Enkripsi dan Dekripsi *Short Message Service* Menggunakan Algoritma *Caesar Chiper*

TUGAS AKHIR

Oleh :

Imbana Saputra 3311201093

Disusun untuk memenuhi syarat kelulusan Program Diploma III

PROGRAM STUDI TEKNIK INFORMATIKA

POLITEKNIK NEGERI BATAM

BATAM

2015

LEMBAR PENGESAHAN

Enkripsi dan Dekripsi *Short Message Service* Menggunakan Algoritma *Caesar Chiper*

Oleh:

Imbana Saputra-3311201093

Tugas Akhir ini telah diterima dan disahkan sebagai persyaratan untuk memperoleh gelar Ahli Madya

Di

PROGRAM STUDI DIPLOMA 3 TEKNIK INFORMATIKA

POLITEKNIK NEGERI BATAM

Batam,..... 2015

Disetujui oleh;

Pembimbing,

Meyti Eka Apriyani, MT

NIK 111081

LEMBAR PERNYATAAN

Dengan ini, saya:

NIM : 3311201093

Nama : Imbana Saputra

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

ENKRIPSI DAN DEKRIPSI *SHORT MESSAGE SERVICE* MENGGUNAKAN ALGORITMA C*AESAR CHIPER”*.

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 ijin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap peryataan diatas, 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 Tugas Akhir ini.

Batam, 15 Januari 2015

Imaban Saputra

3311201093

KATA PENGANTAR

Puji syukur kehadirat Tuhan YME. yang telah memberikan Rahmat dan Karunia-Nya. Sehingga penyusun dapat menyelesaikan laporan Proyek Akhir III Mobile Application dengan judul "**Enkripsi dan Dekripsi *Short Message Service* Menggunakan Algoritma *Caesar Chiper*".** Dalam penyusunan laporan Proyek Akhir ini, penulis telah dibantu oleh beberapa pihak, dan pada kesempatan ini izinkan penyusun untuk mengucapkan terima kasih kepada :

1. Tuhan Yang Maha Esa yang telah memberikan kemudahan dalam mengerjakan Tugas Akhir ini.
2. Ibu Meyti Eka Apriyani, MT, sebagai Dosen Pembimbing yang telah memberikan masukan, dukungan dan bimbingan pada proposal ini.
3. Orang tua dan teman-teman yang senantiasa memberi dorongan dalam menyelesaikan laporan ini.

Penulis menyadari bahwa laporan Tugas Akhir masih belum sempurna oleh karena itu penyusun mengharapkan saran dan kritik yang bersifat membangun demi sempurnanya proposal ini.

Penulis berharap semoga laporan ini dapat berguna dan bermanfaat bagi penyusun dan pembaca pada umumnya.

Batam, Januari 2015

Penulis

ABSTRAK

Pengiriman pesan menggunakan SMS berbasis android merupakan metode penyampaian pesan yang masih digunakan. Pesan yang dikirim ke penerima harus aman tanpa ada yang memanipulasi, karena berisikan pesan rahasia antara pengirim dan penerima. Untuk itu dibutuhkanlah ilmu kriptografi pesan yang mampu menjaga kerahasian pesan dari orang lain. Enkripsi dan deskripsi merupakan ilmu kriptografi yang tepat untuk dijadikan pegamanan pesan, agar pesan aman dari ancaman manipulasi.

Untuk melakukan proses enkripsi dan deskripsi di gunakanlah algoritma caesar chiper yang mempunyai kelebihan dalam proses enkripsi dan deskripsinya, yaitu jumlah karakter yang akan dienkripsi akan sama dengan hasil karakter setelah dienkripsi. selain itu, proses enkripsi dan deskripsi ini berbasis android. Android merupakan teknologi baru pada masa sekarang yang merndominasi dunia telepon selular diseluruh dunia. Android merupakan operasi sistem yang sifatnya *open source* sehingga memudahkan pengembang aplikasi android.

Dengan algoritma *caesar chiper* merahasiakan pesan dapat diimplemantasikan dengan baik,baik dalam proses enkripsi maupun deskripsi. Dengan bentuk pesan enkripsi berupa *monoalfabetik*

Kata kunci *: Algoritma caesar chiper, Android, deksripsi, enkripsi, monoalfabetik, SMS*

ABSTRACK

Delivery of SMS messages using an android-based message delivery method is still used. Messages sent to recipients should be safe without being manipulated, because it contains a secret message between sender and receiver.so it need the science of cryptography message that capable of maintaining the confidentiality of the messages of others. Encryption and description of the science of cryptography right to be patrolling the message, in order to secure from the threat manipulation messages.

To make the process of encryption and description in use caesar cipher algorithm has advantages in the process of encryption and description, ie the number of characters to be encrypted will be equal to the character after encrypted. in addition, the encryption process and this description based on android. Android is a new technology at the present time that rule the world of mobile telephony worldwide. Android is an operating system that is open source so as to facilitate android application developers.

With caesar cipher algorithm can be implemented with a secret message, both in the process of encryption and description. With this form of message encryption in the form monoalfabetik

Key words: *Caesar chiper algorhtym, android, decription, encryption, monoalfabetik, SMS.*

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1 Latar Belakang

Kehidupan manusia saat ini tanpa disadari dilingkupi oleh teknologi informasi salah satu unsur didalamnya adalah kriptografi. Mulai dari transaksi dimesin ATM, transaksi dibank, transaksi dengan kartu kredit, percakapan melalui telepon genggam, mengakses internet. Begitu pentingnya kriptografi untuk keamanan informasi, sehingga jika berbicara mengenai masalah keamanan informasi, maka orang tidak dapat memisahkannya dengan dunia kriptografi.

SMS atau *short message service* merupakan fitur pada handphone yang pada saat ini masih digunakan untuk mengirim dan menerima pesan singkat, walaupun teknologi telepon seluler terus mengalami perkembangan, akan tetapi SMS masih banyak digunakan salah satunya pada perangkat *smart phone* yakni android. Jika pengguna bertukar pesan dengan orang lain, maka pesan yang dikirim sampai ke pihak yang dituju dengan aman. Aman dalam hal ini berarti bahwa selama pengiriman pesan, pesan tersebut tidak dibaca oleh orang yang berhak atau dalam kriptografi *Chippertext-only attacker* seseorang yang ingin mengetahui isi pesan rahasia tersebut. Sebab pesan tersebut berisikan suatu pesan rahasia, sehingga bocorlah kerahasiaan pesan yang dikirim. Seperti *Short Message Service Center* (SMSC) yang berfungsi mencatat komunikasi yang terjadi antara pengirim dan penerima. Dengan tersimpannya SMS pada SMSC, maka seorang operator dapat memperoleh informasi atau membaca SMS di dalam SMSC tersebut.[1]

Berdasarkan permasalahan diatas penulis memilih salah satu algoritma dalam kriptografi yaitu *Caesar Chiper*. Dalam kriptografi banyak algoritma salah satunya yakni Algoritma *Caesar Chiper*, *Caesar Chiper* merupakan algoritma kriptografi klasik yang metode enkripsi dan deskripsi pesan dengan Subtitusi yang mana algortima ini tergolong algoritma yang tingkat keamananya cukup lemah

Dalam hal ini penulis akan menggunakan kode ASCII dalam proses subtitusinya, agar lebih aman dengan tujuan utamanya adalah membuat pesan singkat dengan merahasiakan isi dari pesan. Oleh karena itu, penulis memilih judul "Enkripsi dan Denkripsi *Short Message Service* Menggunakan Algoritma *Caesar Chiper*".

1.2 Rumusan Masalah

Rumusan masalah yang muncul dari latar belakang yang telah di sajikan diatas adalah sebagai berikut :

1. Bagaimana melakukan pengamanan informasi atau pesan pada media SMS berbasis Android
2. Bagaimana implementasi metode algoritma *Caesar Chiper* sehingga pesan teks SMS tersebut terjaga keamanan saat terkirim

1.3 Batasan Masalah

Batasan masalah yang dikerjakan dalam Tugas Akhir ini adalah:

1. Algoritma Kriptografi yang digunakan adalah Algoritma *Caesar Chiper*
2. Aplikasi SMS dibuat dengan menggunakan bahasa pemrograman Java dan *framework* Android SDK.
3. Aplikasi SMS dibangun untuk *smartphone* Android versi 2.3 (Gingerbread) ke atas.

1.4 Tujuan Penelitian

Dalam pembuatan laporan Tugas Akhir ini penulis mempunyai tujuan yaitu membangun aplikasi kriptografi (enkripsi dan dekripsi), sehingga pesan yang penting atau rahasia tetap terlindungi saat dikirim dan diterima dari penyadap atau pihak yang tidak bertanggung jawab.

1. Menerapkan Algoritma *Caesar Chiper* untuk pengamanan informasi dan pesan pada media SMS berbasis Android.
2. Mengimplementasikan metode Algoritama *Caesar Chiper* saat pengiriman dan pembacaan pesan SMS

1.5 Sistematika Penulisan

Sistematika penulisan yang digunakan dalam Tugas Akhir ini sebagai berikut:

BAB I PENDAHULUAN

Pendahuluan yang menjelaskan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan dan sistematika penulisan.

BAB II LANDASAN TEORI

Landasan teori yang berisi mengenai penelitian yang berkaitan, SMS (*Short Message Sevice*), perangkat pada aplikasi, Kriptografi, Algoritma Caesar, Android, Eclipse.

BAB III ANALISIS DAN PERANCANGAN

Analisis yang terdiri dari deskripsi umum sistem, kebutuhan fungsional dan kebutuhan non fungsional, karakteristik pengguna, fitur utama perangkat lunak, *Use Case,* dan perancangan yang terdiri dari pembahasan mengenai *collaboration*, *squence*, diagram kelas, dan rancangan antarmuka.

BAB IV IMPLEMENTASI DAN PENGUJIAN

Dalam bab ini membahas mengenai implementasi dan pengujian.

BAB V PENUTUP

Kesimpulan dan saran yang berisi tentang kesimpulan dari hasil pembangunan aplikasi yang dibuat pada Tugas Akhir serta saran pengembangan aplikasi mengenai penyempurnaan ide yang dapat dilakukan terhadap aplikasi yang dibuat.

BAB II
LANDASAN TEORI

2.1 Penelitian Yang Berkaitan

Beberapa penelitian yang berhubungan dengan sms enkripsi telah banyak di lakukan oleh peneliti-peneliti seperti yang telah dilakukan oleh M. ACEP SYAIFULLAH- 2014, dengan judul " MOBILE APPLICATION SMS ENKRIPSI"[1] .

Berikut adalah tabel perbedaan antara penggunaan algoritma pada *RC6* dengan algoritma *caesar chipher*.

Tabel 2. 1 Perbedaan Algoritma RC6 dengan Caesar Chiper

Perbedaan	RC6	Caesar Chiper
Sandi	hexadesimal	*Monoalfabetik*
Kunci	transposisi	Subtitusi
Algoritma	Modern	Klasik

2.2 SMS (Short Message Service)

Short Message Service disingkat dengan SMS, merupakan pesan singkat berupa teks yang dikirim dan diterima antar sesama pengguna telepon, pada awalnya pesan ini digunakan antar telepon genggam, namun dengan berkembangannya teknologi, pesan tersebut dapat dilakukan melalui komputer ataupun telepon rumah.

Secara umum sebuah telepon selular hanya dapat melakukan pengiriman satu buah paket SMS dalam satu pesan, namun dengan kemajuan teknologi yang ada sekarang, beberapa telepon selular mampu mengirimkan beberapa paket SMS dalam satu pesan. Yang dilakukan telepon selular agar dapat melakukan pengiriman beberapa paket dalam satu kali pengiriman pesan adalah melakukan konkatinasi, jadi sebenarnya hal yang dilakukan sama dengan mengirimkan

beberapa pesan hanya saja dengan melakukan konkatinasi, beberapa pesan yang disatukan tersebut dapat terlihat menjadi satu buah pesan. Dengan adanya fitur konkatinasi, sebuah SMS seolah-olah dapat mengirim pesan dengan panjang lebih dari 160 karakter (7 bit karakter) dalam satu buah pesan, namun pada fitur konkatinasi ini dibutuhkan sebuah informasi tambahan pada pesan untuk menyambungkan beberapa pesan menjadi satu buah pesan, oleh karena itu panjang satu buah pesan akan menjadi lebih kecil.

