

Dalam pengimplementasian ini akan diperoleh beberapa datayang bersifat informasi tentang bisa atau tidaknya sistim bekerja beserta kelemahannya. Dalam hal ini pengujian dilakukan hanya pada PC atau Laptop yang di dalamnya sudah terinstal Eclipse Galileo dan sejenisnya.

Untuk melakukan pengujian Aplikasi Friend Finder digunakan sebuah Laptop dengan spesifikasi *hardware* dan *software* sebagai berikut :

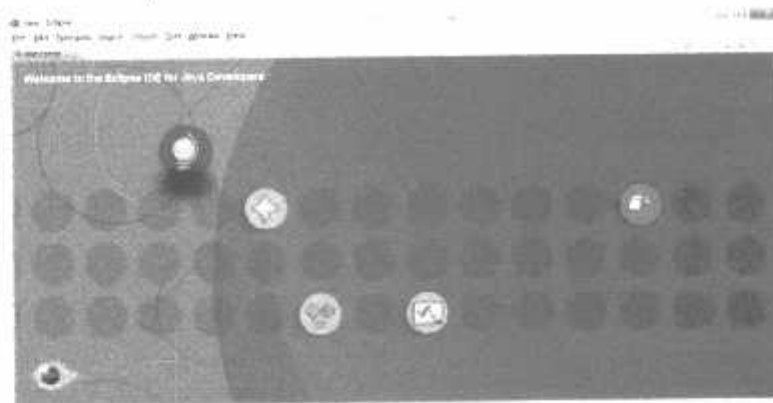
- Prosesor Intel Core i3
- RAM 4 GB
- VGA NVIDIA GeForce 2 GB
- Sistem Operasi menggunakan Windows 7 64 Bit
- Eclipse Galileo versi 2.3.3 (Gingerbread)

4.2 Persiapan Pembuatan Aplikasi

4.2.1 Eclipse dan ADT

Eclipse yang digunakan untuk membuat aplikasi ini merupakan Eclipse Galileo. Sebeum membuat aplikasi pada android perlu dilakukan pengaturan terhadap Eclipse terlebih dahulu, yaitu dengan menginstall ADT yang merupakan plug- in dari Eclipse agar bisa membuat project Android. Plug – in ini didapatkan dengan cara mendownload dari <https://dl-ssl.google.com/android/eclipse/> ADT yang digunakan adalah ADT versi 21.

3. Tampilan awal Eclipse.



Gambar 4. 3 *Tampilan Awal Eclipse.*

4. Workspace.



Gambar 4. 4 *Tampilan Workspace*

4.2.1.2 Mengkonfigurasi Eclipse dengan ADT

Untuk dapat menggunakan Eclipse dalam membangun Aplikasi Android, Terlebih dahulu harus dilakukan konfigurasi *Android Development Tools (ADT)* dalam Eclipse.

1. Jalankan Eclipse (jika belum berjalan)
2. Pilih pada menubar → *Help* → *Install New Software*



Gambar 4.5 Tampilan Instal Android Development Tools (ADT)

3. Muncul *Install window pops-up*, yang mengijinkan untuk menginstall *plug-in* baru ke dalam Eclipse.



Gambar 4.6 Tampilan pops-up plug-in ADT

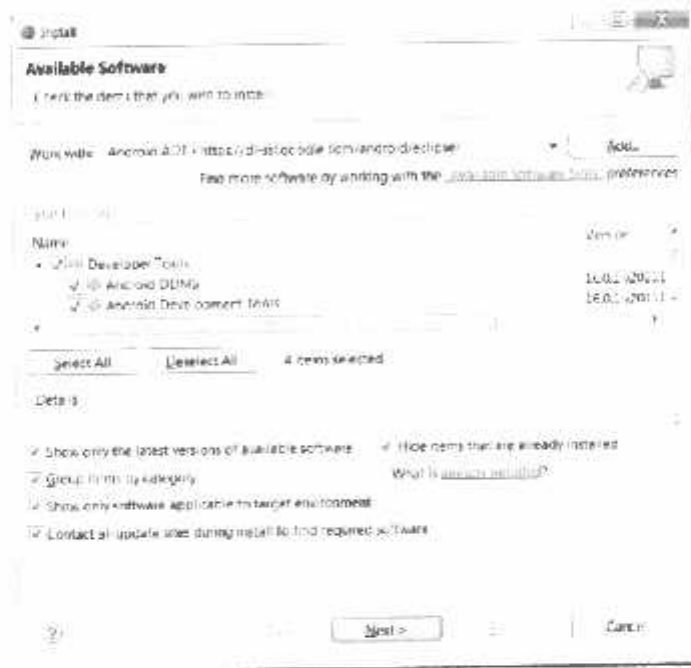
4. Tuliskan nama pada *Name Field*, misal: **Android ADT**

5. Pada *Location Field*, tuliskan: <https://dl-ssl.google.com/android/eclipse/>



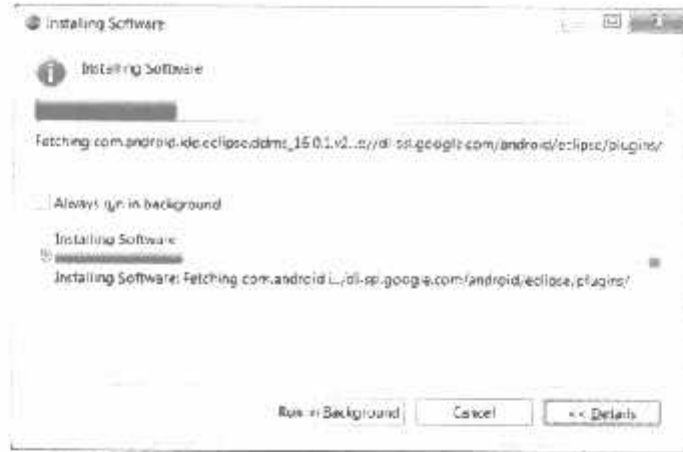
Gambar 4.7 Tampilan pops-up Repository plug-in.

