

TUGAS AKHIR

**MEMBANGUN APLIKASI SMS SECURITY MENGGUNAKAN METODE
KRIPTOGRAFI BERBASIS ANDROID**



Oleh :

Irvander Grey Genggang

NIM : 11 024 011

Dosen Pembimbing

Alfrets Septy Wauran, ST.,MCSE

NIP. 19780927 200501 1 002

KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI

POLITEKNIK NEGERI MANADO

JURUSAN TEKNIK ELEKTRO

2015

"MOTTO"

“STOP DREAMING AND START DOING”



IRVANDER GREY GENGGANG

PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini :

Nama : Irvander Grey Genggang

NIM : 11 024 011

Program Studi : D-IV Teknik Informatika

Menyatakan dengan sebenarnya bahwa Tugas Akhir yang saya tulis ini benar-benar merupakan hasil karya sendiri, bukan merupakan pengambilalihan tulisan atau pemikiran orang lain. Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa sebagian atau keseluruhan Tugas Akhir ini hasil karya orang lain, saya bersedia menerima sanksi atas perbuatan tersebut.

Manado, Agustus 2015

Yang menyatakan,

Irvander Grey Genggang

ABSTRAK

Layanan komunikasi SMS (Short Message Service) merupakan layanan komunikasi favorit karena semua ponseluler memiliki layanan ini dan yang terpenting biayanya yang tergolong murah. Sayangnya aplikasi SMS bawaan ponseluler bukan merupakan jalur yang aman dalam pertukaran informasi. Pesan yang di kirimkan harus melewati SMS Center yang berfungsi mencatat komunikasi yang terjadi. SMS bawaan ponseluler jugatidak menjamin keamanan dan keutuhan isipesan tidak di ketahui, di curi, bahkan di manipulasi oleh orang lain.

Metode kriptografi di implementasikan dalam aplikasi ini agar dapat menenkripsi dan mendeskripsikan pesan yang akan di kirimkan ke penerima menggunakan kunci dengan algoritma kunci simetris, yaitu menggunakan kunci yang sama dalam proses enkripsi dan dekripsi. Algoritma kunci simetris dalam aplikasi yaitu algoritma Rivest Code 6 (RC6).

Pengujian aplikasi SMS secure di lakukan pada ponseluler android Samsung GT-S7392 yang di dalam nya menggunakan firmware Android 2.3. Pengujian terdiridari 4 bagian utamaitu pengujian enkripsi pesan, pengiriman SMS, pembacaan SMS, dan dekripsi pesan. Pengujian ini di lakukan dengan mengirimkan pesan pada nomor sendiri, sehingga dapat di baca kembali isipesan yang diterima. Selama di lakukan pengujian dengan mengirim pesan terenkripsi secara berulang-ulang dengan isipesan dan kunci yang berbeda-beda, tidak ditemukan masalah. Semua fungsi dapat berjalan dengan baik.

Untuk mengatasi celah keamanan pada layanan SMS ini di butuhkan aplikasi SMS yang mampu menenkripsi dan mendeskripsi isipesan SMS sehingga hanya orang yang memiliki kunci yang sama yang dapat membaca maknadaripesan.

Kata Kunci— Android, Dekripsi, Enkripsi, Kriptografi, SMS, Algoritma Rivest Code 6 (RC6)

HALAMAN PENGESAHAN

**“MEMBANGUN APLIKASI SMS SECURITY MENGGUNAKAN
METODE KRIPTOGRAFI BERBASIS ANDROID”**

OLEH:

Irvander Grey Genggang

11 024 011

*Tugas Akhir ini telah diterima dan disahkan sebagai persyaratan untuk
Menyelesaikan pendidikan Diploma IV Teknik Elektro Program Studi
Teknik Informatika
Politeknik Negeri Manado*

Ketua Panitia Tugas Akhir,

Fanny Jouke Doringin, ST.,MT

NIP.19670430 199203 1 003

Manado, Agustus 2015

Dosen Pembimbing,

Alfrets Septy Wauran, ST.,MCSE

NIP. 19780927 200501 1 002

Mengetahui

Ketua Jurusan Teknik Elektro,

Ir. Jusuf Luther Mappadang, MT

NIP. 19610601199003 1 002

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Tuhan Yang Maha Esa yang telah memberikan rahmat serta tuntunan-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir dengan judul “Membangun Aplikasi SMS Security Menggunakan Metode Kriptografi Berbasis Android” ini dengan baik.

Adapun Tugas Akhir ini disusun dalam rangka memenuhi persyaratan kelulusan bagi mahasiswa, khususnya Jurusan Teknik Elektro Program Studi D-IV Teknik Informatika di Politeknik Negeri Manado. Selain itu, tugas akhir ini juga dimaksudkan untuk mengembangkan pemahaman mahasiswa dibidang kriptografi dalam mengamankan informasi.

Dari awal penyusunan tugas akhir sampai telah selesai, tentunya penulis mendapatkan banyak motivasi dan dorongan positif serta sumbangsih dari berbagai pihak. Oleh karena itu, selayaknya diucapkan rasa terima kasih yang tulus, kepada :

1. Bapak Ir. Jemmy J. Rangan, MT. selaku direktur Politeknik Negeri Manado yang telah membimbing dan memimpin segala kegiatan perkuliahan.
2. Bapak Ir. Jusuf L. Mappadang, MT. selaku ketua jurusan Teknik Elektro yang telah membantu terlaksananya penulisan tugas akhir bagi mahasiswa.
3. Bapak Ir. Nikita A.E Sajangbati, MT. selaku kaprodi D-IV Teknik Informatika yang telah membantu dan mengarahkan penulisan tugas akhir bagi mahasiswa.
4. Bapak Fanny Doringin, ST.,MT. selaku ketua panitia Tugas Akhir yang telah membantu dan mengatur penyelenggaraan ujian tugas akhir.
5. Bapak Mochdar Patabo,MT. selaku sekretaris panitia Tugas Akhir yang telah membantu dan mengatur penyelenggaraan ujian tugas akhir.

6. Dosen Pembimbing, Bapak Alfrets Septy Wauran, ST., MCSE yang selalu membimbing dan mengarahkan penulis sehingga Tugas Akhir ini dapat diselesaikan dengan baik.
7. Mama tercinta yang tanpa hentinya memberikan motivasi dan masukan sehingga segalanya saya jalani dengan baik.
8. Ripkah Legi dan teman-teman yang sudah membantu memberikan saran maupun tenaga dalam pembuatan Tugas Akhir ini.
9. Tak lupa pula penulis ingin mengucapkan terima kasih kepada pihak-pihak terkait lainnya yang telah banyak membantu dalam menyelesaikan Tugas Akhir.

Tak ada gading yang tak retak demikian pula dengan Tugas Akhir ini belumlah sempurna, sehingga kritik, saran dan masukan yang konstruktif sangat di harapkan untuk penyempurnaan penulisan berikutnya.

Agustus 2015

Irvander Grey Genggang

ABSTRAK

Layanan komunikasi SMS (Short Message Service) merupakan layanan komunikasi favorit karena semua telpon seluler memiliki layanan ini dan yang terpenting biayanya yang tergolong murah. Sayangnya aplikasi SMS bawaan telpon seluler bukan merupakan jalur yang aman dalam pertukaran informasi. Pesan yang di kirimkan harus melewati SMS Center yang berfungsi mencatat komunikasi yang terjadi. SMS bawaan telpon seluler juga tidak menjamin keamanan dan keutuhan isi pesan tidak di ketahui, di curi, bahkan di manipulasi oleh orang lain.

Metode kriptografi di implementasikan dalam aplikasi ini agar dapat menenkripsi dan mendeskripsikan pesan yang akan di kirimkan ke penerima menggunakan kunci dengan algoritma kunci simetris, yaitu menggunakan kunci yang sama dalam proses enkripsi dan deskripsi. algoritma kunci simetris dalam aplikasi yaitu algoritma Rivest Code 6 (RC6).

Pengujian aplikasi SMS secure di lakukan pada telpon seluler android Samsung GT-S7392 yang di dalamnya menggunakan firmware Android 2.3. Pengujian terdiri dari 4 bagian utama yaitu pengujian enkripsi pesan, pengiriman SMS, pembacaan SMS, dan dekripsi pesan. Pengujian ini dilakukan dengan mengirimkan pesan pada nomor sendiri, sehingga dapat dibaca kembali isi pesan yang diterima. Selama dilakukan pengujian dengan mengirim pesan terenripsi secara berulang-ulang dengan isi pesan dan kunci yang berbeda-beda, tidak ditemukan masalah. Semua fungsi dapat berjalan dengan baik

Untuk mengatasi celah keamanan pada layanan SMS ini di butuhkan aplikasi SMS yang mampu mengenkripsi dan mendeskripsi isi pesan SMS sehingga hanya orang yang memiliki kunci yang sama yang dapat membaca makna dari pesan.

Kata Kunci— Android, Dekripsi, Enkripsi, Kriptografi, SMS, Algoritma Rivest Code 6 (RC6)

DAFTAR ISI

HALAMAN JUDUL	
MOTTO	
SURAT PERNYATAAN	
LEMBAR PENGESAHAN	i
KATA PENGANTAR	ii
ABSTRAK	iv
DAFTAR ISI	v
DAFTAR GAMBAR	vii
DAFTAR TABEL	ix
BAB I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
1.6 Sistematika Penulisan	4
BAB II TINJAUAN PUSTAKA	
2.1 Aplikasi.....	5
2.2 Pengertian Android.....	5
2.3 Pengertian Eclips	14
2.4 Pengertian Java	14
2.5 Kelebihan Android.....	15
2.6 Kekurangan Android	16
2.7 Asal Usul Logo Android.....	16
2.8 Bahasa Pemograman.....	16

2.9 Fitur Android SDK	17
2.10 Fitur Fitur Yang Tersedia Di Android	17
2.11 SMS (Short Message Service)	18

BAB III METODE PENELITIAN

3.1 Tinjauan Sistem	22
3.2 Metode Pengumpulan Data.....	23
3.3 Metode Perancangan Program Aplikasi	23
3.4Metode Analisis	27
3.5Metode Kriptografi	28
3.6 Algoritma RC6.....	31
3.7Diagram Use Case	33
3.8Spesifikasi Use Case.....	34
3.9Deskripsi Perancangan Pesan	35
3.10 Diagram Sequence	36

BAB IV IMPLEMENTASI DAN TESTING

4.1 Implementasi Program.....	38
4.2 Testing Program.....	43

BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan	53
5.2 Saran	53

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar 2.1 Android Beta	6
Gambar 2.2 Android 1.0 Astro.....	6
Gambar 2.3 Android 1.1 Bender	7
Gambar 2.4 Android 1.5 Cupcake.....	7
Gambar 2.5 Android 1.6 Donut.....	8
Gambar 2.6 Android 2.0/2.1 Éclair	8
Gambar 2.7 Android 2.2 Froyo (Froze Yoghurt).....	9
Gambar 2.8 Android 2.3 Gingerbread.....	10
Gambar 2.9 Android 3.0/3.1 Honeycomb	10
Gambar 2.10 Android 4.0 ICS (Ice Cream Sandwich)	11
Gambar 2.11 Android versi 4.1 (Jelly Bean)	11
Gambar 2.12 Android versi 4.2 (Jelly Bean)	12
Gambar 2.13 Android 4.4 KitKat.....	12
Gambar 2.14 Android 5.0 Lollipop.....	13
Gambar 3.1 Diagram Alir Penelitian Tugas Akhir	22
Gambar 3.2 Flowchard.....	23
Gambar 3.3 Proses Pembuatan Program	25
Gambar 3.4 Testing Enkripsi Pesan	26
Gambar 3.5 Testing Deskripsi Pesan	27
Gambar 3.6 Proses Enkripsi/Dekripsi Kriptografi Kunci	29
Gambar 3.7 Diagram Use Case.....	33
Gambar 3.8 Diagram Sequence Tulis Pesan	36
Gambar 3.9 Diagram Sequence Baca Pesan	37
Gambar 4.1 Menu Awal Aplikasi SMS Secure	38
Gambar 4.2 Menu Tulis Pesan SMS Secure	39
Gambar 4.3 Menu List Pesan SMS Secure	40
Gambar 4.4 Menu Baca Pesan SMS Secure	41

Gambar 4.5	Menu About SMS Secure.....	42
Gambar 4.6	Tampilan Awal Device 1	43
Gambar 4.7	Tampilan Menu Tulis Pesan.....	44
Gambar 4.8	Tampilan Proses Mengenkripsi Pesan.....	45
Gambar 4.9	Tampilan Proses Pengiriman Pesan	46
Gambar 5.0	Tampilan Awal Device 2	47
Gambar 5.1	Tampilan List Pesan Masuk	48
Gambar 5.2	Tampilan Pilihan Pesan	49
Gambar 5.3	Tampilan Pesan Masuk Device 2	50
Gambar 5.4	Tampilan Proses Deskripsi Pesan	51
Gambar 5.5	Tampilan Hasil Deskripsi Pesan	52

DAFTAR TABEL

Tabel 3.1 Spesifikasi Use Case Diagram Sistem	31
Tabel 3.2 Deskripsi Perancangan Class Diagram	32

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Beberapa tahun ini perkembangan teknologi komunikasi semakin pesat. Telepon seluler merupakan salah satu hasil dari perkembangan teknologi komunikasi. Telepon seluler mempermudah orang untuk berkomunikasi satu sama lain. Dengan adanya teknologi ini di dunia terasa sempit karena seseorang dapat berkomunikasi dengan orang lain yang jaraknya jauh. Di dalam telepon seluler ini ada beberapa fungsi komunikasi yang dapat digunakan antara lain telepon, video call, SMS, MMS, chatting, internet, dan lain-lain. Di antarlayanan komunikasi tersebut, layanan SMS yang menjadikan komunikasi favorit karena sudah dipastikan semua telepon seluler memiliki layanan ini dan yang paling penting adalah biayanya yang tergolong murah. Sayangnya aplikasi SMS bawaan telepon seluler bukan merupakan jalur yang aman dalam pertukaran informasi dan tidak menjamin integritas serta keamanan pesan yang dikirimkan. Pesan yang dikirimkan menggunakan aplikasi SMS bawaan telepon seluler masih berupa text terbuka yang belum terproteksi selain itu pengiriman SMS yang dilakukan tidak sampai ke penerima secara langsung, akan tetapi pengiriman SMS harus melewati 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, hal ini dapat dibuktikan dari beberapa kasus yang ditangani pihak kepolisian, kejaksaan atau KPK, dimana pihak-pihak tersebut meminta transkrip SMS ke operator untuk dijadikan bahan penyidikan di persidangan.