Pada sebuah aplikasi penerimaan SMS pada telepon selular dikenal nomor p*ort*, nomor *port* ini digunakan sebagai pengenal apabila terdapat dua buah atau lebih aplikasi penerimaan SMS pada sebuah telepon selular. Aplikasi penerimaan SMS tersebut akan menunggu pesan yang ditujukan pada nomor *port* tersebut. Untuk mengirimkan pesan pada *port* yang spesifik, pengirim harus menyertakan nomor *port* pada pesan yang dikirimkannya. Jika pengirim tidak menyertakan nomor *port*, seperti halnya yang dilakukan oleh aplikasi standar setiap telepon selular, maka pesan akan ditujukan ke aplikasi standar yang dimiliki oleh telepon selular atau aplikasi yang memiliki nomor *port* 0. Informasi nomor *port* tersebut dibawa bersama paket pesan yang dikirimkan oleh pengirim, oleh karena itu jika pengirim menyertakan informasi nomor *port* tujuan, maka panjang maksimal pesan yang dapat dikirimkan akan berkurang karena sebagian terpakai oleh informasi nomor *port*.[3]

2.3 Kriptografi

Kriptografi adalah suatu ilmu pengetahuan yang mempelajari beberapa teknik yang berkaitan dengan keamanan informasi, teknik-teknik yang digunakan pada umumnya menggunakan dasar pengetahuan matematika. Kriptografi bukanlah satu-satunya jalan dalam menjaga keamanan dokumen tetapi kriptografi menyediakan kumpulan teknik untuk menjaga keamanan dokumen[2].

Secara garis besar kriptografi dibagi menjadi 2 jenis, yakni kriptografi klasik dan kriptografi moderen. Perbedaan mendasar yang terdapat pada ke dua jenis tersebut adalah pada kriptografi moderen, algoritma kriptografi umumnya

beroperasi pada mode bit sedangkan pada kriptografi klasik beroperasi pada mode karakter. Teknik kriptografi moderen, secara umum dibagi menjadi 2 jenis, yaitu:

1. Algoritma kriptografi kunci simetris

   Pada algoritma kriptografi ini, kunci yang digunakan dalam proses dekripsi dan enkripsi merupakan kunci yang sama. Berdasarkan pemrosesan bit, algoritma kunci simetris dibagi menjadi dua bagian, yaitu; algoritma *block chiper* yang melakukan pemrosesan bit per-blok dan algoritma *stream chiper* yang memproses blok secara mengalir atau per-bit.

2. Algoritma kriptografi kunci publik

   Proses enkripsi dan dekripsi pada algoritma kriptografi kunci publik menggunakan kunci yang berbeda. Seperti namanya algoritma ini menggunakan kunci enkripsi yang bersifat publik atau tidak rahasia, namun menggunakan kunci dekripsi yang bersifat rahasia. Kunci dekripsi pada umumnya merupakan hasil perhitungan dari kunci enkripsi yang bukan merupakan pemetaan satu ke satu, sebuah kunci dekripsi dapat memiliki beberapa kunci enkripsi. Dalam penggunaannya, algoritma kriptografi kunci publik tidak hanya digunakan untuk menyembunyikan pesan, tetapi dapat juga digunakan untuk melakukan otentikasi dokumen

Tujuan kriptografi adalah untuk mencegah dan mendeteksi orang yang tidak bertanggung jawab melakukan hal-hal yang mengganggu seperti membaca data rahasia atau mengubah suatu data penting. Untuk tujuan itu, kriptografi menyediakan empat aspek keamanan yaitu; kerahasiaan, integritas data, otentikasi dan penyangkalan.

Algoritma kripografi melibatkan proses pengubahan pesan menjadi tersembunyi atau tidak dikenali isi dan maksudnya. Pesan yang belum diubah tersebut disebut dengan plainteks dan pesan yang telah diubah disebut dengan chiperteks. Proses pengubahan plainteks menjadi chiperteks disebut dengan enkripsi dan proses pengembalian chiperteks menjadi plainteks disebut dengan dekripsi.

Block chiper adalah suatu tipe algoritma kriptografi kunci simetri yang mengubah plainteks yang dibagi dalam blok-blok dengan panjang yang sama menjadi chiperteks yang memiliki panjang blok yang sama. Ukuran panjang blok dapat beragam bergantung kepada algoritma yang digunakan, ukuran yang sering digunakan adalah 64 bit dan menuju 128 bit. Seperti semua algoritma kunci simetri, proses enkripsi yang dilakukan akan menggunakan suatu input dari user yang disebut sebagai kunci rahasia.

Dalam melakukan perancangan *block chiper*, beberapa prinsip harus dipertimbangkan. Prinsip-prinsip tersebut yaitu:

1. Prinsip *Confusion* dan *Diffusion* dari Shannon.

   Tujuan dari prinsip *confusion* adalah untuk menyembunyikan hubungan apapun yang ada antara plainteks, chiperteks, dan kunci, sehingga dapat membuat kripanalis kesulitan dalam menemukan pola-pola pada chiperteks. Tujuan dari prinsip *diffusion* adalah menyebarkan pengaruh satu bit plainteks atau kunci ke sebanyak mungkin chiperteks, sehingga dengan berubahnya satu bit plainteks dapat mengubah chiperteks yang sulit untuk diprediksi.

2. *Iterated Chiper*

   Untuk menambah keamanan, pada algoritma-algoritma *block chiper* dilakukan iterasi pada pemrosesan setiap blok, pada setiap rotasi dari iterasi tersebut digunakan fungsi transformasi yang sama namun memakai kunci yang berbeda yang disebut dengan kunci internal. Kunci internal pada umumnya merupakan hasil dari kunci yang dimasukan oleh pengguna yang dikomputasi menggunakan suatu fungsi tertentu. Dengan adanya iterasi tersebut keamanan akan semakin terjamin, namun performansi akan berkurang karena adanya waktu lebih yang dibutuhkan untuk melakukan iterasi. *Block chiper* yang menerapkan konsep iterasi ini disebut juga dengan *iterated block chiper.*

3. Kunci Lemah

   Suatu hal yang perlu dihindari dalam melakukan perancangan algoritma kriptografi adalah kunci yang dapat menghasilkan *chiperteks* yang mirip atau serupa dengan plainteks.

Dalam bidang ilmu kriptografi terdapat algortima yang menjadi fungsi dasarnya, yaitu:

1. Enkripsi, merupakan pengamanan data yang dikirimkan agar terjaga kerahasiaannya. Pesan asli disebut plaintext, yang diubah menjadi kode-kode yang tidak dapat dimengerti. Dalam hal ini enkripsi disebut juga dengan chiper atau kode.
2. Dekripsi, merupakan kebalikan dari enkripsi. Pesan yang telah dienkripsi dikembalikan ke bentuk asalnya (teks-asli), disebut dengan dekripsi pesan. Algoritma yang digunakan untuk dekripsi tentu berbeda dengan algoritma yang digunakan untuk dekripsi.
3. Kunci, yang dimaksud di sini adalah kunci yang dipakai untuk melakukan enkripsi dan dekripsi. Kunci terbagi menjadi dua bagian yaitu kunci rahasia (*private key*) dan kunci umum (*public key*)

2.4 Algoritma Caesar Chiper

Substitusi kode yang pertama dalam dunia penyandian dikenal dengan Kode Kaisar, karena penyandian ini terjadi pada saat pemerintahan Yulius Caesar. Dengan mengganti posisi huruf awal dengan alphabet atau disebut denganalgoritma ROT3. Caesar Chiper merupakan salah satu algoritma chiper tertua dan paling diketahui dalam perkembangan ilmu kriptografi. *Caesar chiper* merupakan salah satu jenis chipper tiap huruf disubstitusi dengan huruf ketiga berikutnya dari susunan alfabet yang sama. Dalam hal ini kuncinya adalah jumlah pergeseran huruf.[2]

Teknik seperti ini disebut juga sebagai chiper abjad tunggal. Algoritma kriptografi *Caesar Chiper* sangat mudah untuk digunakan. Inti dari algoritma kriptografi ini adalah melakukan pergeseran terhadap semua karakter pada

plainteks dengannilai pergeseran yang sama. Adapun langkah-langkah yang dilakukan untuk membentuk chiperteks dengan Caesar Chiper adalah :

1. Menentukan besarnya pergeseran karakter yang digunakan dalam membentuk chiperteks ke plainteks.
2. Menukarkan karakter pada plainteks menjadi chiperteks dengan berdasarkan pada pergeseran yang telah ditentukan sebelumnya.

Misalnya diketahui bahwa pergeseran sama dengan 3, maka huruf A akan digantikan oleh huruf D, huruf B menjadi huruf E, dan seterusnya.Teknik penyandian ini termasuk sandi tersubtitusi pada setiap huruf pada plaintext digantikan oleh huruf lain yang dimiliki selisih posisi tertentu dalam alphabet. Jika pergeseran yang dilakukan sebanyak tiga kali, maka kunci untuk dekripsinya adalah 3. Pergeseran kunci yang dilakukan tergantung keinginan pengiriman pesan. Dapat saja kunci yang dipakai a = 7, b = 9, dan seterusnya. Cara kerja sandi ini dapat diilustrasikan dengan membariskan dua set alfabet. Alfabet sandi disusun dengan cara menggeser alfabet biasa ke kanan atau ke kiri dengan angka tertentu (angka ini disebut kunci). Misalnya sandi Caesar dengan kunci 3, adalah sebagai berikut:

Alfabet Biasa → ABCDEFGHIJKLMNOPQRSTUVWXYZ
Alfabet Sandi → DEFGHIJKLMNOPQRSTUVWXYZABC

Untuk menyandikan sebuah pesan, cukup mencari setiap huruf yang hendak disandikan di alfabet biasa, lalu tuliskan huruf yang sesuai pada alfabet sandi.Untuk memecahkan sandi tersebut gunakan cara sebaliknya. Contoh penyandian sebuah pesan adalah sebagai berikut.

Teks terang → kirim pasukan ke sayap kiri
Teks tersandi → NLULP SDVXNDQ NH VDBDS NLUL

2.5 Android

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar (*smartphone*) dan komputer tablet. Android awalnya dikembangkan oleh Android, Inc., dengan dukungan finansial dari Google, yang kemudian membelinya pada tahun 2005.[3]

Sistem operasi ini dirilis secara resmi pada 5 November 2007, bersamaan dengan didirikannya *Open Handset Alliance*, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler. Ponsel Android pertama mulai dijual pada bulan Oktober 2008. Dilain pihak, Google merilis kode-kode Android dibawah lisensi Apache, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler. Di dunia ini terdapat dua jenis distributor system operasi Android. Pertama yang mendapat dukungan penuh dari Google atau *Google Mail Service* (GSM) dan kedua adalah yang benar-benar bebas distributornya tanpa dukungan langsung Google atau dikenal sebagai *Open Handset Distribution* (OHD)[3].

