BAB III

ANALISIS DAN PERANCANGAN

III.1. Analisis

III.1.1 Analisis Masalah

Seiring dengan perkembangan teknologi, keamanan dalam berteknologi merupakan hal yang sangat penting. Salah satu cara mengamankan data teks adalah dengan menggunakan metode kriptografi. Hal ini dikarenakan metode kriptografi sangat mudah diimplementasikan. Meskipun kriptografi adalah salah satu cara untuk mengamankan data teks, namun masih ada kekurangannya yaitu metode kriptografi telah diketahui oleh banyak orang. Karena itu dibutuhkan suatu metode, untuk mendapatkan keamanan lebih dalam informasi atau data teks, khususnya yang bersifat rahasia.

III.1.2 Analisis Spesifikasi

Dalam pembuatan aplikasi ini dibutuhkan spesifikas, antara lain:

1. Menggunakan *intel core i3*.
2. Menggunakan *memory* 2 GB.
3. Program aplikasi dibuat menggunakan *Eclipse*.
4. Metode yang digunakan pada proses kriptografi adalah *AES 128*
5. Data yang dienkripsi adalah teks.

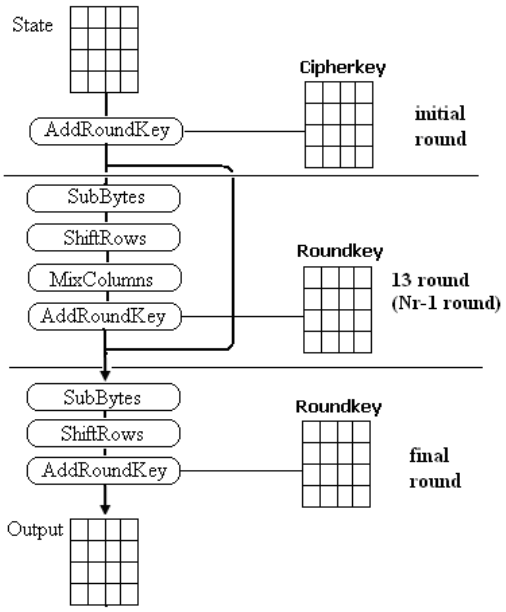
Nantinya aplikasi yang dibuat ini dapat berjalan dengan smartphone android yang memiliki spesifikasi sebagai berikut:

1. Menggunakan sistem operasi andoid 4.4 KitKat atau yang lebih tinggi
2. Membutuhkan RAM minimal 512.

III.2. Penerapan Metode

Algoritma AES menggunakan substitusi dan permutasi, dan sejumlah putaran (cipher berulang), dimana setiap putaran menggunakan kunci yang berbeda (kunci setiap putaran disebut round key). AES menetapkan panjang kuncinya 128, 192, dan 256 bit. Karena itu, maka dikenal AES-128, AES-192, dan AES-256.

Algoritma AES menggunakan substitusi dan permutasi, dan sejumlah putaran (cipher berulang), dimana setiap putaran menggunakan kunci yang berbeda (kunci setiap putaran disebut round key).



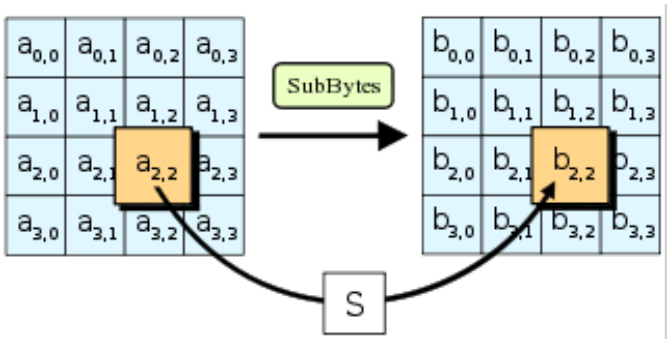
Gambar III.1 Diagram Proses Enkripsi AES

Garis besar Algoritma AES yang beroperasi pada blok 128 bit dengan kunci 128 bit adalah sebagai berikut (di luar proses pembangkitan round key):

1. AddRoundKey: melakukan XOR antara state awal (plainteks) dengan cipher key. Tahap ini juga disebut initial round. Putaran sebanyak Nr - 1 kali. Proses yang dilakukan pada setiap putaran.
2. SubBytes: Substitusi byte dengan menggunakan tabel substitusi (S-Box). Tabel 2 merupakan tabel S-Box SubBytes. Untuk satiap byte pada array state, misalkan S[r,c] = xy yang dalam hal ini xy adalah digit heksadeimal dari nilai S[r,c], maka nilai substitusinya, yang dinyatakan dengan S'[r,c], adalah elemen di dalam S-Box yang merupakan perpotongan baris x dengan kolom y. Gambar 2 merupakan gambar proses tranformasi SubBytes.