6. Tampilan Software ADT yang tersedia di Google.



Gambar 4.8 Tampilan Instalasi ADT plug-in.

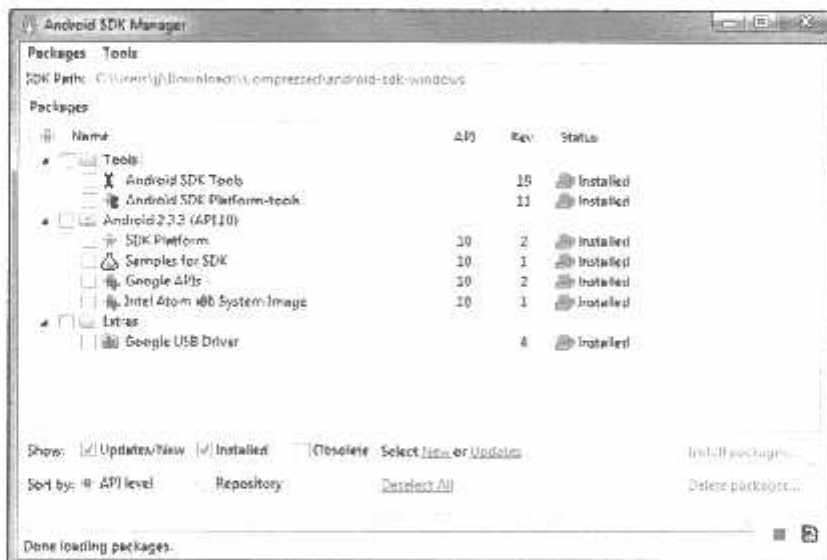
7. Select All, klik next> untuk install
8. License Agreement, Klik Finish
9. Install dalam akan berjalan, plugin ADT akan terinstall



Gambar 4.9 Tampilan Proses Instalasi ADT plug-in pada Eclipse.

4.2.2 Android SDK

Android SDK merupakan komponen library yang digunakan eclipse untuk membuat aplikasi Android. Android SDK yang digunakan merupakan Android SDK revision 21, versi ini digunakan karena mengacu pada ADT yang digunakan merupakan versi 21. Di dalam Android SDK juga harus dilakukan instalasi komponen yaitu Android versi 2.3.3 karena aplikasi ini akan digunakan untuk Android dengan minimal versi 2.3.3. Gambar berikut ini dapat menjelaskan versi dan API Android yang harus diinstal.



Gambar 4.10 Android SDK

4.2.2.1 Mengkonfigurasi lokasi SDK

1. Pada menu, pilih window → preferences.
2. Pilih Android pada tab di kiri
3. Atur lokasi SDK, Kemudian OK

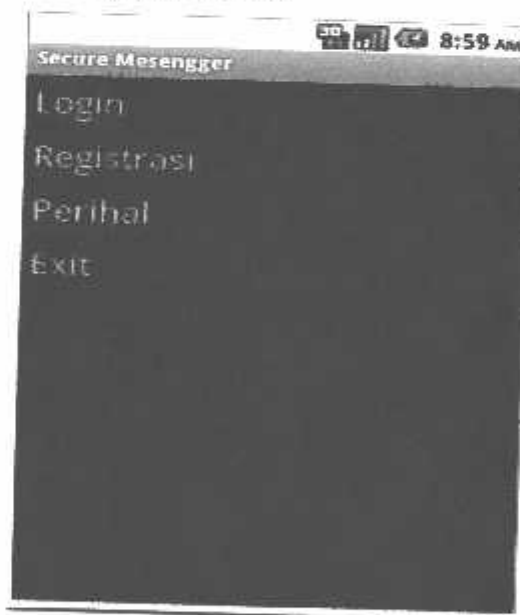


Gambar 4.11 Tampilan Lokasi SDK Android.

4.3 Tampilan Aplikasi

4.3.1 Tampilan awal

Tampilan ini merupakan tampilan awal dari aplikasi Secure Messenger, untuk lebih jelasnya dapat dilihat pada gambar berikut.