Aplikasi SMS bawaan telepon seluler juga tidak menjamin keamanan dan keutuhan isi pesan dan diketahui, dicuri, bahkan di manipulasi isi pesan oleh orang lain/ penyadap. Ada

beberapa risiko yang dapat mengancam keamanan pesan pada layanan SMS antara lain SMS spoofing, SMS snooping, dan SMS interception.

Dengan demikian, dibutuhkan suatu metode dan aplikasi yang dapat mempertimbangkan solusi encrypted end to end dengan melakukan enkripsi terhadap pesan SMS. Enkripsi dimaksudkan untuk melindungi dan menyamarkan informasi agar tidak terlihat oleh pihak atau orang yang tidak seharusnya. Salah satu metode enkripsi yang digunakan yaitu menggunakan algoritma enkripsi dan kunci yang dapat diubah sesuai kesepakatan.

Teknik ini disebut sebagai algoritma kunci simetris yaitu menggunakan kunci yang sama dalam proses enkripsi dan dekripsi. Algoritma kunci simetris dalam aplikasi yaitu algoritma Rivest Code 6 (RC6).

Aplikasi yang akan dibangun merupakan aplikasi yang di terapkan pada ponseluler yang bersifat operasi android dan memiliki fungsi untuk melakukan enkripsi dan dekripsi. Aplikasi dapat melakukan pengiriman dan menerima pesan.

Dengan adanya sistem keamanan ini, pesan yang bersifat personal atau rahasia dapat tersampaikan secara aman.

Sebelumnya, pada tahun 2012, Sdr. Arief Rahmantod dalam penelitiannya untuk Tugas Akhir program studi S1 Teknik Informatika dengan judul "Secure Message Dengan Algoritma RC-6". Telah membuat suatu aplikasi yang dapat mengirim dan menerima pesan yang terenkripsi.

Tapi masih banyak kekurangan pada fitur aplikasinya seperti menghapus dan menerima pesan. Untuk itu penulis mengembangkan penelitian dari Sdr. Arief Rahmantod dengan maksud untuk melengkapi fitur-fitur pada aplikasi.

1.2 Perumusan Masalah

Berdasarkan latar belakang permasalahan yang telah diuraikan di atas, maka rumusan masalah dalam penelitian ini adalah, sebagai berikut ;

- 1.2.1. Bagaimanamembangunaplikasi yang dapatmenjagakeutuhan, kemanandankerahasianisipesan ?
- 1.2.2. Bagaimanamembangunaplikasi yang dapatmengirimdanmenerimapesan ?
- 1.2.3. Bagaimanamembangunaplikasi SMS yang dapatmengkripsidandeskripsipesandenganalgoritmakunci RC6 di android ?

1.3 BatasanMasalah

Untuk membatasi masalah yang diangkat agar tidak menyimpang dari topik yang akandibahas, makapenulis menekankan masalah penelitianyang dibahas adalahterbataspada ;

- 1.3.1. Aplikasiinihanyabisa di jalankan di android.
- 1.3.2. Aplikasiiniuntukmengirimdanmenerimapesan.
- 1.3.3. Kunciuntukmenenkripsidanmendeskripsiharussama.
- 1.3.4. Pesanhanyadapat di deskripsikandenganmenggunakanaplikasiini.

1.4 TujuanPenelitian

Adapuntujuan yang ingindicapaimelaluipenelitianiniadalah, sebagaiberikut;

- 1.4.1. Aplikasidapatmengirimdanmenerimapesan
- 1.4.2. Aplikasidapatmengkripsikanpesandengankuncialgoritma RC 6
- 1.4.3. Aplikasitidakdapat di bacaoleh orang yang bukanseharusnya.

1.5 ManfaatPenelitian

Setelahsemuatujuan yang disebutkan di atasdapat tercapai, makamanfaat yang bisadiperolehmelaluipenelitianiniadalah, sebagaiberikut :

- 1.5.1. Pesantidakdapat di ketahuioleh orang lain.
- 1.5.2. Pesantidakdapat di manipulasioleh orang lain.

1.5.3. Berguna bagi pengguna yang ingin mengamankan pesan yang dikirimkan.

1.6 Sistematika Penulisan

Penulisan laporan penelitian tugas akhir ini sistematikanya akan dibuat sebagai berikut :

Bab 1, **PENDAHULUAN** yang terdiri atas Latar Belakang, Rumusan Permasalahan, Tujuan, Batasan Masalah, dan manfaat Penelitian.

Bab 2, **TINJAUAN PUSTAKA** pada bagian ini akan diuraikan kerangka teoritis tentang pembuatan aplikasi SMS security.

Bab 3, **METODE PENELITIAN**, pada bagian ini akan di jelaskan metode- metode yang di gunakan dalam membangun aplikasi.

Bab 4, **IMPLEMENTASI DAN TESTING**, pada bagian ini akan menjelaskan implementasi dan testing dari pembuatan aplikasi SMS security dengan metode kriptografi berbasis android.

Bab 5, **PENUTUP**, berisi kesimpulan yang dibuat penulis selama melakukan penelitian tugas akhir ini, kemudiannya dalam pembuatan program aplikasi dan uji coba yang jika terdapat beberapa kelemahan ataupun kekurangan dari program aplikasi ini akan dituliskan dalam bagian Saran untuk pengembangan program aplikasi ini di kemudian hari

BAB II

TINJAUAN PUSTAKA

2.1 Aplikasi

Aplikasi atau juga disebut program aplikasi adalah program yang dibuat oleh pemakai yang ditujukan hanya untuk melakukan suatu tugas khusus.

2.2 Pengertian Android

Android merupakan sebuah system operasi telpon seluler komputer tablet layar sentuh (touch screen) yang berbasis linux. Namun seiring perkembangannya android berubah menjadi platform yang begitu cepat dalam melakukan inovasi. Android awalnya dikembangkan oleh android inc, dengan dukungan finansial dari google, yang kemudian membelinya pada tahun 2005. Sistem operasi ini diliris resmi pada tahun 2007 bersamaan dengan didirikannya Open hardset Alliance, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak dan telekomunikasi yang bertujuan untuk memajukan standart terbuka perangkat seluler. Android, inc didirikan di Palo Alto California, pada bulan oktober 2003 oleh Andy Rubin (pendiri Danger), Rich Miner (pendiri Wildfire Communications, inc), Nick Sears (mantan VP T-Mobile), dan Christ White (kepala desain dan pengembangan antarmuka WebTV) untuk mengembangkan perangkat seluler pintar yang lebih sadar akan lokasi dan preferensi penggunanya. Berikut ini adalah ulasan singkat mengenai perkembangan tipe android dari waktu ke waktu :

2.2.1 Android Beta

Pertama kali dirilis pada 5 November 2007, kemudian pada 12 November 2007 Software Development Kit (SDK) dirilis oleh Google.



Gambar 2.1 Android Beta

2.2.2 Android 1.0 Astro

Pertama kali dirilis pada 23 September 2008. Sebenarnya Android versi pertama ini akan dinamai dengan nama “Astro” tapi karena alasan hak cipta dan trademark nama “Astro” tidak jadi disematkan pada versi pertama dari OS Android ini. HTC Dream adalah ponsel pertama yang menggunakan OS ini.



Gambar 2.2 Android 1.0 Astro

2.2.3 Android 1.1 Bender

Pertama kali dirilis pada 9 Februari 2009. Versi Android kedua ini juga mengalami masalah penamaan yang sama dengan versi pertamanya. Pada awalnya Android ini akan diberi nama “Bender” akan tetapi karena alasan melanggar trademark, nama “Bender” tidak jadi disematkan pada versi Android ini. Awalnya versi OS Android ini dirilis untuk perangkat T-Mobile G1 saja. Versi ini merupakan update untuk memperbaiki beberapa bugs, mengganti API dan

menambahkan beberapa fitur. Berikut penampakan home screen OS Android v.1.1.



Gambar 2.3 Android 1.1 Bender

2.2.4 Android 1.5 Cupcake

Pertama kali dirilis pada 30 April 2009. Nah, mulai versi Android ini penamaan menggunakan nama makan pencuci mulut (dessert) mulai digunakan, karena ini merupakan versi yang ketiga maka penamaan diawali dengan huruf “C” dan jadilah “Cupcake” menjadi nama resmi dari versi OS Android ketiga ini. OS ini berbasiskan pada kernel Linux 2.6.27 dan menambahkan beberapa update serta UI baru dari versi Android sebelumnya. Mulai terdapat “widget” yang dapat dibesar kecilkan. Kemudian ditambah kemampuan untuk meng-upload video dan gambar ke Youtube dan Picasa.



Gambar 2.4 Android 1.5 Cupcake

2.2.5 Android 1.6 Donut

Dirilis pertama kali pada 15 September 2009. Terdapat peningkatan pada fitur pencarian dan UI yang lebih user friendly. Pada versi ini juga sudah mendukung teknologi CDMA/EVDO, 802.1x, VPNs. Kemudian support layar dengan resolusi WVGA. Berikut penampakan Android v1.6 Donut.



Gambar 2.5 Android 1.6 Donut

2.2.6 Android 2.0/2.1 Éclair

Dirilis pertama kali pada 9 Desember 2009. Terjadi penambahan fitur untuk 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. Beberapa versi updatenya antara Android v.2.0 kemudian v2.0.2 dan terakhir v.2.1.



Gambar 2.6 Android 2.0/2.1 Éclair

2.2.7 Android 2.2 Froyo (Froze Yoghurt)

Dirilis pertamakali pada 20 Mei 2010 pada smartphone Google Nexus One. Pada versi ini sudah support terhadap Adobe Flash Player 10.1. Peningkatan pada kecepatan membuka dan menutup aplikasi, serta penggunaan SD Card sebagai tempat penyimpanan aplikasi. Ketika Android Froyo hadir mulai muncul banyak diskusi yang membahas mengenai persaingan antara Android dengan iOS yang akan semakin ketat di masa yang akan datang. Beberapa versi update yang dirilis antara lain Android v.2.2.1 hingga v.2.2.3.



Gambar 2.7 Android 2.2 Froyo (Froze Yoghurt)

2.2.8 Android 2.3 Gingerbread

Pertama kali diperkenalkan pada 6 Desember 2010. Terjadi banyak peningkatan pada versi Android yang satu ini dibandingkan dengan versi sebelumnya. Dirancang untuk memaksimalkan kemampuan aplikasi dan game. Serta mulai digunakannya Near Field Communication (NFC). Perbaikan terhadap dukungan layar resolusi WXGA dan di atasnya. Beberapa versi update yang dirilis antara lain v.2.3.3 hingga v.2.3.7. Sampai saat ini Android Gingerbread merupakan versi Android yang memiliki pengguna terbanyak dibandingkan dengan seri Android lainnya, yaitu mencapai 65% dari seluruh versi Android yang dirilis.



Gambar2.8 Android 2.3 Gingerbread

2.2.9 Android 3.0/3.1 Honeycomb

Pertama kali diperkenalkan pada 22 Februari 2011 dan Motorola Xoom adalah yang pertama kali menggunakannya. Android versi ini merupakan OS yang didesain khusus untuk pengoptimalan penggunaan pada tablet PC.



Gambar 2.9 Android 3.0/3.1 Honeycomb

2.2.10 Android 4.0 ICS (Ice Cream Sandwich)

Pertama kali dirilis pada 19 Oktober 2011. Smartphone yang pertama kali menggunakan OS Android ini adalah Samsung Galaxy Nexus. Secara teori semua perangkat seluler yang menggunakan versi Android sebelumnya, Gingerbread, dapat di-update ke Android Ice Cream Sandwich.



Gambar 2.10 Android 4.0 ICS (Ice Cream Sandwich)

2.2.11 Android versi 4.1 (Jelly Bean)

Android Jelly Bean yang diluncurkan pada acara Google I/O lalu membawa sejumlah keunggulan dan fitur baru. Penambahan baru diantaranya meningkatkan input keyboard, desain baru fitur pencarian, UI yang baru dan pencarian melalui Voice Search yang lebih cepat. Tak ketinggalan Google Now juga menjadi bagian yang diperbarui. Google Now memberikan informasi yang tepat pada waktu yang tepat pula. Salah satu kemampuannya adalah dapat mengetahui informasi cuaca, lalu-lintas, ataupun hasil pertandingan olahraga. Sistem operasi Android Jelly Bean 4.1 muncul pertama kali dalam produk tablet Asus, yakni Google Nexus 7.



Gambar 2.11 Android versi 4.1 (Jelly Bean)

2.2.12 Android versi 4.2 (Jelly Bean)

Fitur photo sphere untuk panorama, daydream sebagai screensaver, power control, lock screen widget, menjalankan banyak user (dalam tablet saja), widget terbaru.

Android 4.2 Pertama kali dikenalkan melalui LG Google Nexus 4



Gambar 2.12 Android versi 4.2 (Jelly Bean)

2.2.13 Android 4.4 KitKat

Pada dasarnya adalah versi *update* (pembaruan) dari Android versi 4.3 Jelly Bean. Dengan begitu, di dalam sistem operasi Android KitKat pastinya sudah terdapat sejumlah pembaruan dan penyempurnaan dari sistem operasi Android generasi sebelumnya



Gambar 2.13 Android 4.4 KitKat

2.2.14 Android Lollipop

Android Lollipop adalah versi stabil terbaru dari sistem operasi Android yang dikembangkan oleh Google, yang pada saat ini mencakup versi antara 5.0 dan 5.1. Diresmikan pada 25 Juni 2014 saat Google I / O, dan tersedia secara resmi melalui *over-the-air* (OTA) update pada tanggal 12 November 2014, untuk memilih perangkat yang menjalankan distribusi Android dilayani oleh Google (seperti perangkat Nexus dan Google Play edition). Kode sumbernya dibuat tersedia pada 3 November 2014.

Salah satu perubahan yang paling menonjol dalam rilis Lollipop adalah user interface yang didesain ulang dan dibangun dengan yang dalam bahasa desain disebut sebagai "material design". Perubahan lain termasuk perbaikan pemberitahuan, yang dapat diakses dari lockscreen dan ditampilkan pada banner di bagian atas screen. Google juga membuat perubahan internal untuk platform, dengan Android Runtime (ART) secara resmi menggantikan Dalvik untuk meningkatkan kinerja aplikasi, dan dengan perubahan yang ditujukan untuk meningkatkan dan mengoptimalkan penggunaan baterai, yang dikenal secara internal sebagai Project Volta



Android 5.0, Lollipop

Gambar 2.14 Android 5.0 Lollipop

2.3. Pengertian 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:

2.3.1 Multi-platform: Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.

2.3.2 Multi-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.

2.3.3 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*.