Fitur yang tersedia di Android adalah:

- Kerangka aplikasi: itu memungkinkan penggunaan dan penghapusan komponen yang tersedia.
- Dalvik mesin virtual: mesin virtual dioptimalkan untuk perangkat mobile.
- Grafik: grafik di 2D dan grafis 3D berdasarkan pustaka OpenGL.
- SQLite: untuk penyimpanan data.
- Mendukung media: audio, video, dan berbagai format gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
- GSM, Bluetooth, EDGE, 3G, dan WiFi (hardware dependent)
- Kamera, Global Positioning System (GPS), kompas, dan *accelerometer* (tergantung hardware)

2.6 Eclipse

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (*platform-independent*).[3]

Berikut ini adalah sifat dari Eclipse:

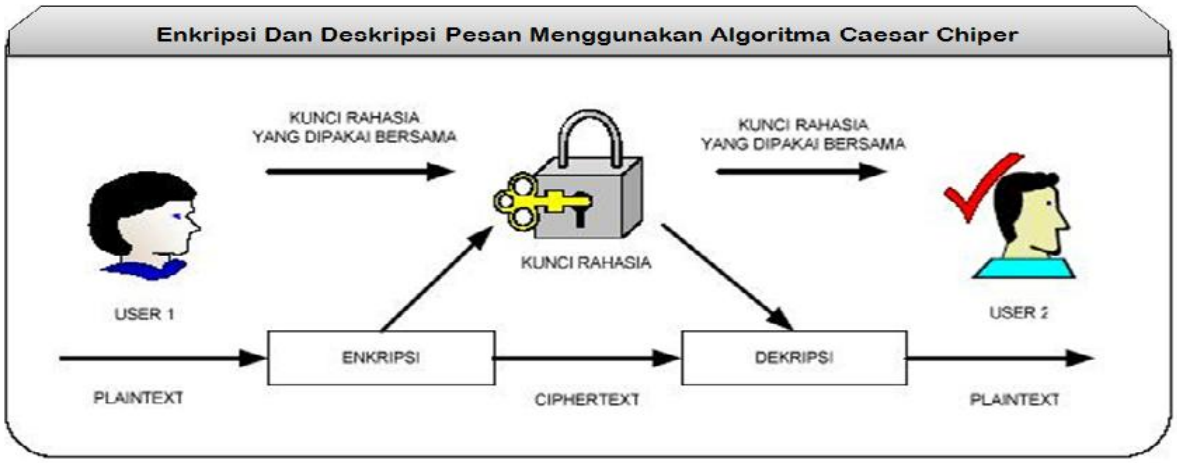
- *Multi-platform*: Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
- *Mulit-language*: Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Cobol,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

BAB III
ANALISIS DAN PERANCANGAN

3.1 Deskripsi Umum Sistem

Secara umum arsitektur desain sistem untuk Tugas Akhir ini dapat dilihat pada Gambar 3.1 berikut ini :



Gambar 3. 1 Desain Sistem

Data yang akan digunakan dalam sistem enkripsi sms menggunakan metode caesar chiper yaitu:

1. Data pesan SMS

   Data pesan SMS adalah pesan yang dimasukkan oleh pengirim yang ingin ditujukan pada penerima. Data pesan SMS ini adalah pesan yang belum terenkripsi. Data pesan ini merupakan pesan teks, namun pada awal proses enkripsi, data inilah yang akan diubah menjadi bentuk *substitusi monoalfabetik.*.

2. Data pesan SMS terenkripsi

   Data pesan SMS terenkripsi adalah data pesan SMS yang telah terenkripsi oleh algoritma *Caesar Chiper* berdasarkan kunci yang dimasukkan oleh pengguna. Data pesan SMS terenkripsi berbentuk *substitusi monoalfabetik*, oleh karena itu pada umumnya tidak dapat dibaca.

3. Data kunci enkripsi

   Data ini berasal dari pengirim. Data ini digunakan untuk melakukan proses enkripsi. Seperti halnya data pesan SMS, data ini pada awalnya berupa teks, diubah menjadi bentuk *substitusi monoalfabetik*. Pada pemakaiannya untuk proses enkripsi, data ini diubah menjadi kumpulan kunci internal.

4. Data kunci dekripsi

   Data ini berasal dari penerima, penggunaan data ini dalam proses dekripsi sama dengan data kunci enkripsi pada proses enkripsi. Jika data ini sama dengan data kunci enkripsi, maka hasil data pesan SMS keluaran yang dihasilkan akan sama dengan data pesan SMS.

5. Data pesan keluaran

   Data ini adalah hasil keluaran akhir yang didapatkan oleh penerima setelah memasukan data kunci dekripsi. Pada awalnya data ini akan didapat dalam bentuk *substitusi monoalfabetik*, namun akan diubah dalam bentuk teks agar lebih mudah untuk dibaca oleh penerima.

3.2 Analisis Spesifikasi dan Kebutuhan Perangkat Lunak

Perangkat lunak yang akan dibangun memiliki dua buah fitur utama, yaitu:

1. Melakukan enkripsi SMS pada telepon selular dengan algoritma *Caesar Chiper*. Pada perangkat lunak yang akan dibangun, pengguna harus dapat melakukan pembuatan SMS yang kemudian dapat dienkripsi dan pesan SMS yang telah terenkripsi tersebut harus dapat dikirimkan ke tujuan dengan baik oleh perangkat lunak yang akan dibangun.
2. Melakukan dekripsi dari SMS terenkripsi yang diterima oleh telepon selular dengan algoritma *Caesar Chiper*. Perangkat lunak harus dapat menerima pesan yang telah terenkripsi dan perangkat lunak juga harus dapat mendekripsi dengan baik pesan yang telah terenkripsi tersebut, jika kunci yang dimasukan benar.

3.3 Kebutuhan Fungsional

Kebutuhan fungsional perangkat lunak diantaranya:

1. Sistem memiliki kemampuan untuk dapat melakukan pengiriman pesan yang berbentuk *monoalfabetik*.
2. Sistem dapat melakukan enkripsi SMS dengan menggunakan algoritma *Caesar Chiper*.
3. Sistem harus dapat melakukan penyimpanan pesan.
4. Dalam melakukan penyimpanan pesan, pesan yang akan / sudah terkirim dengan pesan yang diterima harus dapat dibedakan.
5. Sistem harus dapat menerima pesan. Untuk dapat menerima pesan ini, sistem harus dapat berjalan terus dan dapat memberikan pemberitahuan jika pesan datang.
6. Sistem harus mampu melakukan dekripsi. Pesan yang telah terenkripsi harus dapat dikembalikan menjadi pesan semula jika masukkan kunci dari pengguna benar.
7. Sistem memiliki fasilitas untuk melakukan pemilihan properti algoritma *Caesar Chiper*.

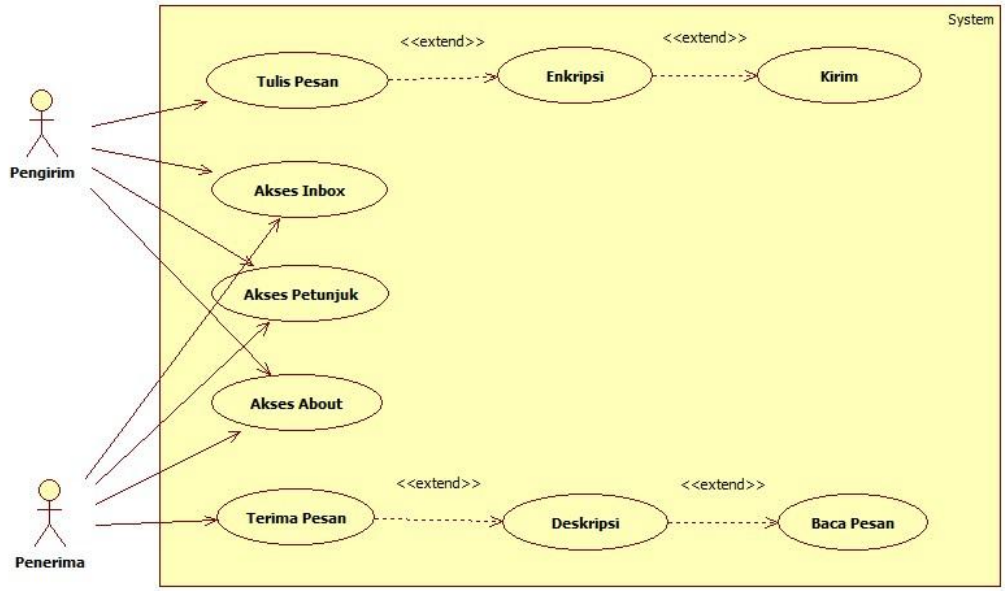
3.4 Kebuthan Non Fungsional

Selain kebutuhan fungsional diatas, aplikasi yang akan dibangun harus dapat memenuhi beberapa kebutuhan non-fungsional yang dapat membantu pengguna dalam menggunakan aplikasi dan juga dapat memudahkan dalam pengembangan lebih lanjut. Kebutuhan non-fungsional itu antara lain:

1. Sistem akan memiliki antar muka yang menarik dan juga mudah untuk dimengerti.
2. Sistem akan memiliki menu bantuan agar memudahkan dalam penggunaan.
3. Sistem mudah untuk dikembangkan lebih lanjut.

3.5 Diagram Use Case

Dalam Tugas Akhir ini, sistem akan menampilkan program berdasarkan diagram *use case* pada Gambar 3.2 berikut:



Gambar 3. 2 Use Case Diagram

3.6 Skenario Use Case

3.4.1 *Use Case* Tulis Pesan

Aktor : Pengirim pesan.

Kondisi Awal : Pengirim belum menulis pesan.

Kondisi Akhir : Pengirim sudah menulis pesan.

Skenario : Pengirim menulis pesan pada aplikasi untuk dikirim.

3.4.2 *Use Case Enkripsi Pesan*

Aktor : Pengirim pesan.

Kondisi Awal : Pengirim belum mengenkripsi pesan.

Kondisi Akhir : Pengirim sudah mengenkripsi pesan.

Skenario : Pengirim mengenkripsi pesan yang akan dikirm agar pesan teracak dan tidak dapat dibaca.

3.4.3 *Use Case Akses Inbox*

Aktor : Pengirim atau Penerima pesan.

Kondisi Awal : Belum melihat atau menerima pesan.

Kondisi Akhir : Sudah melihat atau menerima pesan.

Skenario : Pengirim atau penerima melihat membaca pesan di dalam kotak masuk.

3.4.4 *Use Case Terima Pesan*

Aktor : Penerima pesan.

Kondisi Awal : Belum menerima pesan.

Kondisi Akhir : Sudah menerima pesan.

Skenario : Penerima mendapatkan pesan dari pengirim.

3.4.5 *Use Case Dekripsi Pesan*

Aktor : Penerima pesan.

Kondisi Awal : Belum mengenkripsi pesan.

Kondisi Akhir : Sudah mengenkripsi pesan.

Skenario : Penerima mendekripsi pesan yang teracak agar dapat terbaca.

3.4.6 *Use Case Akses Petunjuk*

Aktor : Pengirim atau Penerima pesan.

Kondisi Awal : Belum melihat petunjuk aplikasi.

Kondisi Akhir : Sudah melihat petunjuk aplikasi.

Skenario : Pengirim/penerima mendapat keterangan mengenai petunjuk aplikasi bagaimana cara penggunaannya.

3.4.7 *Use Case Akses About*

Aktor : Pengirim atau Penerima pesan.

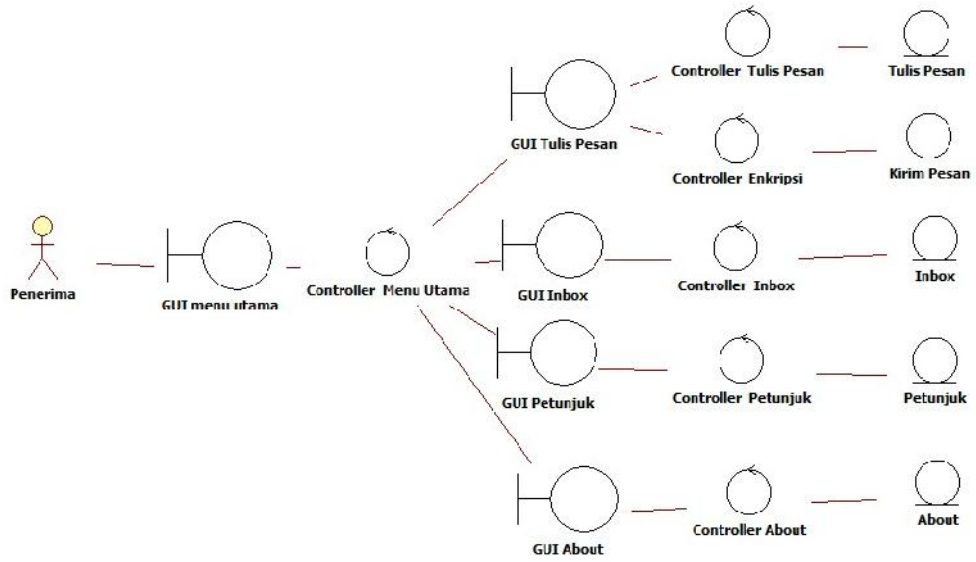
Kondisi Awal : Belum melihat about aplikasi.