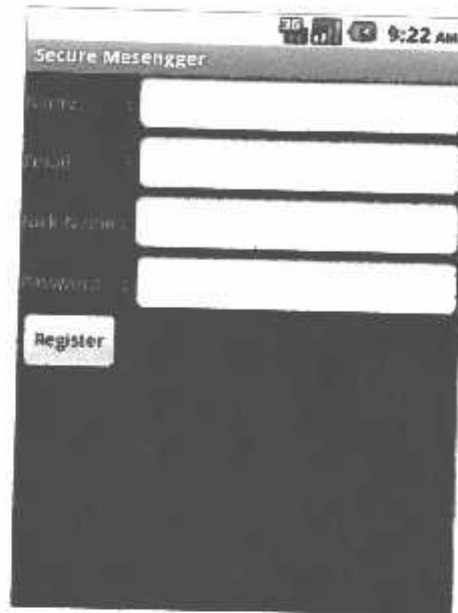


Gambar 4.12 Tampilan Awal Aplikasi

4.3.2 Tampilan Registrasi

Tampilan ini berfungsi menyediakan menu yang salah satu di dalamnya terdapat registrasi. Registrasi berfungsi untuk mendaftarkan pengguna, supaya bisa masuk ke dalam layanan aplikasi tersebut

Untuk lebih jelasnya dapat dilihat pada gambar 4.17 dibawah ini:



Gambar 4.13 Menu Registrasi

Untuk menampilkan registrasi pada aplikasi kita harus terlebih dahulu menginputkan *Sourcecode* kita, ke dalam *main.xml*.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    android:id="@+id/widget188"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    xmlns:android="http://schemas.android.com/apk/res/android"
    >
    <LinearLayout
        android:id="@+id/widget80"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        >
```

Gambar 4.14 Tampilan Sourcecode Registrar

4.3.3 Tampilan Login

Tampilan login ini berfungsi untuk menyediakan menu yang salah satu didalamnya berfungsi untuk memasukan pengguna didalam menu aplikasi tersebut. Yang mana didalamnya terdapat User Name dan Password.

Untuk lebih jelasnya dapat dilihat pada gambar 4.19 dibawah ini:



Gambar 4.15 Menu Login

Untuk menampilkan login pada aplikasi kita harus terlebih dahulu menginputkan

Sourcecode kita, ke dalam main.xml.

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    android:id="@+id/widget88"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    xmlns:android="http://schemas.android.com/apk/res/android"
    >
    <LinearLayout
        android:id="@+id/widget89"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        >
        <TextView
            android:id="@+id/textView1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="User Name : "
```

Gambar 4.16 Tampilan Sourcecode Login

4.3.4 Tampilan Ruang Chat

Tampilan perihal ini berfungsi untuk menyediakan menu yang di dalamnya berfungsi untuk melakukan komunikasi chatting antar sesama pengguna aplikasi tersebut. Dengan begitu para pengguna layanan tersebut bisa dengan bebas melakukan pembicaraan.

Untuk lebih jelasnya dapat dilihat pada gambar 4.21 dibawah ini:



Gambar 4.17 chat

Untuk menampilkan perihal pada aplikasi kita harus terlebih dahulu menginputkan *Sourcecode* kita, ke dalam main.xml.