2.4 Pengertian Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam. Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal. Aplikasi-aplikasi berbasis java umumnya dikompilasi ke dalam p-code (bytecode) dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM). Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (*general purpose*), dan secara khusus didisain untuk

memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan aplikasi java mampu berjalan di beberapa platform sistem operasi yang berbeda, java dikenal pula dengan slogannya, "Tulis sekali, jalankan di mana pun". Saat ini java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis web.

2.5 Kelebihan Android

Akses mudah ke market Android. Dengan Google play store saat ini anda bisa mendownload berbagai aplikasi dan game gratis untuk ponsel android anda. Tersedia juga aplikasi dan game premium yang berbayar. Menggunakan multi-touch pada layar

Bersifat terbuka. Karena berbasis Linux yang open source, Android bebas dikembangkan oleh siapa saja. Fasilitas penuh USB. Anda bisa mengisi baterai, mass storage, diskdrive, dan USB tethering. Semua dalam satu USB. Merakyat. Dikatakan merakyat karena Android punya banyak produsen mulai dari hp China sampai yang terbaru Galaxy S III. Jadi dapat dinikmati semua golongan. Berbeda dengan iOS yang terbatas untuk iPhone dan penikmatnya pun hanya golongan yang berduit karena harganya yang cukup mahal. Kemudahan dalam hal notifikasi. Dalam hal pemberitahuan, Android menyediakan fasilitas yang bisa dibilang lengkap seperti SMS, Email, Artikel terbaru dari RSS Reader, bahkan dapat disinkronisasi dengan jejaring sosial seperti Facebook dan Twitter. Instal ROM modifikasi. Kadang kita mendapati ROM yang tidak resmi. Maksudnya ROM yang dirilis tidak sesuai dengan spesifikasi ponsel yang kita miliki. Jalan terakhir adalah dengan modifikasi. Banyak custom ROM yang bisa anda dapat untuk ponsel anda dan dijamin tidak akan membahayakan perangkat anda. Google mania. Anda penggemar Google? Ya, anda dapat menikmati semua fasilitas Google mulai dari Gmail sampai Google Reader dengan satu sistem operasi yaitu Android.

2.6 Kekurangan Android

Terhubung dengan internet. Bisa dibilang OS ini memerlukan koneksi internet yang aktif. Setidaknya GPRS agar perangkat siap online sesuai dengan kebutuhan. Iklan. Ya, yang mudah dan gratis akan selalu diboncengi iklan. Perusahaan perangkat kadang lambat mengeluarkan update resmi untuk Android milik anda. Meski tidak ada perbedaan mencolok dalam hal user interface. Boros pulsa. Karena Android memerlukan banyak koneksi internet. Dipastikan juga akan menguras pulsa anda untuk melakukan koneksi internet. Kecuali dengan hotspot. Boros Baterai. Banyak aktifitas, banyak tenaga. Seperti Android yang memerlukan kapasitas baterai besar guna mendukung penggunaan fitur yang ada di dalamnya secara maksimal. Sebagai penyedia layanan langsung, kadang pengguna kesulitan dalam berhubungan dengan pihak Google.

2.7 Asal Usul Logo Android

Siapa yang tak kenal Maskot Android Hijau ini, selain bentuknya lucu dan menarik ternyata asal usul logo tersebut cukup unik, Logo android Berasal dari Sebuah Game Arcade Di era tahun 90an yang berjudul Gauntlet : The Third Encounter Arcade , memang game ini tak begitu di kenal di eranya namun salah satu Tokoh robot abu-abu yang memiliki dua antena bernama android ini menjadi maskot yang sangat di kenal di seluruh dunia sebagai OS.

2.8 Bahasa Pemrograman

Aplikasi Android dikembangkan dalam bahasa pemrograman Java dengan menggunakan kit pengembangan perangkat lunak Android (SDK). SDK ini terdiri dari seperangkat perkakas pengembangan,[58] termasuk debugger, perpustakaan perangkat lunak, emulator handset yang berbasis QEMU, dokumentasi, kode sampel, dan tutorial. Didukung secara resmi oleh lingkungan pengembangan terpadu (IDE) Eclipse, yang menggunakan plugin Android Development Tools (ADT). Perkakas pengembangan lain yang tersedia di antaranya adalah Native Development Kit untuk aplikasi atau ekstensi dalam C atau C++, Google App

Inventor, lingkungan visual untuk pemrogram pemula, dan berbagai kerangka kerja aplikasi web seluler lintasplatform.

2.9 Fitur Android SDK

Fitur-fitur terpenting dari Android antara lain: - Kemudahan untuk merilis aplikasi yang telah dibuat, tidak adanya proses perizinan, distribusi, atau pengembangan biaya atau persetujuan proses rilis – Adanya akses hardware Wi-fi - GSM, EDGE, dan 3G untuk transfer telepon atau data yang memungkinkan kita untuk membuat atau menerima panggilan atau pesan SMS, atau untuk mengirim dan mengambil data di jaringan mobile - Adanya fasilitas API untuk menggunakan sensor hardware, termasuk accelerometer dan kompas - Komprehensif API untuk layanan berbasis lokasi seperti GPS - Librari yang berguna untuk Bluetooth untuk peer-to-peer mentransfer data - IPC message passing - Kendali kontrol multimedia hardware, termasuk pemutaran dan merekam dengan kamera dan mikropon - Kemampuan untuk mengintegrasikan hasil pencarian aplikasi ke dalam sistem pencarian - Adanya sebuah browser open-source HTML5 WebKit berbasis terpadu - Fasilitas hardware-accelerated grafis Mobile yang optimal, termasuk library berbasis grafis jalur 2D dan dukungan untuk grafis 3D menggunakan OpenGL ES 2.0 - Fasilitas media yang dilengkapi library dan media untuk bermain dan merekam berbagai audio / video atau masih dalam format gambar - Fasilitas dukungan Native Google Maps, Geocoding, dan Layanan Berbasis Lokasi

2.10 Fitur yang tersedia di Android antara lain:

- a. Framework aplikasi yang mendukung penggantian komponen dan reusable
- b. Dalvik virtual machine
- c. Integrated browser
- d. Grafik berdasarkan OpenGL
- e. SQLite untuk penyimpanan data
- f. Multimedia support

- g. Lingkungan Development yang lengkap dan kaya termasuk perangkat emulator, tools untuk debugging, profil dan kinerja memori, dan plugin untuk IDE Eclipse

2.11 SMS (Short Message Service)

Short Message Service atau layanan pesan singkat adalah sebuah layanan yang dilaksanakan dengan sebuah telepon genggam untuk mengirim atau menerima pesan-pesan pendek. Pada mulanya SMS di rancang sebagai bagian dari GSM tetapi sekarang sudah di tempatkan pada jaringan bergerak lainnya termasuk UMTS. Sebuah pesan SMS maksimal terdiri dari 140 bytes, dengan kata lain sebuah pesan bisa memuat 140 karakter 8-bit, 160 karakter 7-bit atau 70 karakter 16-bit untuk bahasa Jepang, bahasa Mandarin dan bahasa Korea yang memakai hanzi (aksara Kanji / Hanja). Selain 140 bytes ini ada data-data lain yang termasuk. Adapun beberapa metode untuk mengirim pesan yang lebih dari 140 bytes, tetapi seorang pengguna harus membayar lebih mahal. SMS bisa pula untuk mengirim gambar, suara dan film. SMS bentuk ini disebut MMS.

2.11.1 Awal Perkembangan

Pada tahun 1980-an gagasan menambahkan fungsi pertukaran pesan teks untuk telepon selular pada awalnya diusulkan oleh komunitas layanan ponsel. Pada bulan Desember 1982, Usulan pesan singkat untuk telepon selular dikembangkan dan diusung oleh CEPT Group GSM (Global System for Mobile Communications) berhasil disetujui, usulan tersebut adalah menambahkan fasilitas dan layanan pesan teks kedalam telepon seluler seperti fungsi fax agar pemngguna ponsel dapat pula berkomunikasi secara tulisan dengan layanan baru ini. Pada tahun 1980-an konsep ini juga direncanakan agar dapat melakukan pertukaran pesan secara langsung dan dalam cakupan yang luas seperti layanan panggilan ke operator yang berbeda.

Kemudian pada tahun 1984, konsep ini dikembangkan lagi oleh perusahaan GSM Perancis yang bekerja sama dengan perusahaan GSM dari Jerman, khususnya oleh Friedhelm Hillebrand dan Bernard Ghillebaert.

Perkembangan yang diusulkan adalah layanan pesan yang singkat, karena pada saat itu fungsi utama GSM adalah membuat dan menerima panggilan suara.

Untuk mengoptimalkan fungsi layanan pesan singkat atau SMS ini, traffic atau arus komunikasi untuk SMS ini hanya menggunakan sedikit jalur sinyal, karena untuk melakukan panggilan dan menerima panggilan telepon, GSM harus menggunakan sinyal yang kuat. Dengan kata lain, SMS terbatas dalam penggunaan sinyal, karena fungsi utama system GSM adalah untuk membuat dan menerima panggilan. Oleh sebab itu, SMS hanya terbatas 160 karakter saja, yang mengharuskan pengguna telepon seluler untuk memadatkan (mempersingkat) kata yang digunakan.

Permasalahan di negara-negara yang menginginkan pemutakhiran layanan ini adalah terfokus pada software berbeda yang dimiliki perusahaan-perusahaan yang memproduksi system seluler GSM dan operator-operator (service provider) yang belum semuanya mendukung fasilitas layanan pesan singkat ini, namun pada perkembangannya layanan pesan teks dapat dikembangkan pada software yang berbeda dan memungkinkan untuk melakukan pertukaran pesan dengan lancar dari operator dan jenis ponsel yang berbeda bahkan untuk Negara yang berbeda. Dan untuk lebih memudahkan proses layanan ini dibuatlah konsep untuk menghindari hambatan akibat pesatnya arus layanan yang mulai populer ini yakni SMS center.

Penkembangan SMS kemudian berlanjut dengan dibuatnya pengaturan tentang standar protokol kemudian dilanjutkan dengan pengaturan untuk jaringan pada tahun 1985-an yang pada awalnya hanya untuk di beberapa negara terutama kawasan Eropa. Dan untuk memastikan bahwa teknologi komunikasi ini benar-benar memenuhi standar, Implementasi dari konsep awal SMS kemudian dilakukan pada tahun 1990-an. Hasilnya, SMS pertama berhasil dikirim oleh jaringan SEMA group menggunakan PC ke handset Orbitel 901 dan diterima oleh jaringan GSM VODAVONE di Inggris (UK) pada 3 desember 1992. Percobaan selanjutnya adalah SMS pertama dari telepon seluler GSM untuk telepon selular lainnya berhasil dilakukan oleh Riku Pihkonen, seorang mahasiswa teknik di NOKIA pada tahun 1993. SMS kemudian berhasil untuk dikomersilkan dan

kemudian berkembang pesat pada tahun 1993. Dimulai dari di Swedia oleh Aldicson dan TeliaSonera, dan kemudian diikuti oleh Fleet Call di US, Telenor di Norwegia dan BT celnet (sekarang O2) di Inggris (UK).

2.11.2 Perkembangan SMS Untuk Sistem Lain

Pada awalnya SMS khusus dirancang dan dibuat hanya untuk ponsel GSM, namun kemudian dengan berkembang pesatnya teknologi-teknologi dalam kebutuhan komunikasi modern, layanan SMS mulai dikembangkan untuk berbagai jaringan selain GSM seperti CDMA (Code Division Multiple Access) dan jaringan lain yang berkembang pada masa itu. Pada dasarnya CDMA berbeda dengan GSM, perbedaan terdapat pada frekuensi yang mengharuskan CDMA untuk mengadaptasi dan menanamkan fungsi SMS dari jaringan GSM. Perkembangan ini tentu dipengaruhi oleh gaya komunikasi baru yakni komunikasi dengan teks yang populer dan berorientasi pada keuntungan tinggi dari penyediaan layanan ini. Di sisi lain, perusahaan yang juga mengembangkan teknologi ini mulai memunculkan alternative layanan pesan teks yang berbeda seperti J-Phone yang mulai mengembangkan layanan J-Phone's Skymail selain itu perusahaan lain seperti NTT Docomo mulai mengembangkan NTT Docomo's Short Mail sebagai pengganti SMS namun terbatas hanya untuk jaringan (system ponsel) yang sama. Kemudian ada pula perkembangan seperti email messaging dari ponsel seperti NTT Docomo's i-mode dan RIM Blackberry (Blackberry Messenger). Dan diketahui bahwa NTT dan RIM memakai standar untuk mail protokol yang sejenis.

Resiko-resiko yang dapat mengancam keamanan pesan pada layanan SMS :

a. SMS Spoofing

merupakan pengiriman sms di mana nomor pengirim yang tertera bukanlah nomor pengirim yang sebenarnya. Mekanisme SMS spoofing ini dimungkinkan karena lemahnya proteksi koneksi SMSCgateway. Penyusup dapat merekam login dan password dari pesan yang berasal dari SMS gateway menuju SMSC. Walaupun tak terlalu mudah namun ini dapat dilakukan

dalam beberapa kasus. Dalam hal ini penyusup mengatur sebuah gateway palsu yang berlaku seperti gateway sesungguhnya. Gateway palsu ini dapat mengirim semua jenis pesan pendek kepada user MS melalui SMSC. Pada teknik spoofing ini pesan dikirim dengan memanipulasi nomor MSISDN asal (originate) pada field yang disediakan sehingga pesan akan tampak datang dari nomor pengirim lainnya. Kemungkinan spoofing yang lain adalah dengan membuat simulator SMSC yang berlaku seperti SMSC asli. Dengan cara ini gateway akan kebanjiran pesan, sebagai contoh aplikasi bank menggunakan gateway dapat dengan mudah diperoleh informasi account bahkan dapat digunakan untuk transaksi bank tanpa proses authorisasi.

b. SMS Snooping

SMS snooping lebih sering terjadi karena kelalaian pengguna telepon seluler. Contohnya ketika seseorang meminjamkan telepon selulernya pada orang lain untuk menggunakan telepon selulernya. Pada saat itu orang tersebut dapat dengan sengaja atau tidak membuka isi pesan yang ada pada inbox SMS. Pesan yang bersifat personal atau rahasia dapat dibaca dengan mudah oleh orang lain melalui cara ini.

c. SMS Interception

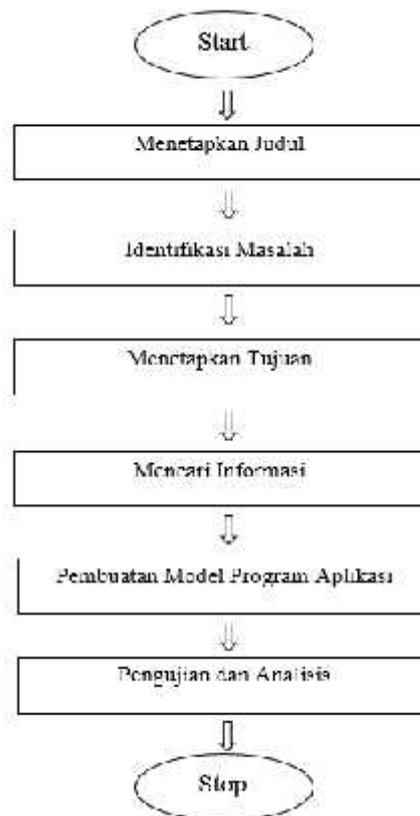
SMS bekerja pada jaringan nirkabel yang memungkinkan terjadinya pencurian isi pesan SMS ketika dalam proses transmisi dari pengirim ke penerima. Celah keamanan terbesar pada layanan komunikasi SMS adalah pada saat SMS tersebut sedang dikirim melalui jaringan SMS tersebut.

