BAB III

ANALISIS DAN DESAIN SISTEM

III.1. Analisis Masalah

Handphone merupakan salah satu teknologi yang sangat diminati masyarakat dalam membantu pekerjaan, pendidikan yang memberikan informasi secara mudah dan praktis. Perkembangan teknologi *handphone* semakin meningkat sejak dirilisnya berbagai macam smartphone yang diiringi dengan berbagai macam aplikasi. Tidak sedikit orang yang ingin membuat aplikasi untuk mencari keuntungan dan kepuasan tersendiri bagi sipembuat aplikasi tersebut. terutama aplikasi android dimana dalam pembuatannya bersifat *open source* dan gratis

SMS merupakan aplikasi yang mudah digunakan untuk mengirim pesan dengan hanya mengetik pesan dan mengirimkan pesan tersebut ke nomor tujuan dengan harga yang murah tanpa adanya koneksi antara kedua pengguna. Tetapi aplikasi SMS belum terdapat keamanan tambahan untuk mengamankan pesan tersebut sampai ke nomor tujuan, baik dari jaringan maupun dari aplikasi tersebut, ini sangat berbahaya apabila ada informasi penting yang dapat dibaca dan disalah gunakan oleh orang yang tidak bertanggung jawab. Untuk mengatasi masalah ini penulis akan mengamankan pesan tersebut dengan mengubahnya ke dalam kalimat atau bilangan acak, teknik tersebut di dalam dunia komputer dikenal sebagai teknik kriptografi. Penulis akan menggunakan dua metode kriptografi yaitu metode *vigenere cipher* dan *one time pad* agar dalam pengamanan pesan tersebut sulit untuk dipecahkan.

III.2. Spesifikasi Perangkat

Spesifikasi perangkat adalah rincian perangkat yang digunakan dalam pembuatan aplikasi. Adapun perangkat yang digunakan oleh penulis adalah sebagai berikut:

1. Perangkat Lunak (*Software*)
   a. Sistem operasi windows 7 dan android 4.4.2, yaitu digunakan dalam perancangan dan tes untuk aplikasi mobile
   b. Android-sdk_r24.4.1, sebagai emulator android pada OS windows 7
   c. Eclipse indigo versi 3.7, sebagai editor kode program dan tampilan
   d. Java SE development Kit 6, sebagai bahasa pemograman dan compiler java
2. Perangkat Keras (*Hardware*)
   a. Laptop dengan processor intel core i3
   b. Smartphone android dengan OS 4.4.2

III.3. Teknik Pemecahan Masalah

Teknik pemecahan masalah yang diambil penulis terdapat beberapa langkah untuk menghasilkan perangcangan pengamanan SMS yang baik. Adapun langkah-langkah tersebut adalah sebagai berikut:

1. Hal pertama yang harus dilakukan adalah menganalisa tentang aplikasi SMS, metode yang digunakan dan cara kerja aplikasi tersebut.
2. Menentukan perangkat-perangkat apa saja yang dibutukan dalam melakukan perancangan aplikasi tersebut, baik perangkat keras maupun perangkat lunak.
3. Membuat desain sistem atau gambaran yang akan diterapkan pada aplikasi perancangan pengamanan SMS tersebut.
4. Mencoba dan menguji aplikasi tersebut dengan hasil yang diharapkan.

III.4. Penerapan Metode

Dalam penelitian pengamanan SMS ini penulis menerapkan dua metode kriptografi klasik yaitu metode *vigenere cipher* dan *one time pad*, dengan cara menggabungkan dua metode tersebut agar hasil *ciphertext* yang diperoleh lebih sulit untuk dipecahkan.

1. Metode *Vigenere Cipher*

Metode *vigenere cipher* adalah metode kriptografi klasik yang menggunakan abjad majemuk. Seperti yang sudah dijelaskan sebelumnya metode *vigenere cipher* hampir mirip dengan metode caesar perbedaannya terdapat pada pergeseran huruf, dimana metode caesar menggeser *plaintext* satu kali menurut key dibandingkan dengan metode *vigenere cipher* menggeser semua karakter di dalam *plaintext* menurut *key*. Rumus yang digunakan pada proses enkripsi dan deskripsi adalah sebagai berikut:

Proses Enkripsi:

Ci = (Pi + Ki) mod 26, atau Ci = (Pi + Ki) – 26 untuk Pi + Ki >= 26

Proses Dekripsi:

Pi = (Ci – Ki) mod 26, untuk Ci >= Ki

Pi = (Ci - Ki) + 26, untuk Ci <= Ki

Keterangan:

C = *Chiphertext*

P = *Plaintext*

K = *Kunci*

a. Enkripsi *Vigenere Cipher*

Enkripsi adalah langkah yang digunakan untuk mengamankan data informasi atau pesan dengan cara mengacak pesan tersebut menggunakan *key*. Keamanan pesan tergantung pada *key* yang digunakan, proses ini adalah proses mengubah *plaintext* menjadi *ciphertext*.

Enkripsi *vigenere cipher* dapat dijelaskan menggunakan tabel bujur sangkar yang terdapat 26 baris dan 26 kolom, dimana baris paling atas sebagai plaintext dan kolom paling kiri sebagai *key* seperti yang dijelaskan pada tabel III.1.