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <AbsoluteLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     android:layout_width="match_parent"
5     android:layout_height="match_parent" >
6
7     <Button
8         android:layout_width="145dp"
9         android:layout_height="wrap_content"
10        android:layout_x="85dp"
11        android:layout_y="118dp"
12        android:onClick="clickHandler"
13        android:text="Buat" >
14
15    </Button>
16    <EditText
17        android:layout_height="wrap_content"
18        android:text=""
19        android:id="@+id/roomName"
20        android:layout_width="fill_parent"
21        android:layout_x="1dp"
22        android:layout_y="28dp"
23
24    </EditText>
25 </AbsoluteLayout>

```

Gambar 4.18 Tampilan Sourcecode chat room

4.3.5 Tampilan Room

Tampilan ini muncul saat *user* menekan salah satu *marker* yang terdapat di dalam peta. Untuk lebih jelasnya pada gambar 4.19 berikut:



Gambar 4.19 Room

4.4 Pengujian Hasil

Setelah pembuatan rancang bangun, dihasilkan suatu aplikasi, dimana setelah dilakukan pengujian pada program ini melalui prosedur - prosedur dalam bahasa pemrograman *Java*, program ini dapat berjalan dengan baik. Pengujian hasil difokuskan pada pengujian aplikasi.

4.4.1 Lingkungan Perangkat Keras (Hardware) Mobile Phone

Beberapa perangkat keras (*hardware*) yang digunakan dalam pengujian aplikasi ini yaitu dengan menggunakan *handphone* berbasis Android, spesifikasi dari *handphone* tersebut yaitu:

Handphone berbasis Android :

- Type : Samsung Android Galaxy Young GT-S5360
- Network : HSDPA /EDGE / GPRS

- ISP : Telkomsel
- OS : Android OS – v2.3.6 (*Gingerbread*)

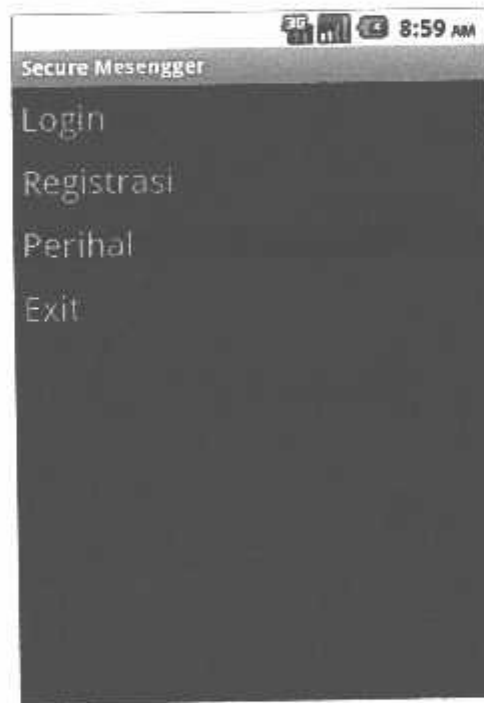
4.4.2 Pengujian Aplikasi Secure Messenger di Handphone

Pada pengujian Aplikasi Secure Messenger akan dilakukan pada *handphone* berbasis Android yaitu, *Samsung Galaxy Mini*. Dimana dalam pengujian untuk mengetahui tingkat keberhasilan dari aplikasi ini.

4.4.2.1 Pengujian Aplikasi

Pada pengujian Aplikasi, aplikasi dapat berjalan dengan baik dengan menampilkan aplikasi pada *device*. Untuk lebih jelasnya pada gambar berikut:

Gambar ini adalah tampilan awal menu *Secure Messenger*, dimana terdapat Login, Registrasi, Perihal, dan juga Exit.



Gambar 4.20 Tampilan Menu Awal

Tampilan berikut ini adalah menu registrasi yang berfungsi untuk mendaftarkan pengguna supaya bisa masuk ke dalam aplikasi secure messenger.



Gambar 4.21 Tampilan *Registrasi*

Tampilan berikutnya adalah menu Login yang berfungsi untuk memasukan nama pengguna ke dalam aplikasi Secure Messenger.



Gambar 4.22 Tampilan *Login*

Tampilan selanjutnya adalah menu chat yang berfungsi untuk memulai perbincangan antara pengguna aplikasi tersebut.



Gambar 4.23 Tampilan chat

Tampilan yang berikut ini adalah tampilan pada RSA yang berfungsi untuk mengamankan chatting.



Gambar 4.24 Tampilan RSA

Tampilan ini adalah hasil contoh dari RSA dari kata "SAYA" yang sudah diubah menjadi kode-kode dalam bilangan ASCII.



Gambar 4.25 Tampilan hasil *RSA*

BAB V PENUTUP

5.1 Kesimpulan

Dari pembahasan dan pembuatan Tugas Akhir yang berjudul "RANCANG BANGUN APLIKASI *SECURE MESSENGER* MENGGUNAKAN *ECLIPSE GALILEO* BERBASIS ANDROID" ini dapat diambil kesimpulan sebagai berikut :

- Dalam pembuatan aplikasi ini, Secure Messenger menggunakan pengaman RSA yang dalam tingkat keamanannya termasuk yang paling terbaru dan canggih.
- Setiap huruf yang dikirim melalui aplikasi ini akan diubah dalam bentuk bahasa *kriptografi* kemudian akan diubah lagi menjadi seperti semula sesudah sampai ke penerima.
- Tulisan dan ikon pada aplikasi ini akan tampak lebih simpel dan pemakaiannya pada android maka akan ringan.
- Keamanan dalam privasi akan selalu tetap terjaga, karena setiap huruf akan selalu diubah menjadi kode-kode rahasia.

5.2 Saran

Aplikasi ini masih memiliki banyak kekurangan yang harus diperbaiki maupun dikembangkan, adapun saran-saran yang ingin disampaikan yaitu:

1. Menyediakan fasilitas *switch channel* sehingga dengan mudah dapat dilakukan navigasi jika sedang chatting dengan beberapa teman pada channel yang berbeda.
2. Menyediakan fitur fitur *control panel* untuk pengaturan pribadi seperti jenis huruf, warna huruf, ukuran huruf, dan profil *picture*.
3. Menyediakan fasilitas rekam history, sehingga percakapan dapat disimpan di persistent database.

DAFTAR PUSTAKA

- 1) Naazrudin Safaat. 2012. *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung : Informatika.
- 2) Winarno, Edy, Ali Zaki dan SmitDev Community. 2012. *Hacking dan Programing dengan Android SDK untuk Advanced*. Jakarta : Elex Media Komputindo.
- 3) Siregar Michael Ivan. 2011. *Membongkar Source Code Berbagai Aplikasi Android*. Yogyakarta : Gava Media.
- 4) <http://en.wikipedia.org/wiki/WAMP>, diakses tanggal 18 April 2012.
- 5) <http://agusharyanto.net/wordpress/?tag=belajar-android>, diakses tanggal 19 April 2012.