BAB III METODE PENELITIAN

3.1 Tinjauan Sistem

Layanan pesan singkat (SMS) tidak menjamin integritas dan keamanan pesan yang di sampaikan. Pesan yang bersifat personal atau rahasia tidak di jamin sampai ke penerima tanpa di ketahui, di curi, maupun di manipulasi informasinya oleh orang lain. Di harapkan dengan menggunakan aplikasi sms security dengan metode kriptografi, dapat membantu dalam menjaga kerahasiaan dan keutuhan pesan yang dikirimkan.

Adapun langkah kerja atau metode pembuatan program aplikasi ini di lakukan secara bertahap sebagaimana di uraikan sebagai berikut;



Gambar 3.1 Diagram Alir Penelitian Tugas Akhir

3.2 Metode Pengumpulan Data

Metode yang digunakan dalam pengumpulan data pada penyusunan laporan penelitian ini terdiri dari beberapa metode, yaitu :

3.2.1 Metode Observasi

Metode observasi merupakan suatu cara pengumpulan data dengan pengamatan secara langsung pada obyek. Dalam hal ini melakukan observasi pada beberapa caramenjaga keutuhan dan kerahasiaan isi pesan.

3.2.2 Metode Studi Kepustakaan

Metode ini merupakan sebuah cara dalam pengumpulan data dengan mempelajari bahan pustaka baik berupa dokumen tertulis maupun gambar.

3.2.3 Metode Dokumentasi

Yaitu suatu teknik atau metode penelitian yang digunakan untuk memperoleh informasi dengan cara mengumpulkan data yang lama dan yang ada dalam perpustakaan tersebut. Dalam hal ini penulis mendapatkan dokumentasi berupa buku aplikasi berbasis android.

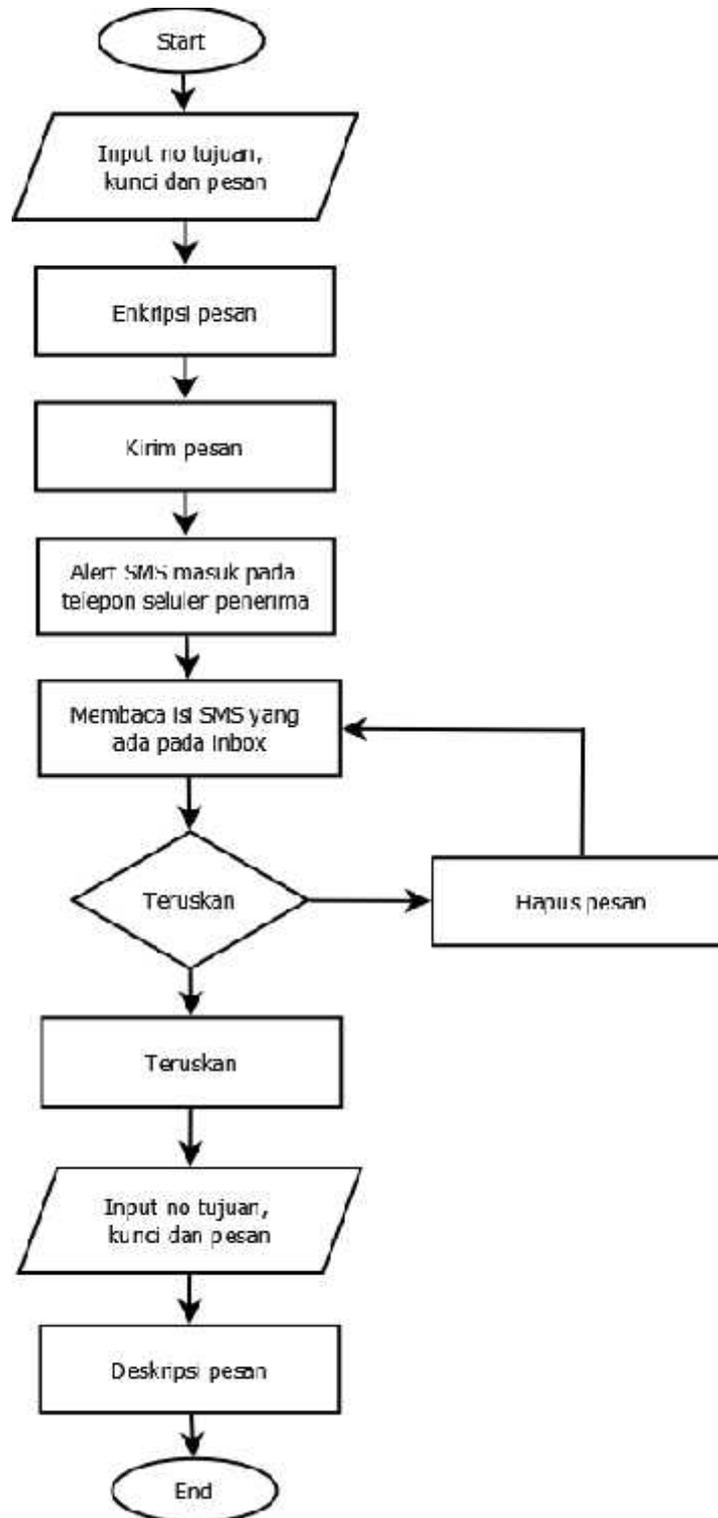
3.3 Metode Perancangan Program Aplikasi

Langkah-langkah pembuatan program aplikasi secara umum akan mengikuti tahapan sebagai berikut :

3.3.1 Mendefinisikan Masalah

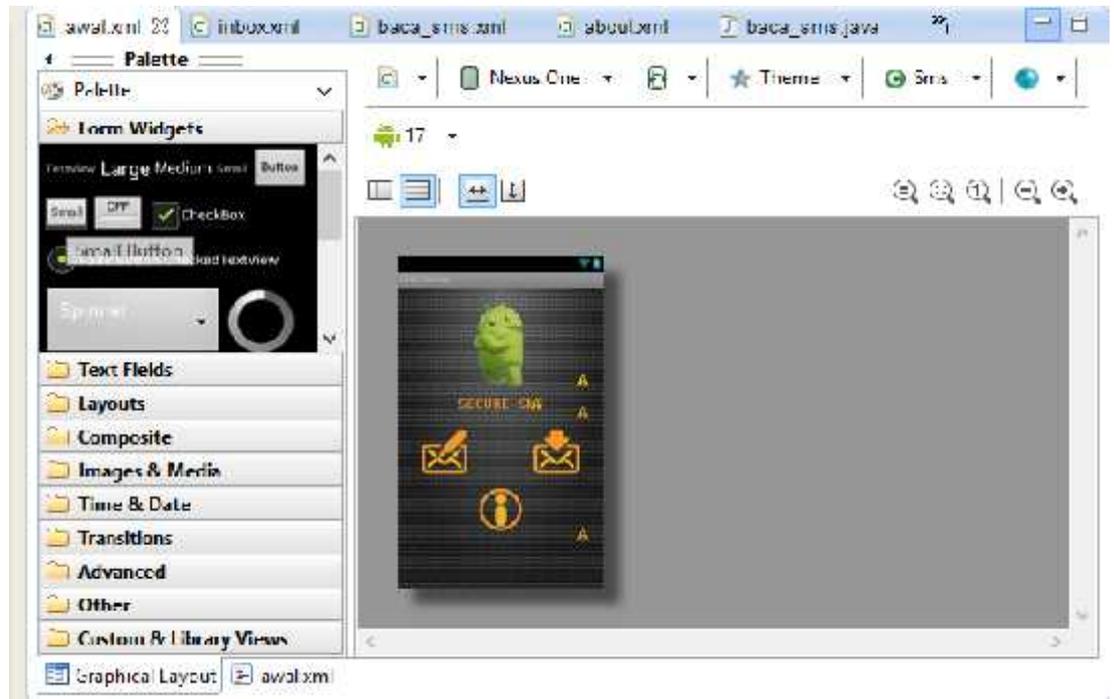
Pesan yang dikirimkan oleh aplikasi SMS bawaan telpon seluler belum terproteksi dan pengiriman SMS harus tercatat di SMS Center. pesan yang dikirimkan juga tidak aman karena ada beberapa resiko seperti SMS Spoofing, SMS Snooping, dan SMS Interception.

3.3.2 Membuat Flowchart



Gambar 3.2 Flowchart

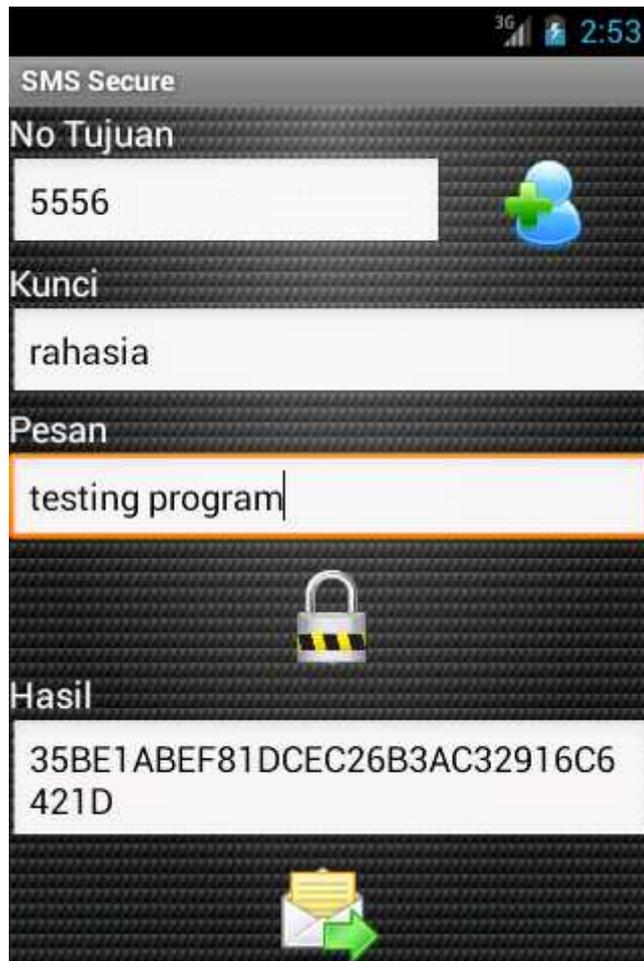
3.3.3 Membuat Program



Gambar 3.3 Proses Pembuatan Program

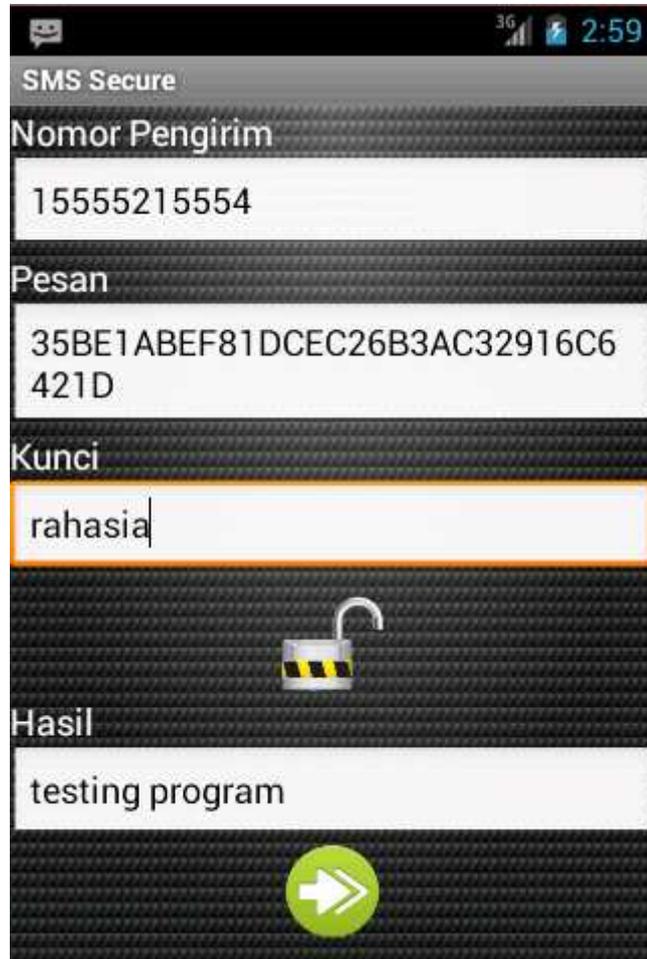
3.3.4 Melakukan tes program

Pengirim



Gambar 3.4 Testing Enkripsi Pesan

Penerima



Gambar 3.5 Testing Deskripsi Pesan

3.4 Metode Analisis

Metode analisis data digunakan untuk menjawab masalah penelitian. Dengan menganalisa maka data tersebut mempunyai arti dan makna yang berguna dalam memecahkan masalah penelitian. Untuk melakukan pengembangan sistem dibutuhkan tahapan-tahapan tertentu. Beberapa kebutuhan sistem yang akan di definisikan antara lain:

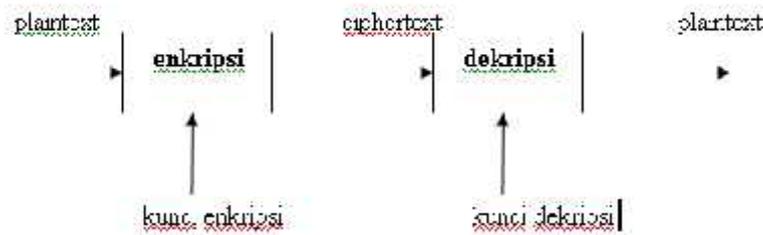
- a. Memiliki kemampuan untuk mengirimkan dan menerima pesan melalui SMS
- b. Memiliki kemampuan untuk mengenkripsi pesan
- c. Memiliki kemampuan untuk mendekripsi pesan
- d. Memiliki kemampuan untuk membaca pesan masuk yang ada di dalam inbox

Karena aplikasi ini tergolong aplikasi sederhana maka tidak dibutuhkan suatu kondisi yang rumit. Berikut adalah gambaran sistem secara umum:

- a. Pengirim akan mengirim pesan melalui layanan SMS Secure
- b. Pesan ini akan dienkripsi terlebih dahulu dengan kunci sebelum dikirim
- c. Pesan ini akan dikirim berupa pesan teks (SMS) Pesan ini nantinya akan diterima oleh penerima pesan dalam keadaan terenkripsi.
- d. Karena pesan yang diterima dalam keadaan terenkripsi maka harus ada pendekripsi pesan supaya pesan yang diterima memiliki makna.