Kondisi Akhir : Sudah melihat about aplikasi.

Skenario : Pengirim/penerima mendapat keterangan mengenai tentang aplikasi siapa pengembangnya

3.7 Analisis Kelas

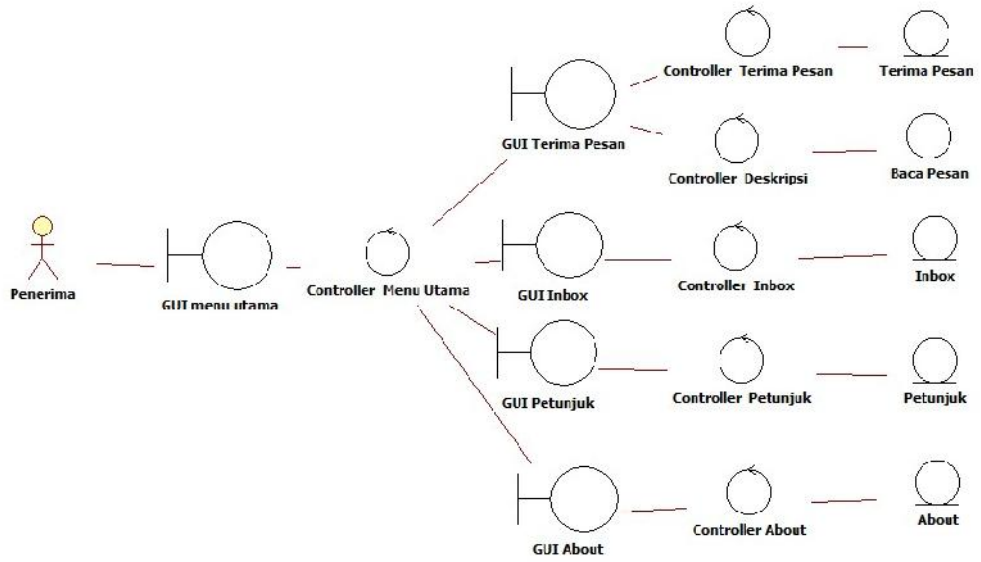
3.5.1 Analisis Kelas Pengirim



Gambar 3. 3 Analisis Diagram Pengirim

Pada analisis kelas diatas pengirim akan melihat GUI menu utama dan GUI sub menu seprti GUI tulis pesan jika pengirim ingin menulis pesan makan controller akan memproses tulis pesan, begitu juga dengan yang lainnya.

3.5.2 Analisis Kelas Penerima

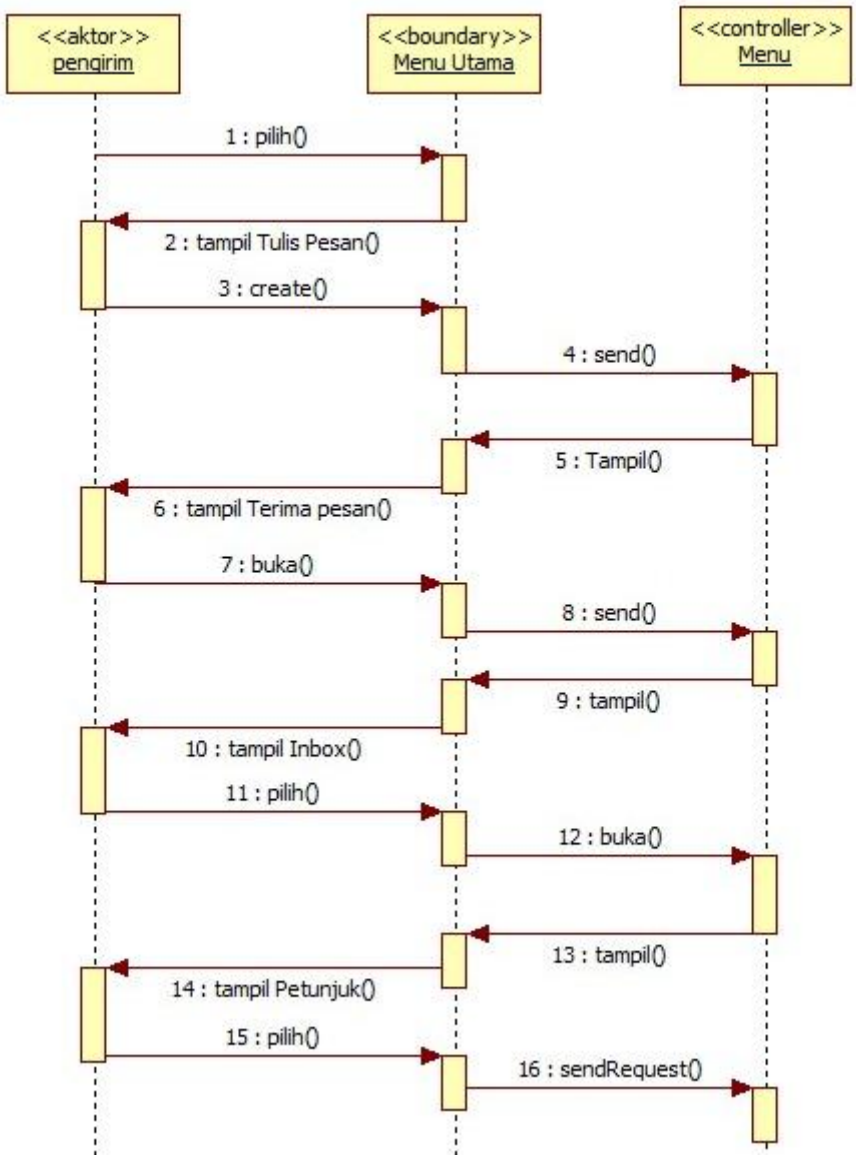


Gambar 3. 4 Analisis Diagram Pengirim

Pada analisis kelas diatas, penerima akan melihat GUI menu utama dan GUI sub menu seperti GUI terima pesan *inbox*, petunjuk, about.

3.8 Sequence Diagram

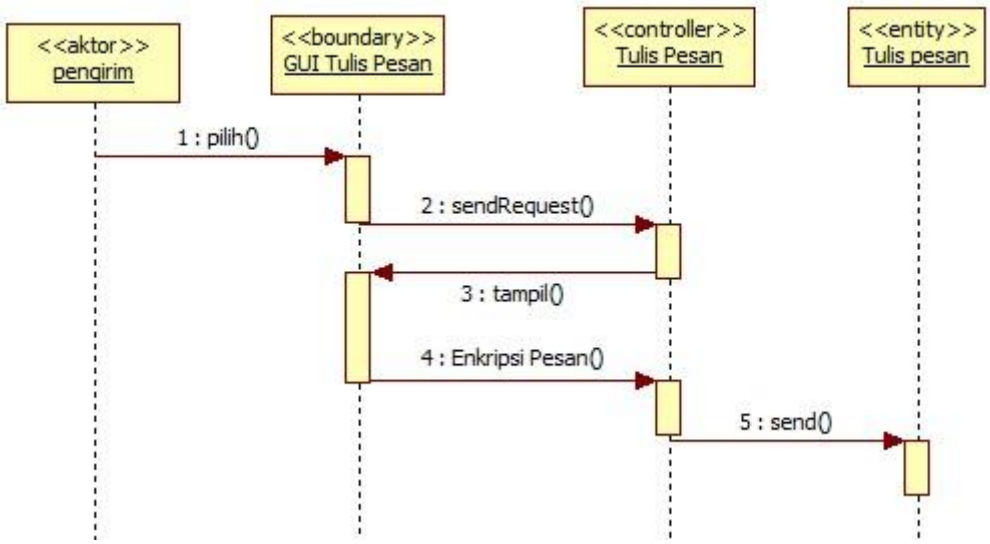
3.6.1 Sequence Diagram Menu Utama



Gambar 3. 5 Sequence Diagram Menu Utama

Sequence diagram Menu Utama, menjelaskan *button* apa saja yang tersedia di halaman utama. Pada halaman utama terdapat *button* untuk menuju ke GUI Tulis Pesan , GUI *inbox*, dan GUI Petunjuk.

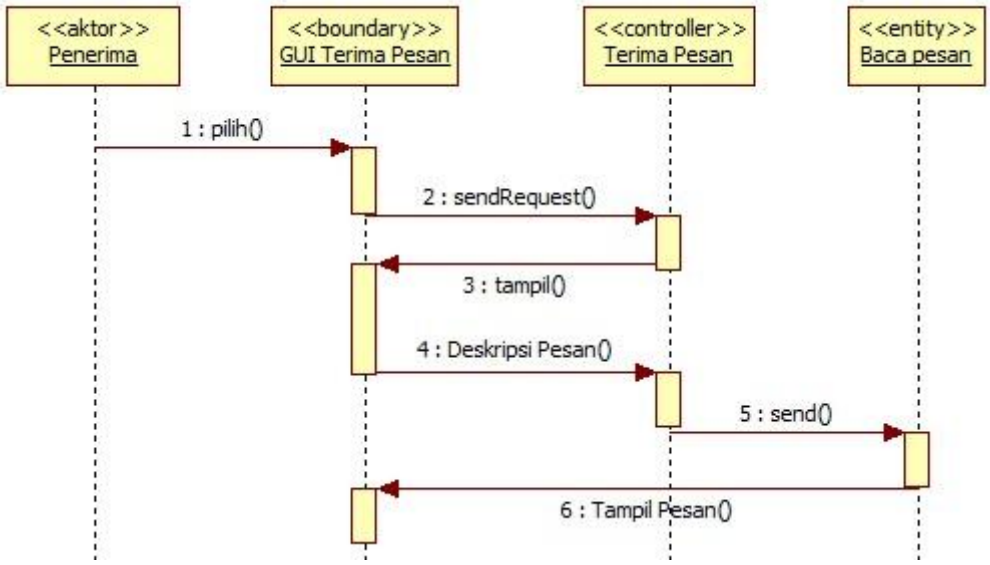
3.6.2 Sequence Diagram Tulis Pesan



Gambar 3. 6 Sequence Diagram Tulis Pesan

Pada sequence diagaram tulis pesan, menjelaskan proses menulis pesan yang didalam nya terdiri dari inputan no hanphone, kunci, pesan , dan enkripsi, selanjutnya di kirim.

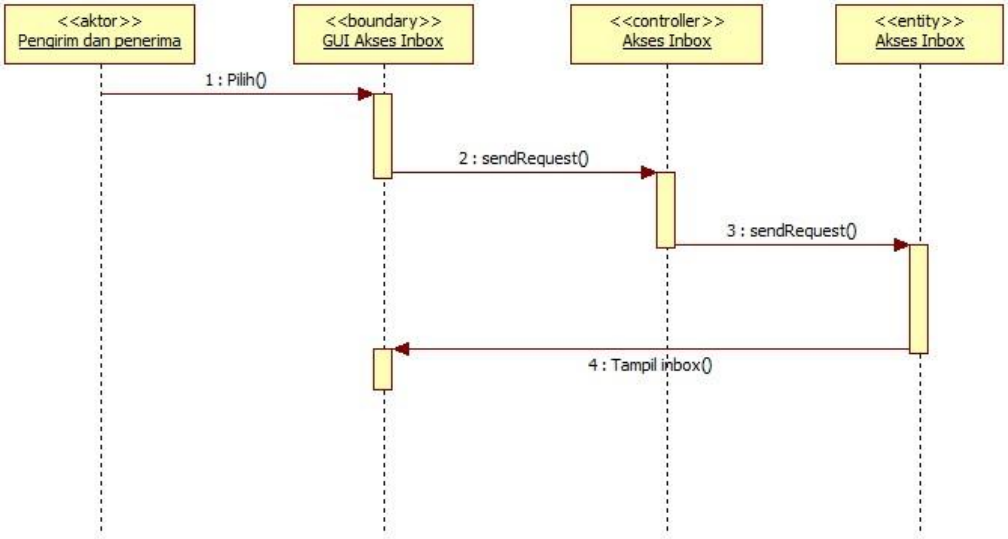
3.6.3 Sequence Diagram Terima Pesan



Gambar 3. 7 Sequence Diagram Terima Pesan

Pada sequence diagram Terima Pesan, menjelaskan proses membuka pesan yang terdiri dari memasukan kunci, dan di dekripsi, kemudian pesan dapat di baca.