- 6) <http://www.maniacms.web.id/2012/01/pengertian-wamp-server.html>, diakses tanggal 12 Juli 2012
- 7) Siregar Michael Ivan. 2010. *Mengembangkan Aplikasi Enterprise Berbasis Android*. Yogyakarta : Gava Media.

LAMPIRAN



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Nomor Surat : ITN-340/EL-FTI/2012
Lampiran : -
Perihal : BIMBINGAN SKRIPSI

Kepada : Yth. Bapak/Ibu **Dr. Eng. Aryuanto Soetedjo, ST, MT**
Dosen Teknik Elektro S-1
ITN MALANG

Dengan Hormat

Sesuai dengan permohonan dan persetujuan dalam Proposal Skripsi untuk mahasiswa :

Nama : **ANDIKA SUBRATA**
Nim : **0712628**
Fakultas : **Teknologi Industri**
Program Studi : **Teknik Elektro S-1**
Konsentrasi : **Teknik Komputer & Informatika**

Maka dengan ini pembimbingan tersebut kami serahkan sepenuhnya kepada Saudara/i selama masa waktu :

" Semester Ganjil Tahun Akademik 2012 -2013 "

Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.



Mengetahui

Program Studi Teknik Elektro S-1

Ir. Yusuf Ismail Nakhoda, MT

NIP. Y. 1018800189



FORMULIR BIMBINGAN SKRIPSI

Nama : Andika Subrata
Nim : 07.12.628
Masa Bimbingan : Semester Ganjil Tahun Akademik 2012-2013
Judul Skripsi : RANCANG BANGUN APLIKASI *SECURE MESSENGER* PADA *ECLIPSE GALILEO* BERBASIS ANDROID

No	Tanggal	Uraian	Paraf Pembimbing
1	30/12/12	Revisi BAB I	
2	31/12/12	Revisi BAB II	
3	3/1/13	Revisi BAB III	
4	8/1/13	Revisi BAB IV	
5	5/2/13	Revisi material seluruh hasil	
6			
7			
8			
9			
10			

Malang,
Dosen Pembimbing

Dr. Eng. Arvanto Soetedjo. ST, MT
NIP. R/1030800417

Form S-4b



FORMULIR BIMBINGAN SKRIPSI

Nama : Andika Subrata
Nim : 07. 12. 628
Masa Bimbingan : Semester Ganjil Tahun Akademik 2012-2013
Judul Skripsi : RANCANG BANGUN APLIKASI *SECURE MESSENGER* PADA *ECLIPSE GALILEO* BERBASIS ANDROID

No	Tanggal	Uraian	Paraf Pembimbing
1	30/12/12	REVISI BAB I	
2	31/12/12	REVISI BAB II	
3	5/1/13	REVISI BAB III	
4	12/1/13	REVISI BAB IV	
5	8/2/13	REVISI mabalah struktur hasil	
6			
7			
8			
9			
10			

Malang, 8 Februari 2013
Dosen Pembimbing II

Michael Ardita, ST. MT
NIP. P. 1031000434



**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

NAMA : Andika Subrata
NIM : 07.12.628
JURUSAN : Teknik Elektro S-1
KONSENTRASI : Teknik Komputer dan Informatika
MASA BIMBINGAN : Semester Ganjil Tahun Akademik 2012-2013
JUDUL : **RANCANG BANGUN APLIKASI SECURE MESSENGER
MENGUNAKAN ECLIPSE GALILEO BERBASIS ANDROID**

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :

Hari : Senin
Tanggal : 18 Februari 2013
Dengan Nilai : 77,65 (B+) \checkmark

PANITIA UJIAN SKRIPSI

Ketua Majelis Penguji

M. Ibrahim Ashari, ST, MT
NIP.P.1030100358

Sekretaris Majelis Penguji

Dr. Eng. Arvanto Soetedio, ST, MT
NIP.P.1030800417

ANGGOTA PENGUJI

Dosen Penguji I

Sotyo Hadi, ST.
NIP.Y. 1039700309

Dosen Penguji II

Ir. Eko Nurcahyo, MT
NIP. Y. 1028400087



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : AHDIKA SUBRATA
NIM : 0912628
Perbaikan meliputi :

- 1) Perjelas antara Rumusan Masalah & Batasan Masalah, + Latar Belakang.
- 2) Tambah sumber Referensi.
- 3) Bab IV. Jaga penggunaan Ekuivalen, + Proses Penyederhanaan.
- 4) Bab II. Tambahan contoh proses pengaliran / perubahan
di bawah huruf mayori angka.

Malang, 18-02-2013

(T. EKO N.)



FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Komputer dan Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : Andika Subrata
NIM : 07.12.628
Jurusa : Teknik Elektro S-1
Konsentrasi : Teknik Informatika & Komputer
Masa Bimbingan : Semester Ganjil Tahun Akademik 2012 - 2013
Judul Skripsi : **RANCANG BANGUN APLIKASI SECURE MESSENGER
MENGUNAKAN ECLIPSE GALILEO BERBASIS ANDROID**

No	Penguji	Tanggal	Uraian	Paraf
1	Penguji II	18 Februari 2013	Perjelas antara rumusan masalah, batasan masalah dan latar belakang	
2	Penguji II	18 Februari 2013	Tambah sumber setiap gambar	
3	Penguji II	18 Februari 2013	BAB III tambah contoh proses perubahan huruf menjadi angka	
4	Penguji II	18 Februari 2013	BAB IV jangan menggunakan emulator ditambah proses penyesuaian	

Disetujui :

Penguji I

Sothyohadi, ST
NIP.Y. 1039700309

Penguji II

Ir. Eko Nurcahyo, MT
NIP. Y. 1028700172

Mengetahui :

Dosen Pembimbing I

Dr. Eng. Arvianto Soetedjo, ST, MT
NIP.P.1030800417

Dosen Pembimbing II

Maichael Ardita, ST, MT
NIP.Y. 1031000434

SURAT PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini :

Nama : Andika Subrata
NIM : 0712628
Program Studi : Teknik Elektro
Konsentrasi : Teknik Informatika dan Komputer

Dengan ini menyatakan bahwa Skripsi yang saya buat adalah hasil karya sendiri, tidak merupakan plagiasi dari karya orang lain. Dalam Skripsi ini tidak memuat karya orang lain, kecuali dicantumkan sumbernya sesuai dengan ketentuan yang berlaku.

Demikian surat pernyataan ini saya buat, dan apabila di kemudian hari ada pelanggaran atas surat pernyataan ini, saya bersedia menerima sanksinya.

Malang, 24-Maret-2013

Yang membuat Pernyataan,



Andika Subrata
0712628

LAMPIRAN