3.5 Metode Kriptografi

Kriptografi (atau kriptologi dari bahasa Yunani *kryptos*, "tersembunyi, rahasia", dan *graphein*, "menulis", atau *logi*, "ilmu") merupakan keahlian dan ilmu dan cara-cara untuk komunikasi aman pada kehadirannya di pihak ketiga. Secara umum kriptografi adalah ilmu dan seni untuk menjaga kerahasiaan berita. Selain pengertian tersebut terdapat pula pengertian ilmu yang mempelajari teknik-teknik matematika yang berhubungan dengan aspek keamanan informasi seperti kerahasiaan data, serta autentikasi data, keabsahan data, integritas data, serta autentifikasi data. Tidak semua aspek keamanan informasi ditangani oleh kriptografi.



Gambar 3.6 Proses Enkripsi/Dekripsi Kriptografi Kunci

Ada empat tujuan mendasar dari ilmu kriptografi ini yang juga merupakan aspek keamanan informasi yaitu ;

- a) **Kerahasiaan**, adalah layanan yang digunakan untuk menjaga isi dari informasi dari siapapun kecuali yang memiliki otoritas atau kunci rahasia untuk membuka/ mengupas informasi yang telah disandi.
- b) **Integritas data**, adalah berhubungan dengan penjagaan dari perubahan data secara tidak sah. Untuk menjaga integritas data, sistem harus memiliki kemampuan untuk mendeteksi manipulasi data oleh pihak-pihak yang tidak berhak, antara lain penyisipan, penghapusan, dan pensubsitusian data lain kedalam data yang sebenarnya.
- c) **Autentikasi**, adalah berhubungan dengan identifikasi/ pengenalan, baik secara kesatuan sistem maupun informasi itu sendiri. Dua pihak yang saling berkomunikasi harus saling memperkenalkan diri. Informasi yang dikirimkan melalui kanal harus diautentikasi keaslian, isi datanya, waktu pengiriman, dan lain-lain.
- d) **Non-repudiasi**, atau nirpenyangkalan adalah usaha untuk mencegah terjadinya penyangkalan terhadap pengiriman/ terciptanya suatu informasi oleh yang mengirimkan/ membuat.

Terminologi dalam Kriptografi

Ada beberapa istilah-istilah yang penting dalam kriptografi, yaitu :

1. **Pesan** (Plaintext dan Ciphertext) : Pesan (message) adalah data atau informasi yang dapat dibaca dan dimengerti maknanya. Pesan asli disebut plainteks (plaintext) atau teks-jelas (cleartext). Sedangkan pesan yang sudah disandikan disebut cipherteks (chipertext).
2. **Pengirim** dan **Penerima** : Komunikasi data melibatkan pertukaran pesan antara dua entitas. Pengirim (sender) adalah entitas yang mengirim pesan kepada entitas lainnya. Penerima (receiver) adalah entitas yang menerima pesan.
3. **Penyadap** (eavesdropper) adalah orang yang mencoba menangkap pesan selama ditransmisikan.
4. **Kriptanalisis** dan **Kriptologi** : Kriptanalisis (cryptanalysis) adalah ilmu dan seni untuk memecahkan chiperteks menjadi plainteks tanpa mengetahui kunci yang digunakan. Pelakunya disebut kriptanalis. Kriptologi (cryptology) adalah studi mengenai kriptografi dan kriptanalisis.
5. **Enkripsi** dan **Dekripsi** : Proses menyandikan plainteks menjadi cipherteks disebut enkripsi (encryption) atau enciphering. Sedangkan proses mengembalikan cipherteks menjadi plainteks semula dinamakan dekripsi (decryption) atau deciphering.
6. **Cipher** dan **Kunci** : Algoritma kriptografi disebut juga cipher yaitu aturan untuk enchipering dan dechipering, atau fungsi matematika yang digunakan untuk enkripsi dan dekripsi. Kunci (key) adalah parameter yang digunakan untuk transformasi enciphering dan dechipering. Kunci biasanya berupa string atau deretan bilangan.

3.6 Algoritma RC6

Algoritma RC6 merupakan algoritma sederhana, fungsi yang digunakan merupakan fungsi yang sederhana dan hanya mengandalkan prinsip iterated cipher untuk keamanan. Tampilan hasil enkripsi dan data hasil enkripsi yang di terima harus di perhatikan, hal ini dikarenakan pada hasil enkripsi, setiap karakternya akan memiliki panjang 8 bit, sedangkan sebagian telpon selular hanya dapat menampilkan karakter dengan panjang 7 bit. Dengan demikian dalam penerapan algoritma RC6 pada SMS karakter-karakter yang akan dienkripsi diubah kedalam nilai ASCII, dimana nilai karakter dalam tabel ASCII ditambah tabel karakter special adalah 0 sampai dengan 255, artinya satu karakter ASCII akan diwakili oleh 8 bit, dimana $2^8 = 256$. Sehingga, dalam 1 blok plaintext (32 bit) akan menyampaikan 4 karakter dan setiap kali iterasi, maka akan diambil 16 karakter dari plainteks.

Apabila panjang plainteks atau panjang kunci kurang dari 16 karakter, maka akan dilakukan padding, yaitu dengan menambahkan bit "0" (nol) di akhir teks, sehingga panjang teks mencakupi 16 karakter. Layar pada sebagian besar telpon seluler hanya dapat menampilkan karakter dengan panjang 7 bit dan pesan yang telah terenkripsi akan membentuk binary, sehingga layar tidak akan menampilkan dengan semestinya. Oleh karena itu, pada aplikasi yang akan dibangun, untuk menampilkan pesan yang telah terenkripsi, ditambahkan informasi karakter yang terdapat pesan tersebut dengan format heksadesimal agar dapat ditampilkan layar dan informasinya lebih terbaca.

Algoritma RC6 yang akan digunakan dalam aplikasi enkripsi SMS yang akan dibangun dengan w sebesar 32 bit, r sebesar 20 kali putaran dan panjang kunci beragam lebih dari 1 karakter (8 bit). Langkah-langkah algoritma RC6 dalam pelaksanaan tugas akhir ini akan dikelompokkan kedalam beberapa bagian, yaitu ;

1. Pembangkit Subkunci

Kunci dari pengguna ini akan dimasukan oleh pengguna saat akan melakukan proses enkripsi dan deskripsi. Kunci ini memiliki tipe data String dan memiliki panjang 16 byte (16 karakter).

2. Baca masukan untuk proses enkripsi

Yang dilakukan pada tahapan ini adalah membaca text yang menjadi masukan pada proses enkripsi, yaitu field dari aplikasi enkripsi SMS. Pada proses enkripsi pesan, field-nya adalah isi pesan.

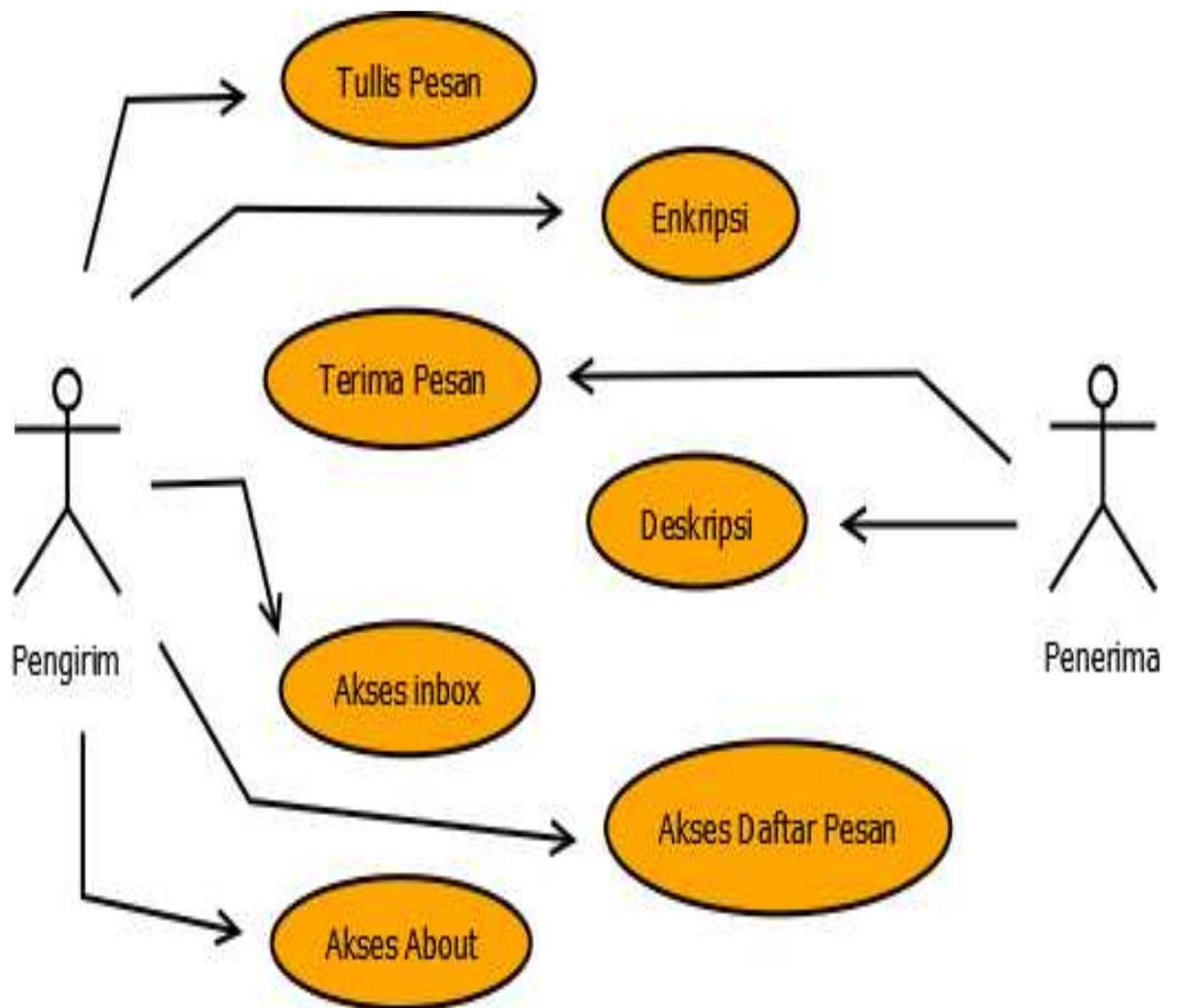
3. Enkripsi meliputi whitening awal, iterasi dan whitening akhir.

4. Baca masukan untuk proses deskripsi

Yang dilakukan pada tahapan ini adalah membaca teks yang menjadi masukan pada proses deskripsi, yaitu record dari hasil pesan yang telah dienkripsi pada pengirim dan menjadi field pesan pada penerima.

5. Deskripsi merupakan kebalikan dari proses enkripsi.

3.7. Diagram Use Case



Gambar 3.7 Diagram Use Case

3.8. Spesifikasi Use Case

Aktor	Deskripsi
Pengirim	<p>Tulis Pesan</p> <ul style="list-style-type: none"> - Pengirim menulis pesan pada aplikasi untuk dikirim
Pengirim	<p>Enkripsi Pesan</p> <ul style="list-style-type: none"> - Pengirim mengenkripsi pesan yang akan dikirim agar pesan teracak dan tidak dapat di baca
Pengirim/Penerima	<p>Akses Inbox</p> <ul style="list-style-type: none"> - Pengirim/penerima membaca pesan di dalam kotak masuk - <include> Daftar Pesan Inbox
Penerima	<p>Terima Pesan</p> <ul style="list-style-type: none"> - Penerima mendapatkan pesan dari pengirim
Penerima	<p>Deskripsi Pesan</p> <ul style="list-style-type: none"> - Penerima mendeskripsi pesan yang teracak agar tidak terbaca
Pengirim/Penerima	<p>Akses About</p> <p>-Pengirim/penerima mendapat keterangan mengenai aplikasi seperti versi dan pembuat aplikasi</p>

