MEMBANGUN APLIKASI SMS MENGGUNAKAN ALGORITMA CAESAR CIPHER BERBASIS ANDROID

Muhamad Dinar Syafanda

Program Studi Teknik Informatika, STMIK Widya Cipta Dharma
Jl. Prof. M. Yamin No. 25 Samarinda Kalimantan Timur 75123
Telp: (0541) 736071, Fax: (0541) 203492
E-mail:dinarsyafa@gmail.com

ABSTRAK

Layanan aplikasi SMS bawaan yang digunakan pada perangkat android merupakan jalur yang tak aman dalam pertukaran informasi. Oleh karena itu, dibangunlah suatu aplikasi SMS yang dapat menenkripsi dan mendekripsi informasi tersebut dengan menggunakan algoritma *Caesar Cipher*. Dalam algoritma ini tidak hanya menggunakan karakter alfabet tetapi juga karakter ataupun simbol yang terdapat pada ASCII (*American Standard Code for Information Interchange*).

Adapun metode pengembangan sistem yang digunakan dalam penelitian ini yaitu *waterfall.* Di dalam metode ini terdapat beberapa tahapan diantaranya, analisis, desain, pengkodean, pengujian dan pemeliharaan.

Adapun cara pembuatan aplikasi SMS menggunakan algoritma *Caesar Cipher* berbasis android dengan melalui beberapa tahapan yang terdapat pada metode *waterfall*. Pada tahap pertama melakukan analisis pengguna, analisis sistem serta analisis perangkat lunak dan perangkat keras. Selanjutnya melalui tahap desain untuk memberikan gambaran umum mengenai sistem yang diusulkan dengan menggunakan UML (*Unified Modeling Language*) dan *flowchart.* Setelah itu melakukan tahap pengkodean dengan membuat program berdasarkan desain yang telah dibuat. Kemudian melalui tahapan pengujian *black box* dan *white box* terhadap program yang sudah dibuat. Apabila terjadi kesalahan maka lakukan tahap pemeliharaan dengan memperbaiki program. Saat aplikasi siap untuk digunakan, diharapkan aplikasi ini dapat digunakan sebagai salah satu media komunikasi rahasia oleh masyarakat.

Kata Kunci : algoritma caesar cipher, SMS, android, enkripsi, dekripsi, kriptografi

1. PENDAHULUAN

Salah satu aplikasi yang sering digunakan pada perangkat android saat ini adalah aplikasi SMS. Aplikasi SMS ini dapat mengirim pesan singkat kepada pengguna ponsel lainnya. Layanan SMS dengan menggunakan aplikasi bawaan ponsel merupakan jalur yang tak aman dalam melakukan pertukaran informasi. Pesan singkat yang dikirim menggunakan aplikasi SMS bawaan ponsel masih berupa teks terbuka yang tak terptoteksi, selain itu proses pengiriman pesan tidak sampai ke penerima secara langsung. Melainkan pengirim SMS harus melewati SMSC (*Short Message Service Center*) yang berfungsi mencatat komunikasi yang terjadi antara pengirim dan penerima. Dengan tersimpannya SMS pada SMSC, maka seorang operator dapat memperoleh informasi atau membaca SMS pada SMSC tersebut.

Oleh karena itu, dibutuhkan suatu metode yang dapat di implementasikan kedalam sebuah aplikasi atau program untuk dapat mengenkripsi dan memprotect pesan singkat tersebut. Enkripsi adalah proses mengubah pesan asli ke sebuah sandi atau kode yang tidak terbaca dan tidak dapat dimengerti, untuk bisa dibaca oleh orang yang kita inginkan diperlukan proses dekripsi yaitu membuka *plaintext* atau pesan asli dari ciphertext atau kode. Enkripsi dimaksudkan melindungi dan menyamarkan informasi agar tidak terbaca oleh orang lain atau pihak yang bukan semestinya.

Algortima caesar cipher merupakan subsitusi kode pertama dalam dunia penyandian terjadi pada pemerintahan Yulius Caesar, dengan mengganti posisi huruf awal dari alfabet. Namun, dalam penelitian ini yang berbeda dalam penggunaan algoritma tersebut yaitu pergantian karakter awal dilakukan pada karakter yang terdapat pada ASCII (*American Standard Code for Information Interchange*). Algoritma ini disebut sebagai algoritma kunci simetris (symmetric key) yaitu suatu enkripsi dengan menggunakan kunci yang sama untuk melakukan proses enkripsi dan dekripsi.

Berdasarkan pemaparan diatas, maka dibangun sebuah aplikasi SMS berbasis android yang mampu

melakukan enkripsi dan dekripsi dengan menggunakan algoritma caesar cipher.

2. RUANG LINGKUP PENELITIAN

2.1 Rumusan Masalah

Berkaitan dengan latar belakang diatas, maka hal yang perlu dirumuskan dalam masalah ini yaitu :

"Bagaimanakah Membangun Aplikasi SMS Menggunakan Algoritma Caesar Cipher Berbasis Android ?"

2.2 Batasan Masalah

Batasan masalah pada penelitian ini adalah :

1. Kunci yang digunakan berupa angka dari 1 sampai 255
2. Diterapkan pada ponsel berbasis android
3. Aplikasi ini dibuat dengan versi android 4.2.2
4. Menggunakan algoritma caesar cipher
5. Penulisan pesan yang dapat digunakan hanya berbentuk simbol yang tertera pada tabel ASCII
6. Aplikasi ini tidak menampilkan simbol yang sesuai dengan tabel ASCII jika simbol yang digunakan memiliki kode ASCII dari 128 sampai 160
7. Pesan hanya dapat dikirim ke satu nomor tujuan dalam sekali kirim
8. Aplikasi ini tidak ada pengaturan untuk memilih sim card yang digunakan untuk mengirim pesan
9. Hanya dapat menghapus satu pesan dalam sekali hapus
10. Dua belah pihak pengguna harus sama – sama menggunakan aplikasi ini
11. Aplikasi ini tidak memberitahukan pihak penerima pesan tentang kunci
12. yang di gunakan oleh pihak pengirim
13. Aplikasi ini tidak memberitahukan pihak penerima pesan untuk menggunakan aplikasi yang sama.

3. BAHAN DAN METODE

Adapun bahan dan metode yang digunakan dalam sistem ini adalah :

3.1. Aplikasi

Menurut Madcom (2008), Pemrograman aplikasi atau biasa disebut dengan aplikasi merupakan program yang berjalan pada sistem operasi dan dibuat untuk membuat pengguna mengerjakan suatu untuk meningkatkan produktivitasnya.

Menurut Dhanta (2009), Aplikasi adalah software yang dibuat oleh suatu perusahaan computer untuk mengerjakan tugas-tugas tertentu, misalnya Microsoft Word, MicrosoftExcel.

Beberapa aplikasi yang digabung bersama menjadi suatu paket kadang disebut sebagai suatu paket atau suite aplikasi (application suite). Contohnya adalah Microsoft Office dan Kingsoft Office Suite, yang menggabungkan suatu aplikasi pengolah kata, lembar kerja, serta beberapa aplikasi lainnya.

Aplikasi-aplikasi dalam suatu paket biasanya memiliki antarmuka pengguna yang memiliki kesamaan sehingga memudahkan pengguna untuk mempelajari dan menggunakan tiap aplikasi. Sering kali, mereka memiliki kemampuan untuk saling berinteraksi satu sama lain sehingga menguntungkan pengguna. Contohnya, suatu lembar kerja dapat dibenamkan dalam suatu dokumen pengolah kata walaupun dibuat pada aplikasi lembar kerja yang terpisah.

3.2 SMS (*Short Message Service*)

Menurut Rosidi (2004), *Short Message Service* (SMS) merupakan sebuah layanan yang banyak di aplikasikan pada sistem komunikasi tanpa kabel, memungkinkan dilakukannya pengiriman pesan dalam bentuk *alphanumeric* antara terminal pelanggan atau antara terminal pelanggan dengan sistem eksternal seperti email, *paging*, *voice mail*, dan lain-lain.

3.3 Algoritma *Caesar Cipher*

Menurut Ariyus (2008), subsitusi kode yang pertama dalam dunia penyandian terjadi pada pemerintahan Yulius Caesar yang dikenal dengan kode kaisar, dengan mengganti posisi huruf awal dari alfabet atau disebut juga dengan algoritma ROT3.

Tabel 1. Contoh *Caesar Cipher* (ROT3)

Plain Text	Encoded Text
ABC	DEF
HELLO	KHOOR
ATTACK	DWWDFN

Secara lebih detail, coba perhatikan contoh berikut :

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Jika penggeseran yang dilakukan sebanyak tiga kali maka kunci untuk dekripsinya adalah 3. Penggeseran kunci yang dilakukan tergantung keinginan pengirim pesan. Bisa saja kunci yang dipakai a = 7, b = 9, dan seterusnya.

Proses penyandian (enkripsi) dapat secara matematis menggunakan operasi modulus dengan mengubah huruf-huruf menjadi angka, A = 0, B = 1,..., Z = 25. Contoh algoritma kode kaisar : untuk teks-asli diberikan symbol "P" dan teks kodenya "C" dan kunci "K". Jadi rumusnya dapat di buat sebagai berikut :

$$C=E(P)=(P+K) \bmod (26)$$

Pada contoh di atas kita bisa memasukkan kunci dengan nilai tiga sehingga menjadi :

$$C=E(P)=(P+3) \bmod (26)$$

Rumus dekripsinya menjadi seperti berikut :

$$P=D(C)=(C - K) \bmod (26)$$

Dari contoh di atas, dengan memasukkan kunci tiga, maka :

$$P=D(C)=(C – 3) \bmod (26)$$

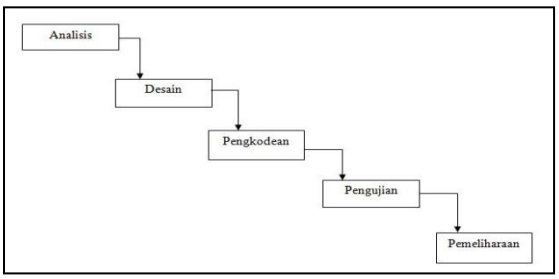
Kode kaisar dipecahkan dengan cara *brute force attack*, suatu bentuk serangan yang dilakukan dengan mencoba-coba berbagai kemungkinan untuk menemukan kunci. Bisa saja menggunakan *exhaustive key search*, karena jumlah kunci sangat sedikit (hanya ada 26 kunci). Meski sedikit, kunci suatu kode cukup merepotkan kriptanalis, karena untuk menemukannya denga *brute force attack* dibutuhkan waktu yang cukup lama.

3.4 *Android*

Menurut Safaat (2012), android merupakan subset perangkat lunak untuk perangkat mobile yang meliputi sistem operasi, milddeware dan aplikasi inti yang di realise oleh Google. Sedangkan android SDK (Software Development Kit) menyediakan tools dan API (Aplication Programing Interface) yang diperlukan untuk mengembangkan aplikasi pada platform android dengan menggunakan bahasa pemrograman Java.

3.5 Model *Waterfall*

Menurut Rosa dan Shalahuddin (2015), model SDLC air terjun (waterfall) sering juga disebut model sekuensial linier (sequential linear) atau alur hidup klasik (classic life cycle). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan pemeliharaan (maintenance). Berikut adalah gambar model air terjun :



Gambar 1. Ilustrasi model *waterfall*

1. Analisis kebutuhan perangkat lunak

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user.

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, dan prosedur pengodean.

3. Pembuatan kode program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian

Pengujian fokus pada perangkat lunak dari segi lojik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (error) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke user. Perubahan bisa terjadi Karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tetapi tidak untuk membuat perangkat lunak baru.

Dari kenyataan yang terjadi sangat jarang model air terjun dapat dilakukan sesuai alurnya karena sebab berikut :

a. Perubahan spesifikasi perangkat lunak terjadi di tengah alur pengembangan.
b. Sangat sulit bagi pelanggan untuk mendefinisikan semua spesifikasi di awal alur pengembangan.
c. Pelanggan tidak mungkin bersabar mangakomodasi perubahan yang diperlukan di akhir alur pengembangan.

Dengan berbagai kelemahan yang dimiliki model air terjun tapi model ini telah menjadi dasar dari model-model yang lain dalam melakukan perbaikan model pengembangan perangkat lunak.

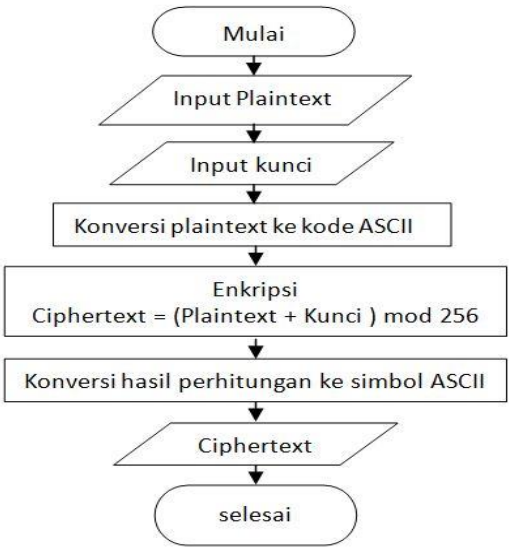
Model air terjun sangat cocok digunakan kebutuhan pelanggan sudah sangat dipahami dan kemungkinan terjadinya perubahan kebutuhan selama pengembangan perangkat lunak kecil.

Hal positif dari model air terjun adalah struktur tahap pengembangan sistem jelas, dokumentasi di hasilkan di setiap tahap pengembangan, dan sebuah tahap dijalankan setelah tahap sebelumnya selesai dijalankan (tidak ada tumpang tindih pelaksanaan tahap).

4 RANCANGAN SISTEM

4.1 *Flowchart* Enkripsi *Caesar Cipher*

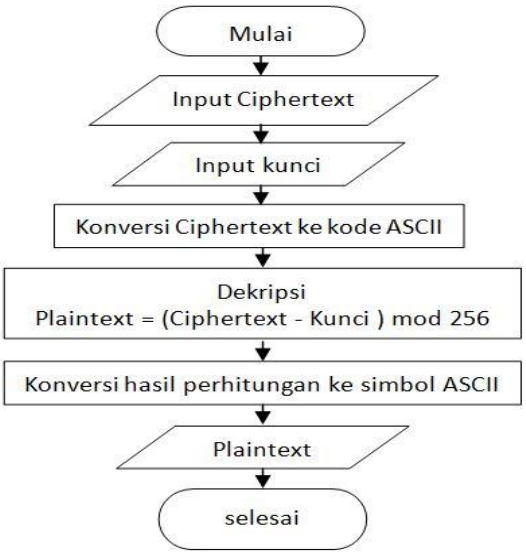
Adapun *flowchart* enkripsi *caesar cipher* yang digunakan pada aplikasi yang akan dibangun dapat dilihat pada gambar 2.



Gambar 2. *Flowchart* Enkripsi *Caesar Cipher*

4.2 *Flowchart* Dekripsi *Caesar Cipher*

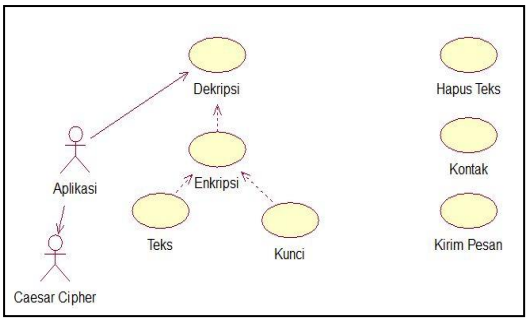
Adapun *flowchart* dekripsi *caesar cipher* yang digunakan pada aplikasi yang akan dibangun dapat dilihat pada gambar 3.



Gambar 3. *Flowchart* Enkripsi *Caesar Cipher*

4.3 *Use Case Diagram*

Use case diagram pada aplikasi SMS menggunakan algoritma *caesar cipher* yang dapat dilihat pada gambar 4.



Gambar 4. *Use Case Diagram*

Berdasarkan gambar 4, dapat diketahui prosedur rancangan dan interaksi yang terdapat di dalam aplikasi. Pada aplikasi terdapat satu class yaitu *caesar cipher* yang terdiri dari dekripsi, enkripsi, teks dan kunci.

Dekripsi hanya dapat terjadi apabila syarat-syarat dekripsi telah terpenuhi, adapun syarat-syarat tersebut yaitu adanya teks hasil enkripsi atau *ciphertext* dan kunci. Sedangkan agar enkripsi dapat terjadi syarat yang harus terpenuhi yaitu adanya teks yang akan di enkripsi dan kunci.

Usecase hapus teks, kontak dan kirim pesan merupakan *usecase* pendukung dari sistem yang berguna untuk manghapus teks, menampilkan daftar kontak dan mengirim pesan hasil enkripsi.

4.4 *Activity Diagram* Enkripsi dan Dekripsi Sistem

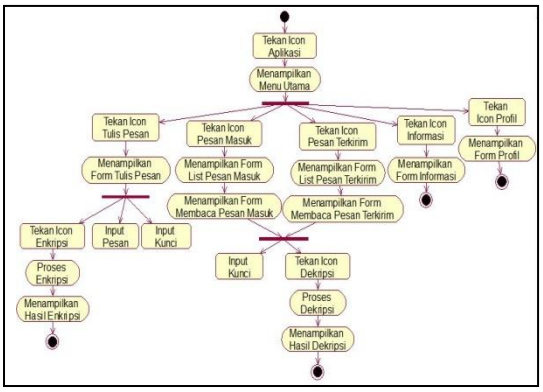
Pada Activity diagram enkripsi dan dekripsi sistem, menggambarkan kejadian atau aktivitas-aktivitas yang terjadi pada sistem yang dibangun. Dimana dimulai dengan tekan icon aplikasi sehingga menampilkan menu utama, di dalam menu utama terdapat empat aktivitas yang dapat dilakukan yaitu : tekan icon tulis pesan, tekan icon pesan masuk, tekan icon pesan terkirim dan tekan icon informasi.

Pada aktivitas tekan icon tulis pesan, aplikasi menampilkan form tulis pesan yang berisikan aktivitas input pesan, input kunci dan tekan icon enkripsi. Dimana yang terjadi pada aktivitas tekan icon enkripsi, aplikasi melakukan proses enkripsi kemudian menampilkan hasil enkripsi (Ciphertext).

Pada aktivitas tekan icon pesan masuk, aplikasi menampilkan list pesan masuk kemudian menampilkan form membaca pesan masuk. Pada form membaca pesan masuk ketika melakukan aktivitas input kunci dan tekan icon dekripsi kemudian aplikasi melakukan proses dekripsi kemudian menampilkan hasil dekripsi. Begitupun yang terjadi pada aktivitas tekan icon pesan terkirim.

Pada aktivitas tekan icon informasi aplikasi menampilkan form informasi. Selanjutnya pada aktivitas tekan icon profil aplikasi menampilkan profil.

Adapun Activity diagram enkripsi dan dekripsi sistem yang dapat dilihat pada gambar 5.



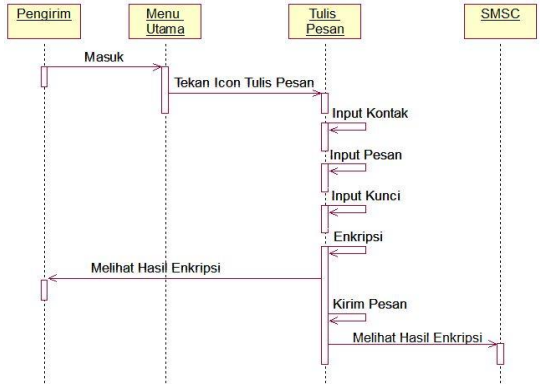
Gambar 5. *Activity Diagram* Enkripsi dan Dekripsi Sistem

4.5 *Sequence Diagram*

Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah skenario. Diagram ini menunjukkan sejumlah contoh objek dan pesan yang diletakkan diantara objek-objek ini di dalam *usecase*. Adapun *Sequence diagram* yang digunakan pada aplikasi yang akan dibangun yaitu sebagai berikut :

1. *Sequence Diagram* Mengirim Pesan

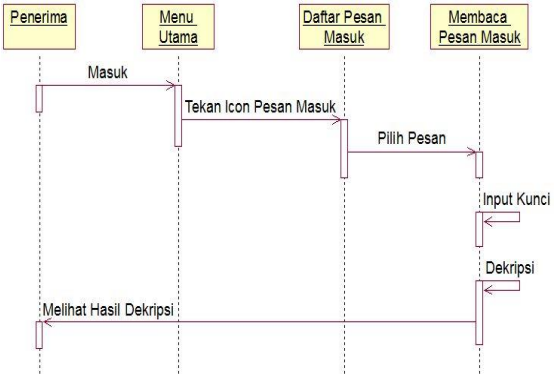
Pada Sequence diagram mengirim pesan, menggambarkan saat *user* (pengirim) pertama masuk dalam aplikasi maka akan ditampilkan menu utama, saat pengirim menekan icon tulis pesan maka form tulis pesan akan tampil. Dalam form tulis pesan, pengirim dapat memasukkan nomor telepon yang ingin dituju, pesan dan kunci. Pengirim harus melakukan proses enkripsi dengan menekan icon enkripsi. Selanjutnya hasil enkripsi dapat dilihat oleh pengirim dan dikirim ke nomor tujuan yang sudah dimasukkan sebelumnya dengan menekan icon kirim. Pada SMSC (*Short Message Service Center*) hanya dapat melihat hasil enkripsi yang dikirim. Adapun *sequence diagram* mengirim pesan yang dapat dilihat pada gambar 6.



Gambar 6. *Sequence Diagram* Mengirim Pesan

2. *Sequence Diagram* Membaca Pesan Masuk

Pada Sequence diagram membaca pesan masuk, menggambarkan saat user (penerima) pertama masuk dalam aplikasi maka akan ditampilkan menu utama, saat penerima menekan icon pesan masuk maka form daftar pesan masuk akan tampil. Dalam form daftar pesan masuk, pengirim dapat memilih pesan yang ingin di dekripsi. Kemudian akan ditampilkan form membaca pesan masuk. Selanjutnya, penerima memasukkan kunci yang digunakan oleh pengirim untuk mendekripsi pesan. Setelah pesan sudah terdekripsi, penerima dapat membaca pesan hasil dekripsi. Adapun sequence diagram membaca pesan masuk yang dapat dilihat pada gambar 7.

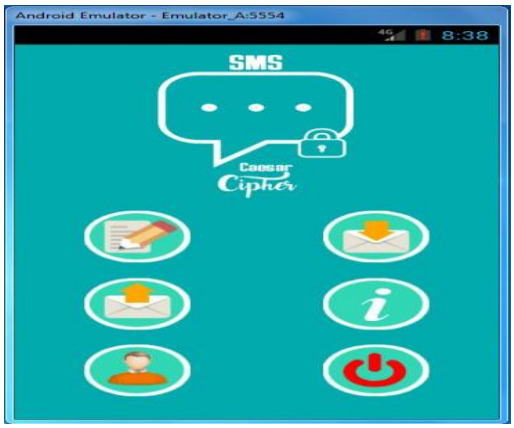


Gambar 7. *Sequence Diagram* Membaca Pesan Masuk

5 IMPLEMENTASI

5.1 Tampilan *Form* Menu Utama

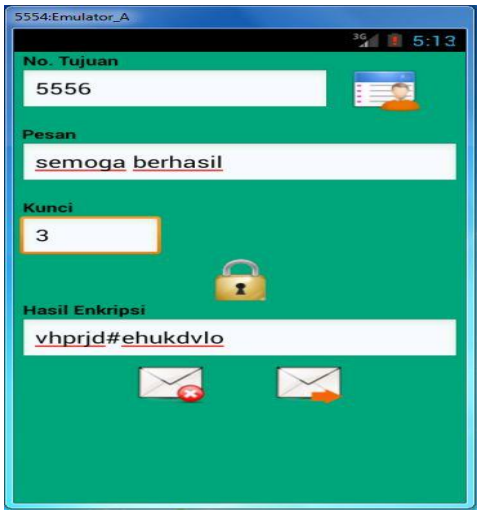
Adapun tampilan form menu utama yang terdapat pilihan menu tulis pesan, pesan masuk, pesan terkirim, informasi, profil dan keluar. Adapun tampilan form menu utama yang dapat dilihat pada gambar 8.



Gambar 8. Tampilan *Form* Menu Utama

5.2 Tampilan *Form* Tulis Pesan

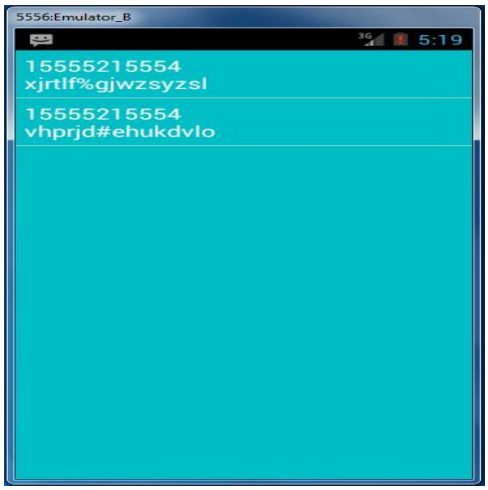
Pada tampilan form tulis pesan terdapat button kontak untuk mengakses daftar no telpon yang tersimpan dalam memori handphone. Kemudian dapat di isi secara manual, *edit text* kunci dapat di isi kunci yang diinginkan pengguna namun hanya dapat menggunakan angka dari 1- 255, *edit text* pesan dapat di isi pesan yang akan dikirim, tekan tombol *button* enkripsi untuk melihat hasil enkripsi nya. *Edit text Ciphertext* akan menampilkan hasil enkripsi. Tekan tombol kirim untuk mengirim pesan. Adapun tampilan halaman tulis pesan yang dapat dilihat pada gambar 9.



Gambar 9. Tampilan *Form* Tulis Pesan

5.3 Tampilan *Form List* Pesan Masuk

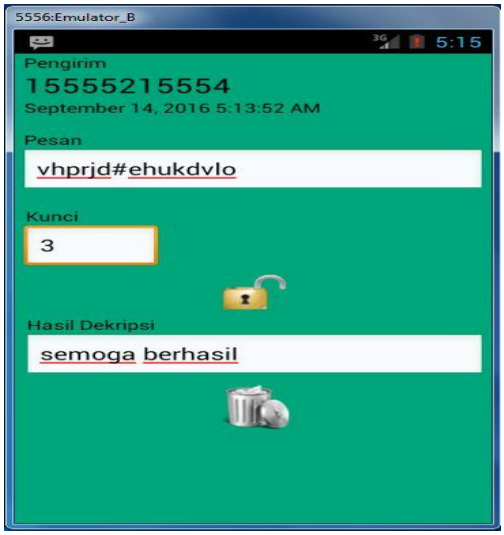
Pada tampilan list pesan masuk menampilkan nomor pengirim dan teks pesan dengan maksimal 20 karakter. Pesan yang ditampilkan tersebut berasal dari memori perangkat yang telah tersimpan sebelumnya. Ketika memilih salah satu list pesan masuk, maka akan menampilkan pesan secara keseluruhan. Adapun tampilan form list pesan masuk yang dapat dilihat pada gambar 10.



Gambar 10. Tampilan *Form List* Pesan Masuk

5.4 Tampilan *Form* Membaca Pesan Masuk

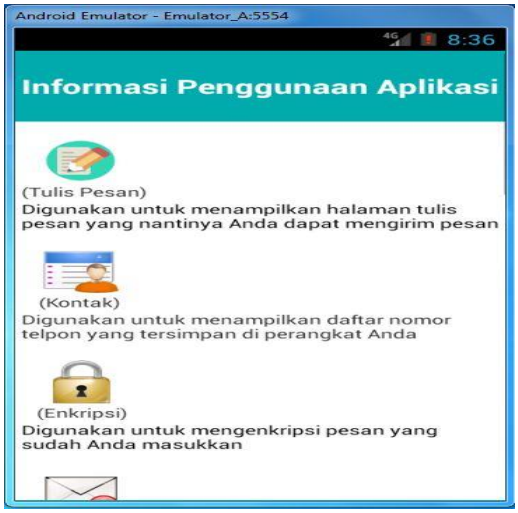
Pada tampilan form membaca pesan masuk akan menampilkan pesan masuk secara keseluruhan serta menampilkan waktu ketika pesan tersebut masuk. Jika pesan tersebut merupakan ciphertext maka pada kolom kunci di isi dengan kunci yang digunakan oleh pengirim pesan maka akan tampil isi pesan sesungguhnya dan jika kunci yang dimasukkan berrbeda dengan yang digunakan oleh pengirim maka output nya tidak sesuai dan berantakan. Adapun tampilan form membaca pesan masuk dapat dilihat pada gambar 11.



Gambar 11. Tampilan *Form* Membaca Pesan masuk

5.5 Tampilan *Form* Informasi

Pada tampilan form informasi yang didalamnya menampilkan tentang cara penggunaan aplikasi atau fungsi tombol-tombol yang terdapat pada aplikasi. Adapun tampilan form informasi yang dapat dilihat pada gambar 12.



Gambar 12. Tampilan *Form* Informasi

5.6 Tampilan *Form* Profil

Adapun tampilan form Profil yang didalamnya berisi tentang informasi pengembang aplikasi yang dapat dilihat pada gambar 13.

Gambar 13. Tampilan *Form* Profil

6 KESIMPULAN

Bedasarkan uraian pada bab-bab sebelumnya maupun pembahasan yang telah dikemukakan maka dapat ditarik kesimpulan sebagai berikut:

1. Cara membangun aplikasi SMS menggunakan algoritma caesar cipher berbasis Android dengan malalui tahapan analisis yang di dalamnya terdapat analisis user, analisis sistem serta analisis teknologi, kemudian tahapan desain yang didalamnya menggambarkan desain perancangan aplikasi yang akan di bangun, setelah itu melalui tahapan pengkodean yang menerapkan dari hasil desain ke dalam bentuk script, kemudian tahapan pengujian yang terdiri dari pengujian white box dan pengujian black box.
2. Berdasarkan pengujian white box dan black box maka aplikasi ini sudah berjalan dengan baik karena white box digunakan untuk pengujian script jika terdapat kesalahan dari user akan tampil pemberitahuan, sedangkan black box digunakan untuk pengujian button pada aplikasi sehingga menampilkan sesuai yang di inginkan oleh user dengan benar.

7 SARAN

Berdasarkan dari kesimpulan yang telah dikemukakan diatas, maka saran-saran yang dapat diberikan adalah sebagai berikut :

1. Aplikasi ini dapat dikembangkan dengan algoritma metode lain untuk lebih meningkatkan keamanan pesan yang dikirim.
2. Sebaiknya untuk pengembang selanjutnya bisa menambahkan pemberitahuan jumlah SMS yang akan dikirim.
3. Desain aplikasi masih kurang menarik belum terdapat audio-visual, oleh karena itu kedepannya desain aplikasi akan jauh lebih menarik dengan di tambahkannya animasi.

8 DAFTAR PUSTAKA

Ariyus, Dony. 2008. Pengantar Ilmu Kriptografi. Andi : Yogyakarta.

Dhanta, Rizky. 2009. Pengantar Ilmu Komputer. INDAH : Surabaya.

Harumy, T. Henny Febriana, Windarto, Agus Perdana dan Sulistianingsih, Indri. 2016. Belajar Algoritma & Pemrograman C++. Deepublish : Yogyakarta

Hermawan S., Stephanus. 2011. Mudah Membangun Aplikasi Android. Andi : Yogyakarta.

Kadir, Abdul dan Triwahyuni, Terra Ch. 2013. Pengantar Teknologi Informasi Edisi Revisi. Andi : Yogyakarta.

Madcom. 2008. Panduan Lengkap Untuk Teknisi Komputer. Andi : Yogyakarta.

Pressman, Roger S. 2010. Software Engineering : A Practicioner's Approach, 7th Edition. McGraw-Hill Inc. : New York.

Rosidi, R,. I. 2004. Membuat Sendiri SMS Gateway (ESME) Berbasis Protocol SMPP. Andi : Yogyakarta.

Safaat, Nazruddin. 2012. Android Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android. Informatika : Bandung.

Schneier, B. 1996.Applied Cryptography: Protocols, Algorithms, and Source

Code in C. 2nd Edition. John Wiley & Sons, Inc: New Jersey.

Shalaluddin, M. dan A. S., Rosa. 2015. Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Informatika : Bandung.

Simarmata, Janner. 2010. Rekayasa Perangkat Lunak. Andi : Yogyakarta.

Utami, Ema dan Raharjo, Suwanto. 2004. Logika Algoritma dan Implementasi dalam Bahasa Pythondi GNU/Linux. Andi : Yogyakarta.

Yakub. 2012. Pengantar Sistem Informasi. Graha Ilmu : Yogyakarta.