```
<?php
//Mengatur koneksi ke basis data
Mysql_connected("localhost","root");
Mysql_select_db("aplidroidmessenger");
//step 3: activate global variable
Ectract($_REQUEST, EXTR_OVERWRITE);
//echo "berhasil c=" . $c . "k=" . $k . "p=" . $p ;
If ($c=="r") {
    $reg = " ";
    $h = ereg_replace("~", $reg, $h);
    $n = ereg_replace("~", $reg, $n);
    If ($s == "user") {$sql = "SELECT * FROM tbl_user where nick = '$k'";
    $result = mysql_query($sql);
    $total = mysql_num_rows($result);
    If ($total == 0) {$sql = "INSERT INTO tbl_user
(name,email,password,nick,status,cdate) ";
    $sql .= "VALUES ('$n','$e','$p','$k','offline',NOW())";
    $result = mysql_query ($sql) or die ("gagal");
    $status = 1;
    $msgReg = "=====\n";
    $msgReg .= " CONGRATULATION\n";
    $msgReg .= "=====\n";
    $msgReg .= "you have been registered\n";
    $msgReg .= "your nick is: $k \n";
    Echo $msgReg;
```

```

}
Else {Echo "nick anda telah teregister" ;
}
Else if ($s == "room") {$sql = "SELECT * FROM tbl_user where nick = '$k'";
$result = mysql_query ($sql) ;
$total = mysql_num_rows ($result) ;
If ($total == 0) {$sql = "INSERT INTO tbl_user
(name,email,password,nick,status,cdate) ";
$sql .= "VALUES ('$n','$e','$p','$k','room',NOW ())"
$result = mysql_query ($sql) or die ("gagal")
$status = 1;
$msgReg = "S U C C E S S";
Echo $msgReg ;
}
}
}
If ($c=="s") {$sql = "SELECT * FROM tbl_user" ;
$sql = "WHERE nick = '$k' AND password = '$p'";
$result = mysql_query ($sql) or die ("3");
$row = mysql_num_rows ($result) ;
If ($row > 0) {$sql ="UPDATE tbl_user set status = 'online' ";
$sql .= "WHERE nick = '$k' ";
$result = mysql_query ($sql) or die ("2");
If ($result) {Echo "11" ;
}
}
Else{Echo "2" ;
}

```

```

}
}
If ($c=="f") {If ($z=="on") {$sql = "SELECT * FROM tbl_user";
$sql .= "WHERE status='online' AND nick != '$k'"
//$sql .= "WHERE status='online' AND NOT nick = '$k'";
$result = mysql_query ($sql) or die ("gagal");
$spce = " _ ";
$con = "1";
$row=mysql_num_rows ($result);
If (!$row == 0) {
While ($data = mysql_fetch_array ($result) ) {
$namea .= $data ["nick"] . $spce ;
}
Echo "$con$namea";
}
Else {Echo "1";
}
}
Else if ($z=="off") {$sql = "SELECT * FROM tbl_user" ;
$sql .= "WHERE status='offline' AND nick != '$k'"
//$sql .= "WHERE NOR nick = '$k'";
$result = mysql_query ($sql) or die ("gagal");
$spce = " _ ";
$con = "0";
$row=mysql_num_rows ($result);
If (!$row == 0) {While ($data = mysql_fetch_array ($result) ) {
$namea .= $data ["nick"] . $spce ;
}
}
}
}

```

```

}Echo "$con$nama";
}
Else {Echo "0";
}
}

If ($c=="a") {$spce = "_";
$con = "1";
$sqla = "SELECT * FROM tbl_user";
$sqla .= "WHERE nick = '$k'";
$result =mysql_num_rows ($hasil) > 0) {
$i=1 ;
While ($data = mysql_fetch_array ($hasil) ) {
$roomid[$i] = $data ["id_room"];
$i++;
}
}
For ($i=1 ; $i <=count ($roomid) ; $i++ ) {$sql_rm = "SELECT *FROM tbl_user";
$sql_rm .= "WHERE id = '$roomid [$i]'" ;
$resultq = mysql_query ($sql_rm) or die ("gagal");
$row=mysql_num_rows ($resultq);
If (!$row == 0) {
While ($data = mysql_fetch_array ($resultq) ) {$nama .= $data ["nama'] .$spce ;
}
}
}
Echo "$con$nama";
}

```