Tabel 3.1 Spesifikasi Use Case Diagram Sistem

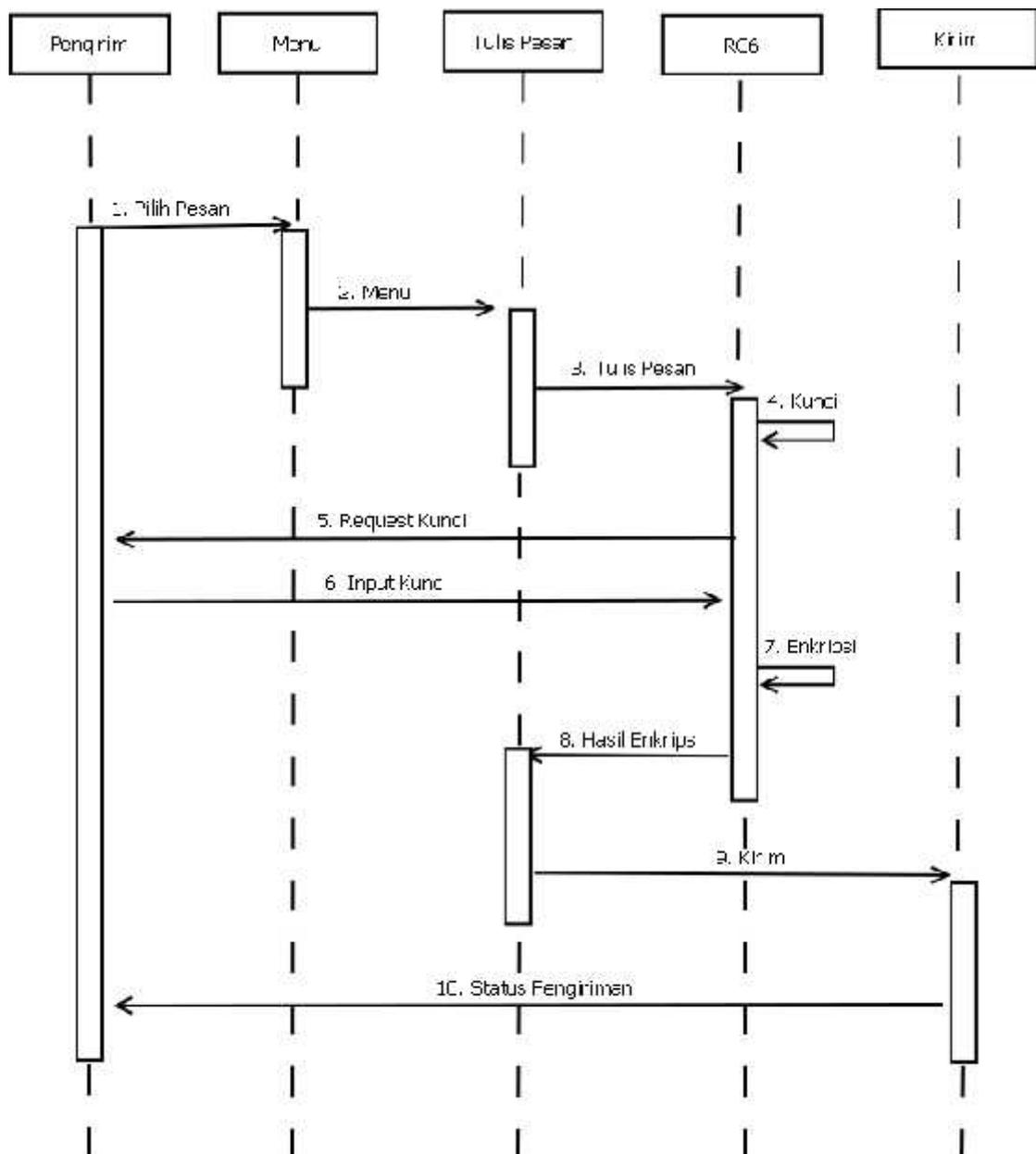
3.9.Deskripsi Perancangan Class

Nama Kelas	Nama File	Keterangan
SMS	SMS.java	Kelas ini merupakan tampilan utama dari aplikasi Enkripsi SMS RC6, ketika aplikasi di jalankan, maka kelas ini yang pertama di panggil
Tulis Pesan	tulis_pesanan.java	Kelas ini merupakan tampilan untuk melakukan penulisan, pengenkripsian, dan pengiriman pesan
Inbox	Inbox.java	Kelas ini merupakan kelas untuk membaca SMS yang dipilih di dalam kelas Inbox, serta mendeskripsi pesan yang terenkripsi
About	About.java	Kelas ini untuk menampilkan tentang dari aplikasi
RC6	RC6.java	Kelasi ini merupakan kelas algoritma enkripsi deskripsi dari RC6, serta

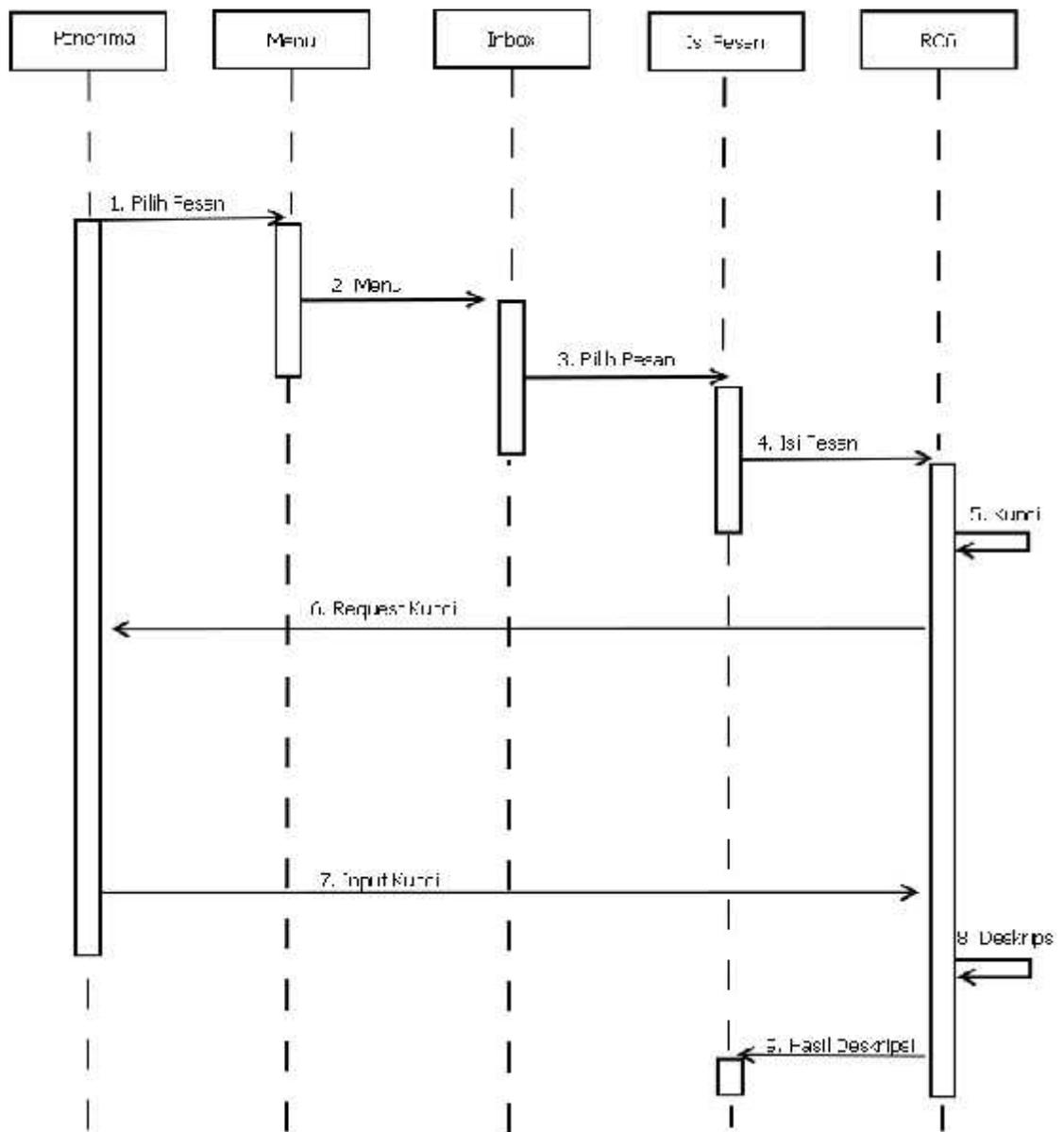
		algoritma dari penjadwalan kunci RC6
--	--	---

Table 3.2 Deskripsi Perancangan Class Diagram

3.10. Diagram Sequence



Gambar 3.8 Diagram Sequence Tulis Pesan



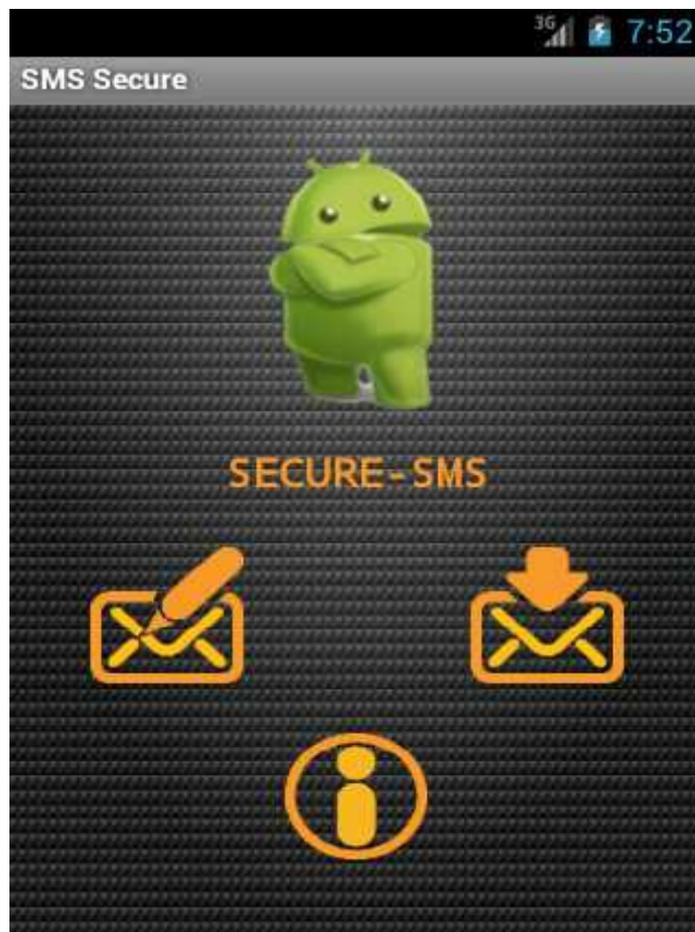
Gambar 3.9 Diagram Sequence Baca Pesan

BAB IV IMPLEMENTASI DAN TESTING

4.1. Implementasi Program

Berikut merupakan implementasi dari aplikasi SMS Security Menggunakan Metode Kriptografi ;

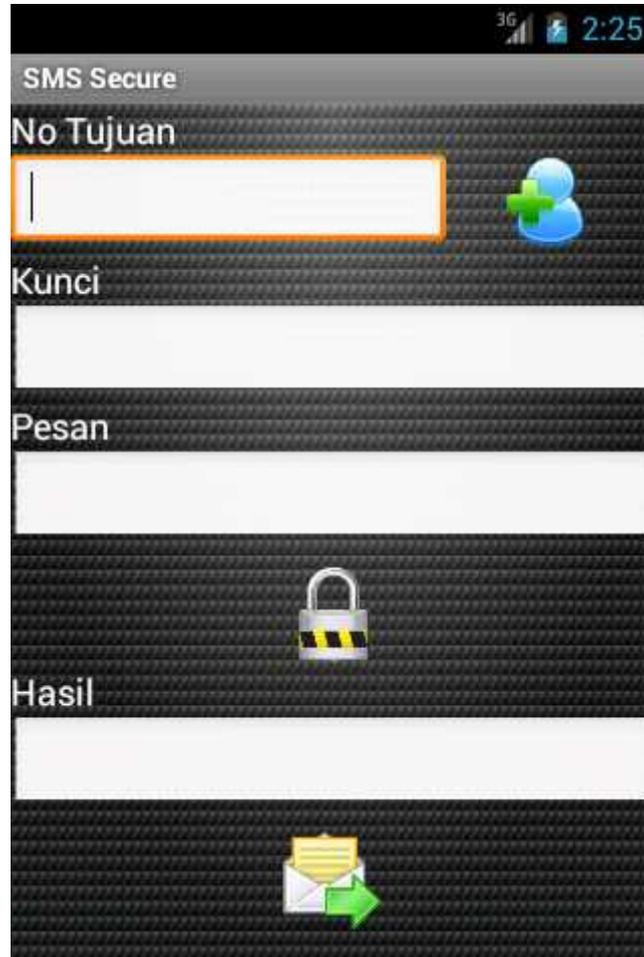
1. Tampilan awal aplikasi SMS security



Gambar 4.1 Menu Awal Aplikasi SMS Security

Tampilan awal aplikasi SMS Security pada saat aplikasi di jalankan. Terdapat 3 menu yaitu ; tulis pesan, baca pesan, about/tentang.

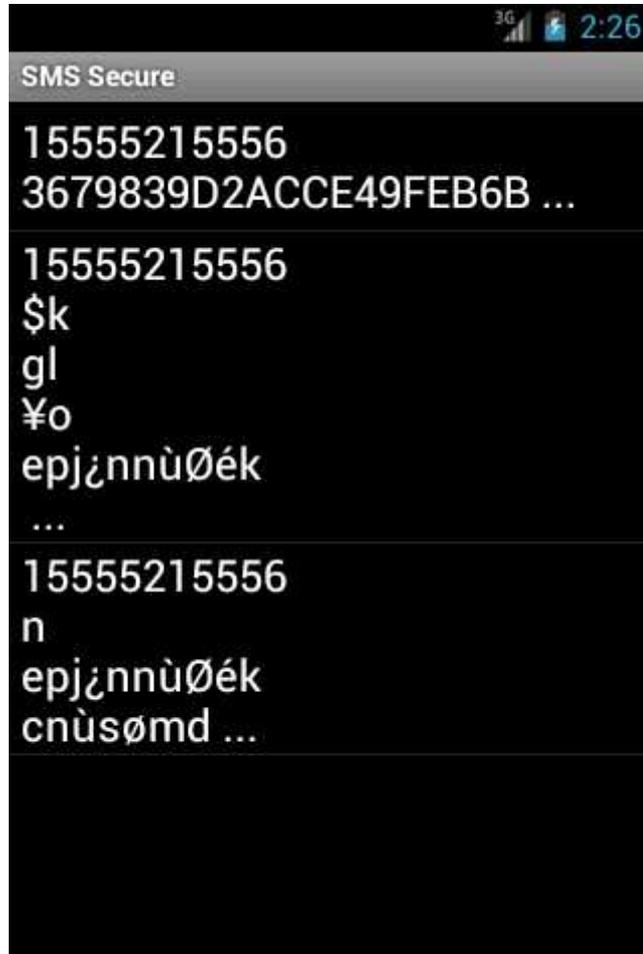
2. Tampilan menu tulispesan aplikasi SMS security



Gambar4.2 Menu TulisPesan SMS Security

Tampilan menu tulispesan. Di manadapatdilihat terdapat input no tujuan, input kunci, input pesan, tombol enkripsi, hasil enkripsi pesan dan tombol kirim pesan ke penerima.

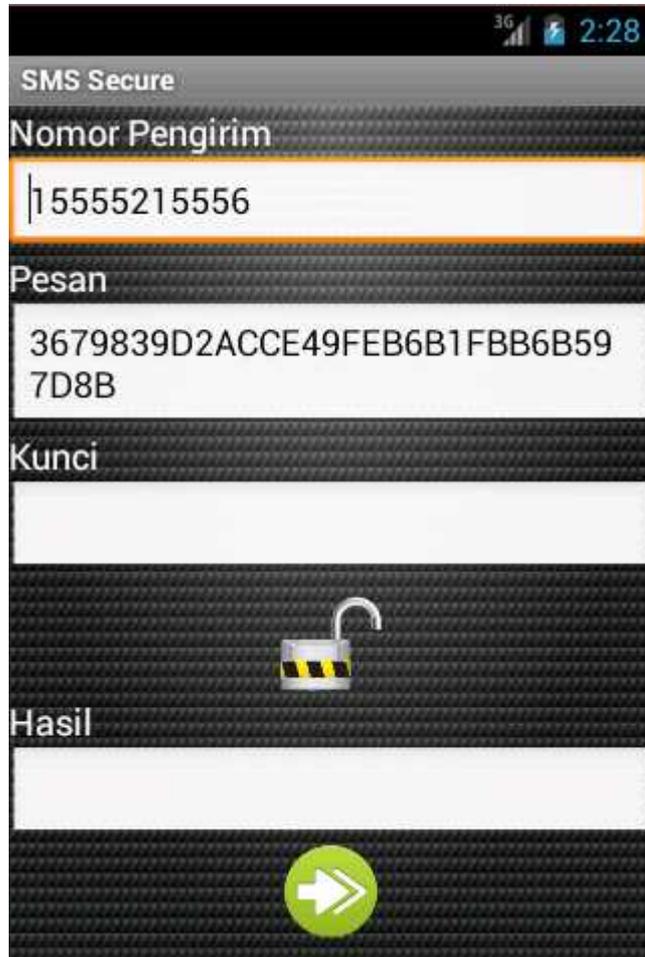
3. Tampilan list pesan aplikasi SMS security



Gambar 4.3 Menu List Pesan SMS Security

Tampilan menu list pesanmasuk. Pesan yang masukakanberurutankebawahberdasarkantanggalpesan yang masukdaripengirim.

4. Tampilan Baca pesan SMS Security



Gambar 4.4 Menu Baca Pesan SMS Security

Tampilan menu bacapesan. Dimanadapatdilihat terdapat tampilan no pengirim. Isi pesan, tampilan input kunci, tombol deskripsi, tampilan hasil deskripsi dan tombol teruskan pesan.

5. Tampilan Menu About SMS Security



Gambar 4.5 Menu About SMS Security

Tampilan menu about. Di manadapatdilihaterdapatnamaaplikasi, versi, namapenulis, nimpenulis, jurusan ,danasalpendidikanpenulis.

4.2. Testing Program

Berikut merupakan testing program mengirim dan menerima pesan menggunakan 2 virtual device ;

1. Device 1 (No telp 5554).



Gambar 4.6 Tampilan Awal Device 1

Tampilan menu awal aplikasi dari pengirim pada saat aplikasi di jalankan. Dapat dilihat tampilan menu tulis pesan, baca pesan dan tentang. Lalu pilih menu tulis pesan untuk menulis pesan yang akan di kirimkan ke penerima (device 2).

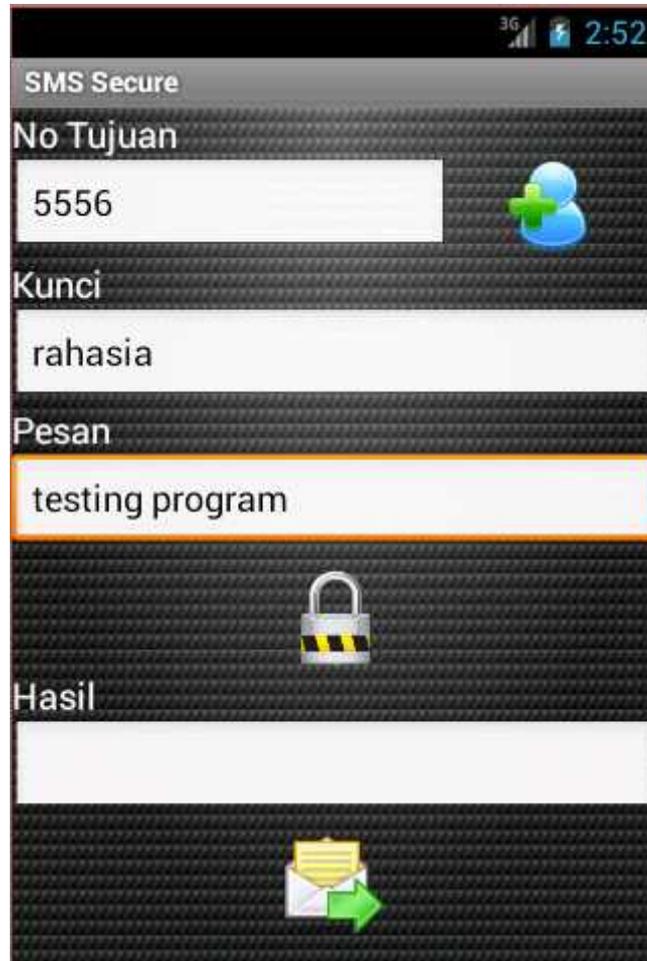
2. Tampilan Menu TulisPesan



Gambar 4.7 Tampilan Menu TulisPesan

Tampilan menu tulispesan. dimanadapat di lihatpengirimharusmemasukannomortujuan “5556”, kata kunci “rahasia”, isipesan “testing program”.

3. Tampilan Proses Mengkripsi Pesan.



Gambar 4.8 Tampilan Proses Mengenkripsi Pesan

Dapat dilihat tampilan di atas nomor tujuan, kunci dan isi pesan telah dimasukkan.

Setelah itu tekan tombol kripsi untuk mengenkripsi pesan agar tidak dapat dibaca oleh pihak lain/penyadap.

4. Tampilan Proses Pengiriman Pesan



Gambar 4.9 Tampilan Proses Pengiriman Pesan

Dapat dilihat hasil dari proses enkripsi pesan. Pesan berupa sandi yang sulit untuk di baca oleh pihak lain/pendadap. Setelah itu klik tombol kirim untuk mengirim pesan ke penerima (device 2).

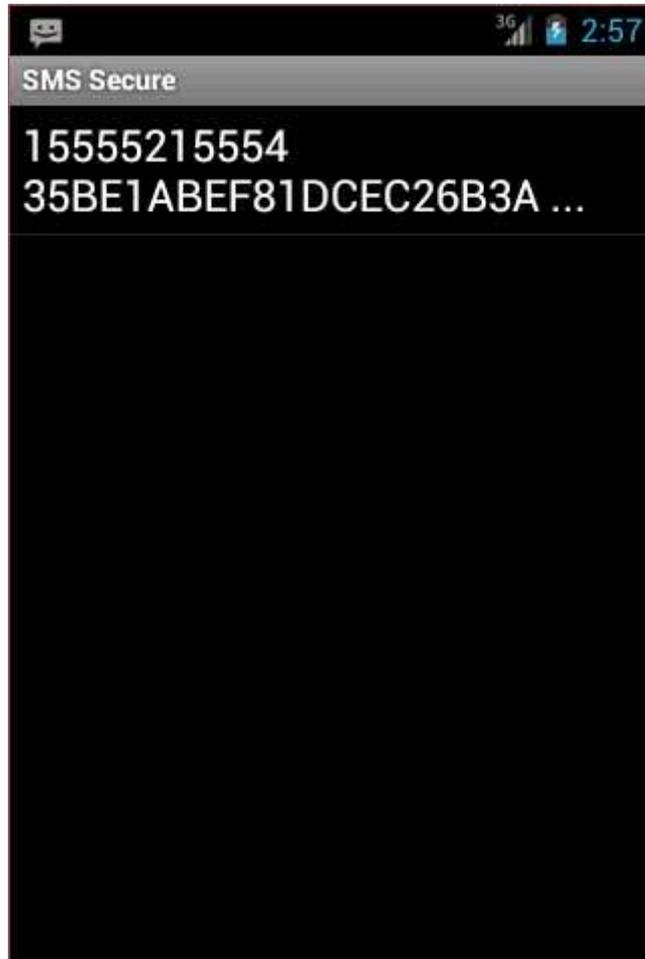
5. Device 2 (No telp 5556).



Gambar 5.0 Tampilan Awal Device 2

Tampilan penerim pesan (Device 2). Dapat dilihat pesan yang dikirimkan oleh pengirim (Device 1) telah masuk ke penerima. Klik tombol baca pesan untuk membaca pesan yang dikirim pengirim.

6. Tampilan List PesanPenerima



Gambar 5.1 Tampilan List PesanMasuk

Tampilan list pesan penerima (Device 2). Dapat dilihat pesan dari pengirim (5554) telah masuk ke penerima (5556). Double klik pada pesan yang di kirimkan untuk masuk ke menu baca pesan.

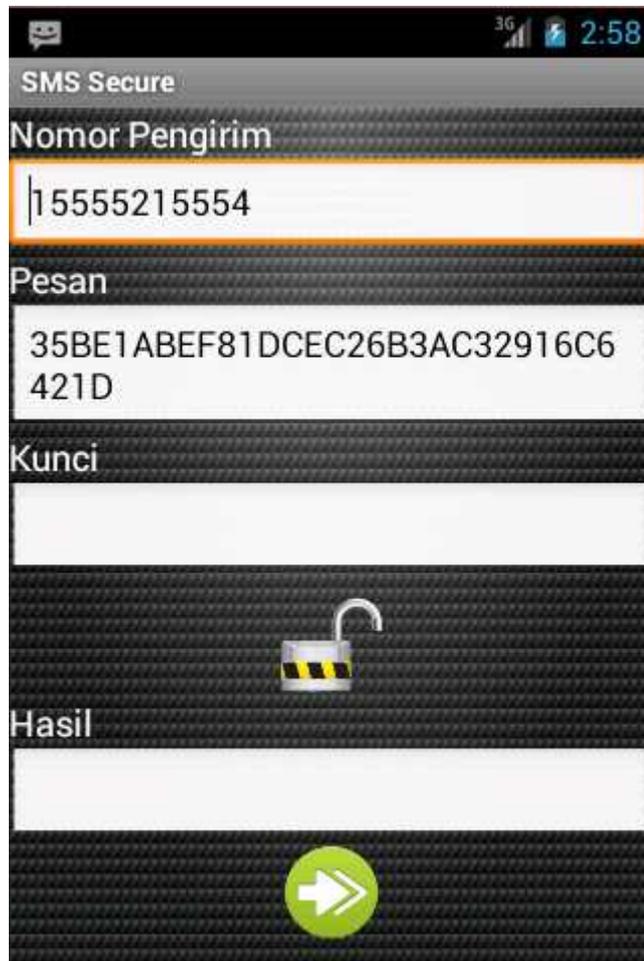
7. Tampilan Pilihan Hapus dan Teruskan



Gambar 5.2 Tampilan Pilihan Pesan

Klik pesan masuk pada list pesan. lalu muncul tampilan pilihan hapus pesan untuk menghapus pesan yang masuk dari pengirim dan teruskan pesan untuk meneruskan pesan ke orang lain.

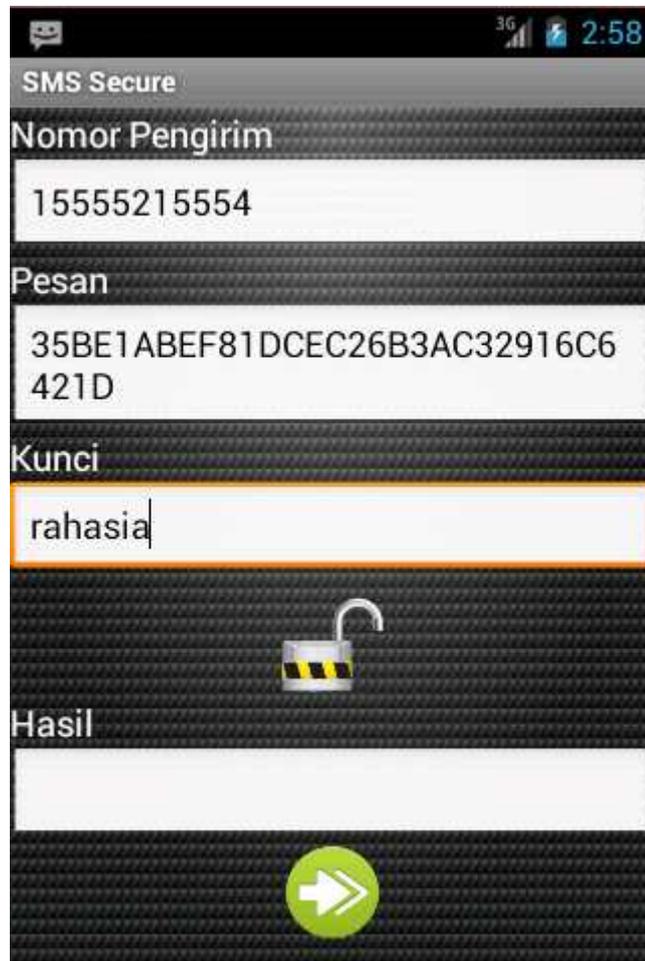
8. Tampilan Baca Pesan Pada Device 2



Gambar 5.3 Tampilan Baca Pesan Device 2

Tampilan menu bacapesanpadapenerima (Device 2). Dapat dilihat pesan berasal dari nomor pengirim dan isi pesan masih berupa sandi yang sulit di baca. Untuk bisa membaca pesan harus menggunakan kunci yang sama seperti kunci pengirim.

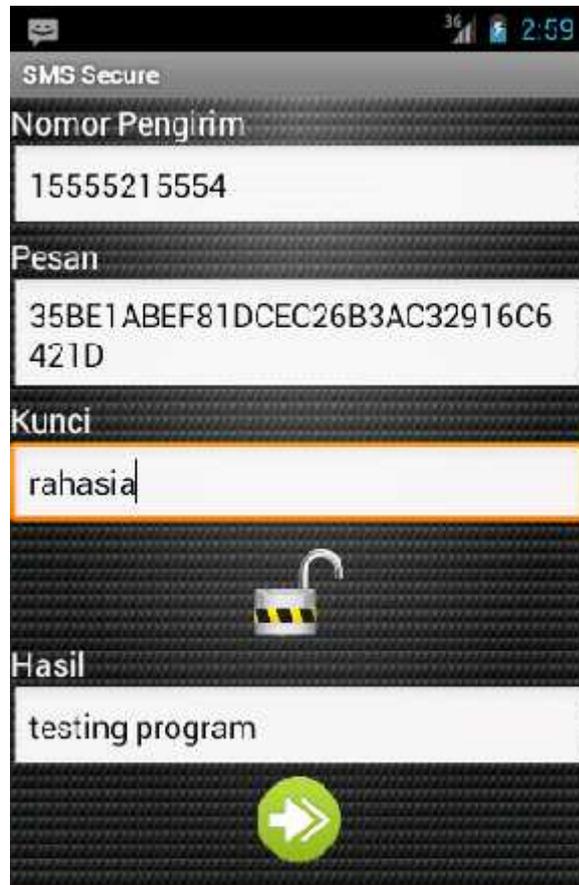
9. Tampilan Proses Deskripsi



Gambar 5.4 Tampilan Proses Deskripsi Pesan

Tampilan proses deskripsi pesan pada penerima (Device 2). Masukkan kunci "rahasia" seperti kunci pada pengirim untuk dapat membaca pesan. Setelah itu klik tombol deskripsi untuk mendeskripsikan isi pesan.

10. Tampilan Hasil Deskripsi Pesan pada device 2



Gambar 5.5 Tampilan Hasil Deskripsi Pesan

Tampilan hasil dari deskripsi pesan. dimana hasil sama dengan isi pesan yang dikirimkan pengirim (Device 1). Pengujian dilakukan pada telepon seluler Android Samsung GT-S7392 di dalamnya menggunakan firmware Android 2.3. Pengujian terdiri dari 4 bagian utama yaitu pengujian enkripsi pesan, pengiriman SMS, pembacaan SMS, dan dekripsi pesan. Pengujian ini dilakukan dengan menggunakan android virtual manager. Selanjutnya dilakukan pengujian dengan mengirim pesan terenkripsi secara berulang

-ulang dengan isip esandankunci yang berbeda-beda,
tidak ditemukan masalah. Semua fungsi dapat berjalan dengan baik.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Pesan yang bersifat personal atau rasis tidak aman jika dikirimkan melalui aplikasi SMS biasa. Orang lain dapat dengan mudah mencuri informasi dari SMS tersebut. Untuk itu dibutuhkan aplikasi yang dapat mengirim dan menerima pesan dengan metode kriptografi, yang mengenkripsi pesan dengan kunci RC 6, agar pesan tidak dapat diketahui, dicuri, atau dimanipulasi oleh orang lain/ penyadap.

SMS Security adalah aplikasi SMS yang di bangun untuk mengatasi masalah keamanan pada layanan SMS. Aplikasi ini mampu berjalan dengan baik pada platform android. Aplikasi ini mempunyai fungsi menulis pesan, mengenkripsi pesan, mengirim pesan, membaca pesan, dan mendeskripsikan pesan.

5.2 Saran

Untuk perbaikan dan pengembangan aplikasi SMS Secure lebih lanjut disarankan sebagai berikut :

- 5.2.1** Aplikasi menampilkan isi SMS masuk dan keluar dalam bentuk thread seperti aplikasi SMS bawaan Android.
- 5.2.2** Aplikasi dapat menyimpan SMS terkirim dan SMS yang belum dikirim.
- 5.2.3** Aplikasi didesain dengan interface yang menarik.
- 5.2.4** Fitur teruskan yang bisa di jalankan langsung di apk.

LAMPIRAN

1. Coding tampilan awal.xml

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

<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="wrap_content"
    android:background="@drawable/background"
android:orientation="vertical">

<LinearLayout
    android:id="@+id/linearLayout4"
android:layout_width="280dp"
android:layout_height="20dp"
android:orientation="vertical">
</LinearLayout>

    <RelativeLayout
        android:id="@+id/relativeLayout4"
android:layout_width="match_parent"
android:layout_height="wrap_content">

<LinearLayout
    android:id="@+id/linearLayout5"
android:layout_width="280dp"
android:layout_height="140dp"
android:layout_alignParentTop="true"
android:layout_centerHorizontal="true"
android:orientation="vertical">

<ImageView
    android:id="@+id/imageView1"
android:layout_width="match_parent"
android:layout_height="wrap_content"
    android:src="@drawable/a2" />
</LinearLayout>
    </RelativeLayout>

    <RelativeLayout
        android:id="@+id/relativeLayout5"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <LinearLayout
            android:id="@+id/linearLayout2"
            android:layout_width="280dp"
            android:layout_height="50dp">
```

```

android:layout_alignParentTop="true"
android:layout_centerHorizontal="true"
        android:orientation="vertical">

        <RelativeLayout
            android:id="@+id/relativeLayout3"
            android:layout_width="match_parent"
            android:layout_height="15dp"
            android:orientation="vertical">
        </RelativeLayout>

<RelativeLayout
    android:id="@+id/relativeLayout6"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:text="SECURE-SMS"
    android:typeface="monospace"
    android:textStyle="bold"
    android:textColor="#f59829"
    android:textSize="20dp" />

</RelativeLayout>
</LinearLayout>
</RelativeLayout>

<RelativeLayout
    android:id="@+id/relativeLayout7"
    android:layout_width="match_parent"
    android:layout_height="234dp">

    <LinearLayout
        android:id="@+id/linearLayout1"
        android:layout_width="280dp"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:orientation="vertical">

        <RelativeLayout
            android:id="@+id/relativeLayout8"
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

<Button
    android:id="@+id/bacasms"
    android:layout_width="70dp"
    android:layout_height="70dp"
    android:layout_alignParentRight="true"
    android:layout_alignParentTop="true"
    android:layout_marginTop="18dp"

```

```

        android:layout_marginRight="18dp"
        android:background="@drawable/inbox" />

<Button
    android:id="@+id/TulisPesan"
    android:layout_width="70dp"
    android:layout_height="70dp"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:layout_marginTop="18dp"
    android:layout_marginLeft="18dp"
    android:background="@drawable/write" />
</RelativeLayout>

    <RelativeLayout
        android:id="@+id/relativeLayout1"
        android:layout_width="match_parent"
        android:layout_height="15dp"
        android:orientation="vertical">

        </RelativeLayout>

<RelativeLayout
    android:id="@+id/relativeLayout2"
    android:layout_width="match_parent"
    android:layout_height="94dp">

<Button
    android:id="@+id/about"
    android:layout_width="65dp"
    android:layout_height="65dp"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:layout_marginBottom="18dp"
    android:background="@drawable/about" />

</RelativeLayout>
</LinearLayout>

</RelativeLayout>

</LinearLayout>

```

2. Coding tampilan main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal"
    android:background="@drawable/background"

```



```

android:layout_height="wrap_content"
android:text="Kunci"
android:textAppearance="?android:attr/textAppearanceMedium" />

<ScrollView
    android:id="@+id/scrollView2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <LinearLayout
        android:id="@+id/linearLayout2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">

        <EditText
            android:id="@+id/kunci"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:scrollbarAlwaysDrawVerticalTrack="true"
            android:scrollbarStyle="insideOverlay"
            android:scrollbars="vertical" />

    </LinearLayout>

</ScrollView>

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Pesan"
    android:textAppearance="?android:attr/textAppearanceMedium" />

<ScrollView
    android:id="@+id/scrollView3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <LinearLayout
        android:id="@+id/linearLayout4"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

        <EditText
            android:id="@+id/pesan"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="textMultiLine"
            android:scrollbars="vertical" />

    </LinearLayout>

</ScrollView>

<RelativeLayout

```

```

        android:id="@+id/relativeLayout2"
android:layout_width="match_parent"
android:layout_height="match_parent">

        <Button
        android:id="@+id/enkrip"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="10dp"
        android:background="@drawable/encrypt" />
</RelativeLayout>

<TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hasil"
        android:textAppearance="?android:attr/textAppearanceMedium" />

<ScrollView
        android:id="@+id/scrollView4"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <LinearLayout

                android:id="@+id/linearLayout5"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical">

                <EditText
                android:id="@+id/hasil"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:inputType="textMultiLine"
                android:scrollbarStyle="insideOverlay"
                android:scrollbars="vertical" />

                </LinearLayout>
        </ScrollView>

<RelativeLayout
        android:id="@+id/relativeLayout3"
        android:layout_width="match_parent"
        android:layout_height="match_parent">

        <Button
        android:id="@+id/kirim"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="10dp"
        android:background="@drawable/send" />

```

```
</RelativeLayout>
```

```
</LinearLayout>
```

```
</ScrollView>
```

```
</LinearLayout>
```

3. Coding tampilan inbox.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
    android:background="@drawable/background"
android:orientation="vertical">

<ListView
    android:id="@+id/list"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

</ListView>

</LinearLayout>
```

4. Coding Tampilan baca_sms.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent"
    android:background="@drawable/background"
android:orientation="vertical">

<ScrollView
    android:id="@+id/scrollView1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

<LinearLayout
    android:id="@+id/linearLayout1"
    android:layout_width="match_parent"
    android:layout_height="989dp"
    android:orientation="vertical"
>
```

```

<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Nomor Pengirim"
    android:textAppearance="?android:attr/textAppearanceMedium"
/>

<EditText
    android:id="@+id/noPengirim"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:inputType="phone"
>

    <requestFocus />
</EditText>

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Pesan"
    android:textAppearance="?android:attr/textAppearanceMedium"
/>

    <ScrollView
    android:id="@+id/scrollView2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    >

        <LinearLayout
    android:id="@+id/linearLayout2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">

            <EditText
    android:id="@+id/pesan2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:editable="false"
    android:inputType="textMultiLine"
    android:enabled="true"
    />

                </LinearLayout>
        </ScrollView>

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Kunci"
    android:textAppearance="?android:attr/textAppearanceMedium"
/>

```

```

        <ScrollView
            android:id="@+id/scrollView3"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
        >

        <LinearLayout
            android:id="@+id/linearLayout3"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
        >

        <EditText
            android:id="@+id/kunci2"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
        />
            </LinearLayout>
        </ScrollView>

        <RelativeLayout
            android:id="@+id/relativeLayout2"
            android:layout_width="match_parent"
            android:layout_height="match_parent">

            <Button
                android:id="@+id/dekrip"
                android:layout_width="50dp"
                android:layout_height="50dp"
                android:layout_alignParentTop="true"
                android:layout_centerHorizontal="true"
                android:layout_marginTop="10dp"
                android:background="@drawable/decrypt" />
        </RelativeLayout>

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hasil"
        android:textAppearance="?android:attr/textAppearanceMedium"
    />

    <ScrollView
        android:id="@+id/scrollView4"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
    >

    <LinearLayout
        android:id="@+id/linearLayout4"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"

```

```

        android:orientation="vertical"
    >

    <EditText
        android:id="@+id/hasil2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:inputType="textMultiLine"
        android:editable="false"
    />
        </LinearLayout>
    </ScrollView>

</LinearLayout>
</ScrollView>

<Button
    android:id="@+id/buttonForward"
    android:layout_width="50dp"
    android:layout_height="50dp"
    android:layout_below="@+id/scrollView1"
    android:layout_centerHorizontal="true"
    android:background="@drawable/forward_icon" />

</RelativeLayout>

```

5. Coding tampilan about.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:background="#ffffff">
    <RelativeLayout
        android:id="@+id/relativeLayout1"
        android:layout_width="match_parent"
        android:layout_height="25dp"
        android:orientation="vertical">
    </RelativeLayout>

    <RelativeLayout
        android:id="@+id/relativeLayout2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
    >

    <ImageView
        android:id="@+id/imageView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"

```

```

        android:src="@drawable/b1"
    />
</RelativeLayout>

<LinearLayout
    android:id="@+id/linearLayout1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
>

<RelativeLayout
    android:id="@+id/relativeLayout3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:text="SMS Secure"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:textColor="#000000"
    />
</RelativeLayout>
</LinearLayout>

<LinearLayout
    android:id="@+id/linearLayout2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
>

<RelativeLayout
    android:id="@+id/relativeLayout4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
>

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:text="Ver 1.0.1"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textColor="#000000"
    />
</RelativeLayout>
</LinearLayout>

<RelativeLayout

```

```

        android:id="@+id/relativeLayout5"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
    >

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:text="Irvander Grey Genggang"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textColor="#000000"
    />
</RelativeLayout>

<RelativeLayout
    android:id="@+id/relativeLayout6"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    >

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:text="11024011"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textColor="#000000"
    />
</RelativeLayout>

    <RelativeLayout
        android:id="@+id/relativeLayout7"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
    >

    <TextView
        android:id="@+id/textView5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:text="Teknik Informatika"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textColor="#000000"
    />
</RelativeLayout>

<RelativeLayout
    android:id="@+id/relativeLayout8"
    android:layout_width="match_parent"

```

```
android:layout_height="wrap_content"
>

<TextView
    android:id="@+id/textView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:text="Politeknik Negeri Manado"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textColor="#000000"
/>
</RelativeLayout>

<RelativeLayout
    android:id="@+id/relativeLayout9"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
>

<TextView
    android:id="@+id/textView7"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:text="2015"
    android:textAppearance="?android:attr/textAppearanceMedium"
    android:textColor="#000000"
/>
</RelativeLayout>
</LinearLayout>
```

DAFTAR PUSTAKA

AndiKurniawanDwi P. (2012). *PenerapanAlgoritmaVigenereCipher padaAplikasi SMS Android*.

[https://id.wikipedia.org/wiki/Android_\(sistem_operasi\)](https://id.wikipedia.org/wiki/Android_(sistem_operasi))

Di akses 2 Agustus 2015

[https://id.wikipedia.org/wiki/Eclipse_\(perangkat_lunak\)](https://id.wikipedia.org/wiki/Eclipse_(perangkat_lunak))

Di akses 29 Juli 2015

https://id.wikipedia.org/wiki/Layanan_pesan_singkat

Di akses 20 Juli 2015

<http://informatika.stei.itb.ac.id/~rinaldi.munir/Kriptografi/2011-2012/Makalah-2012/Makalah-Kripto-2012-031.pdf>

Di akses 25 Juli 2015

<https://herupurwito.wordpress.com/2013/04/11/ccontoh-aplikasi-sms-sederhana-pada-android/>

Di akses 3 Juli 2015

Safaat H., Nazruddin MT. (2013). *AplikasiBerbasis Android*, edisi 1. Bandung: PenerbitInformatika Bandung.