Tabel III.1. Enkripsi *Vigenere Cipher*

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

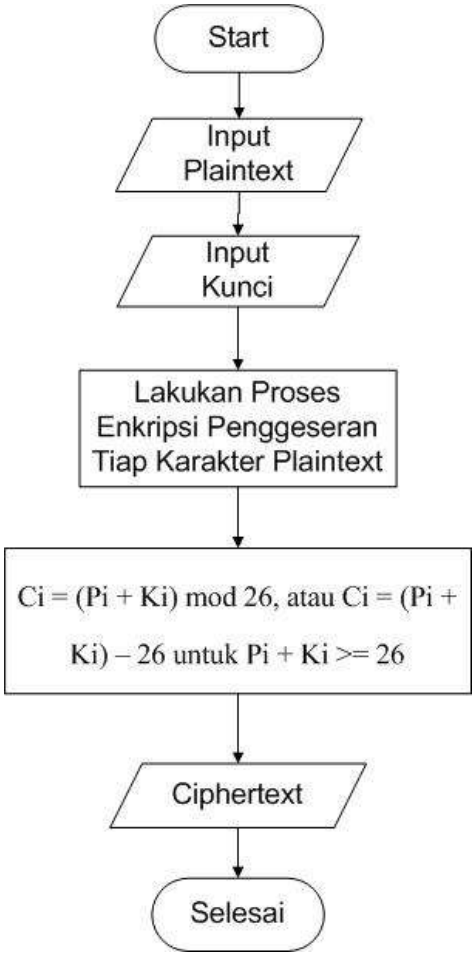
Contoh:

Plaintext = USAHA

Key = HASIL

Ciphertext = BSSPL

Proses dilakukan dengan menghubungkan setiap karakter *plaintext* dengan karakter *key* sesuai dengan nomor urut menggunakan bujur sangkar tersebut. Apabila *plaintext* lebih panjang dari *key* maka karakter *key* akan diulang sesuai dengan panjang *plaintext*. Gambar flowchart enkripsi *vigenere cipher* dapat dilihat pada gambar III.1.



Gambar III.1. Proses Enkripsi *Vigenere Cipher*

b. Deskripsi *Vigenere Cipher*

Deskripsi adalah proses dimana pesan acak akan diubah dalam *plaintext*. Proses deskripsi *vigenere cipher* pada abjad bujur sangkar yaitu:

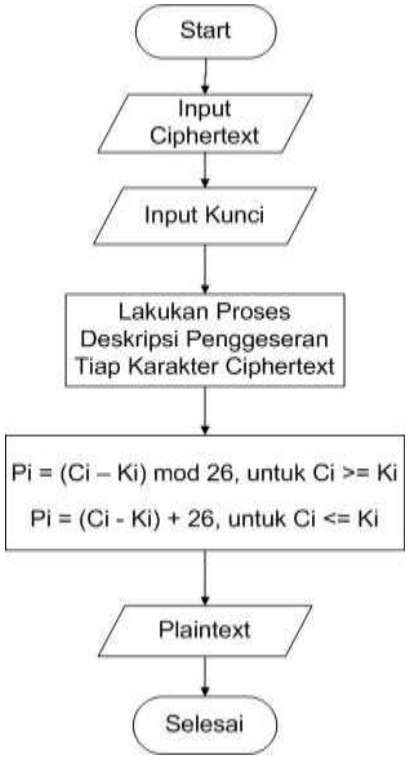
Diketahui:

Ciphertext = BSSPL

Key = HASIL

Lakukanlah menggunakan proses kebalikannya, karakter *ciphertext* pertama B dan *key* H, cari huruf B (*ciphertext*) pada baris H (*key*) dan akan didapat haruf U (*plaintext*) pada baris paling atas. Selanjutnya cari huruf S (*ciphertext*) pada baris A (*key*) dan akan didapat haruf S

(*plaintext*) pada baris paling atas dan seterusnya. Berikut proses deskripsi *vigenere cipher* dalam bentuk flowchart.



Gambar III.2. Proses Deskripsi *Vigenere Cipher*

2. Metode *One Time Pad* (OTP)

One Time Pad (OTP) atau juga disebut dengan *vernam cipher* merupakan metode kriptograpi klasik yang menggunakan kunci simetri yang berarti proses enkripsi dan deskripsinya menggunakan sandi yang sama. Dalam proses perhitungan algoritma *one time pad* tidak terlalu sulit dan mudah dipahami yaitu dengan perhitungan XOR dimana dalam proses perhitungannya akan menghasilkan 1 apabila 1 bertemu 0 atau 0 bertemu dengan1 dan sebaliknya apa bila 1 bertemu dengan 1 atau 0 bertemu 0 akan menghasilkan 0.

a. Enkripsi *One Time Pad*

Dalam proses perhitungan *one time pad* langkah pertama yang harus dilakukan adalah mengubah *plaintext* menjadi bilangan biner (1 dan 0) setelah itu ubah juga bilangan yang

terdapat pada *key* ke dalam bilangan biner (1 dan 0), lalu lakukan perhitungan XOR antara bilangan biner *plaintext* dan bilangan biner *key* setelah dapat bilangan biner baru, ubah bilangan biner tersebut kedalam bilangan ASCII. Dari penjelasan tersebut enkripsi *one time pad* dapat dilakukan menggunakan rumus sebagai berikut :

Ci = Pi XOR Ki

Keterangan :

C = *Ciphertext*

P = *Plaintext*

K = *Key*

i = Index karakter

Berikut contoh proses perhitungan enkripsi *one time pad*, Diketahui *plaintext* yaitu "USAHA" dengan menggunakan *key* "HASIL" maka perhitungan enkripsinya adalah sebagai berikut.