```

If ($c=="0") {$sql = "UPDATE tbl_user set status = 'offline' ";
$sql .= "WHERE nick = '$k' ";
$result = mysql_query ($sql) or die ("2");
}
If ($c=="k") {
// proses MESSAGE pada variable $m
// replace delimiter ~ dengan spasi
$spasi = " ";
$null = "";
$m = ereg_replace ("~", $spasi, $m);
$t = ereg_replace ("~", $null, $t);
// Lakukan INSERT ke tbl_msg
// Insert ini dilakukan sebanyak jumlah penerima pesan,
// Jumlah penerima pesan ditemukan pada proses diatas.
// Pada saat insert, status diset=0
// artinya pesan belum dibaca oleh penerima
If ($m != "") {$sql .= "INSERT INTO tbl_msg (dari, ke, msg, status)";
$sql .= "VALUES ('$n', '$t', '$m', '0') ";
$result = mysql_query ($sql);
}
}
If ($c=="p") {
$pisah = "_";
$batas = ":";
If ($r == "person") {
$sql_tm = "SELECT * FROM tbl_msg";
$sql_msg .= "WHERE dari='$a' ";
}
}

```

```

$ssql_tm .= "AND ke='$n' AND status = '0' ";
$result = mysql_query ($ssql_tm);
If (mysql_num_rows ($result) > 0) {
While ($data = mysql_fetch_array ($result) ) {
$msg = $data ["msg"];
$dari = $data ["dari"];
Echo "$dari$data$msg$pisah";
}
}
$ssql_rd = "UPDATE tbl_msg set status = 1";
$ssql_rd .= "WHERE dari = '$a' AND ke = '$n' ";
$result_rd = mysql_query ($ssql_rd);
}
Else if ($r == "room") {
$ssql_rm = "SELECT * FROM tbl_room ";
$ssql_rm .= "WHERE nama='$a' ";
$result = mysql_query ($ssql_rm);
If (mysql_num_rows ($result) > 0) {
While ($data = mysql_fetch_array ($result) ) {
$idroom = $data ["id"];
}
}
$ssql_rm = "SELECT * FROM tbl_room_user";
$ssql_rm .= "WHERE id_room='$idroom' ";
$result = mysql_query ($ssql_rm);
If (mysql_num_rows ($result) > 0) {
$i=1 ;

```

```

While ($data = mysql_fetch_array ($result) ) {
    $from [$i] = $data ["id_user"];
    $i++;
}
}

For ($i=1 ; $i <=count ($from) ; $i++) {
    $sql_rm ="SELECT * FROM tbl_user ";
    $sql_rm .= "WHERE id='$from[$i]' ";
    $result = mysql_query ($sql_rm);
    If(mysql_num_rows ($result) > 0) {
        While ($data = mysql_fetch_array ($result) ) {
            $name [$i] = $data ["nick"];
        }
    }
}

For ($i=1 ; $i <=count ($name) ; $i++) {
    $sql_tm = "SELECT * FROM tbl_msg";
    $sql_tm .= "WHERE dari='$name[$i]' ";
    $sql_tm .= "AND ke='$n' AND status ='0' ";
    $result = mysql_query ($sql_tm);
    If (mysql_num_rows ($result) 0) {
        While ($data= mysql_fetch_array ($result) ) {
            $msg = $data["msg"];
            $dari =$data ["dari"];
            Echo "$dari$batas$msg$pisah";
        }
    }
}

```

```

$sql_rd = "UPDATE tbl_msg set status = 1";
$sql_rd .= "WHERE dari = '$name [$i]' AND ke = '$n' ";
$result_rd = mysql_query ($sql_rd);
}
}

```

?>

<?php

```

//mengatur koneksi ke basis data
//koneksi ke database
//user name : ODBC
//password : ODBC
Mysql_connect ("localhost","root","");
Mysql_select_db ("aplydroidmessenger");
$sql = "SELECT * FROM tbl_user";
$result = mysql_query ($sql);
$row = mysql_num_rows ($result);
// Bagian ini akan dikerjakan bila
// hasil query mereturn > 0 row
If ($row != 0) {
// membuat title table
// title disimpan pada variable $show
Echo "<center>Aplidroid Messenger – 


---



```

```

$show = "";
// membuat isi table yang akan ditampilkan
// isi table disambung pada variable $show
While ($data = mysql_fetch_array ($result) ) {
$nama = $data ["name"];
$nick = $data ["nick"];
$email = $data ["email"];
$status = $data ["status"];
$show .= "<tr bgcolor=' #CCFFFF'><td align='center'>$nama</td><td
Align='center'> $nick </td><td align='center'>$email</td><td align='center'>
$status </td></tr>";
} // cetak ke layar
// yang dicetak adalah variable $show
Echo $show;
Echo "</table>";
}
Else {Echo "<tr><td>no user </td></tr>";
}
Echo "<br><br><br>";
$sql ="SELECT * FROM tbl_msg";
$result = mysql_query ($sql);
$row = mysql_num_rows ($result);
// Bagian ini hanya akan dikarjakan bila
// hasil query mereturn > 0 row
If ($row !=0) {

```