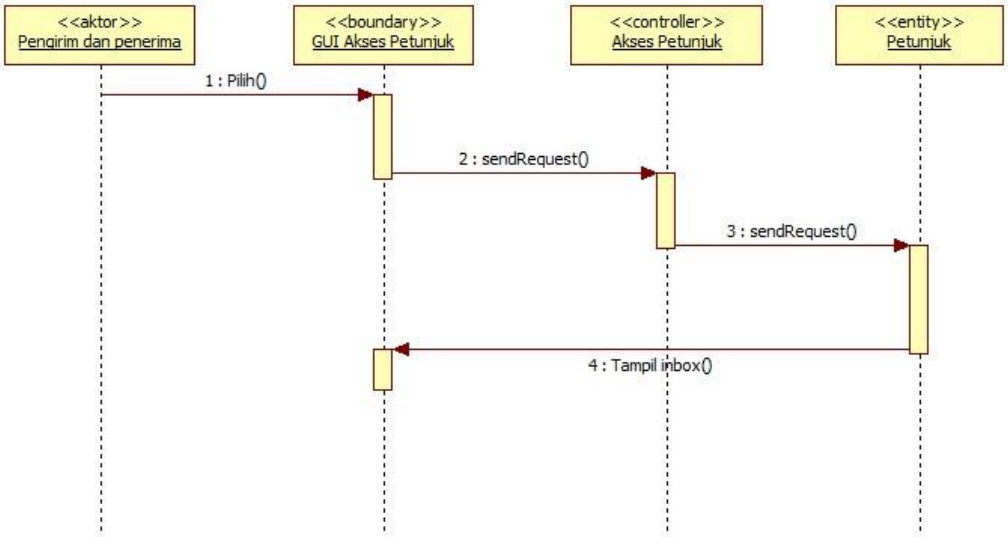
3.6.4 Sequence Diagram Akses *Inbox*



Gambar 3. 8 Sequence Diagram Akses *Inbox*

Pada sequence diagram Akses *Inbox*, pengguna dapat mengakses halaman *Inbox* untuk melihat pesan baru atau pesan lama.

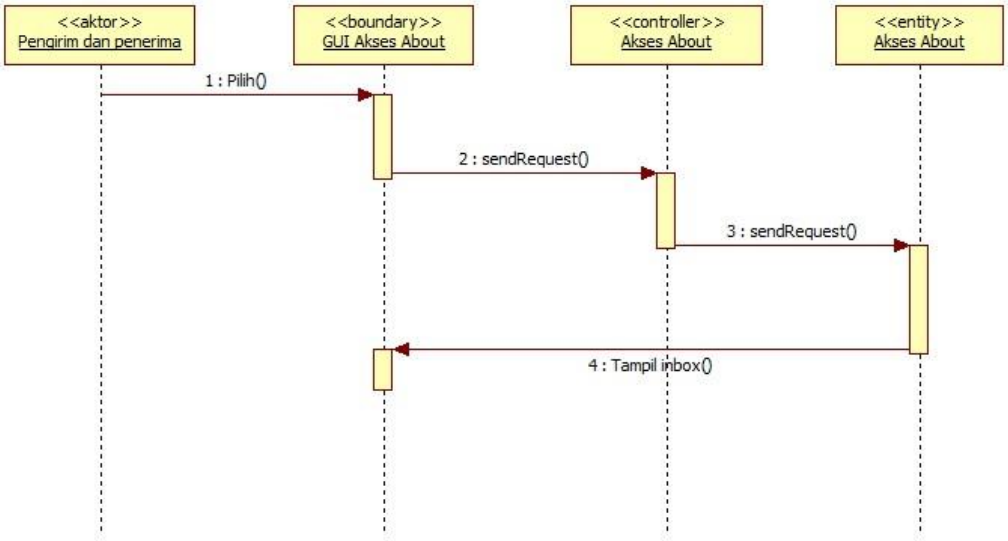
3.6.5 Sequence Diagram Akses Petunjuk



Gambar 3. 9 Sequence Diagram Akses Petunjuk

Apabila pengguna belum mengetahui secara detail cara penggunaan daripada aplikasi yang dibangun maka ada fasilitas akses petunjuk. Pada Akses Petunjuk, pengguna dapat mengakses halaman Petunjuk untuk bantuan cara menggunakan aplikasi SMS Enkripsi.

3.6.6 Sequence Diagram Akses About

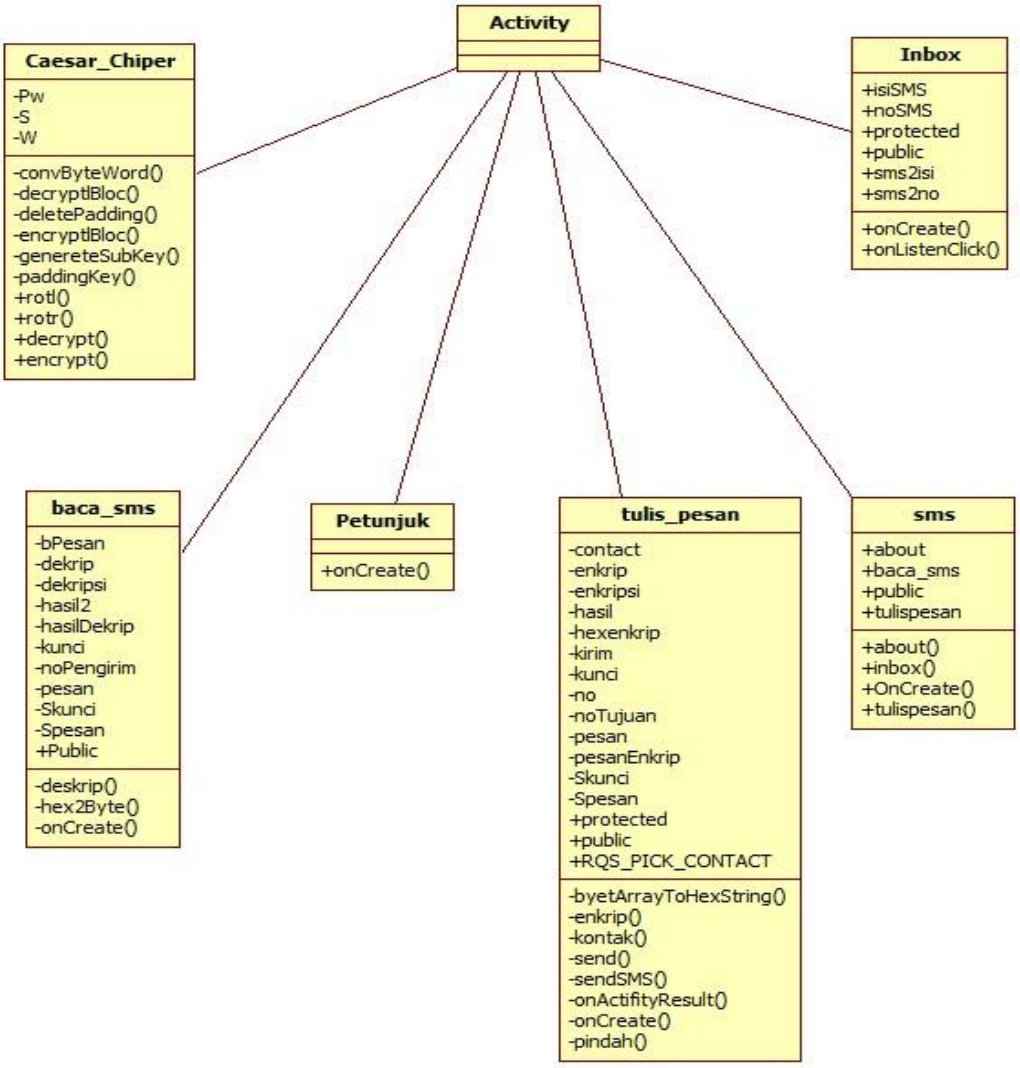


Gambar 3. 10 Sequence Diagram Akses Petunjuk

Apabila pengguna belum mengetahui siapa pengembang aplikasi yang dibangun maka ada fasilitas akses *about*.

3.9 Diagram Kelas

Class diagram merupakan diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam system atau perangkat lunak yang sedang dikembangkan dimana diagram ini memberikan gambaran (diagram statis) tentang system atau perangkat lunak dan relasi-relasi yang ada di dalamnya. Di bawah ini adalah *class diagram* yang isinya berupa *coding-coding* yang mana menjadi dasar pemrograman yang akan di bangun dalam aplikasi nantinya.



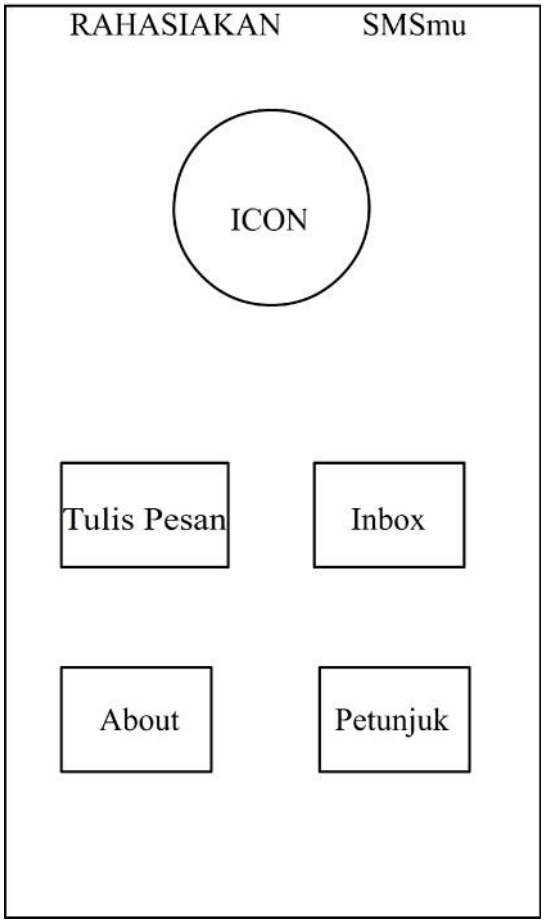
Gambar 3. 11 Diagram Kelas

3.10 Perancangan Antarmuka

Perancangan antarmuka ini digunakan sebagai tolek ukur dari sistem aplikasi yang nantinya akan dibuat. Perancangan ini adalah gambaran umum atau contoh simple dari sistem. Adapun perancangan antarmuka yang ada adalah sebagai berikut:

3.8.1 Perancangan Antarmuka Halaman Utama

Perancangan ini berisi menu-menu yang berada pada halaman muka pada aplikasi yang dibangun.