Plaintext :

U = 01010101

S = 01010011

A = 01000001

H = 01001011

A = 01000001

Key :

H = 01001011

A = 01000001

S = 01010011

I = 01001001

L = 01001100

Plaintext = 01010101 01010011 01000001 01001011 01000001

Key = 01001011 01000001 01010011 01001001 01001100

XOR

Ciphertext = 00011110 00010010 00010010 00000010 00001101

Ciphertext :

00011110 = RS

00010010 = DC2

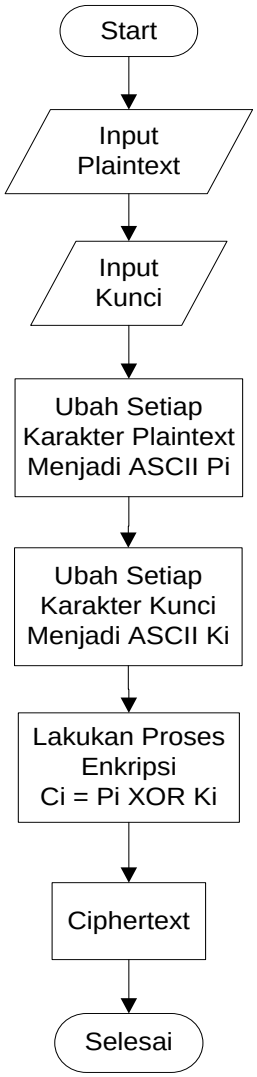
00010010 = DC2

00000010 = SOH

00001101 = CR

Jadi *ciphertext* yang didapat adalah “RS-DC2-DC2-SOH-CR”.

Gambar di bawah digunakan untuk menjelaskan proses enkripsi *one time pad* dalam bentuk flowchart, yaitu pada gambar III.3.



Gambar III.3. Proses Enkripsi *One Time Pad*

b. Deskripsi *One Time Pad*

Pada proses deskripsi ini menggunakan cara yang hampir sama dengan proses enkripsi di atas, berikut penjelasannya, Pertama *ciphertext* yang diperoleh dari hasil enkripsi *one time pad* ubah ke dalam bilangan biner (1 dan 0) lalu ubah juga bilangan pada *key* ke dalam bentuk bilangan biner (1 dan 0) selanjutnya lakukan perhitungan XOR pada bilangan biner *ciphertext* dan bilangan biner *key* maka hasil yang diperoleh adalah plaintext. Berikut contoh proses perhitungan deskripsi *one time pad* yaitu dengan menggunakan rumus Pi = Ci XOR Ki.

Diketahui *ciphertext* yang diperoleh adalah “RS-DC2-DC2-SOH-CR” dengan *key* “HASIL” maka proses perhitungan yang akan menghasilkan *plaintext* adalah sebagai berikut:

Ciphertext :

RS =00011110

DC2 = 00010010

DC2 = 00010010

SOH = 00000010

CR = 00001101

Key :

H = 01001011

A = 01000001

S = 01010011

I = 01001001

L = 01001100

Ciphertext = 00011110 00010010 00010010 00000010 00001101

Key = 01001011 01000001 01010011 01001001 01001100

XOR

Plaintext = 01010101 01010011 01000001 01001011 01000001

01010101 = U

01010011 = S

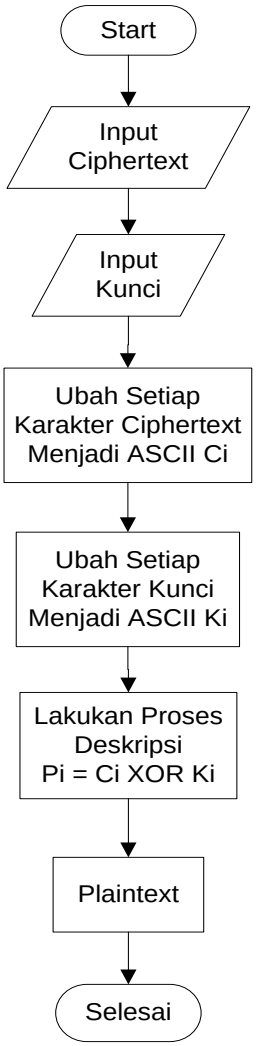
01000001 = A

01001011 = H

01000001 = A

Jadi *plaintext* yang didapat adalah “USAHA”.

Untuk menjelaskan proses deskripsi *one time pad* dalam bentuk flowchart, dapat dilihat pada gambar III.4.



Gambar III.4. Proses Deskripsi *One Time Pad*

Dari penjelasan metode *vigenere cipher* dan *one time pad* di atas penulis akan menggabungkan kedua metode tersebut pada aplikasi SMS sebelum dilakukan pengiriman, yaitu dengan cara mengenkripsi teks menggunakan algoritma vigenere terlebih dahulu kemudian algoritma one time pad.. Berikut contoh enkripsi *vigenere cipher* dan *one time pad:*

1. Enkripsi *vigenere cipher:*

   Plaintext = USAHA

Key = HASIL

Ciphertext Sementara = BSSPL

2. Enkrpsi *one time pad*

Plaintext = BSSPL

Key = HASIL

Ciphertext = LF-DC2-NULL-EM-NULL

Jadi, hasil enkripsi pesan "USAHA" dengan sandi "HASIL" yang akan dikirim menjadi *ciphertext* "LF-DC2-NULL-EM-NULL". Untuk hasil deskripsi dari *ciphertext* tersebut yaitu dengan cara kebalikannya, deskripsi *one time pad* lalu deskripsi *vigenere cipher*.

III.5. Rancangan *Database*

Database adalah informasi-informasi yang disimpan didalam suatu perangkat baik perangkat komputer maupun mobile yang tersusun secara sistematis dan dapat diatur dengan mudah menggunakan aplikasi. Dalam perancangan aplikasi pengamanan SMS sering kali kita lupa mengingat *password* yang diberikan, oleh karena itu penulis menggunakan *database* agar pengguna lebih mudah cara penggunaannya tanpa mengingat sandi.

Aplikasi *database* yang digunakan adalah SQLite. SQLite merupakan sistem *database* yang ukurannya tidak terlalu besar yang sudah ada pada perangkat android dan mirip seperti SQLServer yang menggunakan *insert edit delete* dan sebagainya. Adapun nama *database* yang dibuat adalah "DbSMS", menggunakan satu tabel yaitu dengan nama "tblPassword".

Tabel III.2. Keterangan tblPassword

Nama Field	Tipe Data	Keterangan
_id	*integer*	*Primary key* tabel
tanggal	*text*	Tanggal pesan dikirim/diterima
nama	*text*	Nama pengirim/penerima pesan
nomor	*text*	Nomor *handphone* pengirim/penerima pesan

password	*text*	Kunci atau sandi untuk enkripsi/enkripsi pesan
pesan	*text*	Pesan *Plaintext* (teks terang) yang dikirim

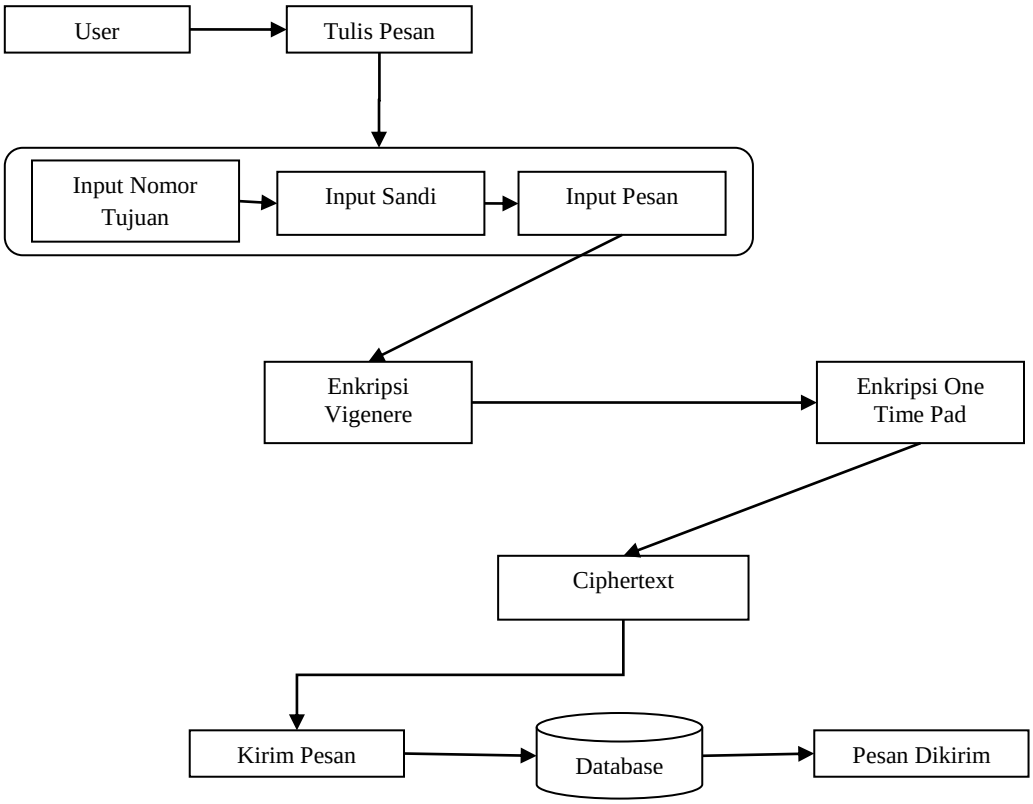
III.6. Desain Sistem

Desain sistem adalah gambaran perancangan tentang sistem, pengaturan yang akan diterapkan pada sitem dan elemen-elemen yang akan disatupadukan menjadi perangkat yang kita inginkan.

III.6.1. Diagram Blok

Diagram blok merupakan penjelasan dari suatu sistem menggunakan blok-blok yang digunakan untuk mencari keluaran, sebab, akibat dari masukan yang dibuat.