```

// membuat title table
// title disimpan pada variable Sshow
Echo "<table bgcolor='black' width='100%'><tr bgcolor='#CCFFFF'>
    <td width='15%' align='center'>DARI</td><td width='15%'
align='center'>KE</td><td
    Width='35%' align='center'>PESAN</td><td width='25%'
align='center'>TIME</td><td
    Width='25%' align='center'>STATUS</td></tr>"; $show = "";
// membuat isi table yang akan ditampilkan
// isi table disambung pada variable Sshow While ($data = mysql_fetch_array($result) ) {
$dari = $data ["dari"];
$ke = $data ["ke"];
$msg = $data ["msg"];
$timestamp = $data ["timestamp"];
$status = $data ["status"];
$show .= "<tr bgcolor=' #CCFFF'><td align='center'>$dari</td><td
Align='center'> $ke</td><td align='center'>$msg</td><td
Align='center'>$timestamp</td><td align='center'>$status </td></tr>";
}
// cetak ke layar
// yang dicetak adalah variable $show
Echo $show;
Echo "</table>";
}
Else {
Echo "<tr><td>no message</td></tr>";
}

```

}

?>

```
<?xml version="1.0" encoding="utf-8"? >
```

```
<AbsoluteLayout
```

```
    Android : id="@+id/widget 74"
```

```
    Android : layout_width="fill_parent"
```

```
    Android : layout_height="fill_parent"
```

```
    xmlns : android="http://schemas.android.com/apk/res/android"><TextView
```

```
    Android : id="@+id/widget 47"
```

```
    Android : layout_width="wrap_content"
```

```
    Android : layout_height="wrap_content"
```

```
    Android : text="welcome"
```

```
    Android : layout_x="88px"
```

```
    Android : layout_y="0px">
```

```
</TextView><Button
```

```
    Android : text="Sign In"
```

```
    Android : layout_width="86px"
```

```
    Android : layout_height="wrap_content"
```

```
    Android : layout_x="97dip"
```

```
    Android : layout_y="127dip"
```

```
    Android : id="@+id/sign_in_button"
```

```
    Android : onClick="clickHandler"></Button><Button
```

```
    Android : text="Register"
```

```
    Android : layout_width="88px"
```

```
    Android : layout_height="wrap_content"
```

```
Android : layout_x="93dip"  
Android : layout_y="52dip"  
Android : id="@+/register_button"  
Android : onClick="clickHandler"></Button>  
</AbsoluteLayout>
```

```
<? Xml version="1.0" encoding="utf-8"?>
```

```
<RevalativeLayout
```

```
Android : id="@+id/widget88"  
Android : layout_width="fill_parent"  
Android : layout_height="fill_parent"  
Xmlns : android="http://schemas.android.com/apk/res/android">
```

```
<LinearLayout
```

```
Android : id="@+id/widget80"  
Android : layout_width="fill_parent"  
Android : layout_height="wrap_content"  
Xmlns : android="http://schemas.android.com/apk/res/android"  
Android : layout_alignParentTop="true"  
Android : layout_alignParentLeft="true">
```

```
<TextView
```

```
Android : id="@+id/textview1"  
Android : layout_width="wrap_content"  
Android : layout_height="wrap_content"  
Android : text="name" : "> </TextView><EditText  
Android : id="@+id/regname"  
Android : layout_width="fill_parent"
```

```
Android : layout_height="wrap_content">
</EditText>
</LinearLayout><LinearLayout>
Android : id="@+id/widget83"
Android : layout_width="fill_parent"
Android : layout_height="wrap_content"
Xmlns : android="http://schemas.android.com/apk/res/android"
Android : layout_below="@+id/widget80"
Android : layout_alignParentLeft="true">
<TextView
Android : id="@+id/textView2"
Android : layout_width="wrap_content"
Android : layout_height="wrap_content"
Android : text="Email" : ">
</TextView><EditText
Android : id="@+id/email"
Android : layout_width="fill_parent"
Android : layout_height="wrap_content"></EditText></LinearLayout><LinearLayout
Android : id="@+id/widget85"
Android : layout_width="fill_parent"
Android : layout_height="wrap_content"
Xmlns : android=" http://schemas.android.com/apk/res/android"
Android : layout_below="@+id/widget83"
Android : layout_alignParentLeft="true">
<TextView
```

```
        Android : id="@+id/textView2"
    Android : layout_width="wrap_content"
    Android : layout_height="wrap_content"
    Android : text="nick name" : ">
</TextView><EditText
    Android : id="@+id/nickName"
    Android : layout_width="fill_parent"
    Android : layout_height="wrap_content">
</EditText></LinearLayout><LinearLayout
    Android : id="@+id/widget86"
    Android : layout_width="fill_parent"
    Android : layout_height="wrap_content"
    xmlns : android="http://schemas.android.com/apk/res/android"
    Android : layout_below="@+id/widget85"
    Android : layout_alignParentLeft="true">
<TextView
    Android : id="@+id/txtView2"
    Android : layout_width="wrap_content"
    Android : layout_height="wrap_content"
    Android : text="password" : ">
</TextView><EditText
    Android : id="@+id/password"
    Android : layout_width="fill_parent"
    Android : layout_height="wrap_content">
</EditText></LinearLayout><Button
```

```
Android : id="@+id/completeReg_button"
Android : layout_width="wrap_content"
Android : layout_height="wrap_content"
Android : text="Register"
Android : layout_below="@+id/widget86"
Android : layout_alignParentLeft="true"
Android : onClick="clickHandler"
</Button><TextView
Android : id="@+id/loginstart"
Android : layout_width="wrap_content"
Android : layout_height="wrap_content"
Android : text=""
Android : layout_below="@+id/button_login"
Android : layout_centerHorizontal="true">
</TextView>
</RelativeLayout>
```