Gambar 3. 12 Perancangan Antarmuka Halaman Utama

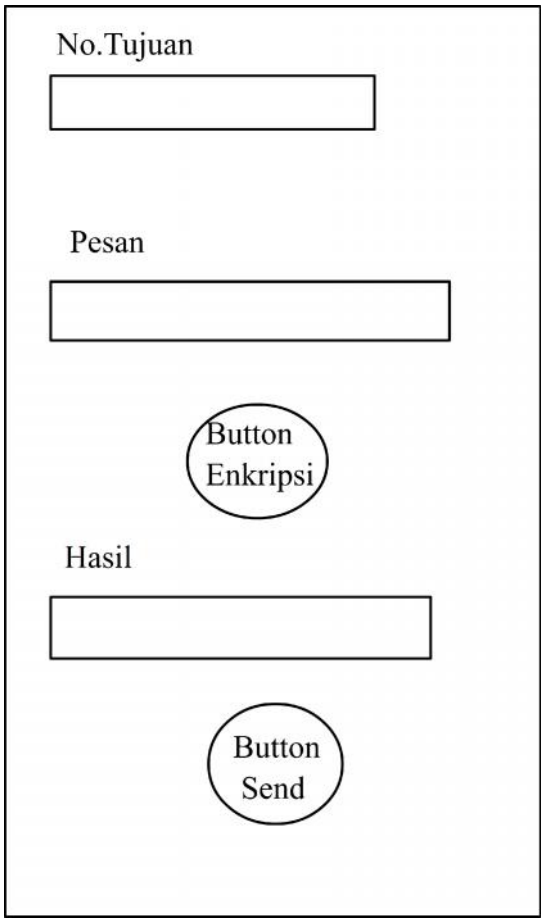
Dari gambar 3.12 diatas, berikut adalah deskripsi unsur pembangun sistem, diantaranya:

Tabel 3. 1 Deskripsi Perancangan Antarmuka Halaman Utama

Id_Objek	Jenis	Nama	Keterangan
Jtulis_pesan	jButton	Tulis Pesan	Menampilkan Halaman Tulis Pesan
JInbox	jButton	*Inbox*	Menampilkan Halaman *Inbox*
jAbout	jButton	About	Menampilkan Halaman *About*
jPetunjuk	jButton	Petunjuk	Menampilkan Halaman Petunjuk

3.8.2 Perancangan Antarmuka Tulis Pesan

Pernacangan ini berisi tentang unsur yang akan digunakan dalam menulis pesan enkripsi. Tersusun atas nomer tujuan, isi pesan dan yang terakhir adalah hasil pesan yang sudah dienkripsi.



Gambar 3. 13 Perancangan Antarmuka Tulis Pesan

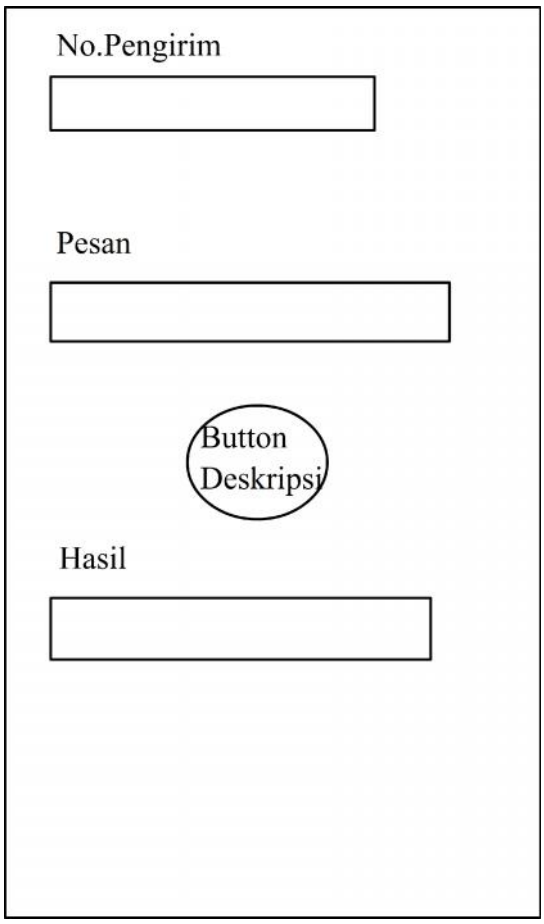
Dari perancangan diatas, juga memiliki unsur pendukung diantaranya:

Tabel 3. 2 Deskripsi Perancangan Antarmuka Tulis Pesan

Id_Objek	Jenis	Nama	Keterangan
jKontak	jButton	Kontak	Menampilkan Halaman Kontak
jNoTujuan	jText	No Tujuan	Masukan No Tujuan
jPesan	jText	Pesan	Masukan Pesan
jEnkrip	jButton	Enkripsi	Menampilkan Enkripsi
jHasil	jText	Hasil	Menampilkan Hasil Pesan Enkripsi
jKirim	jButton	Kirim	Mengirim Pesan

3.8.3 Perancangan Antarmuka *Inbox*

Perancangan antarmuka ini berisi pesan masuk yang dikirim .Berikut adalah tampilan perancangan untuk *inbox.*



Gambar 3. 14 Perancangan Antarmuka *Inbox*

Berikut adalah unsur pembangun dari perancangan antarmuka diatas.

Tabel 3. 3 Deskripsi Perancangan Antarmuka *Inbox*

Id_Objek	Jenis	Nama	Keterangan
jNoPengirim	jText	No Penngirim	Menampilkan No Pengirim
jPesan	jText	Pesan	Menampilkan Pesan
jDekripsi	jButton	Dekripsi	Menampilkan Dekripsi
jHasil	jText	Hasil	Menampilkan Hasil Dekripsi Pesan

3.8.4 Perancangan Antarmuka Petunjuk

Perancangan ini berisi tentang petunjuk langkah penggunaan aplikasi yang sudah dibangun.

Gambar 3. 15 Perancangan Antarmuka Petunjuk

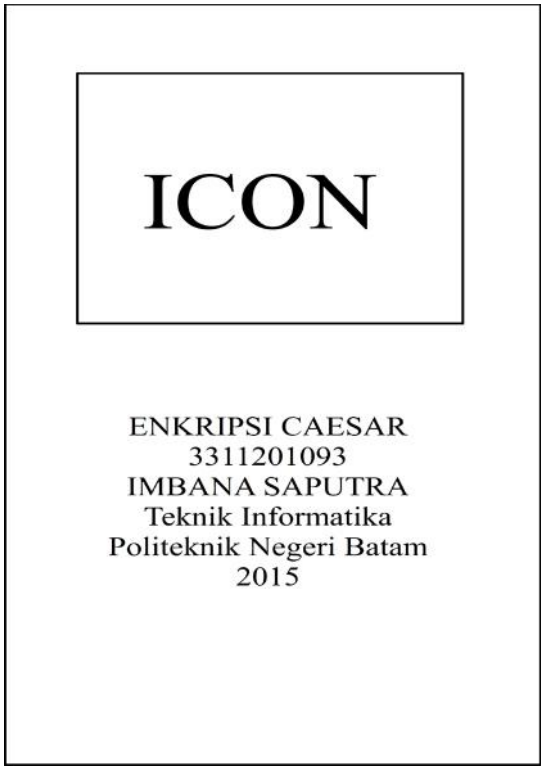
Dari tampilam diatas, berikut adalah deskripsi sistemnya:

Tabel 3. 4 Deskripsi Perancangan Antarmuka Petunjuk

Id_Objek	Jenis	Nama	Keterangan
jPetunjuk	jLabel	Petunjuk	Menampilkan Tampilan Petunjuk

3.8.5 Perancangan Antarmuka About

Perancangan ini berisi tentang about , tentang pengembang aplikasi .



Gambar 3. 16 Perancangan Antarmuka About

Dari tampilam diatas, berikut adalah deskripsi sistemnya:

Tabel 3. 5 Deskripsi Perancangan Antarmuka About

Id_Objek	Jenis	Nama	Keterangan
jAbout	jLabel	*About*	Menampilkan Tampilan About

BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Kelas

Berdasarkan perancanaan yang telah dilakukan, maka hasil implementasi kelas dan antarmuka yang dibuat secara detail dapat dilihat pada Tabel 4.1 berikut:

Tabel 4. 1 Implementasi kelas

No	***Nama Kelas***	***Nama File Fisik***	***Nama File Executable***
1	*Inbox*	*Inbox.java*	*Inbox.class*
2	*Baca_sms*	*Baca_sms.java*	*Baca_sms.class*
3	*Tulis_pesan*	*Tulis_pesan.java*	*Tulis_pesan.class*
4	*Sms*	*Sms.java*	*Sms.class*
5	*About*	*About.java*	*About.class*
6	*Petunjuk*	*Petunjuk.java*	*Petunjuk.class*
7	*Caesar*	*Caesar.java*	*Caesar.class*

Dari perancangan yang telah dilakukan, saat melakukan implementasi menghasilkan 7 kelas yaitu kelas Inbok, Baca_sms, Tulis_pesan, Sms, About, Petunjuk, *Caesar*. Dimana kelas tersebut mewakili fungsional dari aplikasi.

4.2 Implementasi Antarmuka

Berdasarkan dokumen perancangan yang telah dilakukan, maka hasil implementasi dari antarmuka yang dibuat dapat dilihat pada tabel 4.2:

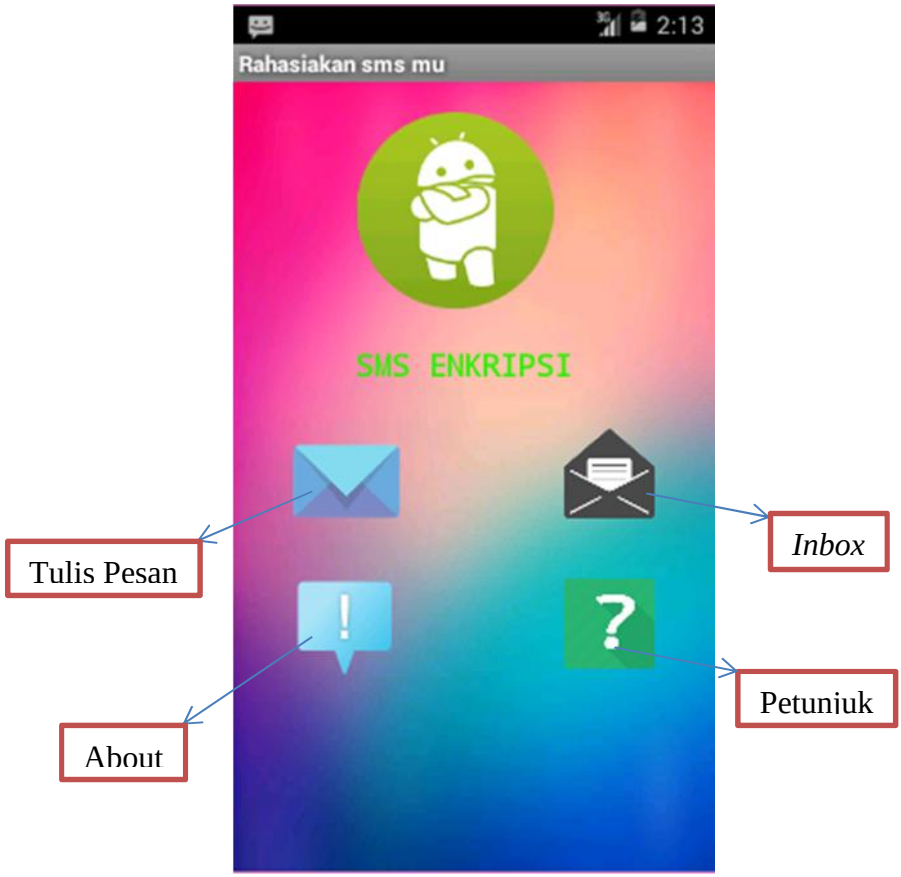
Tabel 4. 2 Implementasi Antarmuka

No	***Antarmuka***	***Nama File Fisik***	***Nama File Executable***
1	*Awal*	*Awal.xml*	*Awal.xml*
2	*Main*	*Main.xml*	*Main.xml*
3	*Baca_sms*	*Baca_sms.xml*	*Baca_sms.xml*
4	*Inbox*	*Inbox.xml*	*Inbox.xml*
5	*About*	*About.xml*	*About.xml*
6	*Petunjuk*	*Petunjuk.xml*	*Petunjuk.xml*

Pada tahap desain dan tahap implementasi tetap teradapat enam antarmuka yaitu Awal, Main, Baca_sms, *Inbox*, About dan Petunjuk

4.2.1 Implementasi Menu Utama

Pada gambar 4.1 merupakan antarmuka Menu Utama.