Pada sistem ini pengguna ingin mengirim pesan ke nomor tujuan menggunakan aplikasi keamanan SMS, Sebelum pesan dikirim pengguna harus mengisi *form* nomor tujuan, sandi, pesan. Setelah *form* tersebut lengkap diisi, pengguna dapat menekan tombol pesan dimana sebelum pesan dikirim aplikasi tersebut akan mengenkripsi pesan menggunakan algoritman *vigenere cipher* dan *one time pad.*

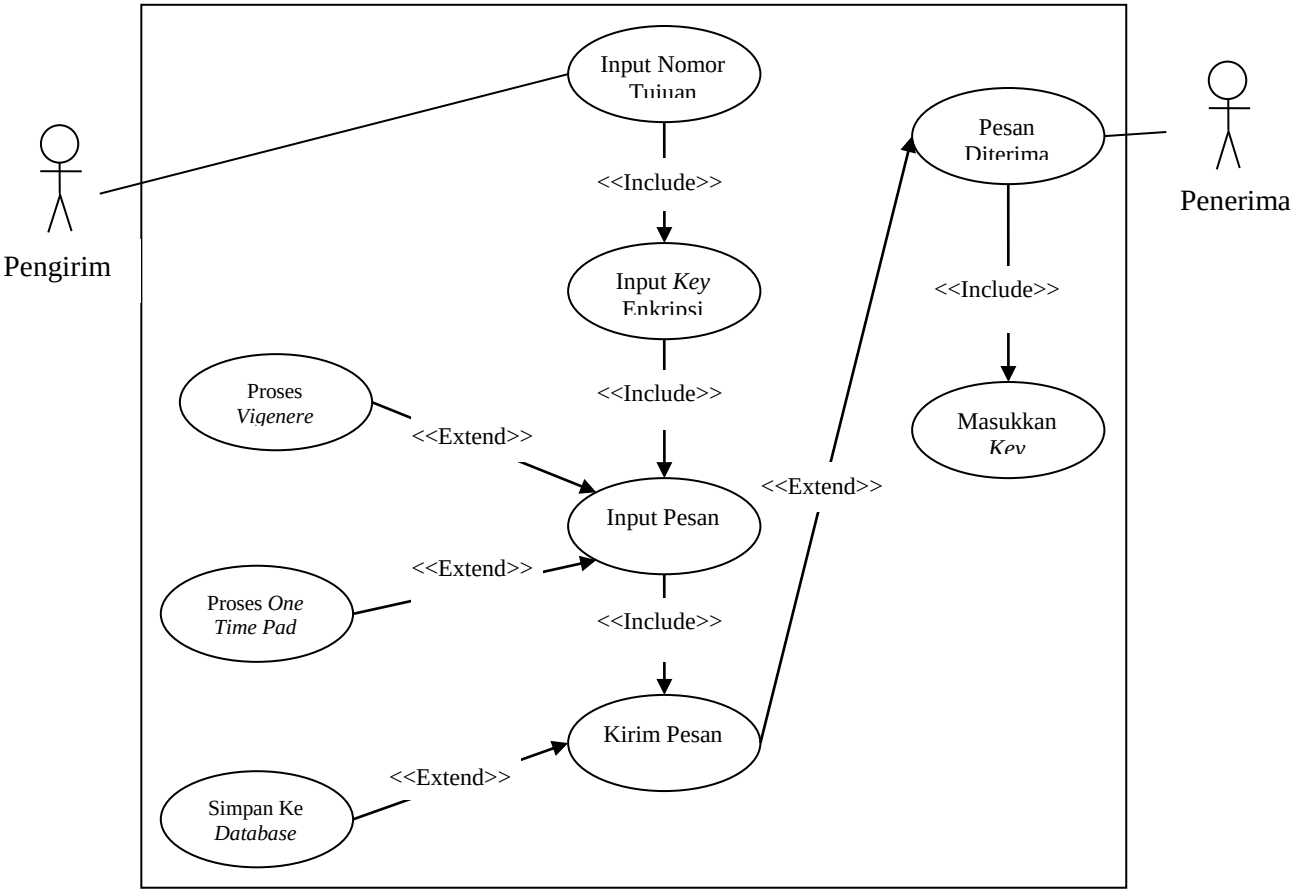


Gambar III.5. Diagram Blok Mengirim Pesan Aplikasi Pengamanan SMS

III.6.2. *Use Case Diagram*

Pada diagram ini memperlihatkan bagaimana kejadian dari aktor pengirim dan penerima. Pengirim akan melakukan beberapa aktivitas di *use case* dalam mengirim pesan mulai dari mengetik pesan, nomor, sandi dan lain lain, begitu juga penerima melakukan input sandi, deskripsi pesan dan sebagainya. Adapun *use case diagram* dalam perangcangan pengamanan SMS tersebut dapat dilihat pada gambar III.6.

juga penerima melakukan input sandi, deskripsi pesan dan sebagainya. Adapun *use case diagram* dalam perangcangan pengamanan SMS tersebut dapat dilihat pada gambar III.6.



Gambar III.6. *Use Case Diagram* Aplikasi Pengamanan SMS

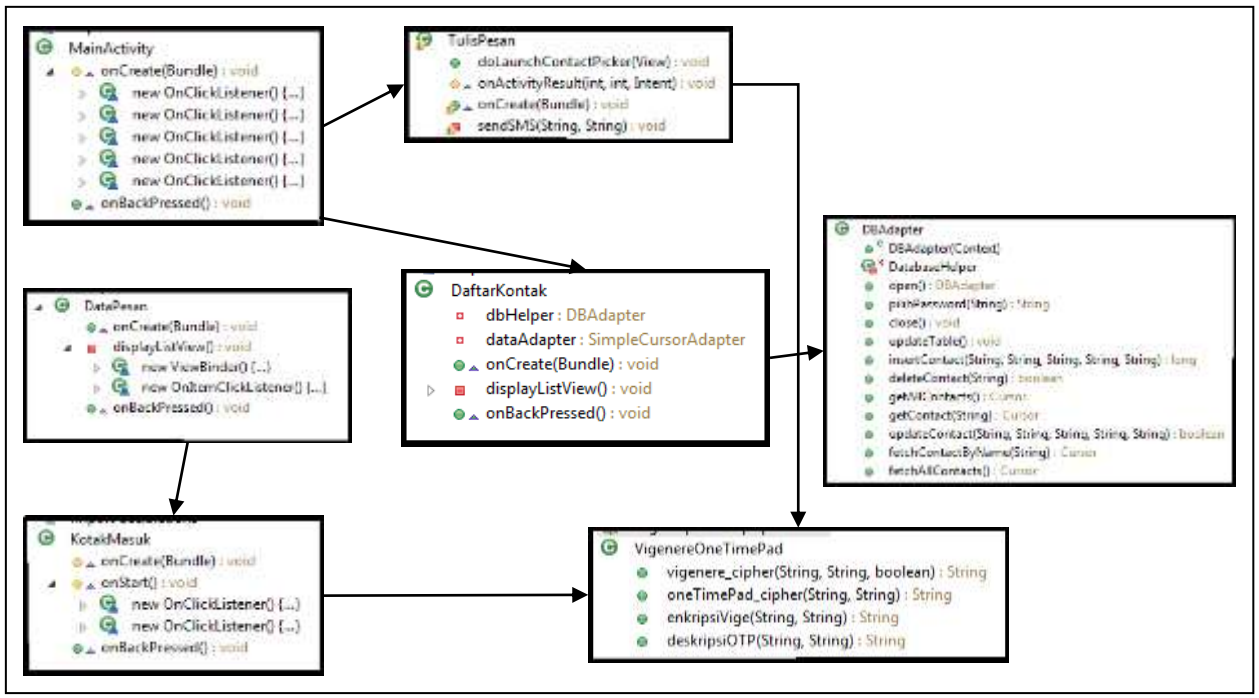
III.6.3. *Class Diagram*

Pembahasan class diagram pada aplikasi SMS terdapat 7 class yaitu MainActivity, TulisPesan, KotakMasuk, DataPesan, DBAdapter, VigenereOneTimePad dan DaftarKontak. Masing-masing class dapat dijelaskan sebagai berikut:

1. MainActivity : Class yang berisikan menu-menu pada aplikasi SMS yang ingin dijalankan.
2. TulisPesan : Class yang digunakan untuk mengirim, mengenkripsi dan menyimpan pesan.
3. KotakMasuk : Class untuk mendeskripsi pesan.
4. DataPesan : Class untuk menyimpan pesan terkirim atau pesan masuk.

5. DBAdapter : Class untuk menyimpan kontak yang telah dikirim pesan ciphertext ke dalam database.
6. VigenereOneTimePad : Class untuk enkripsi dan deskripsi pesan.
7. DaftarKontak : Class untuk menampilkan kontak yang telah terdaftar.

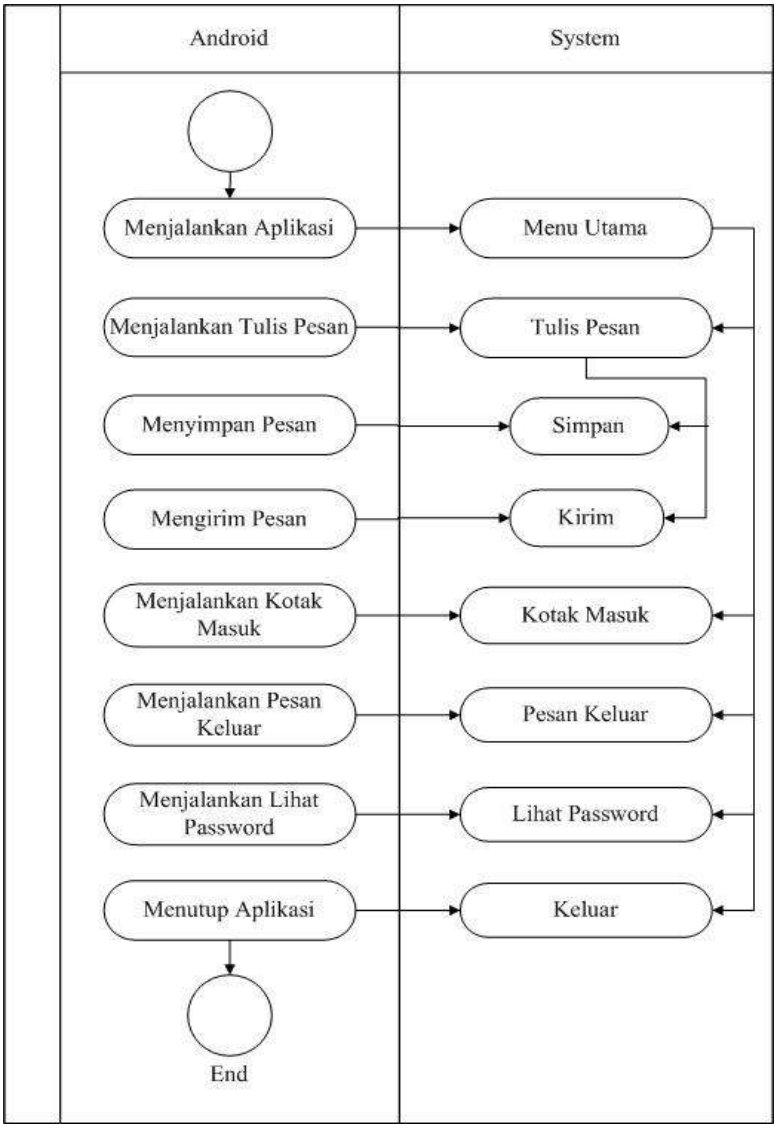
Di bawah ini merupakan gambar class diagram aplikasi SMS.