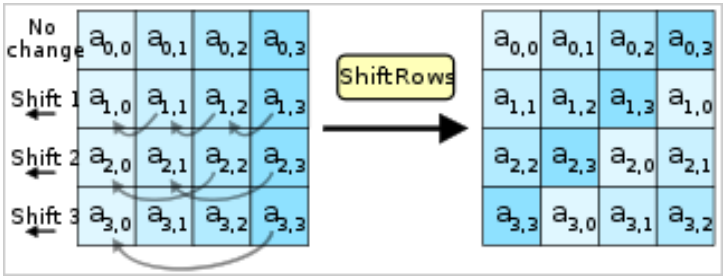
Tabel III.1 *S-Box SubBytes*

		y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
x	0	63	7c	77	7b	f2	6b	6f	c5	30	01	67	2b	fe	d7	ab	76
	1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
	2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
	3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
	4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
	5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
	6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
	7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
	8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
	9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
	a	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
	b	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
	c	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
	d	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
	e	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
	f	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16



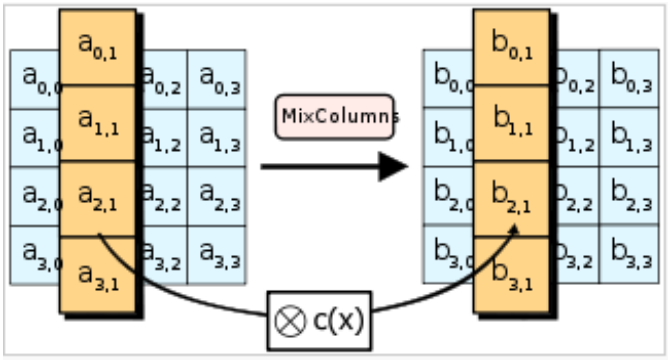
Gambar III.2 Transformasi *SubBytes*

3. ShiftRows: pergeseran baris-baris array state secara wrapping pada 3 baris terakhir dari array state, dimana pada proses ini bit paling kiri akan dipindahkan menjadi bit paling kanan (rotasi bit). Jumlah pergeseran bergantung pada nilai baris (r). Baris r=1 digeser sejauh 1 byte, baris r=2 digeser sejauh 2 byte,dan baris r=3 digeser sejauh 3 byte. Baris r=0 tidak digeser. (Gambar 3).



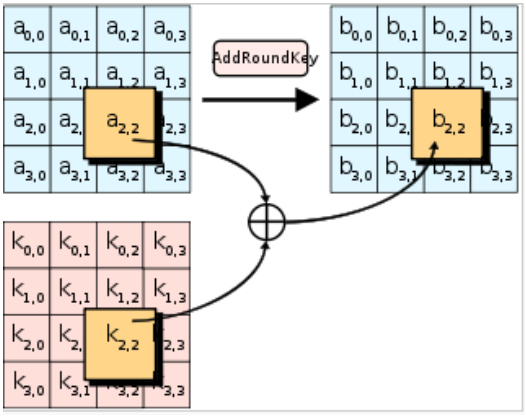
Gambar III.3 Transformasi *ShiftRows*

4. MixColumns: mengacak data di masing-masing kolom array state (Gambar 4). Dalam proses MixColumn terdapat beberapa perkalian, yaitu Matrix Multiplication dan Galois Field Multiplication.



Gambar III.4 Transformasi *MixColumns*

5. AddRoundKey: melakukan XOR antara state sekarang dengan round key.



Gambar III.5 Transformasi *AddRoundKey*

6. Final Round (proses untuk putaran terakhir): SubBytes, ShiftRows, dan AddRoundKey.

III.2.1 Struktur Enkripsi AES

Proses didalam AES merupakan transformasi terhadap *state*. Sebuah teks asli dalam blok (128 *bit*) terlebih dahulu di organisir sbagai *state*. Enkripsi AES adalah transformasi terhadap state secara berulang dalam beberapa ronde. *State* yang menjadi keluaran ronde k menjadi masukkan untuk ronde ke-*k*+1. Pada awalnya teks asli di reorganisasi sebagai sebuah *state*. Kemudian sebelum ronde 1 dimulai blok teks asli dicampur dengan kunci ronde ke-0 (transformasi ini disebut

AddRoundKey). Setelah itu, ronde ke-1 sampai dengan ronde ke-(*Nr* – 1) dengan *Nr* adalah jumlah ronde menggunakan 4 jens transformasi, yaitu *SubBytes*, *ShiftRows*, *MixColumns* dan *AddRoundKey*. Pada ronde terakhir, yaitu ronde ke-*Nr* dilakukan transformasi serupa dengan ronde lain namun tanpa transfomasi *Mix Coloumns*.

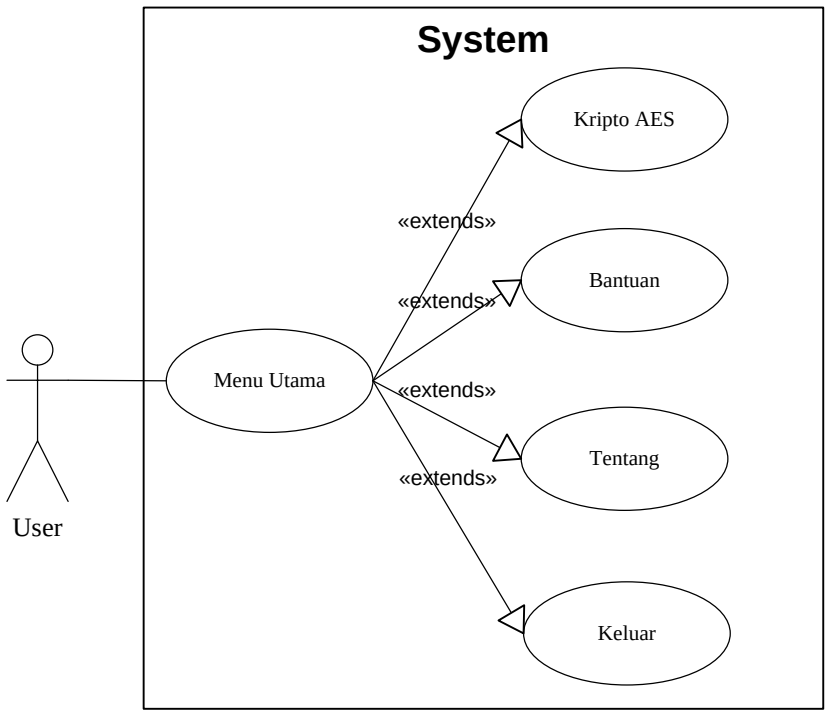
III.2.2 Struktur Dekripsi AES

Secara ringkas algoritma dekripsi merupakan kebalikan dari algoritma enkripsi AES. Algoritma dekripsi AES menggunakan transformasi invers semua transformasi dasar yang digunakan pada algoritma enkripsi AES. Setiap transformasi dasar AES memiliki transformasi invers, yaitu : *InvSubBytes*, *InvShiftRows* dan *InvMixColoumn*. *AddRoundKey* merupakan transformasi yang bersifat *self-invers* dengan syarat menggunakan kunci yang sama.