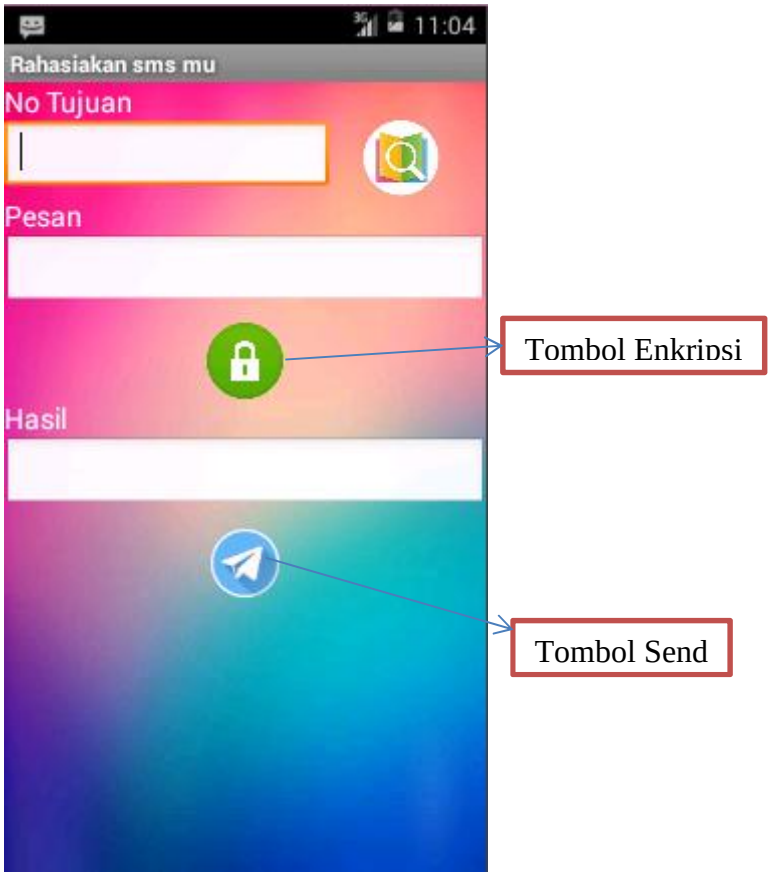


Gambar 4. 1 Antarmuka Menu Utama

Gambar 4.1 menjelaskan komponen-komponen menu utama yang terdapat pada antarmuka aplikasi SMS Enkripsi, dimana user dapat memilih menu yang diingin kan diantaraya menu tulis pesan jika user ingin menulis pesan, menu *inbox* jika user ingin melihat isi pesan, petunjuk jika user ingin mengetahui cara penggunaan aplikasi ini, dan about jika user ingin mengetahui pengembang aplikasi ini.

4.2.2 Implementasi Tulis Pesan

Pada gambar 4.2 merupakan antarmuka tulis pesan aplikasi SMS.

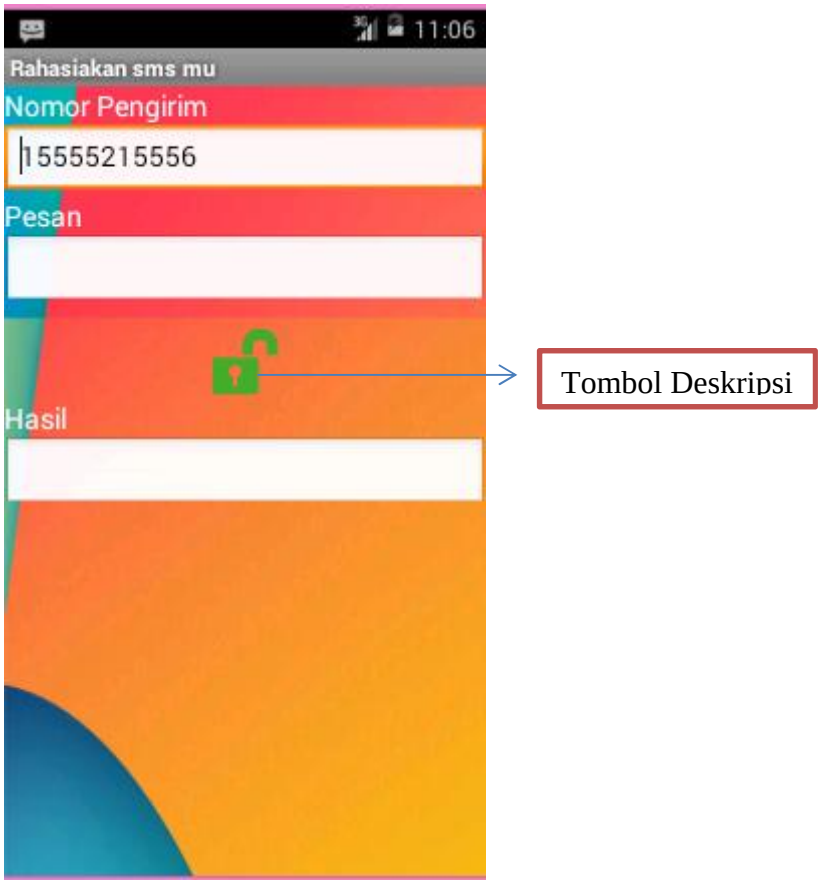


Gambar 4. 2 Antarmuka Tulis Pesan

Gambar 4.2 menjelaskan komponen-komponen yang terdapat pada antarmuka tulis pesan, dimana user memasukan nomor tujuan terlebih dahulu, kemudian menulis pesan dan jika ingin mengenkripsi tekan tombol enkripsi ,dan setelah itu kirim pesan yang terenkripsi dengan menekan tombol send.

4.2.3 Implementasi *Inbox*

Pada gambar 4.3 merupakan antarmuka *inbox* aplikasi SMS.



Gambar 4. 3 Antarmuka *Inbox*

Gambar 4.3 menjelaskan komponen-komponen yang terdapat pada antarmuka *Inbox*. Dimana user memilih pesan yang ingin dideskripsikan dengan menekan tombol untuk deskripsi.

4.2.4 Implementasi About

Pada gambar 4.4 merupakan antarmuka *about* aplikasi SMS.

Gambar 4. 4 Antarmuka About

Gambar 4.4 menjelaskan tentang keterangan mengenai aplikasi dan pengembang aplikasi pada antarmuka About.

4.2.5 Implementasi Petunjuk

Pada gambar 4.5 merupakan antarmuka petunjuk aplikasi SMS.

Gambar 4. 5 Antarmuka Petunjuk

Gambar 4.5 menjelaskan tentang bagaimana menggunakan aplikasi SMS Enkripsi pada antarmuka Petunjuk.

4.3 Hasil Pengujian

Tabel 4. 3 Hasil Pengujian

No	Kelas	Fungsi	Usecase	Skenario	Data Uji	Target	Pengujian	
							Benar	Tidak
1	Sms	Select		⁾User masuk aplikasi SMS Enkripsi	Tampil manual aplikasi SMS Enkripsi	Manual aplikasi SMS tampil	✓	
2	Tulis Pesan	Mengirim Pesan	Menulis Pesan	⁾User masukan no handphone ⁾User menulis pesan ⁾User menekan tombol send	**Data** : Hai, imbana saputra? **Enkripsi** : Lem0$mqfere$wetyxveC	Pesan berhasil dikirim ke penerima dengan pesan berbentuk enkripsi	✓	
3	*Inbox*	Menerima Pesan	Menerima Pesan	⁾User Menerima Pesan ⁾User menekan tombol kunci ⁾User membaca pesan	**Data** : Lem0$mqfere$wetyxveC **Dekripsi** : Hai, imbana saputra?	Pesan berhasil diterima sesuai dengan pesan yang dikirim	✓	

4	Baca SMS	Baca SMS	Membaca SMS	⁾ User Membaca SMS	**Data :** Hello, Apa Kabar?	Pesan berhasil dibaca oleh user	✓	
5	About	Membaca About	Melihat About	⁾ User melihat about dari aplikasi SMS	Enkripsi SMS Caesar 3311201093 Imbana Saputra Teknik Informatika Politeknik Negeri Batam 2014	User berhasil membaca about dari aplikasi SMS Enkripsi	✓	
6	Petunjuk	Membaca petunjuk	Melihat Petunjuk	⁾ User melihat Petunjuk dari aplikasi sms	ENKRIPSI 1. Pilih menu tulis pesan 2. Masukan nomor tujuan 3. Ketik pesan 4. Jika sudah tekan tombol enkripsi 5. Dan kirim pesan, menekan tombol kirim	User Berhasil Membaca Petunjuk	✓	

					DESKRIPSI 1. Buka pesan pada menu *inbox* 2. Pilih pesan yang akan dideskripsi 3. Tekan tombol deskripsi			

Batam, ... 2015

Disetujui oleh;

Penguji,

Meyti Eka Apriyani, MT

NIK 111081

4.4 Tabel Perbandingan

Setiap algortima pasti terdapat perbedaan baik dari kecepatan enkripsi dan deskrpisi maupun ukaran message, pada tabel 4.4 menunjukan perbandingan antara algortima *Ceasar Chiper* dan AES:

Tabel 4. 4 Tabel Perbandingan Algoritma *Caesar Chiper* Dan AES

Parameter	**Algortima AES**	**Algoritma *Caesar Chiper***
Berdasarkan jumlah karakter *ciphertext*	Lebih panjang : Pada proses enkripsi dengan menggunakan kriptografi AES terjadi penambahan bit tiap kelipatan 32 bit, sehingga *output* akan selalu berjumlah kelipatan dari 32 bit.	Lebih singkat : Pada proses enkripsi satu huruf didalam sebuah pesan akan diganti dengan huruf yang berada tiga (3) posisi dalam urutan alphabet huruf tersebut.
Keamanan (dilihat dari proses enkripsi pesan)	Lebih aman	Kurang aman
Output Generator	*Unpredictable* (tidak dapat diprediksi). Contoh : Karakter a = ;â?$Èè??aÖ0ÙÙô5	*Predictable* (dapat diprediksi). Contoh : karakter a = c
Kompleksitas Algoritma	Kurang Kompleks	Lebih Kompleks
Oppurtunities	Lebih Besar	Lebih Kecil

BAB V
PENUTUP

5.1 Kesimpulan

Kesimpulan yang didapat selama pengerjaan tugas akhir ini adalah sebagai berikut:

1. Pada Algoritma *Caesar Chiper* dapat diimplementasikan dengan baik untuk melakukan enkripsi SMS yang bekerja pada jaringan GSM dengan mengirimkan pesan yang berbentuk *Monoalfabetik.*
2. Sesuai dengan saran pengembang sebelumnya, Algoritma *Caesar Chiper* memiliki kelebihan yaitu jumlah karakter yang akan dienkripsi akan sama dengan jumlah karakter yang terenkripsi, dibalik itu kelemahan pada algoritma *Caesar Chiper* adalah kemanannya yang masih sedikit lemah.

5.2 Saran

Berikut adalah saran-saran yang diberikan penulis untuk pengembangan lebih lanjut:

1. Agar dapat menambah keamanan sistem seperti sistem login, karena kemungkinan sistem dapat dibuka oleh orang ketiga karena, kunci sudah ditanam didalam sistem.
2. Dapat di ujicobakan untuk algoritma enkripsi dan deskripsi yang lebih tinggi

DAFTAR PUSTAKA

Referensi yang kami gunakan untuk menyusun proposal ini diantaranya:

[1] Acep, M. Syaifullah, dkk. 2014. *MOBILE APPLICATION SMS ENKRIPSI.* Politeknik Negeri Batam

[2] Munir. Rinaldi. 2007. *Kriptografi.* Institut Teknologi Bandung

[3] Safaat, Nazarudin. 2013. *Aplikasi Berbasis Android*. Bandung: Penerbit Informatika

LAMPIRAN

A. *CAESAR CHIPER* KEY, FUNGSI ENKRIPSI DAN DESKRIPSI

Di dalam *caesar chiper*, setiap unit *plaintext* diganti dengan satu unit *chipertext.* Pada *caesar chiper*, tiap huruf disubtitusi dengan huruf ketiga berikutnya. Namun, dalam hal ini penulis menggunakan pergesaran atau subtitusi sejauh 4 huruf atau karakter dengan modulus 256. Untuk mengenkripsi dan deskripsi pesan yang disusun oleh 256 karakter ASCII.