Gambar III.7. *Class Diagram* Aplikasi Pengamanan SMS

III.6.4. *Activity Diagram*

Activity diagram merupakan penggambaran tentang aktivitas atau proses berjalannya sistem dari suatu aplikasi melalui diagram. Dimana terdapat beberapa tampilan menu pilihan yang akan digunakan pada saat aplikasi dijalankan. Berikut *activity diagram* aplikasi pengamanan SMS pada gambar III.7.



Gambar III.8. *Activity Diagram* Aplikasi Pengamanan SMS

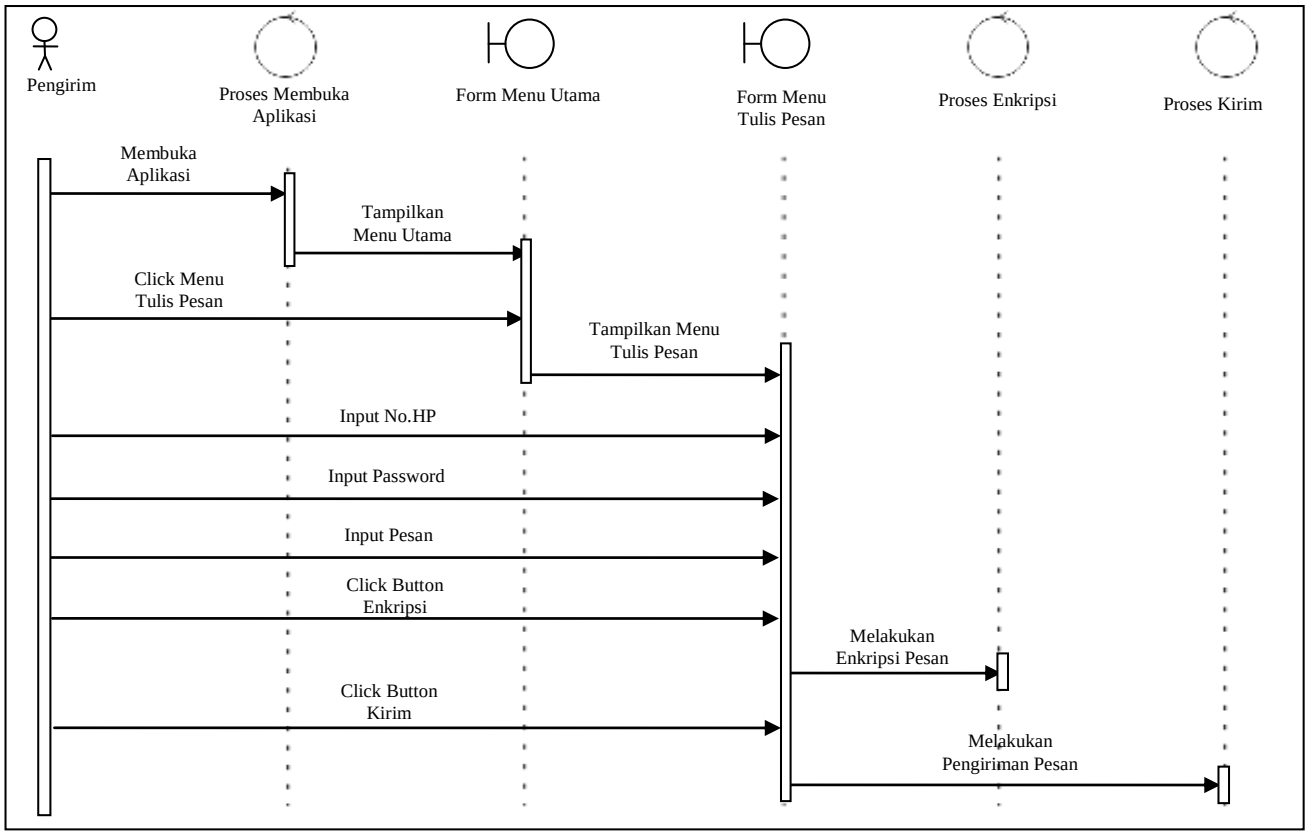
Dari Gambar *Activity Diagram* di atas menjelaskan tentang gambaran *system* aplikasi keamanan SMS pada android saat dijalankan oleh pengguna.

III.6.5. *Sequence Diagram*

Pada perancangan ini untuk menjelaskan interaksi antara pengguna dan aplikasi yaitu menggunakan sequence diagram. Di bawah ini merupakan penjelasan dan gambar sequence diagram untuk pengriman pesan pada aplikasi yang dirancang.

1. Pengirim : yaitu pengguna yang mengirim pesan SMS.

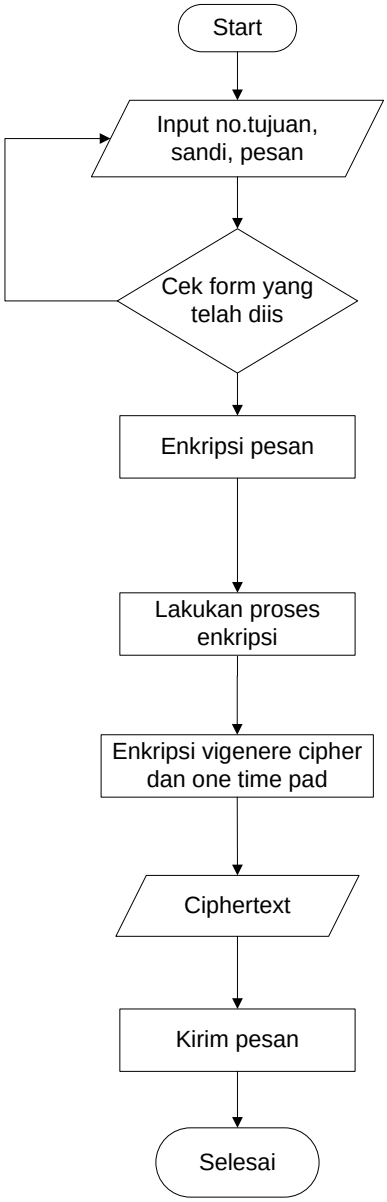
2. Form Menu Utama : yaitu tampilan saat aplikasi pertama dibuka.
3. Form Tulis Pesan : yaitu tampilan untuk mengenkripsi, menyimpan dan mengirim pesan SMS.



Gambar III.9. *Sequence Diagram* Aplikasi Pengamanan SMS

III.6.6. *Flowchart*

F*lowchart* adalah gambaran atau skema dari suatu aplikasi menggunakan grafik atau symbol-simbol yang disatu padukan sehingga menjadi *system* dari suatu aplikasi tersebut. Berikut *flowchart* tentang system aplikasi pengamanan SMS.



Gambar III.10. *Flowchart* Aplikasi Pengamanan SMS

III.7. Desain *Interface*

Di dalam sebuah aplikasi pengamanan SMS terdapat beberapa tampilan desain *form*, menu dan lain-lain apabila kita menggunakan aplikasi tersebut. Berikut beberapa tampilan yang digunakan penulis dalam perancangan aplikasi ini.

1. Rancangan Menu Utama

Tampilan menu merupakan pilihan yang dapat dilakukan pengguna dalam menggunakan aplikasi keamanan SMS tersebut. Rancangannya dapat dilihat pada gambar III.11.

Gambar III.11. Rancangan Menu Utama

Dari tampilan rancangan di atas ada beberapa menu pilihan. Berikut penjelasannya:

1. Tulis pesan adalah menu yang digunakan untuk mengirim pesan kepada orang lain.
2. Pesan masuk adalah menu yang berfungsi untuk melihat pesan yang telah diterima dari orang lain.
3. Pesan keluar merupakan menu yang digunakan untuk melihat pesan yang telah terkirim.
4. Lihat *password* merupakan menu yang digunakan untuk melihat rincian pesan dan *password* yang telah disimpan.
5. Keluar adalah menu untuk menutup aplikasi.

2. Rancangan Tulis Pesan

Tampilan tulis pesan adalah tampilan dimana pesan akan diciptakan atau dibuat dan dikirim ke nomor tujuan. Tampilan buat pesan dapat dilihat pada gambar III.12.

Gambar III.12. Rancangan Menu Tulis Pesan

3. Rancangan Kotak Masuk Dan Pesan Terkirim

Pada rancangan kotak masuk dan pesan terkirim tampilannya sama, lebih sederhana hanya menampilkan nama/nomor, tanggal dan pesan secara berulang-ulang tergantung pesan yang diterima. Berikut tampilan rancangan kotak masuk dan pesan terkirim.

Gambar III.13. Rancangan Menu Kotak Masuk

Gambar III.14. Rancangan Menu Pesan Terkirim

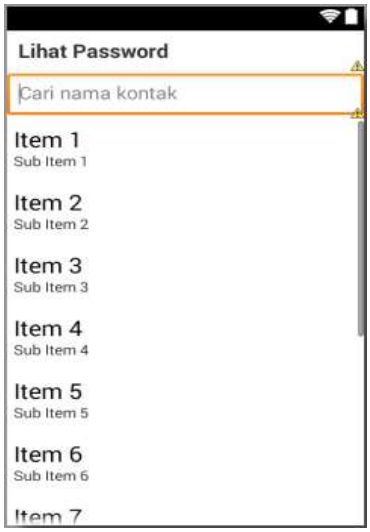
4. Rancangan Deskripsi Pesan

Rancangan deskripsi pesan berfungsi untuk mengubah *ciphertext* menjadi *plaintext*. Pada rancangan ini penulis menggunakan 3 *TextView*, 1 *EditText* dan 3 *Button*, dapat dilihat pada gambar III.15.

Gambar III.15. Rancangan Deskripsi Pesan

5. Rancangan Lihat *Password*

Pada rancangan ini terdapat beberapa informasi termasuk *password*, rancangan ini sangat berguna apabila kita lupa *password* dengan pesan yang kita kirim dengan menyesuaikan informasi lain yaitu tanggal, pesan, nomor tujuan dan lain lain. Berikut tampilan rancangan lihat *password* pada gambar



Gambar III.16. Rancangan Menu Lihat *Password*