III.3. Perancangan

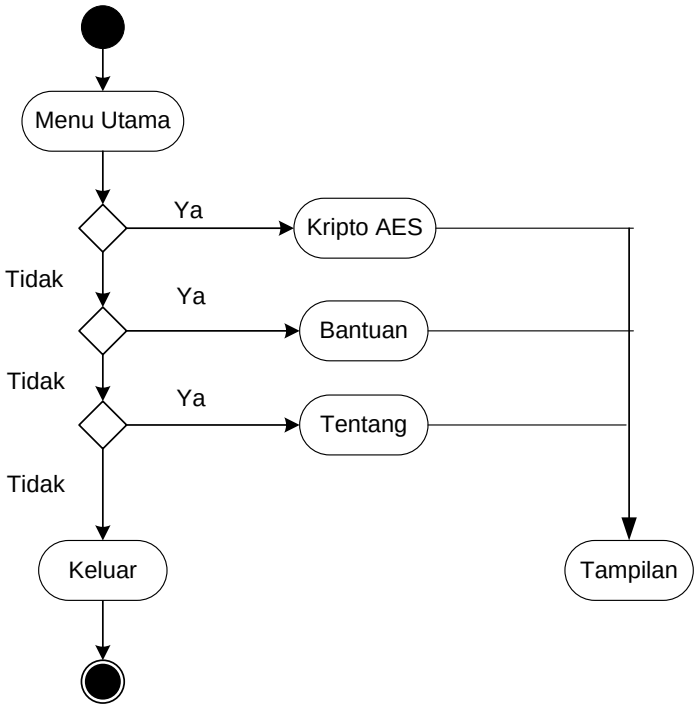
III.3.1 *Use Case Diagram*

Kegiatan interaksi antara aktor terhadap sistem ditunjukan pada *use case diagram*, Aktor yang terlibat dalam kegiatan tersebut adalah *user*. *Use case diagram* perangkat lunak yang dibangun terlihat pada gambar berikut:



Gambar III.5 *Use Case Diagram* Perancangan Aplikasi Enkripsi Teks menggunakan Metode AES berbasis Android

III.3.2 *Activity Diagram*

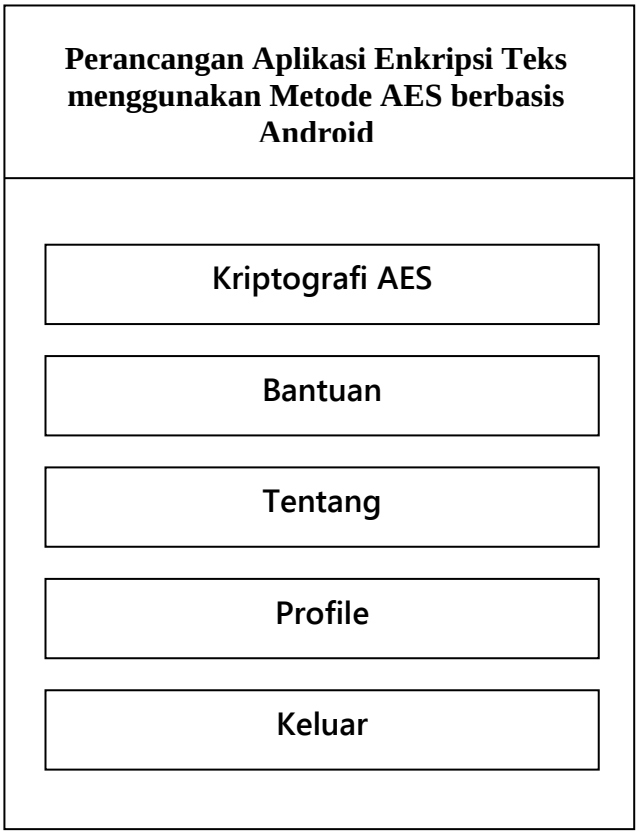


Gambar III.6 *Activity Diagram* Perancangan Aplikasi Enkripsi Teks menggunakan Metode AES berbasis Android

III.4. Desain User Interface

III.4.1 Antarmuka *Form* Home

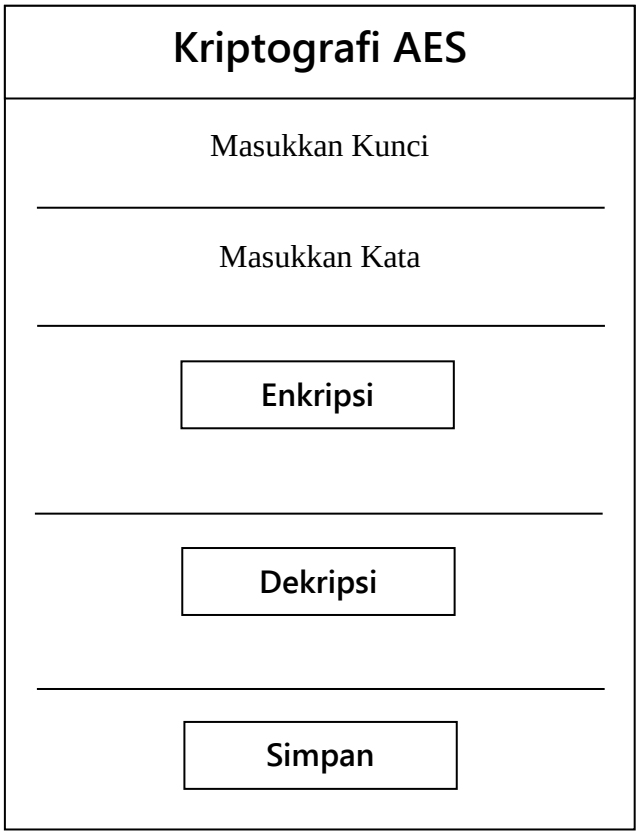
Rancangan *form* ini dibuat sebagai *form home* dimana di *form* ini ada terdapat empat *button* yang akan membuka *form* lain seperti *form* Kriptografi AES, *Form* Bantuan, *form* Tentang, *form Profile* dan *button* keluar.



Gambar III.7 Tampilan *Form* Home

III.4.2 Antarmuka *Form* Kriptografi AES

Rancangan *form* ini dibuat untuk menampilkan form kriptografi AES, dimana form ini terdapat proses untuk enkripsi dan dekripsi AES. Berikut adalah rancangan desain yang penulis buat.



Gambar III.8 Tampilan *Form* Kriptografi AES

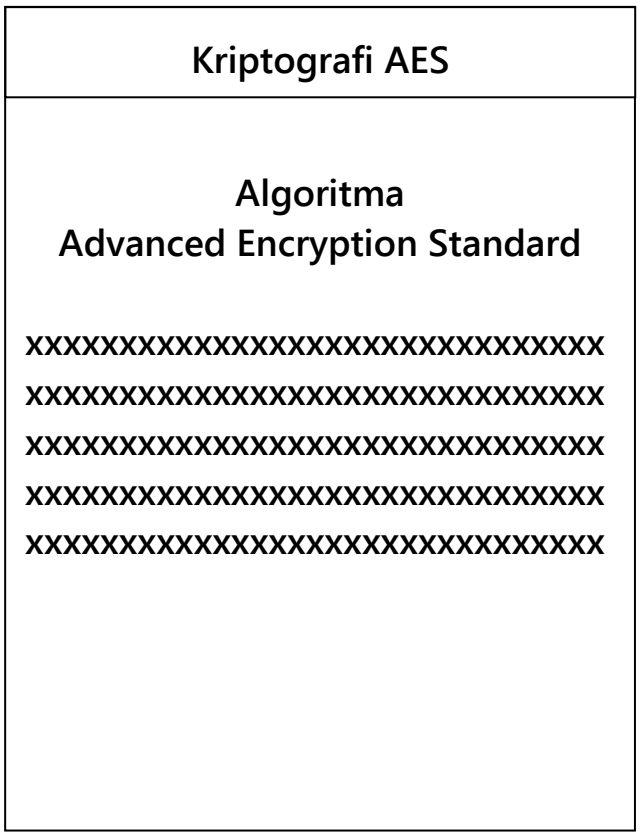
III.4.3 Antarmuka *Form* Bantuan

Rancangan *form* ini dibuat untuk memberikan informasi penjelasan tentang penyelasian proses kriptografi AES. Dengan form bantuan user dapat mengetahui dasar perhitungan dari kriptografi AES dengan pengenalan yang singkat. Berikut adalah rancangan desain yang penulis buat.

Gambar III.9 Tampilan *Form* Bantuan

III.4.4 Antarmuka *Form* Tentang

Rancangan *form* ini dibuat untuk memberikan informasi pengertian atau penjelasan tentang kriptografi AES. Dengan form bantuan *user* dapat mengetahui dasar dari kriptografi AES dengan pengenalan yang singkat. Berikut adalah rancangan desain yang penulis buat.



Gambar III.10 Tampilan *Form* Tentang

III.4.5 Antarmuka *Form Profile*

Rancangan *form* ini dibuat sebagai bertujuan untuk menampilkan informasi tentang pembuat perancangan aplikasi enkripsi teks menggunakan metode aes berbasis android. Berikut adalah gambang rancangan pada *form profile*.

Gambar III.11 Tampilan *Form Profile*