Tabel ASCII:

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Source: www.LookupTables.com

Dengan menggunakan *Caesar Chiper* ASCII , maka pesan:

➔ Hai!

Disandikan dengan *Caesar Chiper* ASCII menjadi:

➔ Lem %

Pada contoh diatas maka didapatlah persamaan dimana

➔ C= E(P) =(P+4) mod 256

Karena ada 256 karakter di dalam ASCII. Penerima pesan mengembalikan lalgi chipertext dengan operasi kebalikan , yang secara matematis dapat dinyatakan dengan persamaan:

➔ P=D(C) =(C-4) mod 256

Chipertext pada pesan “Hai!” dapat dihitung denga persamaan C= E(P) =(P+4) mod 256:

P1=”H”=72 →c1 =E(72)=(72+4) mod 256=76=”L”

P2=”a”=97 →c2 =E(97)=(97+4) mod 256=101=”e”

P2=”i”=105 →c2 =E(105)=(105+4) mod 256=109=”m”

P2=”!”= →c2 =E(33)=(33+4) mod 256=37=”%”

Maka didapatlah c*hipertext*nya:

➔ Lem%

Chipertext diatas dapat dikembalikan menjadi *plaintext* asal dengan persamaan P=D(C) =(C-4) mod 256 menjadi:

➔ Hai!

Secara Umum, untuk pergeseran huruf sejauh *k* dalam hal ini *k* merupakan kunci emkripsi dan deskripsi, fungsi enkripsi:

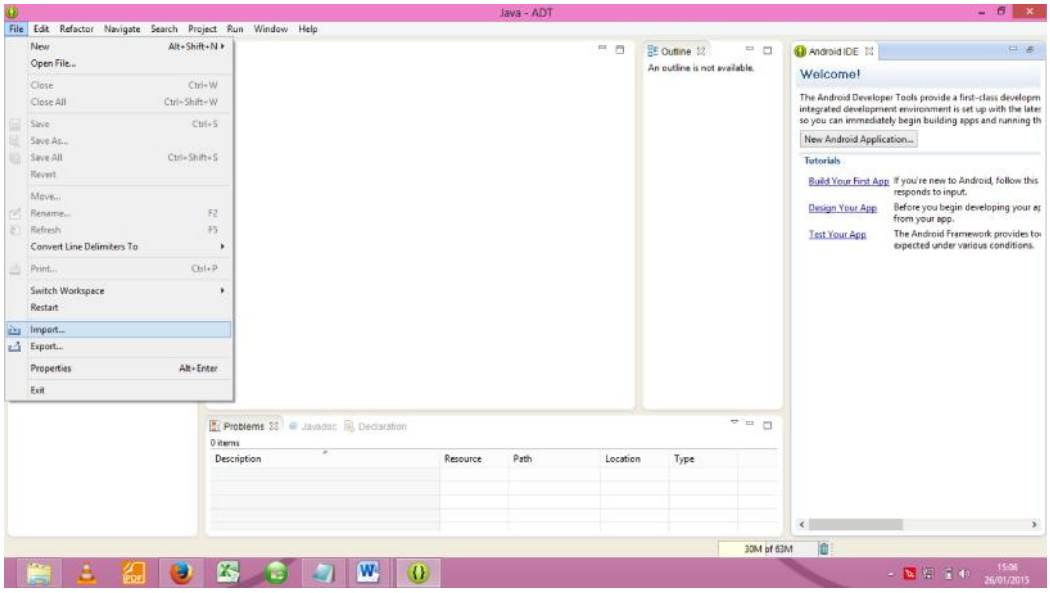
➔ C= E(P) =(P+k) mod 256

Fungsi deskripsi:

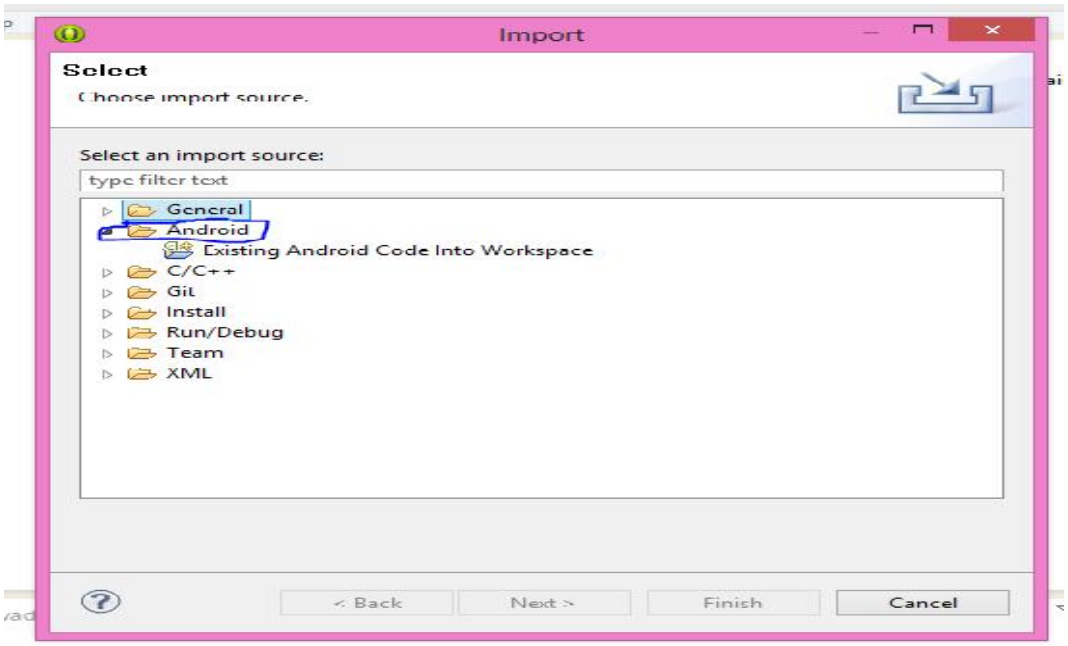
➔ P=D(C) =(C-k) mod 256

LANGKAH-LANGKAH MENGIMPORT LIBRARY

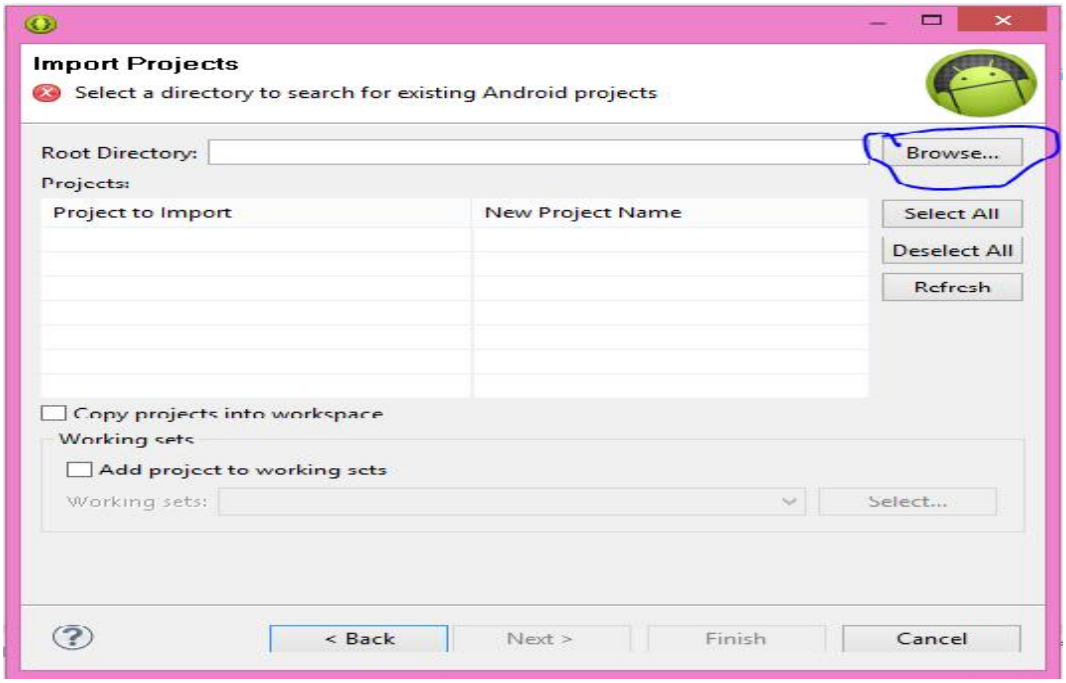
- Buka terlebih dahulu aplikasi eclipsnya, jika sudah pada menu file pilih import, tampak seperti gambar :



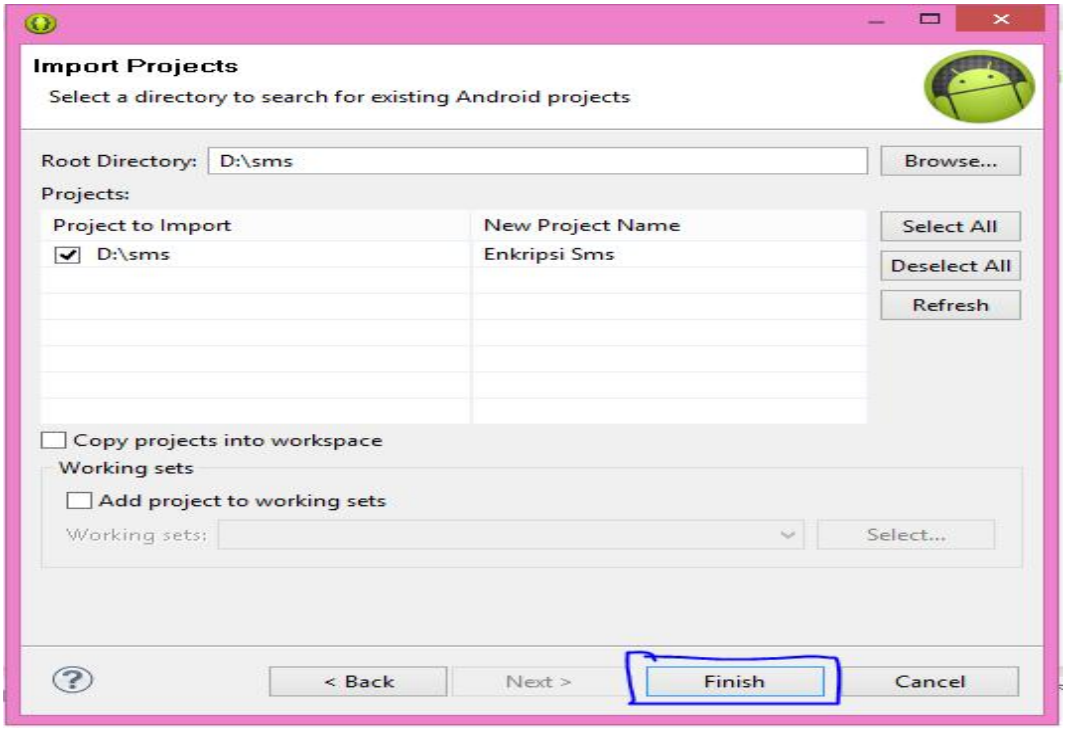
- Setelah itu akan muncul gui import, disana tersedia folder-folder, pilih folder android, dan pilih *Existing Android Code Into Wrokspace*, tampak seprti gambar:



- Akan muncul gui import project, setelah itu browse , dan pilih folder penyimpanan library:



- Jika folder sudah terpilih, pilih menu finish



- Terakhir akan muncul tampilan seperti gambar berikut, jika sudah berhasil: